MOX–Report No. 26/2009

A robust and efficient conservative technique for simulating three-dimensional sedimentary basins dynamics

Matteo Longoni, A.C.I. Malossi, Andrea Villa

MOX, Dipartimento di Matematica "F. Brioschi"
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox@mate.polimi.it http://mox.polimi.it

A robust and efficient conservative technique for simulating three-dimensional sedimentary basins dynamics. *

M. Longoni [♯], A. C. I. Malossi [♮] [†], A. Villa [‡]

September 1, 2009

[♯] MOX– Modellistica e Calcolo Scientifico
Dipartimento di Matematica "F. Brioschi"
Politecnico di Milano
via Bonardi 9, 20133 Milano, Italy
matteo.longoni@mate.polimi.it

[♮] MOX– Modellistica e Calcolo Scientifico
Dipartimento di Matematica "F. Brioschi"
Politecnico di Milano
via Bonardi 9, 20133 Milano, Italy
cristiano.malossi@mate.polimi.it

[‡] Dipartimento di Matematica "F. Eriquez"
Universita' degli studi di Milano
via Saldini 50, 20133 Milano, Italy
andrea.villa@unimi.it

Keywords: Computational geology, Level Set method, Finite-element method.

AMS Subject Classification: 86-08, 65N30

Abstract

The development of new efficient numerical techniques is a key point in computational fluid dynamics, and as a consequence in geological simulations. In this paper we present a model for simulating the dynamic of a three-dimensional stratified sedimentary basins. This kind of problem contains several numerical complexities such as the

*This work has been supported by the Project: Steam3D - ENI

†Currently at Ecole Polytechnique Fédérale de Lausanne (EPFL), IACS–CMCS (Modeling and Scientific Computing), Av. Piccard, Station 8, CH-1015, Lausanne, Switzerland, cristiano.malossi@epfl.ch

presence of high viscosity jumps, or the necessity of tracking multiple surfaces of interface (horizons) independently. To overcome these difficulties, we introduce a new preconditioner, that reduces significantly the amount of time required to solve the finite element linear system resulting from the Stokes problem, and a new tracking method. Using a coupled Level Set–Volume Tracking method, indeed, an unlimited number of layers can be tracked with good mass conservation properties. To prove the efficiency of these new techniques, we present finally the results and the computation performances obtained in simulations of a realistic case with four horizons, together with a complete description of the main physical quantities involved.

1 Introduction

Recent geological studies have shown a strong correlation between the presence of salt domes and the formation of oilfields, and as consequence, during the last few years, the interest towards inner dynamics of salt sedimentary basins has had a marked increase. Indeed, salt is a critical element for the dynamics of sedimentary basins, thanks to its high viscosity and its low permeability.

The development of mathematical models and numerical approximations has lead to impressive results in this research field, as a powerful tool to compare and/or explain seismic data. Nevertheless, the necessity of a three-dimensional approach is inevitable, as the geological structures assume complex morphologies which can not be described completely through two-dimensional models. For this reason it is essential to employ robust and, above all, efficient methods to solve with accuracy and in a reasonable time, models composed by a considerable number of degrees of freedom, such as those associated with three-dimensional discretizations.

In this article we describe the main features of a mathematical model for simulating the evolution of salt sedimentary basins, and its efficient implementation. The most innovative aspects on which we will focus on are the computational efficiency of the proposed method and the ability to describe accurately the interfaces between layers (usually referred as horizons). We postpone to future papers the handling of more complex issues such as faults, non-Newtonian rheologies and sedimentation processes.

The reference model is composed of a series of layers, modeled as incompressible and immiscible fluids in Stokes regime. This choice, justified by the phenomena of interest lifetime, that is comparable to geological ages (as shown for example by [2] and [25]), allows geologists to study the Rayleigh–Taylor instabilities associated with salt diapirism (as described by [11, 13, 21, 28] and [29]). To solve this kind of problem we choose a classic approach that splits the computation of pressure and velocity field from the interface tracking. Moreover, to improve the efficiency and accuracy of the code, we have

introduced some innovations both in the resolution of the linear systems and in the tracking algorithm.
During the past few years, the Finite Element (FE) method has played a predominant role in the solution of the mass and force balance equations (as shown by [11, 12, 13, 14, 17], and [21]), as it permits to solve the fluid dynamic problem in complex geometrical cases with high accuracy. For three-dimensional problems, the number of degrees of freedom (DOF) required for an accurate discretization is so high that the adoption of iterative schemes to solve the linear system is necessary. In the present case, in addition, the value of the viscosity of the sediments, which are modeled as fluids, varies over a range of about five orders of magnitude (as shown by [13] and [21]), and this leads to a bad conditioned algebraic problem. Such a complex problem is in general unsolvable as it is, or at least it requires too many iterations to converge within a reasonable tolerance. To overcome these intricacies, in this paper we introduce an innovative efficient preconditioner, that is able to reduce sensibly the number of iterations required to solve the linear system in presence of high viscosity jumps.
As regards the tracking phase, no technique seems to be prevailing on the others. One of the first method employed to solve this kind of problems is the particle in a cell (PIC) (as shown by [8] and [14]), that has been applied successfully by [23] and then extended by [15]. This method permits to follow the evolution of a large number of layers, by tracking a high number of particles at the same time, and to reconstruct the physical quantities a posteriori, on the mesh used to compute the velocity field. However, despite its strength, this technique is very expensive as the reconstruction of the physical quantities needs to find how many particles lie inside each cell. Furthermore, the PIC method can not represent horizons, that instead provide a direct visualization of the geometry of the sediment layers.
An alternative to the PIC is represented by the Volume of Fluid methods [29]. This algorithm transports the partial volumes but does not contain any direct information regarding interface positions. These are extracted with post processing algorithms that are, usually, only applicable to the two fluid case. A similar way of proceeding, based on characteristic methods, was used by [26, 27] and [28] to simulate the growth of diapirs in two and three dimensions. With respect to PIC, this technique provides some small advantages in terms of computational cost, though neither can handle the reconstruction of the horizons. A third possibility is the Level-Set method (LS), that represents an interface as the zero level of an auxiliary function (see [16] and [22]). The advantage is that this method easily follows a shape that changes topology, for example when it splits into two parts or develops holes, but on the other hand it does not guarantee the conservation of the volumes. There are a few examples of multi-fluid LS approaches, some of them are specifically devoted to the mean curvature flows ([19] and [24]) and cannot be applied readily to an advection-driven case. In [30] a nested LS

approach is proposed: this approach relies on the standard signed distance function and is not conservative.
A different possibility, applied almost exclusively in two dimensions, is the Lagrangian tracking (as reported by [12, 13] and [17]). This technique allows an explicit reconstruction of horizons with a good computational accuracy, paying a relative low computational cost. However, an automatic handling of topological changes is still quite complex, even in two dimensions, therefore this kind of methods have never been applied to realistic three-dimensional geological problems. In our case, we have chosen an Eulerian approach, as it satisfies our objective of having a method that is robust, accurate and above all able to represent horizons evolution. In particular, we implement a coupled Level Set-Volume Tracking (LS-VT) method (see [7]), which is applied for the first time to a realistic three-dimensional geological case, in presence of more than two layers. This new technique is nearly conservative and is able to give a precise and coherent reconstruction of all the horizons.
This paper is organized as follows: in Section 2 we introduce the mathematical model and the classic splitting approach, which consists in separating the Stokes problem from the hyperbolic equation of advection of horizons. Subsequently, in Section 3 we study into detail the algebraic structure of the Stokes problem and we show a way to build an appropriate preconditioner optimized for high-order viscosity jumps. Then, in Section 4 we present the implementation of the tracking method proposed by [7], and we give a brief description of the approach used to reconstruct the Set function. In Section 5, we provide an overview of the algorithm workflow, and finally in Section 6 we illustrate and discuss the results of the simulation of a realistic 3D geological model.

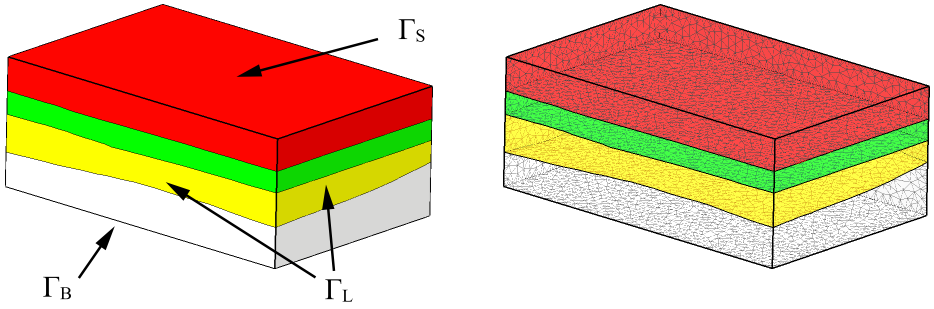
2 Physical and mathematical model

In this section we present the physical model and we introduce some mathematical notation. A time splitting algorithm is introduced along with the time discretization.

2.1 Nomenclature

Let's introduce the geometric model of the sedimentary basin (see Figure 1). The domain $\Omega \in \mathbb{R}^3$ is divided into n_s subdomains Ω_i (without overlapping regions), which represent different layers characterized by a specific value of density ρ_i and dynamic viscosity μ_i. The external boundary Γ of the domain Ω is divided into three parts: the basement Γ_{B} and the free surface Γ_{S}, where we have imposed a Dirichlet condition for the velocity field, and the lateral contour Γ_{L}, that we suppose vertical for simplicity and where we have imposed a vertical no-stress condition. In addition, the horizons

between the subdomains are defined as $\Gamma_{i,j} = \Omega_i \cap \Omega_j$.



