MEDUSA

Model of Early Diagenesis in the Upper Sediment with Adaptable complexity

—

Technical Reference

Guy MUNHOVEN
Université de Liège, Belgium
http://www.astro.ulg.ac.be/~munhoven/

Version 1.0

Contents

Chapter 1

Discretisation of the Equations

1.1 Introduction

The complete sediment column is subdivided into threeor four vertically stacked parts, called realms, as illustrated on Fig. 1.1:

1. REACLAY, the top-most part extending downwards from the sediment top at the sediment-water interface and where the chemical reactions are taken into consideration;
2. TRANLAY, the transition layer of changing thickness just underneath, acting as a temporary storage to connect REACLAY to the underlying CORELAY;
3. CORELAY, a stack of sedimentary layers representing the deep sediment, i. e., the sediment core;
4. additionally an optional Diffusive Boundary Layer (DBL—not to scale on Fig. 1.1) acting as a diffusive barrier to the sediment-water exchange of solutes can be included on top of the REACLAY realm.

REACLAY includes the bioturbated sedimentary mixed-layer, where most of the reactions relevant for early diagenesis take place (organic matter remineralisation, carbonate dissolution etc.).

1.2 Continuity Equations

MEDUSA calls upon the standard diagenesis reaction (Berner, 1980; Boudreau, 1997):

$$\boxed{\frac{\partial \hat{C}_i}{\partial t} + \frac{\partial \hat{J}_i}{\partial z} - \hat{S}_i = 0,} \tag{1.1}$$

where

- t is time
- z is depth, positive downwards
- $\hat{C}_i$ is the concentration of constituent i per unit volume of total sediment (solid + porewater phases), related to the phase-specific concentrations C_i^{f} (solutes) and C_i^{s} (solids) by
  - $\hat{C}_i = \varphi^{\mathrm{f}} C_i^{\mathrm{f}}$ for solutes, where φ^{f} is the porewater fraction of the total sediment volume, linked to porosity, φ, by $\varphi^{\mathrm{f}} = \varphi$
  - $\hat{C}_i = \varphi^{\mathrm{s}} C_i^{\mathrm{s}}$ for solids, where φ^{s} is the solid fraction, linked to porosity, φ, by $\varphi^{\mathrm{s}} = (1 - \varphi)$
- $\hat{J}_i$ is the local transport term (advection, diffusion) for constituent i, i.e., the flux density of constituent i per unit area of total sediment

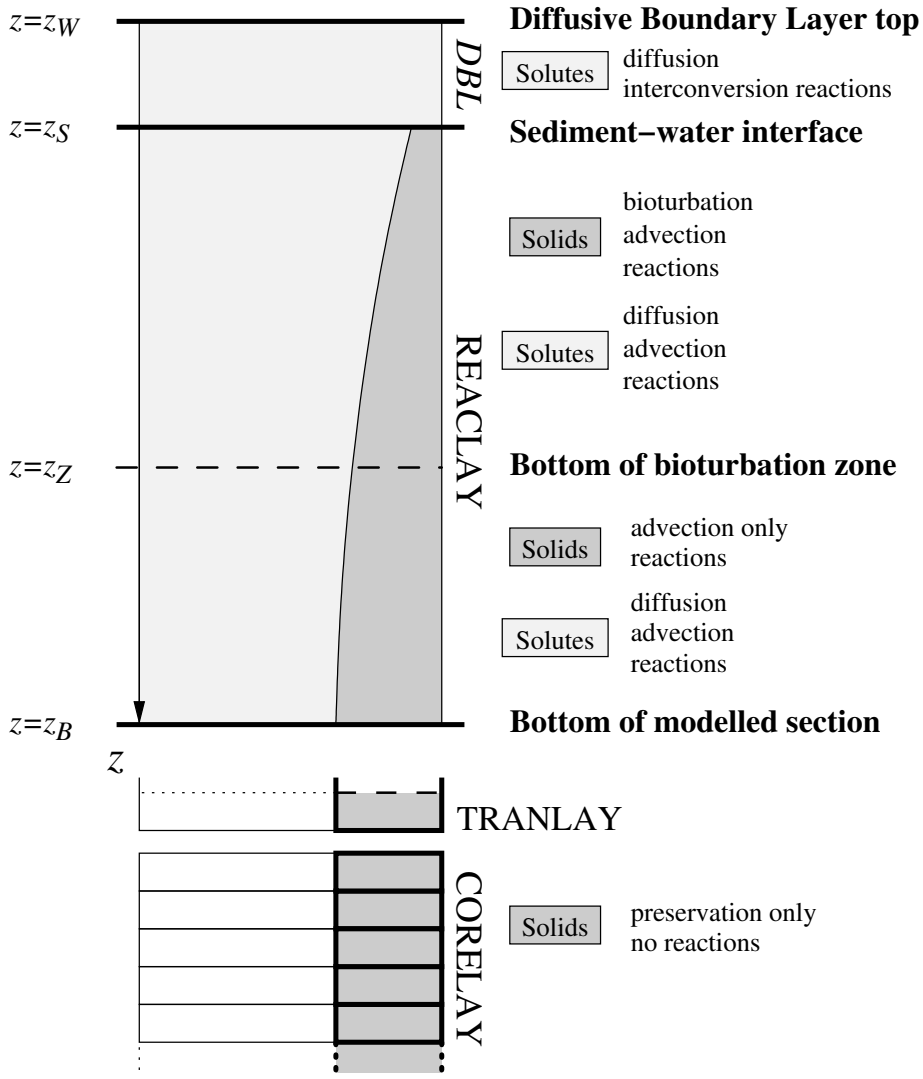

Figure 1.1: Partitioning of the sediment column in MEDUSA: an optional diffusive boundary layer (DBL) on top of the main part of the model sediment where diagenetic reactions and advective-diffusive transport take place (REACLAY), the transition layer (TRANLAY) and the core represented by the stack of layers (CORELAY). The bottom of the bioturbation zone may coincide with the bottom of REACLAY (see text for details).

- $\hat{S}_i$ is the net source-minus-sink balance for constituent i per unit volume of total sediment, equal to
  - $\hat{S}_i = \hat{R}_i^V$ for solids
  - $\hat{S}_i = \hat{R}_i^V + \hat{r}_i^V + \hat{Q}_i^V$ for solutes

  where
  - $\hat{R}_i^V = \hat{P}_i^V - \hat{D}_i^V$, with $\hat{P}_i^V \geq 0$ and $\hat{D}_i^V \geq 0$ being the total production and destruction (or disappearance) rates for constituent i, respectively, per unit volume of total sediment
  - $\hat{r}_i^V$ is the net fast interconversion reaction rate (per unit volume of total sediment), that is going to be filtered out of the equations by an equilibrium consideration
  - $\hat{Q}_i$ represents non-local transport (irrigation) per unit volume of total sediment.

1.2.1 Solids: transport expression and additional equations

For solids, we have

$$\boxed{\hat{J}_i = -\varphi^{\mathrm{s}} D^{\mathrm{bt}} \frac{\partial C_i^{\mathrm{s}}}{\partial z} + \left(\varphi^{\mathrm{s}} w - \beta D^{\mathrm{bt}} \frac{\partial \varphi^{\mathrm{s}}}{\partial z} \right) C_i^{\mathrm{s}}} \tag{1.2}$$

where

- D^{bt} is the total biodiffusion coefficient (inter- plus intraphase)
- w is the solids' advection rate
- β is the interphase fraction of the biodiffusive transport

We take into account the static volume conservation equation

$$\boxed{\sum_{i \in I^{\mathrm{s}}} \vartheta_i C_i^{\mathrm{s}} = 1} \tag{1.3}$$

where

- I^{s} is the inventory of solid constituents
- ϑ_i is the specific volume of the solid i, each supposed to be constant, in which case it is linked to the specific density of i, ρ_i, by $\vartheta_i = 1/\rho_i$.

The total volume flux is thus

$$\sum_{i \in I^{\mathrm{s}}} \vartheta_i \hat{J}_i = \varphi^{\mathrm{s}} w - \beta D^{\mathrm{bt}} \frac{\partial \varphi^{\mathrm{s}}}{\partial z}.$$

In combination with the general equation (1.1) for solids, we can thus write the volume change equation

$$\frac{\partial \varphi^{\mathrm{s}}}{\partial t} + \frac{\partial}{\partial z} \left(\varphi^{\mathrm{s}} w - \beta D^{\mathrm{bt}} \frac{\partial \varphi^{\mathrm{s}}}{\partial z} \right) = \sum_{i \in I^{\mathrm{s}}} \vartheta_i \hat{R}_i. \tag{1.4}$$

For steady-state porosity, we can directly integrate this equation and find

$$\boxed{\varphi^s w - \beta D^{\mathrm{bt}} \frac{\partial \varphi^s}{\partial z} = \sum_{i \in I^{\mathrm{s}}} \vartheta_i \hat{I}_i^{\mathrm{top}} + \int_{z_T}^{z} \sum_{i \in I^{\mathrm{s}}} \vartheta_i \hat{R}_i^V(z')\, dz'.} \tag{1.5}$$

where $\hat{I}_i^{\mathrm{top}}$ is the deposition rate of solid i at the sediment-water interface. Equation (1.5) is used to calculate the advection profile w.

1.2.2 Solutes: transport expressions

For solutes, the local transport includes, in general, advection and diffusion:

$$\hat{J}_i = -\varphi^{\mathrm{f}} \left(\frac{D_i^{\mathrm{sw}}}{\theta^2} + \beta D^{\mathrm{bt}} \right) \frac{\partial C_i^{\mathrm{f}}}{\partial z} + \left(\varphi^{\mathrm{f}} u - \beta D^{\mathrm{bt}} \frac{\partial \varphi^{\mathrm{f}}}{\partial z} \right) C_i^{\mathrm{f}} \tag{1.6}$$

where

- D_i^{sw} is the free seawater diffusion coefficient for constituent i
- θ^2 is tortuosity
- u is the porewater advection rate.

Here, we simplify the solute transport term neglecting porewater advection and the effet of interphase bioturbation upon solutes:

$$\boxed{\hat{J}_i = -\varphi^{\mathrm{f}} \left(\frac{D_i^{\mathrm{sw}}}{\theta^2} \right) \frac{\partial C_i^{\mathrm{f}}}{\partial z}} \tag{1.7}$$

Bioirrigation provides a non-local transport mode for solutes. In MEDUSA, the source-sink approach (Boudreau, 1984) is used to quantify the effect of bioirrigation:

$$\boxed{\hat{Q}_i = \alpha \varphi^{\mathrm{f}} (C_i^{\mathrm{oc}} - C_i^{\mathrm{f}})} \tag{1.8}$$

where

- $\alpha(z)$ is the, possibly depth-dependent, bioirrigation "constant";
- C_i^{oc} the concentration of solute i in the irrigation channels, set equal to the solute's concentration in seawater overlying the sediment.

Extension

By applying the solute transport expression to the porewater solvent, i.e., seawater, whose concentration in the porewater phase, $C_{\mathrm{sw}}^{\mathrm{f}}$, is sensibly constant, we deduce that

$$\hat{J}_{\mathrm{sw}} = \left(\varphi^{\mathrm{f}} u - \beta D^{\mathrm{bt}} \frac{\partial \varphi^{\mathrm{f}}}{\partial z} \right) C_{\mathrm{sw}}^{\mathrm{f}}$$

When introduced into the general equation (1.1), we find which then leads to

$$\frac{\partial \varphi^{\mathrm{f}}}{\partial t} + \frac{\partial}{\partial z} \left(\varphi^{\mathrm{f}} u - \beta D^{\mathrm{bt}} \frac{\partial \varphi^{\mathrm{f}}}{\partial z} \right) = 0. \tag{1.9}$$

By adding equations (1.4) and (1.9), and noting that $\varphi^{\mathrm{s}} + \varphi^{\mathrm{f}} = 1$, we get the generally valid constraint, even for non steady-state conditions:

$$\frac{\partial}{\partial z} \left(\varphi^{\mathrm{s}} w + \varphi^{\mathrm{f}} u \right) = \sum_{i \in I^{\mathrm{s}}} \vartheta_i \hat{R}_i.$$

1.2.3 Boundary-conditions and interface equations

Boundary conditions at the top

Boundary conditions at the top of the sediment are

- prescribed concentration for solutes:

$$\boxed{C_i^{\mathrm{f}}(t, z_{\mathrm{W}}) = C_i^{\mathrm{oc}}(t)} \tag{1.10}$$

  where W denotes the bottom of the free ocean above the sediment (located at top of the DBL if a DBL is present, or at the sediment-water if no DBL is considered);

- prescribed flux (deposition rate) for solids at the sediment-water-interface:

$$\boxed{\hat{J}_i(t, z_{\mathrm{S}}) = \hat{I}_i^{\mathrm{top}}(t)} \tag{1.11}$$

  where S denotes the sediment water-interface (top of the actual sediment, also the bottom of the DBL if a DBL is present).