(a) (b)

Figure 1: (a) External shape of the domain Ω, which contains three horizons and four layers. The external boundary Γ is divided into three parts: the basement Γ_B, the free surface Γ_S and the lateral contour Γ_L. (b) An open three-dimensional view of the sedimentary basin.

To complete our overview let's introduce now some mathematical objects: $\overrightarrow{X} = (x_1, x_2, x_3) \in \Omega$, that indicates a generic point in the spatial domain of coordinates x_i, with $i = 1, 2, 3$, $(\widehat{x}_1, \widehat{x}_2, \widehat{x}_3)$ the axial versors, $\widehat{n}$ the domain outward normal and $t \in (0, T]$ the time coordinate. Then for a generic vector $\overrightarrow{u}$ we denote its cartesian orthogonal components with (u_1, u_2, u_3). Finally, we introduce the mathematical model that describes the geological evolution of the basin, modeled as a stratified fluid, in which the layers are immiscible and have constant properties:

$$\begin{cases} \overrightarrow{\nabla} \cdot \bar{\bar{\sigma}}(\mu) - \overrightarrow{\nabla} P + \rho \overrightarrow{g} = 0 & \text{in} \quad \Omega \times (0, T], \\ \overrightarrow{\nabla} \cdot \overrightarrow{u} = 0 & \text{in} \quad \Omega \times (0, T], \\ \dfrac{\partial \rho}{\partial t} + \overrightarrow{u} \cdot \overrightarrow{\nabla} \rho = 0, \quad \dfrac{\partial \mu}{\partial t} + \overrightarrow{u} \cdot \overrightarrow{\nabla} \mu = 0 & \text{in} \quad \Omega \times (0, T], \\ \rho = \rho^0, \ \mu = \mu^0 & \text{in} \quad \Omega \times \{0\}, \\ \overrightarrow{u} = \overline{u} & \text{on} \quad \Gamma_B \cup \Gamma_S \times [0, T], \\ u_1 = \overline{u}_1, \ u_2 = \overline{u}_2, \ (\bar{\bar{\sigma}} \cdot \overrightarrow{n}) \cdot \widehat{x}_3 = 0 & \text{on} \quad \Gamma_L \times [0, T]. \end{cases} \tag{1}$$

The unknowns are the velocity and pressure fields (respectively $\overrightarrow{u}$ and P), together with the physical quantities (the density ρ and dynamic viscosity μ). Finally $\overrightarrow{g}$ is the gravitational acceleration vector. We also assume for now a Newtonian law for the stress tensor: $\bar{\bar{\sigma}} = \mu(\overrightarrow{\nabla}\overrightarrow{u} + (\overrightarrow{\nabla}\overrightarrow{u})^T)$. This relation may not seem to be truly representative of the rheological complexity of the sediments, however it is widely accepted in literature as a solid base model to study the Rayleigh–Taylor instabilities associated with diapirism ([13, 27],

and [28]). As regards the boundary conditions, $\overline{u}$ is the imposed velocity on Γ_{B} and Γ_{S}. On Γ_{L} only the velocity components orthogonal to $\widehat{x}_3$ are fixed through $\overline{u}_1$ and $\overline{u}_2$. We have also to provide the initial conditions for the physical quantities, through ρ^0 and μ^0.

2.2 Time discretization and splitting algorithm

To solve (1) we must discretize the time interval $[0, T]$; we introduce hence a variable time step, such that $\Delta t^n = t^{n+1} - t^n$, and we discretize the interval as $[0, t^1, t^2, \ldots, T]$. Therefore, we can finally solve the time discrete problem through the following splitting algorithm (already used, for example, by [13]). The apex n indicates the value of a quantity at time t^n:

Algorithm 1. *At the generic time step n we solve problem (1) in two steps:*

1. *given ρ^n and μ^n, we compute $\overrightarrow{u}^n$ and P^n by solving the Stokes problem*

$$\begin{cases} \overrightarrow{\nabla} \cdot \bar{\bar{\sigma}}^n - \overrightarrow{\nabla} P^n + \rho^n \overrightarrow{g} = 0 & \text{in} \quad \Omega, \\ \overrightarrow{\nabla} \cdot \overrightarrow{u}^n = 0 & \text{in} \quad \Omega, \\ \overrightarrow{u}^n = \overline{u} & \text{on} \quad \Gamma_B \cup \Gamma_S, \\ u_1^n = \overline{u}_1, \; u_2^n = \overline{u}_2, \; (\bar{\bar{\sigma}}^n \cdot \overrightarrow{n}) \cdot \widehat{x}_3 = 0 & \text{on} \quad \Gamma_L; \end{cases} \tag{2}$$

2. *given $\overrightarrow{u}^n$ and P^n, we obtain ρ^{n+1} and μ^{n+1} using to the following hyperbolic equations:*

$$\frac{\partial \rho}{\partial t} + \overrightarrow{u}^n \cdot \overrightarrow{\nabla} \rho = 0, \qquad \frac{\partial \mu}{\partial t} + \overrightarrow{u}^n \cdot \overrightarrow{\nabla} \mu = 0, \qquad \text{in } \Omega \times (t^n, t^{n+1}]. \tag{3}$$

The first step of the algorithm is a classic Stokes problem that can be solved efficiently with a FE method. The second step requires the solution of a linear hyperbolic problem.

2.3 Characteristic function for physical quantities

We want now to describe efficiently the internal three-dimensional structure of a domain made of immiscible layers characterized by different physical properties. At this purpose we introduce a set of characteristic functions λ_i (one for each subdomain Ω_i)

$$\lambda_i(\overrightarrow{X}) = \begin{cases} 1 & \text{if } \overrightarrow{X} \in \Omega_i, \\ 0 & \text{if } \overrightarrow{X} \notin \Omega_i, \end{cases}$$

that we can gather in the composition vector $\boldsymbol{\lambda}$. Through these functions we can write the density and the dynamic viscosity fields as

$$\rho = \sum_{i=1}^{n_s} \lambda_i \rho_i, \qquad \mu = \sum_{i=1}^{n_s} \lambda_i \mu_i. \tag{4}$$

Denoting the value of λ_i at time t^0 with λ_i^0, we can rewrite in the same way the initial condition for the physical quantities, as

$$\rho^0 = \sum_{i=1}^{n_s} \lambda_i^0 \rho_i, \qquad \mu^0 = \sum_{i=1}^{n_s} \lambda_i^0 \mu_i.$$

We need now to rephrase the hyperbolic transport equation in a volume transport equation. For $t \in [t^n, t^{n+1}]$ we have

$$\begin{cases} \dfrac{\partial}{\partial t}\boldsymbol{\lambda} + \overrightarrow{u}^n \cdot \overrightarrow{\nabla}\boldsymbol{\lambda} = 0, \\ \boldsymbol{\lambda}(0, \overrightarrow{X}) = \boldsymbol{\lambda}^0(\overrightarrow{X}), \end{cases} \tag{5}$$

where $\boldsymbol{\lambda}^0$ is the composition vector at time t^0. It is easy to show that equation (5) is equivalent to the mass conservation law (3): multiplying (5) by ρ_i and performing a sum on the geological species we obtain

$$\sum_{i=1}^{n_s} \left(\rho_i \frac{\partial \lambda_i}{\partial t} + \rho_i \overrightarrow{u}^n \cdot \overrightarrow{\nabla}\lambda_i \right) = 0.$$

Hence, as the densities of the components ρ_i are constant, we get the first equation of (3). With a similar argument we can also prove the second of (3). Once we have provided the basic description of the splitting algorithm, in the next two sections we will analyze one by one both the steps of this scheme.

3 Velocity field solver

In this section we discuss the difficulties related to the solution of the Stokes problem. The FE formulation is equivalent to an algebraic system of equations, that, especially in 3D, has a very large number of unknowns. Besides, the strong variability of the viscosity coefficient badly affects the conditioning of the FE matrix. This imposes the usage of a preconditioned iterative method, which is the best solution when dealing with 3D problems. This technique has not been studied in detail yet in our particular case. In the following we define the discrete problem and we construct a suitable preconditioning technique.
Let $\mathcal{T}_\Delta^G$ be a simplicial tetrahedral grid containing n_e^G elements e_r^G (with $r = 1, \ldots, n_e^G$) and n_p^G nodes $\overrightarrow{x}_k^G$ (with $k = 1, \ldots, n_p^G$), where the subscript Δ stands for the maximum diameter of the grid elements. From $\mathcal{T}_\Delta^G$ we build the mini-grid $\mathcal{T}_\Delta^M$ by adding n_e^G barycentric nodes (see Figure 2); hence $\mathcal{T}_\Delta^M$ has $n_p^M = n_p^G + n_e^G$ nodes $\overrightarrow{x}_k^M$ and $n_e^M = 4n_e^G$ elements e_r^M. Let's finally recall the discrete variables and introduce the related discrete spaces:

$$\overrightarrow{u}_\Delta^n \in \mathbb{V}_1^M, \quad P_\Delta^n \in \mathbb{V}_1^G,$$

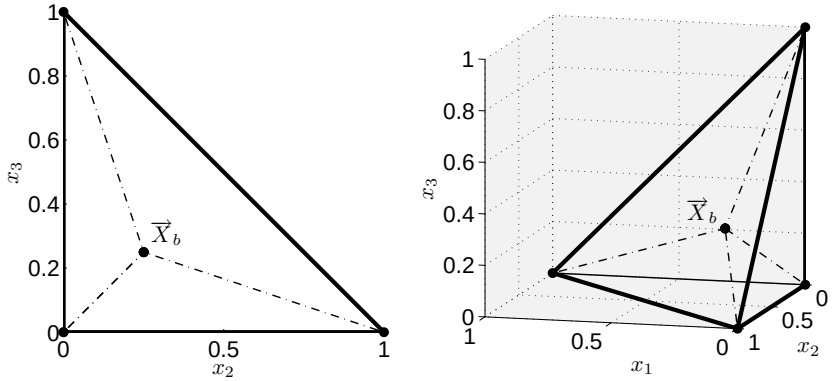


Figure 2: A sketch of $\mathcal{T}_\Delta^M$ grid obtained by the refinement of the $\mathcal{T}_\Delta^G$ grid. An element e_r^G is depicted with a solid line while the four elements e_r^M are depicted with a dot–dashed line. The latter are obtained by adding a barycentric node inside the grid element, connected to the vertices of e_r^G.

where $\mathbb{V}_1^M = X_1^M \cap H_0^1$ and $\mathbb{V}_1^G = X_1^G \cap L_0^2$ and

$$X_1^M = \left\{ \overrightarrow{\varphi}_\Delta \in \mathbb{C}^0(\Omega) : \overrightarrow{\varphi}_\Delta|_{e_r^M} \in \mathbb{P}^1, \quad \forall r = 1, \dots, n_e^M \right\},$$

$$X_1^G = \left\{ \varphi_\Delta \in \mathbb{C}^0(\Omega) : \varphi_\Delta|_{e_r^G} \in \mathbb{P}^1, \quad \forall r = 1, \dots, n_e^G \right\},$$

Then we introduce the weak formulation of problem (2)

$$\begin{cases} a(\overrightarrow{u}_\Delta^n, \overrightarrow{v}_\Delta) + b(P_\Delta^n, \overrightarrow{v}_\Delta) = F(\overrightarrow{v}_\Delta) & \forall \overrightarrow{v}_\Delta \in \mathbb{V}_1^M, \\ b(q_\Delta, \overrightarrow{u}_\Delta^n) = 0 & \forall q_\Delta \in \mathbb{V}_1^G, \end{cases} \tag{6}$$

where

$$a(\overrightarrow{u}_\Delta^n, \overrightarrow{v}_\Delta) = -\int \mu(\overrightarrow{\nabla}\overrightarrow{u}_\Delta^n + (\overrightarrow{\nabla}\overrightarrow{u}_\Delta^n)^T) : (\overrightarrow{\nabla}\overrightarrow{v}_\Delta),$$

$$b(P_\Delta^n, \overrightarrow{v}_\Delta) = \int \overrightarrow{P}_\Delta^n (\overrightarrow{\nabla} \cdot \overrightarrow{v}_\Delta),$$

$$F(\overrightarrow{v}_\Delta) = -\int \rho_\Delta (\overrightarrow{g} \cdot \overrightarrow{v}_\Delta).$$

Equation (6) represents the discretization of the Stokes problem with the mini-elements. Let $\mu > 0$ and $\mu \in L^\infty$, therefore the bilinear form $a(\cdot,\cdot)$ is coercive and from the choice of the discrete spaces the following inf-sup condition holds:

$$\inf_{q \in L^2(\Omega),\, q \neq 0} \sup_{\overrightarrow{v} \in H^1(\Omega),\, \overrightarrow{v} \neq 0} \frac{b(q, \overrightarrow{v})}{|\overrightarrow{v}|_{H^1(\Omega)} \|q\|_{L^2(\Omega)}} \geq \beta.$$

Furthermore, the FE couple we have chosen satisfies the discrete inf-sup condition (see [5])

$$\min_{q_\Delta \in \mathbb{V}_1^G,\, q_\Delta \neq 0} \ \max_{\overrightarrow{v}_\Delta \in \mathbb{V}_1^M,\, \overrightarrow{v}_\Delta \neq 0} \frac{b(q_\Delta, \overrightarrow{v}_\Delta)}{|\overrightarrow{v}_\Delta|_{H^1(\Omega)} \|q_\Delta\|_{L^2(\Omega)}} \geq \beta_\Delta. \tag{7}$$

At last, assuming $\rho \in L^\infty$, the functional $F(\cdot)$ is continuous. Thus, problem (6) has a unique solution (see [18]).

Remark 1. In the bilinear form $a(\cdot,\cdot)$ the velocity gradient is piecewise constant on the elements and we can slip it outside the integral sign. Therefore we have only to provide the viscosity mean value on every element. ■

Let $\{\varphi_k^M\} \in \mathbb{V}_1^M$ and $\{\varphi_k^G\} \in \mathbb{V}_1^G$ be the lagrangian basis defined on the grids $\mathcal{T}_\Delta^M$ and $\mathcal{T}_\Delta^G$ respectively, then the discrete solution can be expanded as

$$\overrightarrow{u}_\Delta^n = \sum_{k=1}^{n_p^M} \overrightarrow{u}_k^n \varphi_k^M, \qquad P_\Delta^n = \sum_{k=1}^{n_p^G} P_k^n \varphi_k^G. \tag{8}$$

Through (8) and defining $\mathbf{V}$ and $\mathbf{P}$ as the vectors of the degrees of freedom of the velocity and pressure fields, we get the following algebraic problem:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V} \\ \mathbf{P} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_v \\ \mathbf{0} \end{bmatrix},$$

where

$$\mathbf{A}_{ij} = a(\varphi_i^M, \varphi_j^M), \quad \mathbf{B}_{ij} = b(\varphi_i^G, \varphi_j^M), \quad \mathbf{F}_{v,i} = F(\varphi_i^M),$$

and $\mathbf{A}$ is a positive definite matrix. We solve this algebraic system by means of the pressure Schur complement

$$(\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T)\mathbf{P} = \mathbf{B}\mathbf{A}^{-1}\mathbf{F}_v.$$

This algebraic complement is solved through a nested cycle iterative system. The inner cycle is necessary to avoid an explicit inversion of matrix $\mathbf{A}$. Let's now attend to the conditioning of $\mathbf{A}$ and of the Schur complement $\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T$ that form the two linear systems of our iterative scheme. Before, we briefly recall some useful results applied to our case. Let $\overrightarrow{v}_\Delta \in \mathbb{V}_1^M$ and $q_\Delta \in \mathbb{V}_1^G$ then we consider the Korn inequality (see [4])

$$\|\overrightarrow{\nabla}\overrightarrow{u}_\Delta\|_{L^2(\Omega)} \leq C_1 \|\overrightarrow{\nabla}\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\overrightarrow{u}_\Delta)^T\|_{L^2(\Omega)}, \tag{9}$$

and we also have

$$\|\overrightarrow{\nabla}\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\overrightarrow{u}_\Delta)^T\|_{L^2(\Omega)} \leq C_2 \|\overrightarrow{\nabla}\overrightarrow{u}_\Delta\|_{L^2(\Omega)}. \tag{10}$$

From [5] we obtain

$$\|\overrightarrow{\nabla} \cdot \overrightarrow{u}_\Delta\|_{L^2(\Omega)} \leq C_3 \|\overrightarrow{\nabla}\overrightarrow{u}_\Delta\|_{L^2(\Omega)}, \tag{11}$$

and the inverse inequality (see [6]) leads to

$$\|\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta}\|_{L^2(\Omega)} \leq \frac{C_4}{\Delta}\|\overrightarrow{u}_{\Delta}\|_{L^2(\Omega)}. \tag{12}$$

We remark that Δ stands for the maximum diameter of the grid elements. Finally we recall the Poincaré inequality (see [6])

$$\|\overrightarrow{u}_{\Delta}\|_{L^2(\Omega)} \leq C_p\|\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta}\|_{L^2(\Omega)}. \tag{13}$$

Let's now estimate the Schur complement conditioning number.

Proposition 1. *The conditioning number of the Schur complement* $\mathcal{K}(\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T)$ *is not dependent upon* Δ*, more explicitly:*

$$\mathcal{K}(\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T) \leq \frac{\sup(\mu)}{\inf(\mu)}\frac{C_6}{C_5}\left(\frac{C_1C_2^2C_3}{\beta_{\Delta}^2}\right)^2.$$

Proof. Combining the discrete inf-sup condition (7) and equation (10) we get:

$$\min_{q_{\Delta}\in\mathbb{V}_1^G}\max_{\overrightarrow{v}_{\Delta}\in\mathbb{V}_1^M}\frac{|(q_{\Delta},\overrightarrow{\nabla}\cdot\overrightarrow{u}_{\Delta})|}{\|\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta}+(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta})^T\|_{L^2(\Omega)}\|q_{\Delta}\|_{L^2(\Omega)}} \geq \frac{\beta_{\Delta}}{C_2}. \tag{14}$$

The norm of the weighted symmetric gradient can be bounded as follows:

$$\begin{aligned}\|\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta}+(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta})^T\|_{L^2(\Omega)} &= \|\frac{1}{\sqrt{\mu}}\sqrt{\mu}\left(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta}+(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta})^T\right)\|_{L^2(\Omega)}\\ &\geq \frac{1}{\max(\sqrt{\mu})}\|\sqrt{\mu}\left(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta}+(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta})^T\right)\|_{L^2(\Omega)},\end{aligned} \tag{15}$$

and plugging equation (15) in (14) we get

$$\min_{q_{\Delta}\in\mathbb{V}_1^G}\max_{\overrightarrow{v}_{\Delta}\in\mathbb{V}_1^M}\frac{|(q_{\Delta},\overrightarrow{\nabla}\cdot\overrightarrow{u}_{\Delta})|}{\|\sqrt{\mu}\left(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta}+(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta})^T\right)\|_{L^2(\Omega)}\|q_{\Delta}\|_{L^2(\Omega)}} \geq \frac{1}{\max(\sqrt{\mu})}\frac{\beta_{\Delta}}{C_2}. \tag{16}$$