Since W and S are different points when a DBL is present, the exact way boundary conditions for solutes are taken into account depends on whether a DBL is considered or not.

If a DBL is considered an additional interface condition, derived from the flux continuity assumption is required at the sediment-water-interface (S). The continuity of the water concentration at S and the water flux across S leads to

$$\left(\varphi^{\mathrm{f}} u - \beta D^{\mathrm{bt}} \frac{\partial \varphi^{\mathrm{f}}}{\partial z}\right)\Bigg|_{z_S^+} = u|_{z_S^-}$$

where '$+$' denotes the sediment-side of the S interface and '$-$' the DBL side (devoid of bioturbation and solids, and thus with $\varphi = 1$). Similarly, the continuity of concentration and flux for any solute i then translates in general to

$$-\varphi^{\mathrm{f}} \left(\frac{D_i^{\mathrm{sw}}}{\theta^2} + \beta D^{\mathrm{bt}}\right) \frac{\partial C_i^{\mathrm{f}}}{\partial z}\Bigg|_{z_S^+} = -D_i^{\mathrm{sw}} \frac{\partial C_i^{\mathrm{f}}}{\partial z}\Bigg|_{z_S^-}$$

i.e.

$$\varphi^{\mathrm{f}} \left(\frac{1}{\theta^2} + \frac{\beta D^{\mathrm{bt}}}{D_i^{\mathrm{sw}}}\right) \frac{\partial C_i^{\mathrm{f}}}{\partial z}\Bigg|_{z_S^+} = \frac{\partial C_i^{\mathrm{f}}}{\partial z}\Bigg|_{z_S^-}$$

Since we neglect the effect of interphase biodiffusion in the solute transport, we consider the following simplified interface equation

$$\boxed{\frac{\varphi^{\mathrm{f}}}{\theta^2} \frac{\partial C_i^{\mathrm{f}}}{\partial z}\Bigg|_{z_S^+} = \frac{\partial C_i^{\mathrm{f}}}{\partial z}\Bigg|_{z_S^-}} \tag{1.12}$$

If no DBL is considered W and S are the same and the boundary condition (1.10) is applied at S and the interface equation (1.12) is not applicable.

Interface equation at the bottom of the bioturbated zone

Flux continuity across the interface that delimits the bioturbated from the non bioturbated part of the sediment (Z) requires that

$$\left(\varphi^{\mathrm{s}} w\right)\big|_{z_Z^+} = \left(\varphi^{\mathrm{s}} w - \beta D^{\mathrm{bt}} \frac{\partial \varphi^{\mathrm{s}}}{\partial z}\right)\Bigg|_{z_Z^-}$$

Accordingly

$$\boxed{\left(\varphi^{\mathrm{s}} D^{\mathrm{bt}} \frac{\partial C_i^{\mathrm{s}}}{\partial z}\right)\Bigg|_{z_Z^-} = 0} \tag{1.13}$$

There is no special constraint that solutes have to fulfil at Z.

Bottom Boundary Conditions

Boundary conditions at the bottom of the modelled sediment (B) melt down to flux constraints, if any:

- for solutes, we adopt a no-flux boundary condition at the bottom, which translates to

$$-\varphi^{\mathrm{f}}\left(\frac{D_i^{\mathrm{sw}}}{\theta^2}+\beta D^{\mathrm{bt}}\right)\frac{\partial C_i^{\mathrm{f}}}{\partial z}+\left(\varphi^{\mathrm{f}}u-\beta D^{\mathrm{bt}}\frac{\partial \varphi^{\mathrm{f}}}{\partial z}\right)C_i^{\mathrm{f}}=0.$$

  This expression finally simplifies here to

$$\boxed{\left.\frac{\partial C_i^{\mathrm{f}}}{\partial z}\right|_{z_B^-}=0} \tag{1.14}$$

  as we neglect porewater advection and interphase biodiffusion for solutes.

- for solids, there is no boundary condition required at the bottom if $w|_{z_B^+}\geq 0$; if $w|_{z_B^+}<0$ (chemical erosion is taking place) a boundary condition with a prescribed input flux, similar to that at the sediment water interface, is required

If the bioturbated part of the modelled sediment section extends to the bottom (i.e., if $\mathrm{B}\equiv\mathrm{Z}$), the Z interface equation for solids must be applied at B.

1.2.4 Differential Algebraic Equations for Solutes

The reaction terms $\hat{r}_i^V$ related to the fast interconversion reactions between solutes are filtered out of the equation system by making the assumption that these reactions are locally at equilibrium. This is done by replacing the total system of solutes' equations with suitably chosen linear combinations of the initial equations so that the $\hat{r}_i^V$ cancel out, and some other equations replaced by the laws of mass-action for the equilibria under consideration: if n solute constituents are involved in $m<n$ equilibria (fast interconversion reactions), then the n partial differential equations for the n solute concentrations are replaced by m equilibrium relationships and $n-m$ linear combination of these equations, chosen to be independent of each other.

1.3 Discretisation of the Continuity Equations

1.3.1 Grid characteristics

The numerical solution of the system of differential algebraic equations (DAEs) is based upon a grid-based discretisation of the equations. We adopt a finite volume approach, where the model domain is overlaid by a grid of nodes, from which a grid of vertices is derived, each vertex being located half-way between neighbouring nodes (so-called vertex-centred grids). This way, the model domain is subdivided into cells (called finite volumes), each one delimited by two consecutive vertices, thus including one node where the representative concentrations for the cell is anchored (and calculated from the DAE system). Both the DBL (if any) and the REACLAY parts of the model sediment are covered by such vertex-centred grids (see Fig. 1.2): for the DBL, that grid is denoted by

$$z_W,\ldots,z_{T-1}$$

for the surface sediment part (REACLAY) by

$$z_T,z_{T+1},\ldots,z_Z,\ldots,z_B.$$

The vertices are thus defined as follows.

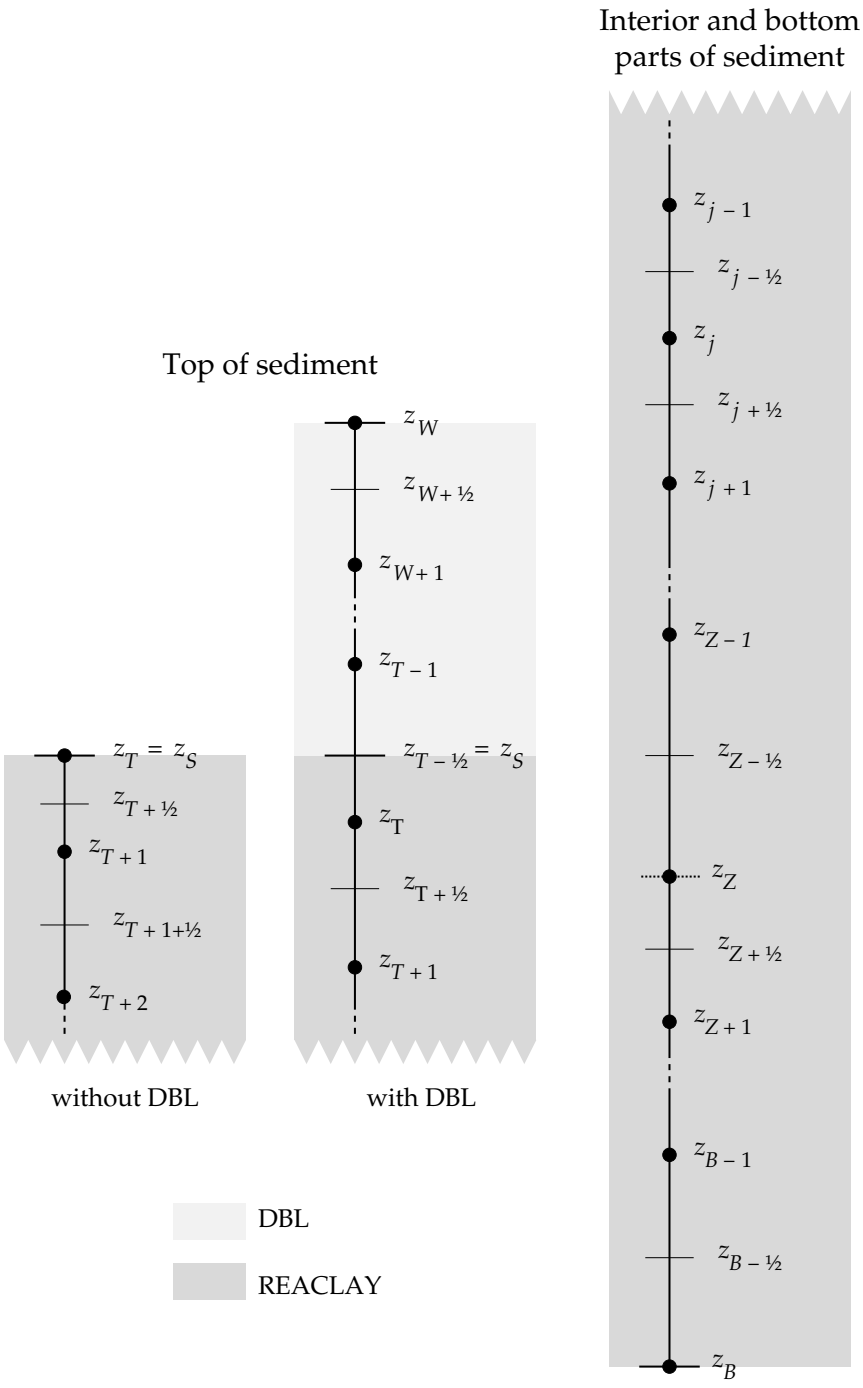

Figure 1.2: Grid characteristics for the Diffusive Boundary Layer (if any) and the REACLAY parts of the sediment.

- If a DBL is included,

$$z_{j+\frac{1}{2}} = \frac{1}{2}(z_j + z_{j+1}), \quad j = W, \ldots, T-2 \text{ and } j = T, \ldots, B-1$$

and $z_{T-\frac{1}{2}} = z_S$; additional vertices are located at z_W (top of the DBL, on the node W) and z_B (bottom of the sediment, on the node B).

If the DBL is represented by a single layer, $T = W + 1$.

- If no DBL is included, then

$$z_{j+\frac{1}{2}} = \frac{1}{2}(z_j + z_{j+1}), \quad j = T, \ldots, B-1$$

and additional vertices are located at $z_T \equiv z_W = z_S$ (top of the sediment, on the node T, which is identical to W and S) and at z_B (bottom of the sediment, on the node B).

The top-most and the bottom-most points of the modelled domain (referred to by T and B, resp.) are both part of the grids' nodes. The diffusive boundary layer is delimited by the z_W and z_S vertices.

We further denote

- the width of the cell around node j by $h_j = z_{j+\frac{1}{2}} - z_{j-\frac{1}{2}}$, except for $j = W$, where we set $h_W = \frac{1}{2}(z_W + z_{W+1}) - z_W = \frac{1}{2}(z_{W+1} - z_W)$ and for $j = B$, where we set $h_B = z_B - \frac{1}{2}(z_{B-1} + z_B) = \frac{1}{2}(z_B - z_{B-1})$;
- the distances between nodes j and $j+1$ by $h_{j+\frac{1}{2}} = z_{j+1} - z_j$.

If no DBL is included, then $W = T$. In this case the node T is placed at the sediment-water-interface. If a DBL is included, the T node is inside the model sediment, else it is located at the top of the model sediment column (REACLAY part). Please refer to chapter 4 for more technical details about the characteristics of the grids and their generation.

1.3.2 General finite volume discretisation

We use an implicit Euler discretisation for the time dimension (variable t, denumerated by superscripts n). In a cell represented by the node j, and delimited by the vertices $j - \frac{1}{2}$ and $j + \frac{1}{2}$, eqn. (1.1) is discretised as

$$\boxed{\frac{\hat{C}_j^{n+1} - \hat{C}_j^n}{t^{n+1} - t^n} + \frac{\hat{J}_{j+\frac{1}{2}}^{n+1} - \hat{J}_{j-\frac{1}{2}}^{n+1}}{h_j} - \hat{S}_j^{n+1} = 0.} \tag{1.15}$$

where $\hat{J}_{j+\frac{1}{2}}^{n+1}$, $\hat{J}_{j-\frac{1}{2}}^{n+1}$ and $\hat{S}_j^{n+1}$ approximate $\hat{J}(t_{n+1}, z_{j+\frac{1}{2}})$, $\hat{J}(t_{n+1}, z_{j-\frac{1}{2}})$ and $\hat{S}(t_{n+1}, z_j)$, respectively.

This equation is used as is at every node j, $W < j < B$. It is slightly adapted at the W and at the B nodes and it is ignored for solids at nodes above the S vertex if a DBL is included. At the node W, the equation actually used writes

$$\boxed{\frac{\hat{C}_W^{n+1} - \hat{C}_W^n}{t^{n+1} - t^n} + \frac{\hat{J}_{W+\frac{1}{2}}^{n+1} - \hat{J}_W^{n+1}}{h_W} - \hat{S}_W^{n+1} = 0.} \tag{1.16}$$

For solutes, the previous equation is only used to calculate $\hat{J}_W^{n+1} + h_W \hat{r}_W^{n+1}$ as the top boundary condition directly sets $\hat{C}_W^{n+1}$ (and $\hat{C}_W^n$). $\hat{J}_W^{n+1} + h_W \hat{r}_W^{n+1}$ is then corrected for the $h_W \hat{r}_W^{n+1}$ contribution by transforming the complete set of solute flux expression using the same set of linear combinations and equilibrium relationships as for the equation system, as described in section 1.2.4, and solving the resulting system for the individual solute concentrations (see section 1.3.5 below for details).

If no DBL is included in the model set-up, then $W \equiv T = S$ and the above equation is also used for solids, with $\hat{J}_W^{n+1} = \hat{J}_S^{n+1}$ set by the solids' boundary conditions; if a DBL is included then $T > W$ and $S = T - \frac{1}{2}$, and the general equation is used for the cell centred on the T node:

$$\boxed{\frac{\hat{C}_T^{n+1} - \hat{C}_T^n}{t^{n+1} - t^n} + \frac{\hat{J}_{T+\frac{1}{2}}^{n+1} - \hat{J}_S^{n+1}}{h_T} - \hat{S}_T^{n+1} = 0.} \tag{1.17}$$

In this case $\hat{J}_S^{n+1}$ is again set from the boundary conditions for solids; for solutes it is derived from the flux continuity equation at the vertex S (see below).

At the node B:

$$\boxed{\frac{\hat{C}_B^{n+1} - \hat{C}_B^n}{t^{n+1} - t^n} + \frac{\hat{J}_B^{n+1} - \hat{J}_{B-\frac{1}{2}}^{n+1}}{h_B} - \hat{S}_B^{n+1} = 0.} \tag{1.18}$$

Here, $\hat{J}_B^{n+1}$ is set from the boundary conditions (see below).

1.3.3 Flux discretisations

For the purpose of the developments in this section, we formally write the flux $\hat{J}$ as

$$\hat{J} = -d\frac{\partial c}{\partial z} + ac$$

where

- c stands for the phase specific concentration of a constituent (${C_i}^\alpha$);
- the "diffusion" coefficient d collects all the factors (including any φ^α) that multiply the derivative $\partial {C_i}^\alpha / \partial z$ in eqns. (1.2) and (1.7) (or (1.6));
- the "advection rate" a collects all the factors (including any φ^α) that multiply the concentration ${C_i}^\alpha$ in eqns. (1.2) and (1.7) (or (1.6)).

Using a centred difference for the discretisation of the gradient in the diffusive part of the flux, and a first-order upwind discretisation for the advective part (including the contribution from the diffusive flux not depending on the gradient, that we get after application of the chain rule to the derivatives of $\varphi^\alpha {C_i}^\alpha$), we get

$$\hat{J}_{j+\frac{1}{2}} = -d_{j+\frac{1}{2}} \frac{c_{j+1} - c_j}{h_{j+\frac{1}{2}}} + a_{j+\frac{1}{2}} \left(\frac{1 - \kappa_{j+\frac{1}{2}}}{2} c_{j+1} + \frac{1 + \kappa_{j+\frac{1}{2}}}{2} c_j\right),$$

where $\kappa_{j+\frac{1}{2}}$ is a parameter that controls the degree of upwinding (decentral weighting) at the vertex $j + \frac{1}{2}$, such that $0 \leq \kappa_{j+\frac{1}{2}} \leq 1$ if $a_{j+\frac{1}{2}} \geq 0$ and $-1 \leq \kappa_{j+\frac{1}{2}} \leq 0$ if $a_{j+\frac{1}{2}} \leq 0$. For a full upwind formulation, $\kappa_{j+\frac{1}{2}} = 1$ if $a_{j+\frac{1}{2}} \geq 0$ and $\kappa_{j+\frac{1}{2}} = -1$ if $a_{j+\frac{1}{2}} \leq 0$. In this case the resulting discretisation is first order and unconditionally positive. Intermediate values $-1 \leq \kappa_{j+\frac{1}{2}} \leq 1$ lead to more central-weighted discretisations of the advection term. This way the order can be increased to two for $\kappa_{j+\frac{1}{2}} = 0$. However, the resulting method is not unconditionally positive any more.

Collecting the terms in c_{j+1} and c_j in the previous equation leads to

$$J_{j+\frac{1}{2}} = -\left(d_{j+\frac{1}{2}} + \frac{h_{j+\frac{1}{2}}}{2} \kappa_{j+\frac{1}{2}} a_{j+\frac{1}{2}}\right) \frac{c_{j+1} - c_j}{h_{j+\frac{1}{2}}} + a_{j+\frac{1}{2}} \frac{c_{j+1} + c_j}{2},$$

showing that upwinding can be seen as adopting centred approximations for both diffusive and advective fluxes, with a diffusion coefficient increased by $\kappa_{j+\frac{1}{2}} a_{j+\frac{1}{2}} h_{j+\frac{1}{2}} / 2$.

Similarly,

$$\begin{aligned} J_{j-\frac{1}{2}} &= -d_{j-\frac{1}{2}}\frac{c_j - c_{j-1}}{h_{j-\frac{1}{2}}} + a_{j-\frac{1}{2}}\left(\frac{1-\kappa_{j-\frac{1}{2}}}{2}c_j + \frac{1+\kappa_{j-\frac{1}{2}}}{2}c_{j-1}\right) \\ &= -\left(d_{j-\frac{1}{2}} + \frac{h_{j-\frac{1}{2}}}{2}\kappa_{j-\frac{1}{2}}a_{j-\frac{1}{2}}\right)\frac{c_j - c_{j-1}}{h_{j-\frac{1}{2}}} + a_{j-\frac{1}{2}}\frac{c_j + c_{j-1}}{2}. \end{aligned}$$

Roos et al. (2008, p. 36) present this in a slightly different way. Instead of formally increasing the diffusion coefficient, they multiply it by a coefficient σ that is function of the ratio (ah/d), i.e., the *cell Péclet number*. The formulation adopted here can be reduced to that of Roos et al. (2008) by setting,

$$\sigma_{j+\frac{1}{2}} = 1 + \frac{a_{j+\frac{1}{2}}h_{j+\frac{1}{2}}}{2d_{j+\frac{1}{2}}}\kappa_{j+\frac{1}{2}}. \tag{1.19}$$

This is of course only feasible if $d \neq 0$.

We consider two different upwinding schemes, characterised by different appraches to set the upwinding coefficient κ.

Full upwind With the full upwind method, we adopt $\kappa_{j+\frac{1}{2}} = -1$ for $a_{j+\frac{1}{2}} < 0$ and $\kappa_{j+\frac{1}{2}} = 1$ for $a_{j+\frac{1}{2}} > 0$. This leads to a completely stable, albeit possibly diffusive, algorithm.

Exponential fitting Let us denote the cell Péclet number by $\mu = ah/d$; at the vertex $j + \frac{1}{2}$, we write it explicitly as $\mu_{j+\frac{1}{2}} = a_{j+\frac{1}{2}}h_{j+\frac{1}{2}}/d_{j+\frac{1}{2}}$. We can then adopt the upwind factor

$$\kappa_{j+\frac{1}{2}} = \frac{\exp(\mu_{j+\frac{1}{2}})+1}{\exp(\mu_{j+\frac{1}{2}})-1} - \frac{2}{\mu_{j+\frac{1}{2}}} = \coth\left(\frac{\mu_{j+\frac{1}{2}}}{2}\right) - \frac{2}{\mu_{j+\frac{1}{2}}}.$$

This scheme is actually the Il'in-Allen-Southwell scheme presented in Roos et al. (2008, p. 41). We may invert eq. (1.19) to express $\kappa_{j+\frac{1}{2}}$ as a function of $\sigma_{j+\frac{1}{2}}$

$$\sigma_{j+\frac{1}{2}} = 1 + \frac{a_{j+\frac{1}{2}}h_{j+\frac{1}{2}}}{2d_{j+\frac{1}{2}}}\kappa_{j+\frac{1}{2}} = 1 + \frac{\mu_{j+\frac{1}{2}}}{2}\kappa_{j+\frac{1}{2}} \;\Rightarrow\; \kappa_{j+\frac{1}{2}} = (\sigma_{j+\frac{1}{2}} - 1)\frac{2}{\mu_{j+\frac{1}{2}}}$$

For the Il'in-Allen-Southwell scheme,

$$\sigma_{j+\frac{1}{2}} = \frac{\mu_{j+\frac{1}{2}}}{2}\coth\left(\frac{\mu_{j+\frac{1}{2}}}{2}\right)$$

and hence,

$$\kappa_{j+\frac{1}{2}} = \coth\left(\frac{\mu_{j+\frac{1}{2}}}{2}\right) - \frac{2}{\mu_{j+\frac{1}{2}}}.$$

The formulation involving exponentials is notoriously unstable for evaluation in floating point arithmetic, and diverges for small $\mu_{j+\frac{1}{2}}$ (for $\mu_{j+\frac{1}{2}} \simeq 10^{-6}$, the result is already off by a factor of 100). The formulation involving coth is better, at least with regard to the order of magnitude of the result. For precise evaluations, one has to use the Laurent series for coth, which, for $\theta = \mu_{j+\frac{1}{2}}/2$, leads to (Abramowitz and Stegun, 1965)

$$\kappa_{j+\frac{1}{2}} = \frac{1}{3}\theta - \frac{1}{45}\theta^3 + \frac{2}{945}\theta^5 - \ldots + \frac{2^{2n}B_{2n}}{(2n)!}\theta^{2n-1} + \ldots \tag{1.20}$$

Series (1.20) converges for $|\theta| < \pi$. Following (Abramowitz and Stegun, 1965, eqn. (4.5.67), p. 85) we have

$$(-1)^{n-1}\frac{2^{2n-1}\pi^{2n}}{(2n)!}B_{2n} = \sum_{k=1}^{\infty}\frac{1}{k^{2n}} = \zeta(2n)$$

where ζ is the Riemann zeta function. The coefficients of the series are thus decreasing fast since $\zeta(2n)$ decreases with n and

$$(-1)^{n-1}\frac{2^{2n}B_{2n}}{(2n)!} = \frac{2\zeta(2n)}{\pi^{2n}}.$$

In addition, we have the following inequality (Abramowitz and Stegun, 1965, eqn. 23.1.15, p. 805)

$$\frac{2(2n)!}{(2\pi)^{2n}} < (-1)^{n-1}B_{2n} < \frac{2(2n)!}{(2\pi)^{2n}}\left(\frac{1}{1-2^{1-2n}}\right).$$

Accordingly

$$\frac{2}{\pi^{2n}} < (-1)^{n-1}\frac{2^{2n}B_{2n}}{(2n)!} < \frac{2}{\pi^{2n}}\left(\frac{1}{1-2^{1-2n}}\right).$$

For $n = 4$, $2/\pi^{2n}$ is already an excellent approximation for the coefficients (error less than 1%), that thus decrease almost by a factor of $\pi^2 \simeq 10$ for consecutive values of n. To order 11, the series writes

$$\kappa_{j+\frac{1}{2}} = \frac{1}{3}\theta - \frac{1}{45}\theta^3 + \frac{2}{945}\theta^5 - \frac{1}{4725}\theta^7 + \frac{2}{93555}\theta^9 - \frac{1382}{638512875}\theta^{11}$$

The calculation of $\kappa_{j+\frac{1}{2}}$ in MEDUSA uses the coth based formula for $\mu > \frac{1}{8}$,[1] and a decreasing number of terms as $\mu \to 0$ (starting with 5 terms, i.e., order 9 at $\mu = \frac{1}{8}$).