From the expansions (8) we obtain

$$\begin{aligned}&(q_{\Delta},\overrightarrow{\nabla}\cdot\overrightarrow{u}_{\Delta}) = \mathbf{P}^T\mathbf{B}\mathbf{V},\\ &\|q_{\Delta}\|_{L^2(\Omega)} = \sqrt{\mathbf{P}^T\mathbf{Q}\mathbf{P}},\\ &\|\sqrt{\mu}\left(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta}+(\overrightarrow{\nabla}\,\overrightarrow{u}_{\Delta})^T\right)\|_{L^2(\Omega)} = \sqrt{\mathbf{V}^T\mathbf{A}\mathbf{V}},\end{aligned}$$

where $\mathbf{Q}$ is the mass matrix of the pressure field and $\mathbf{P}\in\mathbb{R}^{n_p^G}$, $\mathbf{V}\in\mathbb{R}^{3n_p^M}$. Hence, (16) can be rewritten as

$$\min_{\mathbf{P}}\max_{\mathbf{V}}\frac{|(\mathbf{P}^T\mathbf{B}\mathbf{V})|}{\sqrt{\mathbf{V}^T\mathbf{A}\mathbf{V}}\sqrt{\mathbf{P}^T\mathbf{Q}\mathbf{P}}} \geq \frac{1}{\max(\sqrt{\mu})}\frac{\beta_{\Delta}}{C_2}.$$

Now let's introduce $\mathbf{W} = \mathbf{A}^{1/2}\mathbf{V}$, therefore

$$\frac{1}{\max(\sqrt{\mu})}\frac{\beta_\Delta}{C_2} \leq \min_{\mathbf{P}} \max_{\mathbf{W}} \frac{|\mathbf{P}^T\mathbf{B}\mathbf{A}^{-1/2}\mathbf{W}|}{\sqrt{\mathbf{W}^T\mathbf{W}}\sqrt{\mathbf{P}^T\mathbf{Q}\mathbf{P}}}.$$

This equation implies $\mathbf{W} = \mathbf{A}^{-1/2}\mathbf{B}^T\mathbf{P}$, that yields to

$$\frac{1}{\max(\sqrt{\mu})}\frac{\beta_\Delta}{C_2} \leq \frac{\sqrt{|\mathbf{P}^T\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T\mathbf{P}|}}{\sqrt{\mathbf{P}^T\mathbf{Q}\mathbf{P}}}, \tag{17}$$

thus completing the lower bound estimate. As regards the upper bound, combining (9) and (11) with the Cauchy-Schwartz inequality, we have

$$\begin{aligned}|(q_\Delta, \overrightarrow{\nabla} \cdot \overrightarrow{u}_\Delta)| &\leq \|q_\Delta\|_{L^2(\Omega)}\|\overrightarrow{\nabla} \cdot \overrightarrow{u}_\Delta\|_{L^2(\Omega)} \leq C_1C_3\|q_\Delta\|_{L^2(\Omega)}\|\overrightarrow{\nabla}\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\overrightarrow{u}_\Delta)^T\|_{L^2(\Omega)} \\ &\leq \frac{C_1C_3}{\min(\sqrt{\mu})}\|q_\Delta\|_{L^2(\Omega)}\|\sqrt{\mu}\left(\overrightarrow{\nabla}\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\overrightarrow{u}_\Delta)^T\right)\|_{L^2(\Omega)},\end{aligned}$$

or the equivalent form

$$\frac{|(q_\Delta, \overrightarrow{\nabla} \cdot \overrightarrow{u}_\Delta)|}{\|q_\Delta\|_{L^2(\Omega)}\|\sqrt{\mu}\left(\overrightarrow{\nabla}\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\overrightarrow{u}_\Delta)^T\right)\|_{L^2(\Omega)}} \leq \frac{C_1C_3}{\min(\sqrt{\mu})}.$$

Applying the same argument used for the lower bound we get:

$$\frac{\sqrt{|\mathbf{P}^T\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T\mathbf{P}|}}{\sqrt{\mathbf{P}^T\mathbf{Q}\mathbf{P}}} \leq \frac{C_1C_3}{\min(\sqrt{\mu})}. \tag{18}$$

Now, combining the lower bound (17) and the upper bound (18) we obtain

$$\frac{1}{\max(\mu)}\frac{\beta_\Delta^2}{C_2^2} \leq \frac{|\mathbf{P}^T\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T\mathbf{P}|}{\mathbf{P}^T\mathbf{Q}\mathbf{P}} \leq \frac{C_1^2C_3^2}{\min(\mu)},$$

that, multiplied by $(\mathbf{P}^T\mathbf{Q}\mathbf{P})/(\mathbf{P}^T\mathbf{P})$, leads to

$$\frac{1}{\max(\mu)}\frac{\beta_\Delta^2}{C_2^2}\frac{\mathbf{P}^T\mathbf{Q}\mathbf{P}}{\mathbf{P}^T\mathbf{P}} \leq \frac{|\mathbf{P}^T\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T\mathbf{P}|}{\mathbf{P}^T\mathbf{P}} \leq \frac{C_1^2C_3^2}{\min(\mu)}\frac{\mathbf{P}^T\mathbf{Q}\mathbf{P}}{\mathbf{P}^T\mathbf{P}}. \tag{19}$$

From [5] we have

$$C_5\Delta^2 \leq \frac{\mathbf{P}^T\mathbf{Q}\mathbf{P}}{\mathbf{P}^T\mathbf{P}} \leq C_6\Delta^2,$$

that combined with (19) yields to

$$\frac{C_5}{\max(\mu)}\frac{\beta_\Delta^2}{C_2^2}\Delta^2 \leq \frac{|\mathbf{P}^T\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T\mathbf{P}|}{\mathbf{P}^T\mathbf{P}} \leq \frac{C_1^2C_3^2C_6}{\min(\mu)}\Delta^2,$$

and the proof holds. ■

Let's now consider the matrix $\mathbf{A}$.

Proposition 2. *The conditioning number of* $\mathbf{A}$ *is bounded by*

$$\mathcal{K}(\mathbf{A}) \leq \frac{\max(\mu)}{\inf(\mu)} \frac{1}{\Delta^2} \frac{C_8}{C_7} \left(C_1 C_2 C_5 C_p\right)^2 .$$

Proof. We can bound the weighted symmetric gradient through equations (10) and (12) for the upper bound

$$\begin{aligned} \|\sqrt{\mu}\left(\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta)^T\right)\|_{L^2(\Omega)} &\leq \max(\sqrt{\mu})\|\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta)^T\|_{L^2(\Omega)} \\ &\leq C_2 \max(\sqrt{\mu})\|\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta\|_{L^2(\Omega)} \\ &\leq \frac{C_2 C_4}{\Delta} \max(\sqrt{\mu})\|\overrightarrow{u}_\Delta\|_{L^2(\Omega)}, \end{aligned} \tag{20}$$

and through equations (9) and (13) for the lower bound

$$\begin{aligned} \|\sqrt{\mu}\left(\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta)^T\right)\|_{L^2(\Omega)} &\geq \min(\sqrt{\mu})\|\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta)^T\|_{L^2(\Omega)} \\ &\geq \frac{\min(\sqrt{\mu})}{C_1}\|\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta\|_{L^2(\Omega)} \\ &\geq \frac{\min(\sqrt{\mu})}{C_1 C_p}\|\overrightarrow{u}_\Delta\|_{L^2(\Omega)}. \end{aligned} \tag{21}$$

The two inequalities (20) and (21) can be coupled as

$$\frac{\min(\mu)}{C_1^2 C_p^2} \leq \frac{\|\sqrt{\mu}\left(\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta + (\overrightarrow{\nabla}\,\overrightarrow{u}_\Delta)^T\right)\|^2_{L^2(\Omega)}}{\|\overrightarrow{u}_\Delta\|^2_{L^2(\Omega)}} \leq \frac{C_2^2 C_4^2}{\Delta^2} \max(\mu),$$

or, more conveniently, from (8)

$$\frac{\min(\mu)}{C_1^2 C_p^2} \leq \frac{(\mathbf{V}^T \mathbf{A} \mathbf{V})}{(\mathbf{V}^T \mathbf{Q}_V \mathbf{V})} \leq \frac{C_2^2 C_4^2}{\Delta^2} \max(\mu),$$

where $\mathbf{Q}_V$ is the mass matrix of the velocity discrete field. Multiplying this by $(\mathbf{V}^T \mathbf{Q}_V \mathbf{V})/(\mathbf{V}^T \mathbf{V})$ we get

$$\frac{\mathbf{V}^T \mathbf{Q}_V \mathbf{V}}{\mathbf{V}^T \mathbf{V}} \frac{\min(\mu)}{C_1^2 C_p^2} \leq \frac{(\mathbf{V}^T \mathbf{A} \mathbf{V})}{(\mathbf{V}^T \mathbf{V})} \leq \frac{C_2^2 C_4^2}{\Delta^2} \max(\mu) \frac{\mathbf{V}^T \mathbf{Q}_V \mathbf{V}}{\mathbf{V}^T \mathbf{V}}. \tag{22}$$