1.3.4 Boundary conditions

Top boundary conditions

Regardless of whether a DBL is included or not,

- the top boundary condition for solutes is always set at the W node (which is also always a vertex):

$$\boxed{C^{\mathrm{f}}_{i,W} = C_{i,\mathrm{Oc}}(t)} \tag{1.21}$$

  where $C_{i,\mathrm{Oc}}(t)$ is the bottom water concentration of the solute i;

- the top boundary condition for solids is always set at the S vertex (which is only a node if no DBL is included):

$$\boxed{\hat{J}^{n+1}_{i,S} = \hat{I}^{\mathrm{top}}_i(t^{n+1})} \tag{1.22}$$

If a DBL is included, an additional interface equation must be solved for solutes at the S vertex, in order to connect the DBL and REACLAY grids at the sediment-water interface. The flux continuity condition at the sediment-water interface[2]

$$\left.\frac{\varphi^{\mathrm{f}}}{\theta^2}\frac{\partial C^{\mathrm{f}}_i}{\partial z}\right|_{z_S^+} = \left.\frac{\partial C^{\mathrm{f}}_i}{\partial z}\right|_{z_S^-}$$

is discretised as

$$\left.\frac{\varphi^{\mathrm{f}}}{\theta^2}\right|_S \frac{C^{\mathrm{f}}_{i,T} - C^{\mathrm{f}}_{i,S}}{z_T - z_S} = \frac{C^{\mathrm{f}}_{i,S} - C^{\mathrm{f}}_{i,T-1}}{z_S - z_{T-1}}.$$

In order to calculate $C^{\mathrm{f}}_{i,S}$ which is not a concentration value on the grid, we rewrite this expression as

$$\left.\frac{\varphi^{\mathrm{f}}}{\theta^2}\right|_S \frac{z_S - z_{T-1}}{z_T - z_S}\left(C^{\mathrm{f}}_{i,T} - C^{\mathrm{f}}_{i,S}\right) = C^{\mathrm{f}}_{i,S} - C^{\mathrm{f}}_{i,T} + C^{\mathrm{f}}_{i,T} - C^{\mathrm{f}}_{i,T-1}$$

[1] More precisely, for `EXPONENT`$(\mu) < -2$, where `EXPONENT` is a Fortran 90 intrinsic function.

[2] This equation remains, by the way, valid as is even in case porewater advection is not neglected. The continuity condition for the advection rate u across z_S makes the advective terms on both sides cancel each other, regardless of the upwinding procedure adopted.

and

$$\left(\left. \frac{\varphi^{\mathrm{f}}}{\theta^2} \right|_S \frac{z_S - z_{T-1}}{z_T - z_S} + 1 \right) \left(C^{\mathrm{f}}_{i,T} - C^{\mathrm{f}}_{i,S} \right) = C^{\mathrm{f}}_{i,T} - C^{\mathrm{f}}_{i,T-1}.$$

Finally, we get

$$\boxed{C^{\mathrm{f}}_{i,S} = C^{\mathrm{f}}_{i,T} - \frac{C^{\mathrm{f}}_{i,T} - C^{\mathrm{f}}_{i,T-1}}{\left. \frac{\varphi^{\mathrm{f}}}{\theta^2} \right|_S \frac{z_S - z_{T-1}}{z_T - z_S} + 1}} \tag{1.23}$$

Notice that $\mathrm{W} = \mathrm{T} - 1$ if the DBL reduces to a single layer.

Bottom boundary conditions

The bottom boundary conditions directly reflect the physical condition they translate:

- for each solute, we adopt a no-flux boundary condition, i.e., we set

  $$\boxed{\hat{J}_{i,\mathrm{B}} = 0} \tag{1.24}$$

  for every solute i,

- for solids, no boundary condition is required if $a_{\mathrm{B}} \geq 0$; if $a_{\mathrm{B}} < 0$, we use

  $$\boxed{\hat{J}_{i,\mathrm{B}} = a_{\mathrm{B}} \cdot c_{i,\mathrm{Y}}} \tag{1.25}$$

  where $c_{i,\mathrm{Y}}$ is the concentration of the solid i in the transition layer (TRANLAY).

1.3.5 Flux speciation at the free water interface (vertex W)

The conservation equations for solutes at the W do not directly provide estimates for the indivual solute fluxes $\hat{J}_{i,W}$, but only for $\hat{J}'_{i,W} = \hat{J}_{i,W} - h_W \hat{r}_{i,W}$ (we have dropped the $n+1$ superscript for improved readability). In order to correct these $\hat{J}'_{i,W}$ for the unknown $\hat{r}_{i,W}$, we start by formally writing the discrete form of the flux expression for a solute i at the interface located in z_W as

$$\hat{J}_{i,W} = -\hat{D}_{i,W} \frac{C_{i,W+\frac{1}{2}} - C_{i,W-\frac{1}{2}}}{h} + \hat{u}_W \left(\frac{1-\kappa_i}{2} C_{i,W+\frac{1}{2}} + \frac{1+\kappa_i}{2} C_{i,W-\frac{1}{2}} \right)$$

or, equivalently

$$\hat{J}_{i,W} = -\left(\hat{D}_{i,W} + \frac{h\hat{u}_W}{2} \kappa_i \right) \frac{C_{i,W+\frac{1}{2}} - C_{i,W-\frac{1}{2}}}{h} + \hat{u}_W \frac{C_{i,W+\frac{1}{2}} + C_{i,W-\frac{1}{2}}}{2}$$

where

- $\hat{J}_{i,W}$ is the total sediment flux of i;
- $\hat{D}_{i,W}$ is the effective diffusion coefficient of i, including all the necessary corrections for tortuosity, etc. if required;
- h is the difference between the "nodes" of two virtual cells centred in $z_{W+\frac{1}{2}}$ and $z_{W+\frac{1}{2}}$ and the virtual point $z_{W-\frac{1}{2}}$;
- $\hat{u}_W = \varphi^{\mathrm{f}}_W \, u_W$, with u_W is the porewater advection rate at z_W;
- κ_i is the upwinding parameter for i;
- $C_{i,W+\frac{1}{2}}$ is the representative concentration of i at $z_{W+\frac{1}{2}} = \frac{1}{2}(z_W + z_{W+1})$;
- $C_{i,W-\frac{1}{2}}$ is the fictive concentration at the virtual point $z_{W-\frac{1}{2}}$.

We chose

$$\frac{C_{i,W+\frac{1}{2}} + C_{i,W-\frac{1}{2}}}{2} = C_{i,W}$$

and define

$$C'_{i,W} = \frac{C_{i,W+\frac{1}{2}} - C_{i,W-\frac{1}{2}}}{h},$$

i.e., the gradient of the $C_i(z)$ at z_W. Accordingly, we rewrite the expression for $\hat{J}_{i,W}$ as

$$\boxed{\hat{J}_{i,W} = -\left(\hat{D}_{i,W} + \frac{h\hat{u}_W}{2}\kappa_i\right) C'_{i,W} + \hat{u}_W C_{i,W}} \tag{1.26}$$

Contrary to the usual κ_i selection, we only use values between -1 and 0: for $u_W < 0$, we use the standard κ_i, which will range between -1 (full upwind, for strong advection) and 0 (central for negligible advection); for $u_W \geq 0$, we use $\kappa_i = 0$ as the concentrations C_i is supposed to be constant above the sediment-water interface and equal to $C_{i,W} = C_{i,\mathrm{Oc}}$. We further chose $h = 2h_W = z_{W+1} - z_W$, so that the two virtual cells have the same widths.

By combining these expressions for $\hat{J}_{i,W}$ with the set of $\hat{J}'_{i,W}$ values, we get a system of equations for the $C'_{i,W}$ (all the $C_{i,W}$ are known from the prescribed solute boundary conditions). Similarly to the generic equations for the solute concentrations, this system contains the unknown $\hat{r}_{i,W}$, which we are going to filter out adopting the assumption that the corresponding chemical reactions are at equilibrium. To do this, we apply a similar transformation of the equation system, by applying exactly the same linear combinations, but, instead of introducing the equilibrium relationships, we introduce their derivative with respect to z, where the derivatives with respect to the individual concentrations are assimilated to the respective $C'_{i,W}$. The resulting system of equations is then solved for the $C'_{i,W}$ and the individual $\hat{J}_{i,W}$ values are finally derived from eq. (1.26).

This procedure ensures that the calculated fluxes are entirely consistent with the global equation system and are in agreement with the overall mass-balance documented by the system.

Chapter 2

Numerical Methods

2.1 Numerical Solution of the Equation System

The numerical schemes adopted in MEDUSA have been selected with the physical meaningfulness of the results in mind. Accordingly, positiveness of the calculated concentration evolutions was deemed indispensable. For the discretisation of the advective part of the local transport term in the equations, one may chose between a first-order full upwind and a second order exponential fitting scheme, known elsewhere as the Allen-Southwell-Il'in or the Scharfetter-Gummel scheme (Hundsdorfer and Verwer, 2003). It is closely related to the scheme of Fiadeiro and Veronis (1977): on regularly spaced grids both schemes lead to identical discrete forms of the equations. The exponential fitting scheme is, however, better suitable for the flux-conservative finite volume approach adopted in MEDUSA, which uses irregularly spaced grids.

Unlike steady-state models, where solid advection rates are always oriented downwards relative to the sediment-water interface, MEDUSA has to be able to cope with solids' advection rates that may have any orientation and that may even change their orientation with time. Both upwinding schemes automatically handle this complication.

2.1.1 Time Stepping and Boundary Conditions

Convergence of numerical DAE solvers is critically dependent on the consistency of the initial conditions given with the algebraic constraints (equilibrium relationships) included in the model, i. e., the initial conditions must fulfil the constraints. Problems related to possible inconsistencies are avoided in MEDUSA by deriving the values of the equilibrium constants directly from the boundary conditions instead of calculating them independently.