From [5] we have

$$\Delta^2 C_7 \leq \frac{\mathbf{V}^T \mathbf{Q}_V \mathbf{V}}{\mathbf{V}^T \mathbf{V}} \leq C_8 \Delta^2,$$

that coupled with (22) leads to

$$C_7 \frac{\min(\mu)}{C_1^2 C_p^2} \Delta^2 \leq \frac{(\mathbf{V}^T \mathbf{A} \mathbf{V})}{(\mathbf{V}^T \mathbf{V})} \leq (C_2^2 C_4^2 C_8) \max(\mu),$$

and to the proof. ■

These results show that the conditioning of the two linear systems we wish to solve is bad. In particular, in the inner cycle, where $\mathbf{A}$ is involved, the matrix conditioning is affected by both the grid spacing and the viscosity jumps across the interfaces. However in this case several standard techniques such as the incomplete LU factorizations can be profitably applied. The outer cycle, involving the Schur complement, has instead a relatively better conditioning number but is still dependent upon the jumps of the viscosity coefficient. Moreover we can not apply standard preconditioning techniques, since they would require an explicit assembling of the Schur complement, that is memory and time consuming. Therefore it is mandatory to find a spectral equivalent to the Schur complement, and at this scope we introduce the following matrix:

$$\mathbf{M}_{ij} = \int \frac{1}{\mu} \varphi_i^G \varphi_j^G. \tag{23}$$

We now give a rough explanation about the reasons that make this precondition matrix suitable to our purpose. We give only a heuristical explanation of the properties of this matrix, as a complete analysis on the spectral equivalence of the Schur complement with the matrix (23) is still missing. The Schur complement is the product between the matrix $\mathbf{B}$, that represents the divergence operator, the matrix $\mathbf{A}^{-1}$, that is the discrete representation of the inverse of the Laplace operator, and the matrix $\mathbf{B}^T$, that represents the gradient. As the application of the gradient to the divergence yields to the Laplace operator, the Schur complement resembles the identity operator. Indeed it is known (see [5]) that the Schur complement is not dependent on Δ. In our case the matrix $\mathbf{A}^{-1}$ resembles more closely $(\nabla^2)^{-1}/\mu$, therefore we expect that the Schur complement is spectrally similar to the mass matrix, apart from the $1/\mu$ factor. The numerical results in Figure 3 show that the preconditioner is effective and has a better behavior when the grid spacing is reduced. In Figure 4 the spectra of the Schur complement and of the preconditioned Schur complement are compared. It can be noticed that the preconditioning matrix positively affects the ratio between the maximum and minimum eigenvalue, although there is still a dependence on the viscosity jump $g = \max(\mu)/\min(\mu)$. The numerical tests may suggest an asymptotic trend, but a theoretical result regarding the preconditioner performances is still missing and researches in this field are ongoing.

4 Tracking of the layers

As we stated previously, for the tracking phase we adopt the method developed in [7, 20]. This consists in a coupled Level Set–Volume Tracking technique capable of computing the movement of multiple sediment layers. In this section we recall the main feature of that method, we provide some technicalities to reduce the computational burden, and finally their perfor-

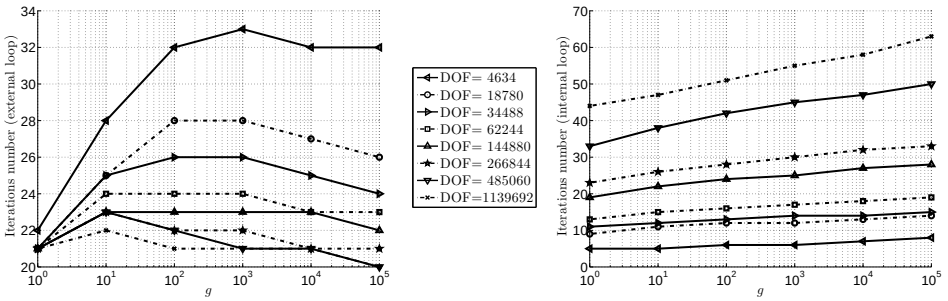


Figure 3: On the left we show the number of iterations of the external cycle at different values of the viscosity jump g: the dependence upon g or Δ^G is not so evident. On the right we show the inner cycle iterations: there is an acceptable rise in the number of iterations.

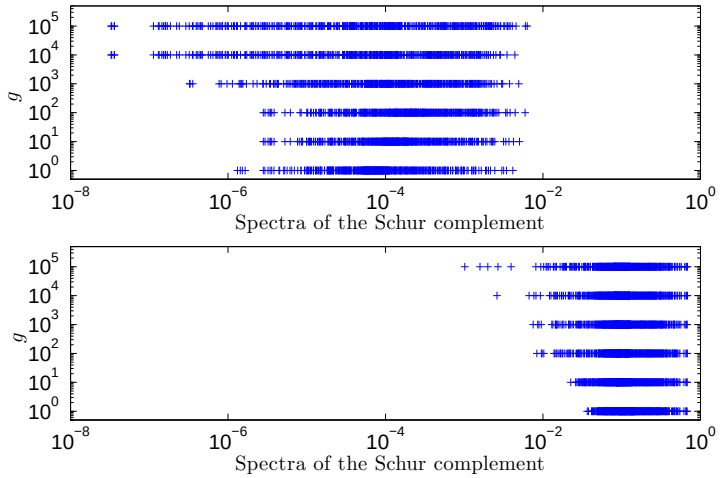
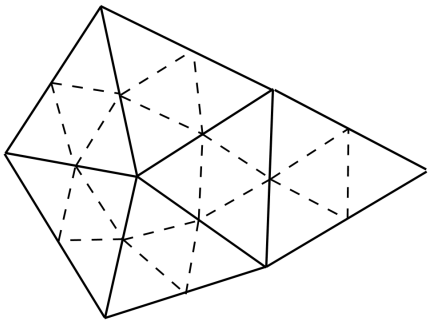


Figure 4: The spectra of the Schur complement (upper figure) for the case with with 4634 DOF, with different viscosity jumps g. In the lower figure we show the preconditioned spectra.

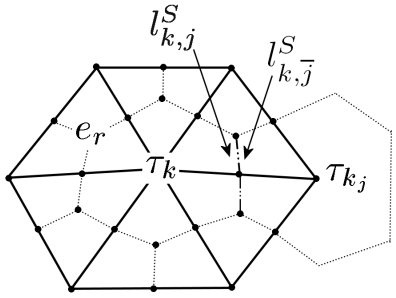
mance efficiency.

Let us introduce the nomenclature: let $\mathcal{T}_\Delta^S$ be a conformal grid obtained as a N_R-times uniform refinement of the grid $\mathcal{T}_\Delta^G$, as depicted in Figure 5. $\mathcal{T}_\Delta^S$ has n_e^S elements e_r^S and n_p^S nodes $\overrightarrow{x}_k^S$. Let's consider the dual mesh provided with n_p^S node-centered cells τ_k^S. As regards its connectivity, let $\{k_j\} = \mathbb{I}_k^C, j = 1, \ldots, |\mathbb{I}_k^C|$ be the set of the indexes of the cells surrounding a τ_k^S cell, and let $\{\tau_{k_j}^S\}$ be the set of cells surrounding τ_k^S (see Figure 5). The common surface between τ_k^S and $\tau_{k_j}^S$ is the interface $l_{k,j}^S$. We can also define the map $\overline{j} = \overline{j}(k, j)$, $j = 1, \ldots, |\mathbb{I}_k^C|$ such that $l_{k_j,\overline{j}}^S = l_{k,j}^S$, in other terms

$l^S_{k_j,\overline{j}}$ is the interface of τ_{k_j} that corresponds to the j-th interface of τ_k. For the sake of clarity, the index i will always identify the sediment layer, k will refer to cells and j to quantities related to the interfaces.



(a)



(b)

Figure 5: We show two 2D sketches of the underlying meshes. Obviously we are handling their 3D counterparts. (a) The grid $\mathcal{T}^G_\Delta$ is identified by a solid line, and the dashed line defines the refinement procedure leading to $\mathcal{T}^S_\Delta$. (b) A sketch of the $\mathcal{T}^S_\Delta$ mesh is shown, together with the cell τ^S_k, its neighbors $\tau^S_{k,j}$ and the related interfaces $l^S_{k,j}$. Lastly, a generic element e^S_r is depicted.

Let's now introduce the discrete functional spaces for the coupled LS–VT method: as it has a two-fold interpretation it has two distinct discrete spaces. Consider the discretized form of $\boldsymbol{\lambda}^n$: $\boldsymbol{\lambda}^n_\Delta \in \mathbb{V}^S_0$ where

$$\mathbb{V}^S_0 = \{\varphi_\Delta \in L^2(\Omega) : \varphi_\Delta|_{\tau^S_k} \in \mathbb{P}^0\}.$$

Let $\mathbb{V}^S_0$ be piecewise constant and equipped with the base of the character-

istic functions $\{\chi_k\}$ of the cells τ_k^S. Therefore we can expand $\boldsymbol{\lambda}_\Delta^n$ as

$$\boldsymbol{\lambda}_\Delta^n = \sum_{k=1}^{n_p^S} \boldsymbol{\lambda}_k^n \chi_k,$$

where $\boldsymbol{\lambda}_k^n$ are the cell mean values at time step n.
We consider then the first grade interpolant of $\boldsymbol{\lambda}_\Delta^n$, $\boldsymbol{\phi}_\Delta^n \in \mathbb{V}_1^S$, with components $\phi_{i,\Delta}^n$, where $\mathbb{V}_1^S = \{\varphi_\Delta \in \mathbb{C}^1(\Omega) : \varphi|_{e_r^S} \in \mathbb{P}^1\}$. Moreover we assume

$$\boldsymbol{\lambda}_k^n = \boldsymbol{\phi}_k^n, \tag{24}$$

where $\boldsymbol{\phi}_k^n$ are the degrees of freedom of the level set field. Equation (24) creates a relation between the LS and VT representations, making possible to accede to the information related to these methods from the solution of a single equation.
We now recall briefly the method described in [7]. Let's consider the following Finite Volume (FV) scheme

$$\boldsymbol{\lambda}_k^{n+1} = \left(1 + \sum_{j=1}^{|\mathbb{I}_k^S|} \nu_{k,j}^n\right) \boldsymbol{\lambda}_k^n - \sum_{j=1}^{|\mathbb{I}_k^S|} \mathbf{F}_{k,j}^n,$$

where $\nu_{k,j}^n$ are the interfaces Courant numbers

$$\nu_{k,j}^n = \frac{\Delta t^n}{|\tau_k^S|} \int_{l_{k,j}^S} \overrightarrow{u}_\Delta^n \cdot \widehat{n}.$$