2.1.2 Solution strategy

The discretisation of the equation systems leads to a coupled non-linear system of equations that is solved iteratively, using a combination of fixed-point and damped Newton-Raphson iterations. Each iteration proceeds in two stages. First, a fixed-point rule is used to update the advection rate profile $w(z)$ (and, if required, the biodiffusion and bioirrigation coefficients D^{bt}, β and α). For the solids' advection profile, the depth-integrated solid phase volume conservation equation (1.5) is used

$$\varphi^{\mathrm{s}}(z)w(z) - \beta(z)D^{\mathrm{bt}}(z)\frac{\partial\varphi^{\mathrm{s}}}{\partial z} = \sum_{i\in\mathcal{I}^{\mathrm{s}}}\vartheta_i\hat{I}_i^{\mathrm{top}} + \int_{z_T}^{z}\sum_{i\in\mathcal{I}^{\mathrm{s}}}\vartheta_i\hat{R}_i(z')\,dz',$$

where $\hat{I}_i^{\mathrm{top}}$ denotes the deposition rate of solid component i per unit surface of total sediment per unit time, entering the surface sediment through the sediment-water interface at the top. The required reaction rate terms are evaluated by using the most recent available concentration profiles (or the initial state).[1]

[1] Please notice that eqn. (1.5) shows that it is not necessary to distinguish between intra- and interphase biodiffusion in the current version of MEDUSA. The biodiffusive component of the transport term is dependent on the total biodiffusion

In a second stage, the concentration profiles are then updated with a damped Newton-Raphson scheme. For this scheme, at iteration k, we first calculate the standard correction $\Delta\mathbf{C}_k$ as the solution of the linear system

$$\mathbf{J}(\mathbf{C}_k)\Delta\mathbf{C}_k = \mathbf{F}(\mathbf{C}_k),$$

where $\mathbf{F}$ denotes the equation system, $\mathbf{J}$ its Jacobian with respect to the concentrations of the model components, which are assembled in the vector $\mathbf{C}_k$. However, only that fraction $\Delta_{kl} = \Delta\mathbf{C}_k/2^l$ ($l = 0, \ldots, L_{\text{Newton}}$) is applied that minimizes $||\mathbf{F}(\mathbf{C}_k + \Delta_{kl})||$. $L_{\text{Newton}} = 4$ by default but his can be changed. The algorithm uses the analytical Jacobian, except for the derivatives of $w(z)$ with respect to the concentrations of the different model components (eqn. 1.5), which are disregarded. These latter are handled by the fixed point-scheme and similarly for D^{bt} and $\alpha(z)$.

The linear system to solve at each Newton-Raphson iteration has a block tridiagonal structure and is solved with the Gauß algorithm as outlined in Engeln-Müllges and Uhlig (1996).

As a convergence criterion, we use a two-stage trigger:

1. first, we require that
$$||\mathbf{F}(\mathbf{C}_k + \Delta_{kl})||^2 < (10^{-6})^2 n_{\text{tot}}$$
where n_{tot} is the total number of components of $\mathbf{F}$ and $||\cdot||$ denotes the L_2 norm; each component of $\mathbf{F}$ may furthermore be scaled by its scale.

2. if the first criterion is met, we further strive to reach
$$\max(|\Delta_{kl}|/\text{Scl}_{i,:}) < 10^{-9}$$
where $\text{Scl}_{i,:}$ is the scale of component i, replicated for its respective evolution equation at each node.

Once the first criterion is fulfilled, iterations are continued until the second one is fulfilled as well, or until a set maximum number of iterations is reached (120 by default). An iteration sequence is assumed to have successfully converged if the first criterion is met; the second one is considered an asset. More details about the way scaling is taken into account in MEDUSA are provided in chapter 3.

For the initialization of the iterative scheme, a sequence of approaches has been implemented:

1. the state of the previous time step or the initial state is used;
2. selected solute profiles are initially set homogeneously equal to the boundary values;
3. a continuation method where the partial specific volumes of all non-inert solids are gradually increased from zero to their actual values;
4. a continuation method where the top solid fluxes are gradually increased from zero to their actual given values;
5. a continuation method where reaction rates are gradually increased from zero to their standard values;
6. a continuation method only used for steady-state calculations where gradually longer time steps are used and
7. a continuation method only used for columns subject to strong chemical erosion, where the amount of eroded material to return to REACLAY is gradually increased to the calculated value.

These are tried out in turn until the convergence criterion is met (see next chapter). The order can be freely chosen, but the order by which the methods are reported above has proven to be one of the most efficient ones.

coefficient anyway. The total advective component obtained from eqn. (1.5) remains unchanged as $\beta(z)$ is modified; only the partitioning between of $\varphi^{\text{s}}(z)w(z)$ and $\beta(z)D^{\text{bt}}(z)\frac{\partial\varphi^{\text{s}}}{\partial z}$ is affected, but does not have any influence on the concentration profiles.

2.2 Primary and secondary variables

N.B.: This has been prepared to some extent in MEDUSA, but not been implemented. Taking advantage of the difference between primary and secondary variables (and their equations) may contribute to strongly reduce computing times for applications that involve, e.g., isotopes, (i.e. secondary variables that do not impinge on volume conservation, but act as passive tracers only).

Primary variables: the smallest subset of variables that influence the compaction (profile of w, etc.) of the sediment; the distribution of primary variable properties may not be influenced by other variable properties (e.g., secondary variables); Secondary variables: variables whose evolution may depend on the primary ones, but that do not impinge on the w profile, on the distribution of primary properties, etc.

This means that the system of equations that controls the distribution of properties p and s may be written in the form

$$f_p = 0 \tag{2.1}$$
$$f_s = 0 \tag{2.2}$$

where p denotes the subset of primary variables, with f_p the subset of equations that determine their evolution, and s the subset of secondary variables. Following the definition of the primary variables, f_p is such that $f_p = f_p(p)$, while $f_s = f_s(p, s)$. Accordingly,

$$\frac{\partial f_p}{\partial s} \equiv 0 \tag{2.3}$$

and Newton iterations required to solve system above thus write

$$\begin{pmatrix} \left.\frac{\partial f_p}{\partial p}\right|_{p_j} & 0 \\ \left.\frac{\partial f_s}{\partial p}\right|_{p_j,s_j} & \left.\frac{\partial f_s}{\partial s}\right|_{p_j,s_j} \end{pmatrix} \begin{pmatrix} p_{j+1} - p_j \\ s_{j+1} - s_j \end{pmatrix} = \begin{pmatrix} -f_p(p_j) \\ -f_s(p_j, s_j) \end{pmatrix}. \tag{2.4}$$

We split them into two separate problems, that can be solved sequentially:

$$\left(\left.\frac{\partial f_p}{\partial p}\right|_{p_j}\right)(p_{j+1} - p_j) = -f_p(p_j) \tag{2.5}$$

and

$$\left(\left.\frac{\partial f_s}{\partial s}\right|_{p_j,s_j}\right)(s_{j+1} - s_j) = -f_s(p_j, s_j) - \left(\left.\frac{\partial f_s}{\partial p}\right|_{p_j,s_j}\right)(p_{j+1} - p_j) \tag{2.6}$$

Chapter 3

Scaling of the equations

In MEDUSA, we have not explicitly scaled the equations. The linear solver used in the Newton-Raphson iterations, does some internal scaling. We do nevertheless take scaling into account in the convergence criterion, if scales are provided. The rationale behind this is outlined in the following section

3.1 Introduction

Following Boudreau (1986), the general diagenesis equation for a solid

$$\frac{\partial \hat{C}}{\partial t} + \frac{\partial}{\partial z}\left(-D_{\mathrm{B}}(z)\frac{\partial \hat{C}}{\partial z} + w\hat{C}\right) - R(\hat{C}) = 0 \tag{3.1}$$

can be transformed into an non-dimensional form by using scaling transformations, based upon characteristic scales. For the length scale, the mixed-layer depth, L, is an obvious choice and for concentrations, a typical value representative of the top of the sediment, where concentrations are often largest, can be chosen. For time, several characteristic scales can e chosen. A first option is the diffusive time scale, which, for a length L is $\theta_{\mathrm{D}} = L^2/D_{\mathrm{B0}}$

$$\begin{aligned}\xi &= z/L\\ \tau &= t/\theta_{\mathrm{D}} = tD_{\mathrm{B0}}/L^2\\ \Gamma &= \hat{C}/\hat{C}_0\end{aligned}$$

With $R(\hat{C}) = -\lambda\hat{C}$, we get

$$\frac{\partial \Gamma}{\partial \tau} + \frac{\partial}{\partial \xi}\left(-f(\xi)\frac{\partial \Gamma}{\partial \xi} + \mathrm{Pe}\,\Gamma\right) + \mathrm{Da(I)}\,\Gamma = 0 \tag{3.2}$$

where

$$\begin{aligned}f(\xi) &= D_{\mathrm{B}}(z)/D_{\mathrm{B0}}\\ \mathrm{Pe} &= wL/D_{\mathrm{B0}}\\ \mathrm{Da(I)} &= \lambda L^2/D_{\mathrm{B0}}\end{aligned}$$

Alternatively, the advective time scale can be chosen instead of the diffusive one. For length a scale L, the advective time scale is $\theta_{\mathrm{w}} = L/w$. Hence, the transformation

$$\begin{aligned}\xi &= z/L\\ \tau &= t/\theta_{\mathrm{w}} = L/w\\ \Gamma &= \hat{C}/\hat{C}_0\end{aligned}$$

The diagenesis equation then transforms to

$$\frac{\partial \Gamma}{\partial \tau} + \frac{\partial}{\partial \xi}\left(-\frac{f(\xi)}{\mathrm{Pe}}\frac{\partial \Gamma}{\partial \xi} + \Gamma\right) + \mathrm{Da(II)}\,\Gamma = 0 \tag{3.3}$$

where

$$\begin{aligned} f(\xi) &= D_B(z)/D_{B0} \\ \mathrm{Pe} &= wL/D_{B0} \\ \mathrm{Da(II)} &= \lambda L/w \end{aligned}$$

3.2 Application

3.2.1 Basics

We reconsider the general diagenesis equation

$$\mathrm{Eq(C)} \equiv \frac{\partial \hat{C}}{\partial t} + \frac{\partial}{\partial z}\left(-D^{\mathrm{ter}}\frac{\partial \hat{C}}{\partial z} - D^{\mathrm{tra}}\varphi^{\mathrm{x}}\frac{\partial C}{\partial z} + w\hat{C}\right) - \hat{R} = 0$$

where the diffusion term has been split to be able to consider both inter- and intraphase diffusive processes.

Let us consider a general scaling transformation

$$\begin{aligned} \xi &= z/L \\ \tau &= t/T \\ \Gamma &= C/C_0 \end{aligned}$$

The total sediment concentration $\hat{C}$ and the phase-specific concentration C for any given component are related by $\hat{C} = \varphi^{\mathrm{x}}C$, where φ^{x} denotes the volume fraction of the phase 'x' (where we adopt $\varphi^{\mathrm{f}} = \varphi$, with the superscript 'f' standing for *fluid* and $\varphi^{\mathrm{s}} = 1-\varphi$, with the superscript 's' standing for the *solid* phase). We further more adopt scales φ_0^{x} such that we may also use $\hat{C}_0 = \varphi_0^{\mathrm{x}}C_0$. In addition, we adopt $\Phi^{\mathrm{x}} = \varphi^{\mathrm{x}}/\varphi_0^{\mathrm{x}}$ which allows us to define

$$\hat{\Gamma} = \hat{C}/\hat{C}_0 = \left(\varphi^{\mathrm{x}}/\varphi_0^{\mathrm{x}}\right)\left(C/C_0\right) = \Phi^{\mathrm{x}}\Gamma.$$

The equation above may then be transformed as follows:

$$\begin{aligned} &\frac{\partial \hat{C}}{\partial t} + \frac{\partial}{\partial z}\left(-D^{\mathrm{inter}}\frac{\partial \hat{C}}{\partial z} - D^{\mathrm{intra}}\varphi^{\mathrm{x}}\frac{\partial C}{\partial z} + w\hat{C}\right) - \hat{R} \\ &= \frac{d\tau}{dt}\frac{\partial}{\partial \tau}(\hat{\Gamma}\hat{C}_0) \\ &\quad + \frac{d\xi}{dz}\frac{\partial}{\partial \xi}\left(-D^{\mathrm{inter}}\frac{d\xi}{dz}\frac{\partial}{\partial \xi}(\hat{\Gamma}\hat{C}_0) - D^{\mathrm{intra}}\varphi^{\mathrm{x}}\frac{d\xi}{dz}\frac{\partial}{\partial \xi}(\Gamma C_0) + w\hat{\Gamma}\hat{C}_0\right) - \hat{R} \\ &= \frac{\hat{C}_0}{T}\frac{\partial \hat{\Gamma}}{\partial \tau} + \frac{1}{L}\frac{\partial}{\partial \xi}\left(-D^{\mathrm{inter}}\frac{\hat{C}_0}{L}\frac{\partial \hat{\Gamma}}{\partial \xi} - D^{\mathrm{intra}}\Phi^{\mathrm{x}}\varphi_0^{\mathrm{x}}\frac{C_0}{L}\frac{\partial \Gamma}{\partial \xi} + w\hat{\Gamma}\hat{C}_0\right) - \hat{R} \\ &= \frac{\hat{C}_0}{T}\frac{\partial \hat{\Gamma}}{\partial \tau} + \frac{\hat{C}_0}{T}\frac{\partial}{\partial \xi}\left(-D^{\mathrm{inter}}\frac{T}{L^2}\frac{\partial \hat{\Gamma}}{\partial \xi} - D^{\mathrm{intra}}\frac{T}{L^2}\Phi^{\mathrm{x}}\frac{\partial \Gamma}{\partial \xi} + w\frac{T}{L}\hat{\Gamma}\right) - \hat{R} \\ &= \frac{\hat{C}_0}{T}\left(\frac{\partial \hat{\Gamma}}{\partial \tau} + \frac{\partial}{\partial \xi}\left(-\frac{D^{\mathrm{inter}}}{L^2/T}\frac{\partial \hat{\Gamma}}{\partial \xi} - \frac{D^{\mathrm{intra}}}{L^2/T}\Phi^{\mathrm{x}}\frac{\partial \Gamma}{\partial \xi} + \frac{w}{L/T}\hat{\Gamma}\right) - \frac{\hat{R}}{\hat{C}_0/T}\right) \end{aligned}$$

Hence, if we further define

$$\Delta^{\mathrm{inter}} = \frac{D^{\mathrm{inter}}}{L^2/T} = \frac{D^{\mathrm{inter}}}{D_0}\,\frac{D_0}{L^2/T} \quad \text{and} \quad \Delta^{\mathrm{intra}} = \frac{D^{\mathrm{intra}}}{L^2/T} = \frac{D^{\mathrm{intra}}}{D_0}\,\frac{D_0}{L^2/T}$$

as well as

$$\omega = \frac{w}{L/T} = \frac{w}{w_0}\frac{w_0}{L/T} \quad \text{and} \quad \hat{P} = \frac{\hat{R}}{\hat{C}_0/T},$$

then

$$\mathrm{Eq}(C) = \frac{\hat{C}_0}{T}\mathrm{Eq}(\Gamma)$$

where

$$\mathrm{Eq}(\Gamma) \equiv \frac{\partial \hat{\Gamma}}{\partial \tau} + \frac{\partial}{\partial \hat{\xi}}\left(-\Delta^{\mathrm{inter}}\frac{\partial \hat{\Gamma}}{\partial \hat{\xi}} - \Delta^{\mathrm{intra}}\Phi^{\mathrm{x}}\frac{\partial \Gamma}{\partial \hat{\xi}} + \omega\hat{\Gamma}\right) - \hat{P} \tag{3.4}$$

According to this scaling, we then have:

$$|\mathrm{Eq}(\Gamma)| < \epsilon \quad \Leftrightarrow \quad |\mathrm{Eq}(\hat{C})| < \epsilon\frac{\hat{C}_0}{T}.$$

If $\hat{R} = -\lambda\hat{C}$, then $\hat{P} = -\lambda T\,\hat{\Gamma}$.

3.2.2 Scaling based up the diffusion time scale

This transformation is analogous to the first one in the previous section, with

$$T = L^2/D_0$$

Hence, if we get

$$\Delta^{\mathrm{inter}} = \frac{D^{\mathrm{inter}}}{D_0} \quad \text{and} \quad \Delta^{\mathrm{intra}} = \frac{D^{\mathrm{intra}}}{D_0},$$

as well as

$$\omega = \frac{w}{D_0/L} = \frac{w}{w_0}\frac{Lw_0}{D_0} = \frac{w}{w_0}\mathrm{Pe} \quad \text{and} \quad \hat{P} = \frac{\hat{R}}{\hat{C}_0 D_0/L^2},$$

then

$$\mathrm{Eq}(C) = \frac{\hat{C}_0 D_0}{L^2}\mathrm{Eq}(\Gamma)$$

where $\mathrm{Eq}(\Gamma)$ follows the same expression as eqn. (3.4) above. According to this scaling, we then have:

$$|\mathrm{Eq}(\Gamma)| < \epsilon \quad \Leftrightarrow \quad |\mathrm{Eq}(\hat{C})| < \epsilon\frac{\hat{C}_0 D_0}{L^2}.$$

This is the scaling procedure currently adopted in MEDUSA.

3.2.3 Scaling based up the reaction time scale

For this scaling, we adopt $T = 1/\lambda$. This time, we further define

$$\Delta^{\mathrm{inter}} = \frac{D^{\mathrm{inter}}}{\lambda L^2} = \frac{D^{\mathrm{inter}}}{D_0}\frac{D_0}{\lambda L^2} = \frac{D^{\mathrm{inter}}}{D_0}\frac{1}{\mathrm{Da(I)}},$$

$$\Delta^{\mathrm{intra}} = \frac{D^{\mathrm{intra}}}{\lambda L^2} = \frac{D^{\mathrm{intra}}}{D_0}\frac{D_0}{\lambda L^2} = \frac{D^{\mathrm{intra}}}{D_0}\frac{1}{\mathrm{Da(I)}},$$

as well as

$$\omega = \frac{w}{\lambda L} = \frac{w}{w_0}\frac{w_0}{\lambda L} \quad \text{and} \quad \hat{P} = \frac{\hat{R}}{\lambda\hat{C}_0},$$

then

$$\mathrm{Eq}(C) = \lambda\hat{C}_0\mathrm{Eq}(\Gamma)$$

where $\mathrm{Eq}(\Gamma)$ is again the same as eqn. (3.4) above.

According to this scaling, we then have:

$$|\mathrm{Eq}(\Gamma)| < \epsilon \quad \Leftrightarrow \quad |\mathrm{Eq}(\hat{C})| < \epsilon\frac{1}{\lambda\hat{C}_0}$$

3.3 Equation scaling for laws of mass-action

Starting from a general form of the equilibrium equation

$$k \prod_{i \in \mathcal{R}} C_i - \prod_{i \in \mathcal{P}} C_i = 0$$

where $\mathcal{R}$ denotes the set of reagents (left-hand side of the equilibrium) and $\mathcal{P}$ the set of products (right-hand side of the equilibrium).

These equations are scaled by a factor

$$\sqrt{k \prod_{i \in \mathcal{R}} C_{i,0}} \times \sqrt{\prod_{i \in \mathcal{P}} C_{i,0}}$$

where $C_{i,0}$ denotes the scale of the reagent i.

Chapter 4

Grid Generation

4.1 Introduction

In order to solve the partial differential equations that describe early diagenetic processes in MEDUSA, the domain of interest which we denote for simplicity by $[0, L]$, is overlaid with a grid or mesh of points denoted z_i $(i = 1, \ldots, n)$, called *nodes*, such that $0 \leq z_1 < \ldots < z_i < \ldots < z_n \leq L$. Each node is representative of a small sub-interval of $[0, L]$, delimited by the mid-points between neigbouring nodes. These mid-points are called *vertices* and represent thus the boundaries between the sub-intervals called *finite volumes* or cells. Concentrations and reaction terms are evaluated at the nodes while the flux terms are evaluated at the vertices. Accordingly, in some instances it is necessary to have a vertex located at 0 or L (e.g., if boundary conditions involve fluxes only). In such cases, we call upon virtual grid points $z_0 < 0$ or $z_{n+1} > L$ outside the domain of interest to define the required vertices, such that $[0, L]$ always includes n nodes z_i. In other instances it is necessary or recommended to have nodes locates at 0 or L (e.g., if boundary conditions involve prescibed concentrations). In this case there are only half cells at 0 or L. Such nodes are thus virtual vertices (cell boundaries).

With a few exceptions, the strategy behind grid generation consists in choosing a function Q to remap a regular grid covering the interval $[0, 1]$ $(\xi_i = i/N, i = 0, \ldots, N)$ onto an irregular grid $q_i = Q(\xi_i)$ $(i = 0, \ldots, N)$ covering the same interval [0,1]. Accordingly, we always have $q_0 = 0$ and $q_N = 1$. This q_i $(i = 0, \ldots, N)$ grid is then scaled and shifted (moved by translation) to fulfil constraints set by the specific problem requirements:

- the extent L of the interval $[0, L]$;
- a vertex or a node is located at the starting point 0;
- a vertex or a node is located at the end point L;
- the number of nodes n to have on the grid.

The final grid z_i $(i = 1, \ldots, n)$, is then derived by

$$z_i = S q_{i-m} + z_{\text{td}},$$

where

- m denotes an index offset (possibly 0) to shift the transformed q_i grid if required;
- S is a scaling factor;
- z_{td} is the translation distance.

The finally generated grid only includes nodes. Vertices are, by definition, located mid-may between nodes, or at either end of the gridded domain.

In order to preserve second order truncation error that can be easily achieved on regular meshes, it is sufficient to use a mapping $Q : \xi_i \rightarrow q_i$ that is twice continuously differentiable on $[0, 1]$.

4.1.1 Node-to-node grids

For node-to-node grids, $z_1 = 0$ and $z_n = L$. Hence,

- q_0 must be remapped onto z_1, requiring that $m = 1$;
- q_N must be remapped onto z_n, requiring that $N + m = n$, i.e., $N = n - 1$
- $z_1 = 0$ and $z_1 = Sq_0 + z_{\text{td}} = z_{\text{td}}$ require that $z_{\text{td}} = 0$
- $z_n = L$ and $z_n = Sq_{n-1} + z_{\text{td}} = S$ require that $S = L$

4.1.2 Node-to-vertex grids

For node-to-vertex grids, $z_1 = 0$. z_n is inside the domain to be gridded and chosen so that with a virtual next point z_{n+1}, mapped from q_N would provide a vertex between z_n and z_{n+1} such that $\frac{1}{2}(z_n + z_{n+1}) = L$. Accordingly:

- q_0 must be remapped onto z_1, requiring that $m = 1$;
- q_N must be remapped onto z_{n+1}, requiring that $N + m = n + 1$, i.e., $N = n$;
- $z_1 = 0$ and $z_1 = Sq_0 + z_{\text{td}} = z_{\text{td}}$ require that $z_{\text{td}} = 0$;
- noting that

$$\begin{aligned} z_n + z_{n+1} &= Sq_{n-1} + Sq_n + 2z_{\text{td}} \\ &= Sq_{N-1} + Sq_N \\ &= SQ(\xi_{N-1}) + S \\ &= S(Q(\xi_{N-1}) + 1) \end{aligned}$$

  the vertex condition $\frac{1}{2}(z_n + z_{n+1}) = L$ finally leads to

$$S = L\frac{2}{Q(\xi_{N-1}) + 1} = L\frac{2}{Q(\frac{n-1}{n}) + 1}.$$

4.1.3 Vertex-to-node grids

For vertex-to-node grids, $z_n = L$. z_1 is inside the domain to be gridded and chosen so that with a virtual preceding point z_0, mapped from q_0 would provide a vertex between z_0 and z_1 such that $\frac{1}{2}(z_0 + z_1) = 0$. Accordingly:

- q_1 must be remapped onto z_1, requiring that $m = 0$;
- q_N must be remapped onto z_n, requiring that $N + m = n$, i.e., $N = n$;
- $z_n = L$ and $z_n = Sq_N + z_{\text{td}} = S + z_{\text{td}}$ require that $z_{\text{td}} = L - S$;
- noting that

$$\begin{aligned} z_0 + z_1 &= Sq_0 + Sq_1 + 2z_{\text{td}} \\ &= Sq_1 + 2(L - S) \\ &= SQ(\xi_1) + 2(L - S) \\ &= S(Q(\xi_1) - 2) + 2L \end{aligned}$$

  the vertex condition $\frac{1}{2}(z_0 + z_1) = 0$ finally then leads to

$$S = L\frac{2}{2 - Q(\xi_1)} = L\frac{2}{2 - Q(\frac{1}{n})}.$$

  Hence,

$$z_{\text{td}} = L - S = L\left(1 - \frac{2}{2 - Q(\frac{1}{n})}\right) = -L\frac{Q(\frac{1}{n})}{2 - Q(\frac{1}{n})}$$

4.1.4 Vertex-to-vertex grids

For vertex-to-vertex grids grids, z_1 is inside the domain to be gridded and chosen so that with a virtual preceding point z_0, mapped from q_0 would provide a vertex between z_0 and z_1 such that $\frac{1}{2}(z_0 + z_1) = 0$; z_n is also inside the domain to be gridded and chosen so that with a virtual next point z_{n+1}, mapped from q_N would provide a vertex between z_n and z_{n+1} such that $\frac{1}{2}(z_n + z_{n+1}) = L$. Accordingly:

- q_1 must be remapped onto z_1, requiring that $m = 0$;
- q_{N-1} must be remapped onto z_n, requiring that $N - 1 + m = n$, i.e., $N = n + 1$;
- noting, as above, that
$$z_0 + z_1 = SQ(\xi_1) + 2z_{\text{td}}$$
and that
$$\begin{aligned} z_n + z_{n+1} &= Sq_{n-1} + Sq_n + 2z_{\text{td}} \\ &= Sq_{N-1} + Sq_N + 2z_{\text{td}} \\ &= S(Q(\xi_{N-1}) + 1) + 2z_{\text{td}} \end{aligned}$$
the vertex conditions $\frac{1}{2}(z_0 + z_1) = 0$ and $\frac{1}{2}(z_n + z_{n+1}) = L$ require that S and z_{td} obey to a linear system
$$\left\{ \begin{array}{rcrcl} Q(\xi_1)S & + & 2z_{\text{td}} & = & 0 \\ (Q(\xi_{N-1}) + 1)S & + & 2z_{\text{td}} & = & 2L \end{array} \right.$$
i.e.,
$$\left\{ \begin{array}{rcrcl} Q(\frac{1}{n+1})S & + & 2z_{\text{td}} & = & 0 \\ (Q(\frac{n}{n+1}) + 1)S & + & 2z_{\text{td}} & = & 2L \end{array} \right.$$
Hence,
$$\begin{aligned} S &= L\frac{2}{Q(\frac{n}{n+1}) - Q(\frac{1}{n+1}) + 1} \\ z_{\text{td}} &= -L\frac{Q(\frac{1}{n+1})}{Q(\frac{n}{n+1}) - Q(\frac{1}{n+1}) + 1} \end{aligned}$$

4.2 Linear grids

For linear grids, the fundamental remapping function is simply
$$Q(\xi_i) = \xi.$$

4.3 Quadratic-linear grids

For quadratic-linear grids (Boudreau, 1997, eqn. 8.156, p. 333), the fundamental remapping function is
$$Q(\xi_i) = \frac{(\xi_i^2 + \xi_c^2)^{\frac{1}{2}} - \xi_c}{(1 + \xi_c^2)^{\frac{1}{2}} - \xi_c}.$$
Here, $0 \leq \xi_c \leq 1$ is a parameter that sets the fraction of the domain where the spacing is quadratic ($\xi \ll \xi_c$) and where it tends to become linear ($\xi \gg \xi_c$).

4.4 Power-linear grids

Power-linear grids are a generalization of the quadratic-linear grids from the previous section. The fundamental remapping function is
$$Q(\xi_i) = \frac{(\xi_i^p + \xi_c^p)^{\frac{1}{p}} - \xi_c}{(1 + \xi_c^p)^{\frac{1}{p}} - \xi_c}.$$

4.5 Geometric progression grids

For geometric progression grids (used by Soetaert et al. (1996)) in general, there are several parameters of importance, besides the number of grid points, which we assume fixed a priori:

- the thickness of the first interval, δ;
- the geometric progression ratio, r;
- the extent of the interval to be gridded, L.

The three parameters are not independent of each other. However, the relationships are different from one grid type to another.

To take advantage of the developments presented in the introduction, let us start to derive the remapping $\xi_i \to q_i$. For any initial scale factor σ and progression ratio r, we have

$$\begin{aligned} q_0 &= 0 \\ q_1 &= \sigma \\ q_2 &= q_1 + \sigma r = \sigma(1+r) \\ q_3 &= q_2 + \sigma r^2 = \sigma(1+r+r^2) \\ &\vdots \\ q_i &= \sigma(1+r+r^2+\ldots+r^{i-1}) \\ &\vdots \\ q_N &= \sigma(1+r+r^2+\ldots+r^{N-1}) \end{aligned}$$

Notice that $q_i = \sigma(1+r+r^2+\ldots+r^{i-1}) = \sigma(1-r^i)/(1-r)$. Hence, if $q_N = 1$ then $\sigma = \frac{1-r}{1-r^N}$, leading to the remapping function

$$Q(\xi_i) = \frac{1-r^i}{1-r^N} = \frac{r^i-1}{r^N-1} = \frac{r^{N\xi_i}-1}{r^N-1}$$

4.5.1 Grid types

Node-to-node

Characteristics: $m = 1$, $N = n-1$, $S = L$, $z_{\text{td}} = 0$. Hence,

$$z_i = L \cdot Q(\xi_{i-1}) = L\frac{r^{i-1}-1}{r^{n-1}-1}, \quad i = 1,\ldots,n$$

Since

$$z_2 = \delta = L\frac{r-1}{r^{n-1}-1}$$

we can rewrite this as

$$z_i = \delta\frac{r^{i-1}-1}{r-1}, \quad i = 1,\ldots,n$$

and we furthermore have

$$L = \delta\frac{r^{n-1}-1}{r-1}$$

Node-to-vertex

Characteristics: $m = 1$, $N = n$, $S = L\frac{2}{Q(\xi_{N-1})+1}$, $z_{\text{td}} = 0$.

$$z_i = L\frac{2}{Q(\xi_{N-1})+1} \cdot Q(\xi_{i-1}) = L\frac{2}{\frac{r^{n-1}-1}{r^n-1}+1}\frac{r^{i-1}-1}{r^n-1}, \quad i = 1,\ldots,n$$

and again

$$z_2 = \delta = L\frac{2}{\frac{r^{n-1}-1}{r^n-1}+1}\frac{r-1}{r^n-1} = L\frac{2(r-1)}{r^n-1+r^{n-1}-1}.$$

We can rewrite this as

$$z_i = \delta\frac{r^{i-1}-1}{r-1}, \quad i = 1,\ldots,n$$

and we furthermore have

$$L = \delta\frac{r^n-1+r^{n-1}-1}{2(r-1)}$$

Vertex-to-node

Characteristics: $m = 0$, $N = n$, $S = L\frac{2}{2-Q(\frac{1}{n})}$, $z_{\text{td}} == -L\frac{Q(\frac{1}{n})}{2-Q(\frac{1}{n})}$.

$$\begin{aligned} z_i &= L\frac{2}{2-Q(\frac{1}{n})}\cdot Q(\xi_i) - L\frac{Q(\frac{1}{n})}{2-Q(\frac{1}{n})} \\ &= L\frac{1}{2-Q(\frac{1}{n})}\cdot(2Q(\xi_i) - Q(\tfrac{1}{n})) \\ &= L\frac{1}{2-\frac{r-1}{r^n-1}}\cdot(2\frac{r^i-1}{r^n-1} - \frac{r-1}{r^n-1}) \\ &= L\frac{2r^i-r-1}{2r^n-r-1} \end{aligned}$$

In this case,

$$\delta = z_1 - z_0 = 2L\frac{r-1}{2r^n-r-1}$$

and we can rewrite the sequence as

$$z_i = \frac{\delta}{2}\frac{2r^i-r-1}{r-1}$$

and we furthermore have

$$L = \delta\frac{2r^n-r-1}{2(r-1)}$$

Vertex-to-vertex

Characteristics: $m = 0$, $N = n+1$, $S = 2L/(Q(\xi_{N-1}) - Q(\xi_1) + 1)$, $z_{\text{td}} = -LQ(\xi_1)/(Q(\xi_{N-1}) - Q(\xi_1) + 1)$

$$\begin{aligned} z_i &= L\frac{2}{Q(\xi_{N-1}) - Q(\xi_1) + 1}\cdot Q(\xi_i) - L\frac{Q(\xi_1)}{Q(\xi_{N-1}) - Q(\xi_1) + 1} \\ &= L\frac{2Q(\xi_i) - Q(\xi_1)}{Q(\xi_{N-1}) - Q(\xi_1) + 1} \\ &= L\frac{2\frac{r^i-1}{r^{n+1}-1} - \frac{r-1}{r^{n+1}-1}}{\frac{r^n-1}{r^{n+1}-1} - \frac{r-1}{r^{n+1}-1} + 1} \\ &= L\frac{2r^i-r-1}{r^{n+1}-r+r^n-1} \\ &= L\frac{2r^i-r-1}{(r+1)(r^n-1)} \end{aligned}$$

In this case,

$$\delta = z_1 - z_0 = 2L\frac{r-1}{(r+1)(r^n-1)}$$

and we can rewrite the sequence as

$$z_i = \frac{\delta}{2}\frac{2r^i - r - 1}{r - 1}$$

and we have

$$L = \delta\frac{(r+1)(r^n - 1)}{2(r-1)}.$$

Furthermore,

$$z_{n+1} + z_n = L\frac{2r^{n+1} - 2r + 2r^n - 2}{(r+1)(r^n - 1)} = 2L\frac{(r+1)(r^n - 1)}{(r+1)(r^n - 1)} = 2L$$

as expected.

4.5.2 Derived quantities

As mentioned above, the three parameters of interest, δ, r and L are interdependent. For each of the four grid-types, we have shown that the three parameters are related by one relationship. Accordingly, one of the three parameters can be derived from the two others. There are two cases that are straightforward to handle:

- for given r and δ, L can be directly derived;
- for given r and L, δ can be directly derived;

The third case, where δ and L are given, is more complicated to handle as it involves a non-linear equation to solve for r. The equation to solve depends on the grid type adopted.