Here $\mathbf{F}_{k,j}^n = \nu_{k,j}^n \Phi(\widehat{\boldsymbol{\lambda}}_{k,j}^n, \widehat{\boldsymbol{\lambda}}_{k_j,\overline{j}}^n)$ are the interface fluxes, and

$$\Phi(\widehat{\boldsymbol{\lambda}}_{k,j}^n, \widehat{\boldsymbol{\lambda}}_{k_j,\overline{j}}^n) = \begin{cases} \widehat{\boldsymbol{\lambda}}_{k,j}^n & \text{if } \nu_{k,j}^n \geq 0, \\ \widehat{\boldsymbol{\lambda}}_{k_j,\overline{j}}^n & \text{if } \nu_{k,j}^n < 0, \end{cases}$$

is the upwind function, where $\widehat{\boldsymbol{\lambda}}_{k,j}^n$ and $\widehat{\boldsymbol{\lambda}}_{k_j,\overline{j}}^n$ are two suitable approximations of the composition near the interface $l_{k,j}^S$ from inside τ_k^S and $\tau_{k_j}^S$ respectively. In other words, the last three equations define a FV scheme in which the interface fluxes are computed by means of an appropriate interface state $\widehat{\boldsymbol{\lambda}}_{k,j}^n$, which is in general different from the cell mean value considered, for instance, in the Godunov method [10]. Our algorithm resembles more closely a MUSCL [1] or a Discontinuous Galerkin method [3] but differs from the latter for the definition of the interface states and the flux limiter. In fact we introduce the following definition of the interface states:

$$\widehat{\lambda}_{i,k,j}^n = \lambda_{i,k}^n + \gamma_{i,k,j}^n \Delta\phi_{i,k,j}^n,$$

where $\widehat{\lambda}^n_{i,k,j}, \lambda^n_{i,k}$ are i-th components of $\widehat{\boldsymbol{\lambda}}^n_{k,j}, \boldsymbol{\lambda}^n_k$. Moreover $\Delta\phi^n_{i,k,j}$ is the i-th component of the vector $\Delta\boldsymbol{\phi}^n_{k,j} = \boldsymbol{\phi}^n_{k,j} - \overline{\boldsymbol{\phi}}^n_{k,j}$ with

$$\boldsymbol{\phi}^n_{k,j} = \frac{1}{|l^S_{k,j}|}\int_{l^S_{k,j}} \boldsymbol{\phi}^n_\Delta, \qquad \overline{\boldsymbol{\phi}}^n_k = \frac{1}{|\tau^S_k|}\int_{\tau^S_k} \boldsymbol{\phi}^n_\Delta.$$

Finally $\gamma^n_{i,k,j}$ are the flux limiters and they are computed with the following maximization problem:

$$\begin{cases} J = \max \displaystyle\sum_{i=1}^{n_1} \gamma^n_{i,k,j}, \\ \displaystyle\sum_{i=1}^{n_s} \gamma^n_{i,k,j}\Delta\phi^n_{i,k,j} = 0, \\ 0 \le \gamma^n_{i,k,j} \le \gamma^{n,\max}_{i,k,j}, \end{cases} \tag{25}$$

with

$$\begin{cases} \gamma^{n,\max}_{i,k,j} = \min\left(1, \dfrac{(1+\sum_{j=1}^{|\mathbb{I}^S_k|}\nu^n_{k,j}) - \nu^n_{k,j}|\mathbb{J}_k|}{\nu^n_{k,j}|\mathbb{J}_k|\Delta\phi^n_{i,k,j}}\lambda^n_{i,k}, \dfrac{1-\lambda^n_{i,k}}{\Delta\phi^n_{i,k,j}}\right) & \text{if } \Delta\phi^n_{i,k,j} > 0, \\ \gamma^{n,\max}_{i,k,j} = \min\left(1, -\dfrac{\lambda^n_{i,k}}{\Delta\phi^n_{i,k,j}}\right) & \text{if } \Delta\phi^n_{i,k,j} < 0, \end{cases} \tag{26}$$

where $\mathbb{J}_k$ is the set of the indices of the outflow interfaces $l^S_{k,j}$ of the k-th cell, i.e.

$$\mathbb{J}_k = \left\{ j \in 1, \dots |\mathbb{I}^S_k| : \nu^n_{k,j} \ge 0 \right\}.$$

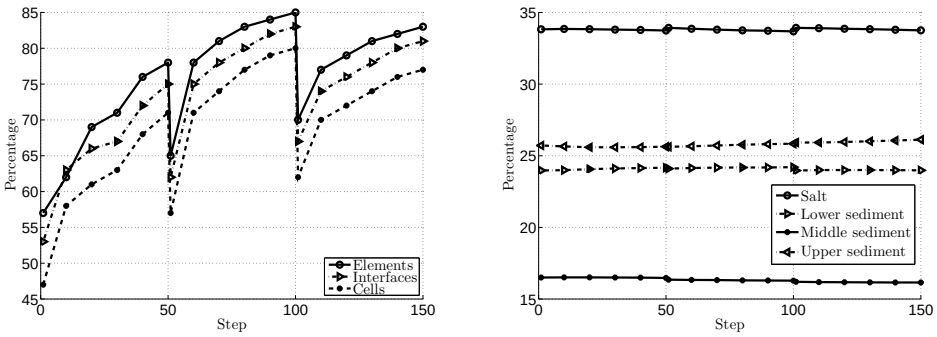
Problems (25) and (26) define implicitly the limiter $\gamma^n_{i,k,j} \in [0,1]$. We point out that if the limiter equals zero, our scheme equals the Godunov method. Now we define some techniques for reducing the computational cost. From equations (25) and (26) it seems that we need to solve a minimization problem on all the interfaces at every time step. Fortunately this is not necessary, as only the cells τ^S_k containing more than two species require an explicit solution of the minimization problem. In fact most of the cells have only one or two species, whereas the remaining cells, that are near to three or more separating interfaces, are a few. Besides, for the two species-cells we can exploit a simplified version of (25):

$$\begin{cases} \gamma^n_{i_1,k,j} &= \min\left(1, \dfrac{\Delta\phi^n_{i_2,k,j}}{\Delta\phi^n_{i_1,k,j}}\gamma^{n,\max}_{i_2,k,j}, \gamma^{n,\max}_{i_1,k,j}\right), \\ \gamma^n_{i_2,k,j} &= \min\left(1, \dfrac{\Delta\phi^n_{i_1,k,j}}{\Delta\phi^n_{i_2,k,j}}\gamma^{n,\max}_{i_1,k,j}, \gamma^{n,\max}_{i_2,k,j}\right), \\ \gamma^n_{i,k,j} &= 1 \qquad \text{with } i \ne i_1, i_2, \end{cases} \tag{27}$$

where i_1 and i_2 are the indices of the two species that fill the cell we are considering. The computational cost could be reduced further; let's consider the following condition:

$$\lambda_k^n = \lambda_{i,k_j}^n \quad \forall j = 1, \ldots, |\mathbb{I}_k^S|. \tag{28}$$

At the beginning of a numerical run most of the cells satisfy this condition as they are far away from the interfaces. Their number tends to decrease during the simulation run while it rapidly increases when the LS function is reinitialized, as we can see in Figure 6. Every cell that satisfies (28) has a trivial solution $\boldsymbol{\lambda}_k^{n+1} = \boldsymbol{\lambda}_k^n$, therefore we have only to update the composition of those cells that does not satisfy (28). In Figure 6 we show some results about the computational cost reduction in a typical geological simulation with four sedimentary layers, that we will discuss in details in the numerical results section. The great majority (more than 98% in current test case)



(a) (b)

Figure 6: In Figure (a) we show the the percentage of the cells, elements, and interfaces inside the active band. The active band is composed by the cells that do not satisfy (28). In this case the grid is coarse therefore a lot of cells lie near an interface. However we can still achieve an average 20% reduction of the computational cost. In Figure (b) we show the percentage volume of the four species. At steps 50 and 100 a reinizialization algorithm is applied.

of the cells has less than two species and the simplified form (27) can be applied. Finally in Figure 6(b) we can appreciate the good conservation properties even during the reinizialization of $\phi_{i,\Delta}^n$ at time steps 50 and 100.

5 Algorithm workflow

In this section we describe briefly the workflow of the algorithm we have developed, which is schematically sketched in Figure 7.

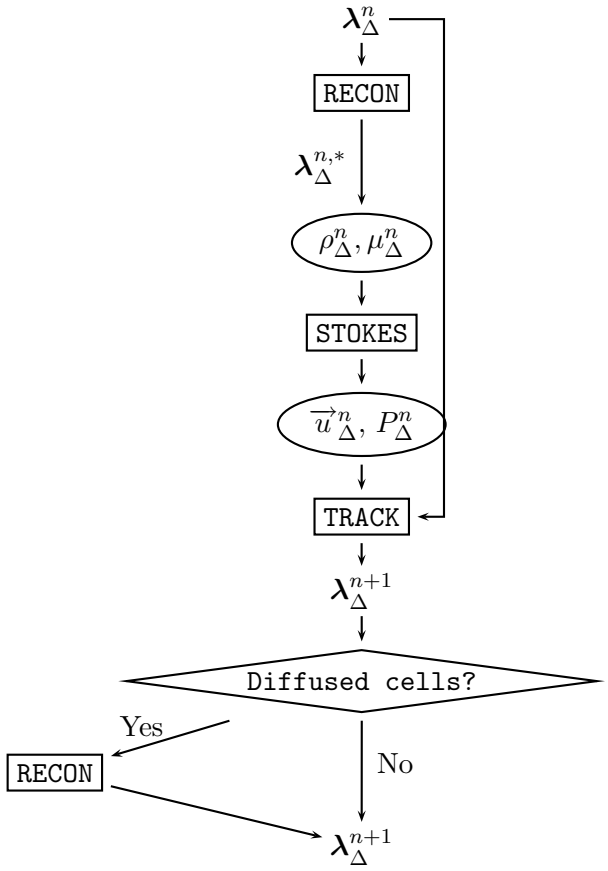


Figure 7: Algorithm workflow.

At the general time step t^n we know the composition vector $\boldsymbol{\lambda}_\Delta^n$, which contains the volume fractions $\lambda_{i,k}^n$ necessary to track the position of the sedimentary layers and in particular of the horizons. From this information, the reconstruction algorithm `RECON` (described below) is applied to all the volume fractions $\lambda_{i,k}^n$, leading to an updated copy $\boldsymbol{\lambda}_\Delta^{n,*}$ of the composition vector. This procedure is important to limit the effects of numerical diffusion: indeed, it reduces the diffusion layer of the properties distribution to the size of a cell, thus enhancing the accuracy of the forthcoming computation of the velocity field. We briefly recall the algorithm `RECON` (described in [7]) in the following:

Algorithm 2. *At the generic time step n, if there is an index $\bar{i}$ such that*

$$\lambda_{\bar{i},k}^n > \frac{1}{2} \text{ and } \lambda_{\bar{i},k_j}^n > \frac{1}{2}, \qquad j = 1, \dots, |\mathbb{I}_k^S|,$$

then set $\lambda^n_{\overline{i},k} = 1$ *and* $\lambda^n_{i,k} = 0$ *with* $i \neq \overline{i}$. *Otherwise maintain the nodal value* $\lambda^n_{i,k}$.

From the new vector $\boldsymbol{\lambda}^{n,*}_\Delta$ we compute the physical quantities ρ^n_Δ and μ^n_Δ through equations (4), and then solve the Stokes problem in the block `STOKES`. This solver generates the pressure and the velocity fields P^n_Δ and $\overrightarrow{u}^n_\Delta$. The latter, together with $\boldsymbol{\lambda}^n_\Delta$, finally leads to the new composition vector $\boldsymbol{\lambda}^{n+1}_\Delta$, by means of the algorithm `TRACK`:

Algorithm 3. *Starting from* $\lambda^n_{i,k}$ *and* $\overrightarrow{u}^n_\Delta$ *compute* $\lambda^{n+1}_{i,k}$ *following these three steps:*

1. *construct the database of the cell to be updated;*
2. *compute the time step satisfying the condition that* $\max_k \max_j \nu^n_{k,j} < \overline{\nu}$ *where* $\overline{\nu}$ *is set equal to* $1/5$. *The number of intermediate time steps necessary to arrive to* t^{n+1} *from* t^n *is also computed;*
3. *compute the interface fluxes, and update the cell partial volumes.*

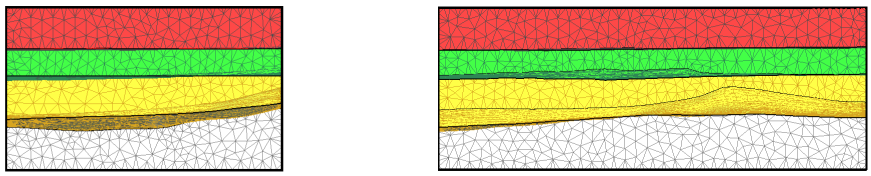
We remark that at this stage, the new composition is tracked starting from the original composition $\boldsymbol{\lambda}^n_\Delta$, to ensure the conservativity of our method. Nevertheless, in the last part of the code, before the beginning of the succeeding time step, we perform a test on the diffusion of the composition, as we need to keep it bounded inside the domain: if the number of diffused cells exceeds an imposed threshold, Algorithm 2 is applied also to the new composition vector $\boldsymbol{\lambda}^{n+1}_\Delta$.

6 Numerical results

The test case we analyze has already been introduced in Figure 1. It consists in a sedimentary basin divided into four layers by three horizons. The basin dimensions are $10.3 \times 15.6 \times 5.8$ km, and the evolution time we consider is equal to 34.35 Mya. Density and viscosity have been taken respectively $2.2 \cdot 10^3$ kg/m^3 and $0.1 \cdot 10^{20}$ Pa s for the salt layer and 2.0–$2.6 \cdot 10^3$ kg/m^3 and 10^{20}–10^{21} Pa s for the overbearing layers, that are reasonable values physically speaking. Among the several simulations run, the one we present here has about 900k unknowns, and requires approximately 4.2Gb of RAM. The computations have been run on an AMD Opteron 8212 Dual-Core 2GHz processor.

With respect to a wide variety of two-dimensional simulations, only a few examples of three-dimensional cases on sedimentary basins already exist in literature, for example Kaus and Podladchikov [9], or Zadeh [28]. The main reason of such a lack of references resides in the dramatic rise of the computational cost that the switch to 3D implies. But then, if compared to a 2D

domain, a three-dimensional model becomes necessary to capture the complete basin dynamics. Anyway, as for the 2D case, the domain boundaries may introduce some undesired effects, worsening the quality of the results. This is mainly due to the imposition of the boundary conditions on a limited domain, that are still to be completely physically comprehended and verified.
In this sense, the simulation of this work not only represents an enrichment of the set of test cases in a three-dimensional domain, but also provides specific features that lead to significant results both from the physical and mathematical point of view. First of all, the implementation of a numerical code for three-dimensional multiflow simulations represents an innovation in the stratified fluid dynamics field, as, with respect to the above-mentioned two-layer 2D simulations, it can handle a model with an arbitrary number of layers. In addition, the model geometry is composed of interfaces representing realistic sedimentary layers, and therefore the perturbation causing the Rayleigh–Taylor instabilities are not imposed *a priori* on a plane surface, but originate from the physical shape of the horizons (see Figure 8).



(a) (b)

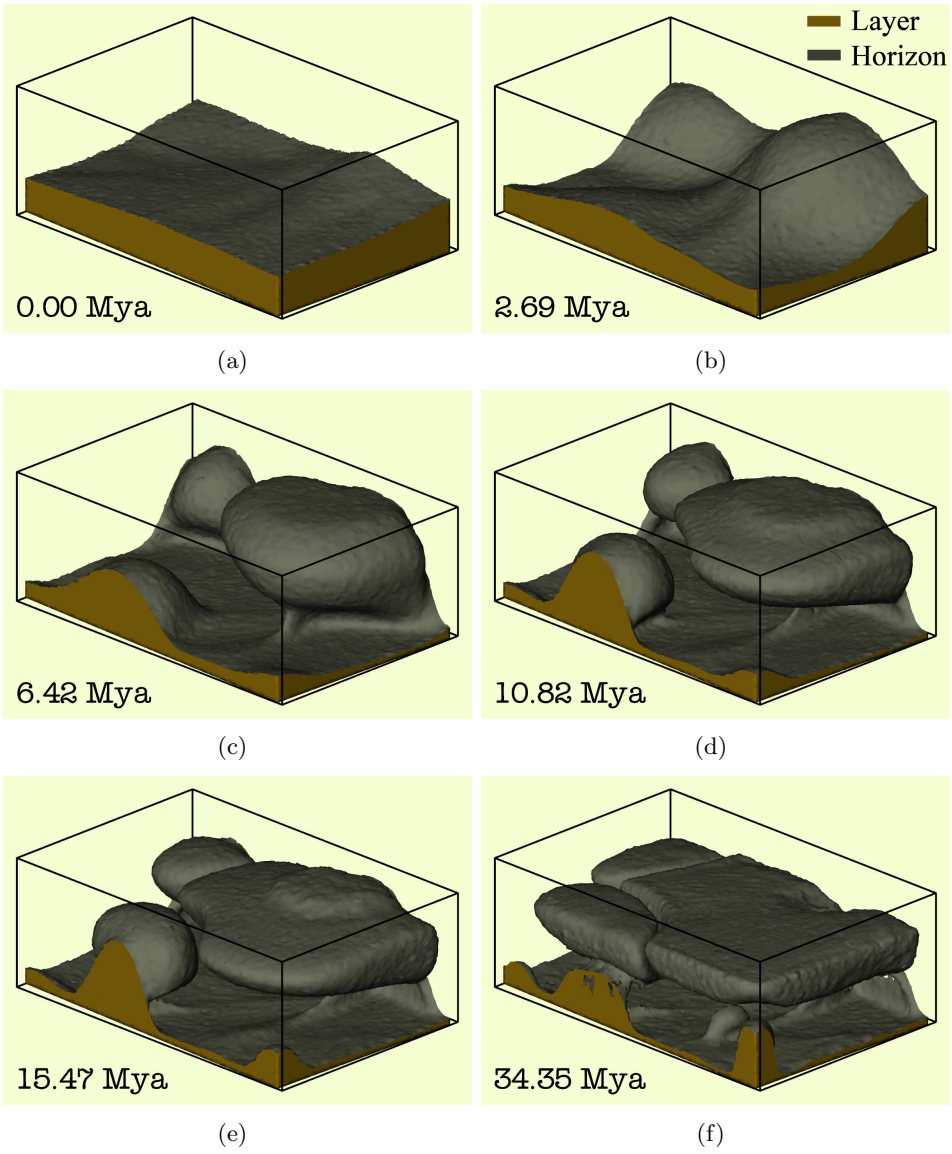
Figure 8: A couple of lateral views of the basin considered in the simulations. Salt is in white, and heavier sediments are in yellow and green. In Figure (b) a bump is visible between the white and the yellow layers, which will lead to a diapir growth as a consequence of the Rayleigh–Taylor instability.