Node-to-node

The equation to solve is

$$f_{\mathrm{nn}}(r) \equiv \frac{r^{n-1} - 1}{r - 1} - \frac{L}{\delta} = 0$$

$r = 1$ appears to be a critical value: for $r = 1$, the equation function evaluates to

$$f_{\mathrm{nn}}(1) = (n - 1) - \frac{L}{\delta}$$

Node-to-vertex

The equation to solve is

$$f_{\mathrm{nv}}(r) \equiv \frac{r^n - 1 + r^{n-1} - 1}{2(r-1)} - \frac{L}{\delta} = 0$$

For $r = 1$, the function evaluates to

$$f_{\mathrm{nv}}(1) = \frac{2n - 1}{2} - \frac{L}{\delta}$$

Vertex-to-node

The equation to solve is

$$f_{\mathrm{vn}}(r) \equiv \frac{2r^n - r - 1}{2(r-1)} - \frac{L}{\delta} = 0.$$

For $r = 1$, the function evaluates to

$$f_{\mathrm{vn}}(1) = \frac{2n - 1}{2} - \frac{L}{\delta}$$

Vertex-to-vertex

The equation to solve is

$$f_{\mathrm{vv}}(r) \equiv \frac{(r+1)(r^n-1)}{2(r-1)} - \frac{L}{\delta} = 0.$$

For $r = 1$, the function evaluates to

$$f_{\mathrm{vv}}(1) = n - \frac{L}{\delta}$$

4.5.3 Solving for r

Preliminaries

For $r > 1$, $r^2 = r \cdot r > r, \ldots, r^i = r \cdot r^{i-1} > r$ and thus

$$\begin{aligned} \frac{r^p - 1}{r-1} &= r^{p-1} + \ldots + r + 1 \\ &> r + \ldots + r + 1 \\ &> (p-1)r + 1 \\ &> (p-1)r \end{aligned}$$

Accordingly, for $r = a/(p-1)$, where $a > (p-1)$,

$$\frac{r^p - 1}{r-1} - a > (p-1) \cdot r - a = (p-1) \cdot a/(p-1) - a = 0.$$

For $r > 1$, and n sufficiently large, we thus have for the different equation functions:

$$f_{\mathrm{nn}}(r) = \frac{r^{n-1} - 1}{r-1} - \frac{L}{\delta} > (n-2)r - \frac{L}{\delta}$$

$$f_{\mathrm{nv}}(r) = \frac{r^n - 1 + r^{n-1} - 1}{2(r-1)} - \frac{L}{\delta} > (\frac{n-1}{2} + \frac{n-2}{2})r - \frac{L}{\delta} = \frac{2n-3}{2}r - \frac{L}{\delta}$$

$$f_{\mathrm{vn}}(r) = \frac{2r^n - r - 1}{2(r-1)} - \frac{L}{\delta} = \frac{r^n - 1 + r(r^{n-1} - 1)}{2(r-1)} - \frac{L}{\delta} > \frac{2n-3}{2}r - \frac{L}{\delta}$$

$$f_{\mathrm{vv}}(r) = \frac{(r+1)(r^n-1)}{2(r-1)} - \frac{L}{\delta} > \frac{r^n - 1}{r-1} - \frac{L}{\delta} > (n-1)r - \frac{L}{\delta}$$

This inequalities can be used to derive bounds, so that a Newton method safe-guarded by a bi-section method can be used.

Solution algorithm

Simple cases:

- $n = 2$
  - $f_{\mathrm{nn}}(r) = \frac{r-1}{r-1} - \frac{L}{\delta} = 1 - \frac{L}{\delta}$. No condition on r, but it is required that $L = \delta$
  - $f_{\mathrm{nv}}(r) = \frac{r^2+r-2}{2(r-1)} - \frac{L}{\delta} = \frac{1}{2}(r+2) - \frac{L}{\delta}$ and thus $r = 2(\frac{L}{\delta} - 1)$. Since $r > 0$, this is only possible if $L > \delta$.
  - $f_{\mathrm{vn}}(r) = \frac{2r^2-r-1}{2(r-1)} - \frac{L}{\delta} = \frac{1}{2}(r+\frac{1}{2}) - \frac{L}{\delta}$ and thus $r = 2\frac{L}{\delta} - \frac{1}{2}$, requiring that $L > 4\delta$.
  - $f_{\mathrm{vv}}(r) = \frac{(r+1)(r^2-1)}{2(r-1)} - \frac{L}{\delta} = \frac{1}{2}(r+1)^2 - \frac{L}{\delta}$ and thus $r = \sqrt{2\frac{L}{\delta}} - 1$ (the negative square root leads to negative r), requiring that $L > 2\delta$.
- $n = 3$
  - $f_{\mathrm{nn}}(r) = \frac{r^2-1}{r-1} - \frac{L}{\delta} = (r+1) - \frac{L}{\delta}$ and thus $r = \frac{L}{\delta} - 1$, requiring that $L > \delta$.

- $f_{\mathrm{nv}}(r) = \frac{r^3-1+r^2-1}{2(r-1)} - \frac{L}{\delta} = \frac{1}{2}(r^2+2r+2) - \frac{L}{\delta}$. The equation to solve is then $r^2+2r+2(1-\frac{L}{\delta})$. This equation has a real solution only if $4-8\cdot(1-\frac{L}{\delta}) \geq 0$, i.e., if $1-\frac{L}{\delta} \leq \frac{1}{2}$, i.e., if $L \geq \frac{1}{2}\delta$. In this case, $r = -1+\sqrt{2\frac{L}{\delta}-1}$. That root r is only positive if $L > \delta$.
- $f_{\mathrm{vn}}(r) = \frac{2r^3-r-1}{2(r-1)} - \frac{L}{\delta} = (r^2+r+\frac{1}{2}) - \frac{L}{\delta}$. The equation to solve is then $r^2+r+(\frac{1}{2}-\frac{L}{\delta})$. This equation has only a solution if $1-4\cdot(\frac{1}{2}-\frac{L}{\delta}) \geq 0$, i.e., if $\frac{1}{2}-\frac{L}{\delta} \leq \frac{1}{4}$, i.e., if $L \geq \frac{1}{4}\delta$. In this case, $r = -\frac{1}{2}+\frac{1}{2}\sqrt{4\frac{L}{\delta}-1}$. That root r is only positive if $L > \frac{1}{2}\delta$.
- $f_{\mathrm{vv}}(r) = \frac{(r+1)(r^3-1)}{2(r-1)} - \frac{L}{\delta} = \frac{1}{2}(r+1)(r^2+r+1) - \frac{L}{\delta}$. The equation to solve is then $(r+1)(r^2+r+1) - 2\frac{L}{\delta} = 0$. The first term is a monotonously increasing polynomial. The equation has thus only one real solution; the other two must be complex conjugate. The product of the three solutions is equal to $-(1-2\frac{L}{\delta})$. The real solution can therefore only be positive if $L > \frac{1}{2}\delta$.
  $f_{\mathrm{vv}}(0) = \frac{1}{2} - \frac{L}{\delta}$: always negative
  $f_{\mathrm{vv}}(1) = 3 - \frac{L}{\delta}$: positive or zero if $\frac{L}{\delta} \leq 3$
  $f_{\mathrm{vv}}(\frac{L}{\delta}/2) > 0$ if $\frac{L}{\delta} > 2$

General case ($n > 3$)

- $f_{\mathrm{nn}}(r) = \frac{r^{n-1}-1}{r-1} - \frac{L}{\delta} = GP_{n-1}(r) - \frac{L}{\delta}$
  $f_{\mathrm{nn}}(0) = 1 - \frac{L}{\delta}$: $f_{\mathrm{nn}}(0) \leq 0$ if $\frac{L}{\delta} \geq 1$
  $f_{\mathrm{nn}}(1) = n-1-\frac{L}{\delta}$: $f_{\mathrm{nn}}(1) > 0 \Leftrightarrow \frac{L}{\delta} < n-1$
  $f_{\mathrm{nn}}(\frac{L}{\delta}/(n-2))) > 0$ if $\frac{L}{\delta} > n-2$
- $f_{\mathrm{nv}}(r) = \frac{r^n-1+r^{n-1}-1}{2(r-1)} - \frac{L}{\delta} = \frac{1}{2}(GP_n(r)+GP_{n-1}(r)) - \frac{L}{\delta} = \frac{1}{2}((r+1)GP_{n-1}(r)+1) - \frac{L}{\delta}$
  $f_{\mathrm{nv}}(0) = 1 - \frac{L}{\delta}$: $f_{\mathrm{nv}}(0) \leq 0$ if $\frac{L}{\delta} \geq 1$
  $f_{\mathrm{nv}}(1) = n-\frac{1}{2}-\frac{L}{\delta}$: $f_{\mathrm{nv}}(1) > 0 \Leftrightarrow \frac{L}{\delta} < n-\frac{1}{2}$
  $f_{\mathrm{n}}(\frac{L}{\delta}/(n-\frac{3}{2})) > 0$ if $\frac{L}{\delta} > n-\frac{3}{2}$
- $f_{\mathrm{vn}}(r) = \frac{2r^n-r-1}{2(r-1)} - \frac{L}{\delta} = \frac{1}{2}(GP_n(r)+rGP_{n-1}(r)) - \frac{L}{\delta} = rGP_{n-1}(r) + \frac{1}{2} - \frac{L}{\delta}$
  $f_{\mathrm{vn}}(0) = \frac{1}{2} - \frac{L}{\delta}$: $f_{\mathrm{nv}}(0) \leq 0$ if $\frac{L}{\delta} \geq \frac{1}{2}$
  $f_{\mathrm{vn}}(1) = n-\frac{1}{2}-\frac{L}{\delta}$: $f_{\mathrm{nv}}(1) > 0 \Leftrightarrow \frac{L}{\delta} < n-\frac{1}{2}$
  $f_{\mathrm{vn}}(\frac{L}{\delta}/(n-\frac{3}{2})) > 0$ if $\frac{L}{\delta} > n-\frac{3}{2}$
- $f_{\mathrm{vv}}(r) = \frac{(r+1)(r^n-1)}{2(r-1)} - \frac{L}{\delta} = \frac{1}{2}(r+1)GP_n(r) - \frac{L}{\delta}$
  $f_{\mathrm{vv}}(0) = \frac{1}{2} - \frac{L}{\delta}$: $f_{\mathrm{nv}}(0) \leq 0$ if $\frac{L}{\delta} \geq \frac{1}{2}$
  $f_{\mathrm{vv}}(1) = n-\frac{L}{\delta}$: $f_{\mathrm{nv}}(1) > 0 \Leftrightarrow \frac{L}{\delta} < n$
  $f_{\mathrm{vv}}(\frac{L}{\delta}/(n-1)) > 0$ if $\frac{L}{\delta} > n-1$

So for each case, it is possible to derive lower and upper bounds for the root r of the equation: $r = 0$ can always provides a lower bound and the third bound listed for each grid type above an upper bound for the root of the equation; $r = 1$ may be used to override either of them, depending on whether it is a lower ($f_{\mathrm{xx}}(1) < 0$) or an upper bound ($f_{\mathrm{xx}}(1) > 0$) of the root.

4.6 General series-based grids

For general series-based grids, we assume that we have a sequence $\delta_i > 0$ $(i = 1, \ldots N)$ and that

$$q_i = \frac{1}{\Delta} \sum_{j=1}^{i} \delta_i \quad i = 1, \ldots N$$

where

$$\Delta = \sum_{j=1}^{N} \delta_i$$

is a normalizing scale such that $q_N = 1$. Furthermore, $q_0 = 0$.

Bibliography

M. Abramowitz and I. Stegun. *Handbook of Mathematical Functions*. Dover, New York, NY, 1965. 9th printing.

R. A. Berner. *Early Diagenesis. A Theoretical Approach*. Princeton Series in Geochemistry. Princeton University Press, Princeton, NJ, 1980.

B. P. Boudreau. *Diagenetic Models and Their Implementation*. Springer-Verlag, Berlin, 1997. doi: 10.1007/978-3-642-60421-8.

B. P. Boudreau. On the equivalence of non-local and radial-diffusion models for porewater irrigation. *J. Mar. Res.*, 42(3):731–735, 1984. doi: 10.1357/002224084788505924.

B. P. Boudreau. The mathematics of tracer mixing in sediments : I. Spatially-dependent, diffusive mixing. *Am. J. Sci.*, 286(3):161–198, 1986. doi: 10.2475/ajs.286.3.161.

G. Engeln-Müllges and F. Uhlig. *Numerical Algorithms with Fortran*. Springer-Verlag, Berlin, 1996. ISBN 3-540-60529-0.

M. E. Fiadeiro and G. Veronis. On weighted-mean schemes for the finite difference approximation to the advection-diffusion equation. *Tellus*, 29:512–522, 1977. doi: 10.1111/j.2153-3490.1977.tb00763.x.

W. Hundsdorfer and J. Verwer. *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*. Springer Series in Computational Mathematics. Springer, Berlin, 2003. ISBN 3-540-03440-4. doi: 10.1007/978-3-662-09017-6.

H.-G. Roos, M. Stynes, and L. Tobiska. *Robust Numerical Methods for Singularly Perturbed Differential Equations*, volume 24 of *Springer Series in Computational Mathematics*. Springer, 2 edition, 2008. ISBN 978-3-540-34466-7. doi: 10.1007/978-3-540-34467-4.

K. Soetaert, P. M. J. Herman, and J. J. Middelburg. A model of early diagenetic processes from the shelf to abyssal depths. *Geochim. Cosmochim. Ac.*, 60(6):1019–1040, 1996. doi: 10.1016/0016-7037(96)00013-0.