The ease of handling complicated geometries is enhanced by the use of unstructured meshes as discretizing tool. Figure 9 shows subsequent steps of the basin evolution, that leads to the shaping of some salt diapirs.
Secondly, the interface surfaces have been drawn with the innovative tracking algorithm, that is able to reconstruct the horizon positions efficiently, both from the geometric and the computational point of view. In particular, as these figures show, it is able to handle and represent topological changes, and so to simulate correctly and in a fully automatic way phenomena of key importance in the sedimentary basin evolution, such as salt diapir detachments (see Figure 9(e)) and horizon intersections (see Figure 10(b)–10(d)). In the following figures we illustrate the evolution of the surface between the salt and the overburden layers, that ends with the formation of three main diapirs.

As well as the interfaces evolution, the distribution of other quantities of physical interest is computed, such as velocity, pressure and strain fields. As an example, in Figure 11, 12 and 13 we show the distribution of the vertical stress component, the vertical velocity and the streamlines on a sectioning plane. In the first we can recognize three different phases in the basin evolution: the most part of the diapir growth happens in the first 11 Mya, followed by a settlement phase lasting 24 Mya, and here-hence the evolution is almost stationary. In the second figure we can appreciate the streamlines representing the flow motion leading to the formation of the main diapir, at a time step of basin major activity.

7 Conclusions

In this article we presented an innovative numerical technique for the mathematical and physical modeling of stratified salt sedimentary basins. In particular we focused on numerical efficiency and computational time reduction, operating both on the solution of the linear system resolution and on the interface tracking algorithm.
Firstly, we faced the problem related to the resolution of the linear system, which is derived from the FE method adopted to solve the set of governing equation. The huge matrix coming from the high number of degrees of freedom shows a dependence from the viscosity, that may have a large range of variability among the different basin layers. The algorithm we propose exploits this dependence of the external loop to build a convenient preconditioning matrix based on the Schur complement, depending on the viscosity itself, that has provided a sensible gain in the calculation time.
Moreover we presented an evolution of the implicit horizon tracking method applied to a three-dimensional case test, with a number of layers major than two. This algorithm puts together the advantages coming from Volume Tracking and Level Set methods, and can both accomplish the reconstruction of the interfaces movement with good accuracy, and solve the problems of the topological changes, extremely significant in a three-dimensional domain. In particular, we focused on an efficient way to compute a simplified solution of the method limiter, that, coupled with the exploitation of the localized distribution of the involved cells, leads to a dramatic reduction of the global computational time.
Finally we applied these techniques to a test case, and the results we presented state the efficiency in simulating the evolution of a realistic sedimentary basin in a three-dimensional domain.
Further development will concern the enrichment of the model and mathematical improvements. Interesting tasks will be the extension of the geometry representation to include faults and sedimentation and compaction processes, and the implementation of more sophisticated physical models



(a) (b) (c) (d) (e) (f)

Figure 9: Progressive evolution of salt diapirs: in about 34 Mya the growth of the three diapirs is almost complete.

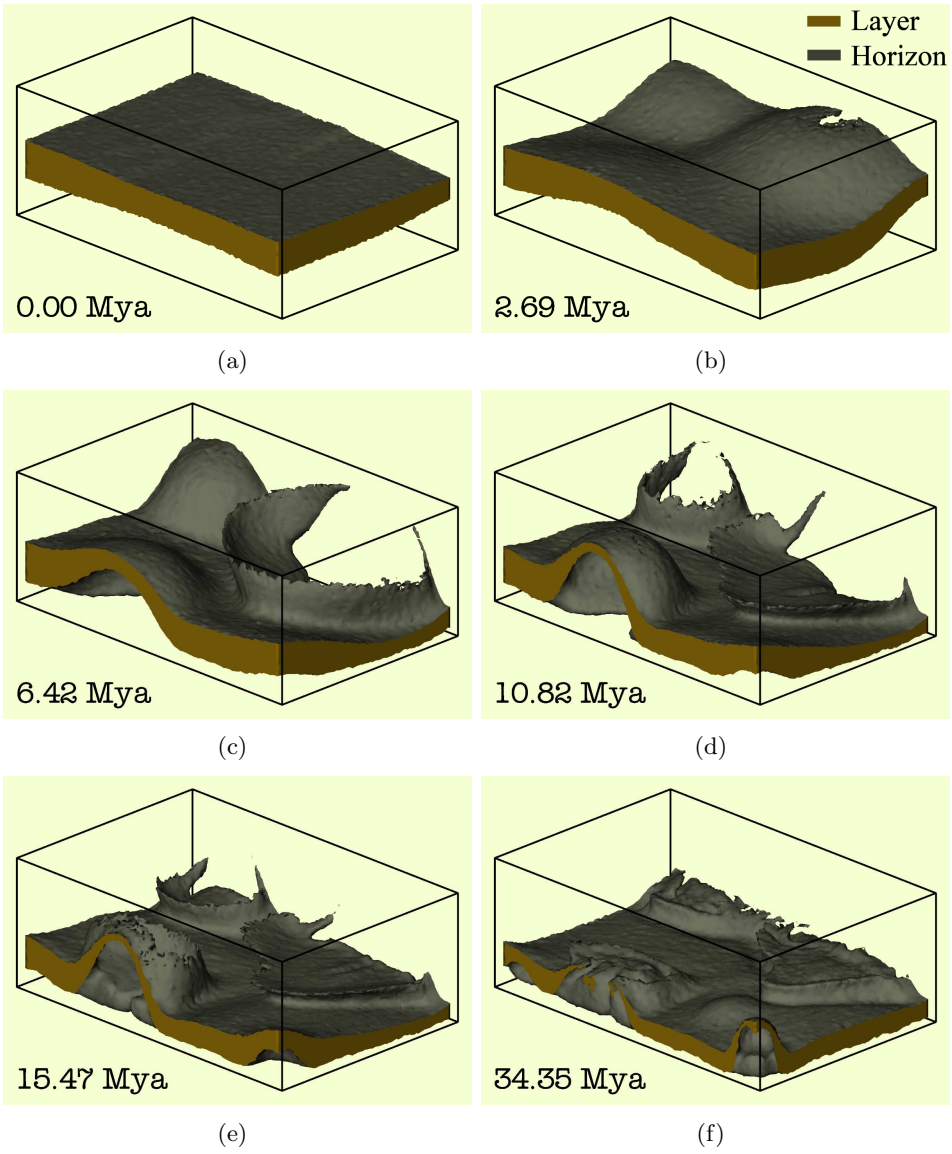


Figure 10: Progressive evolution of the lower sediment: the salt rise perforates the overbearing sediment in three regions.

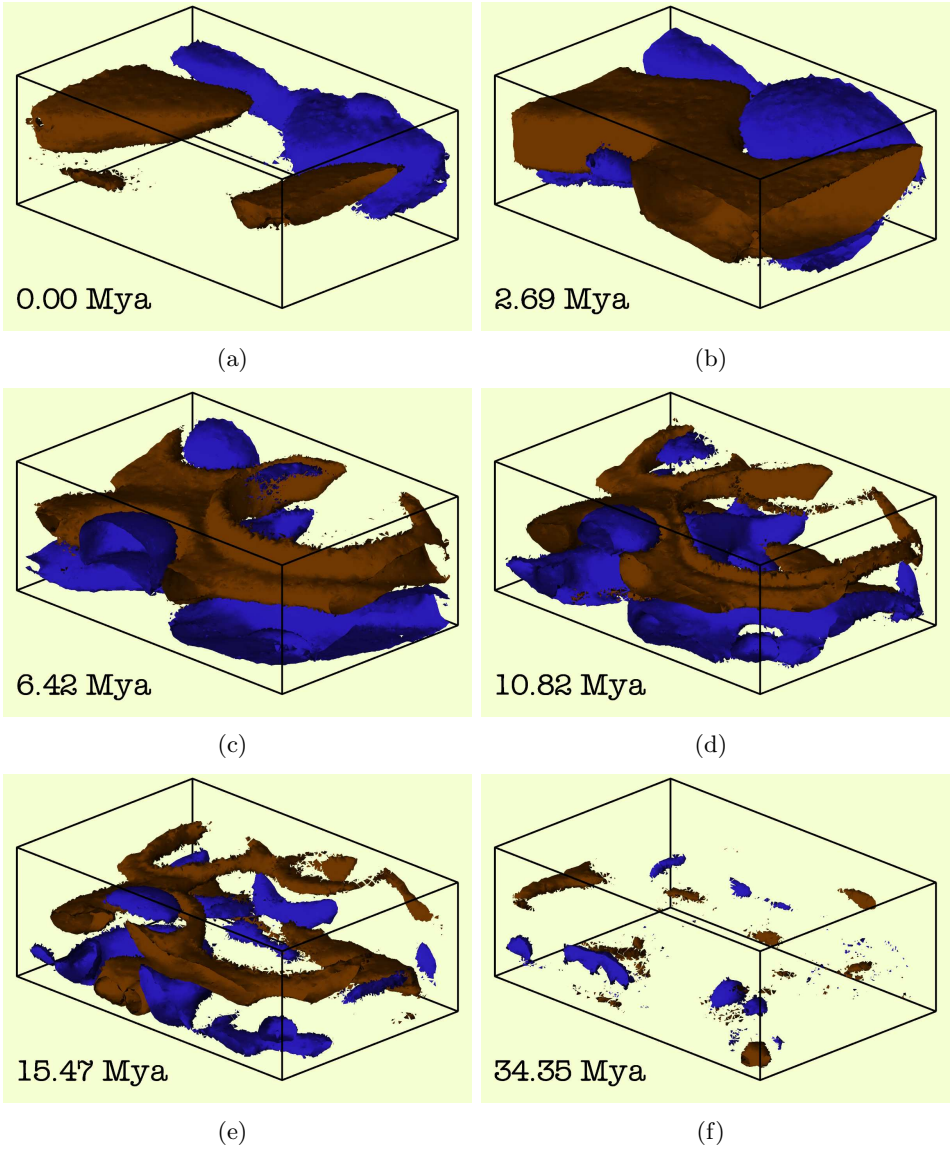
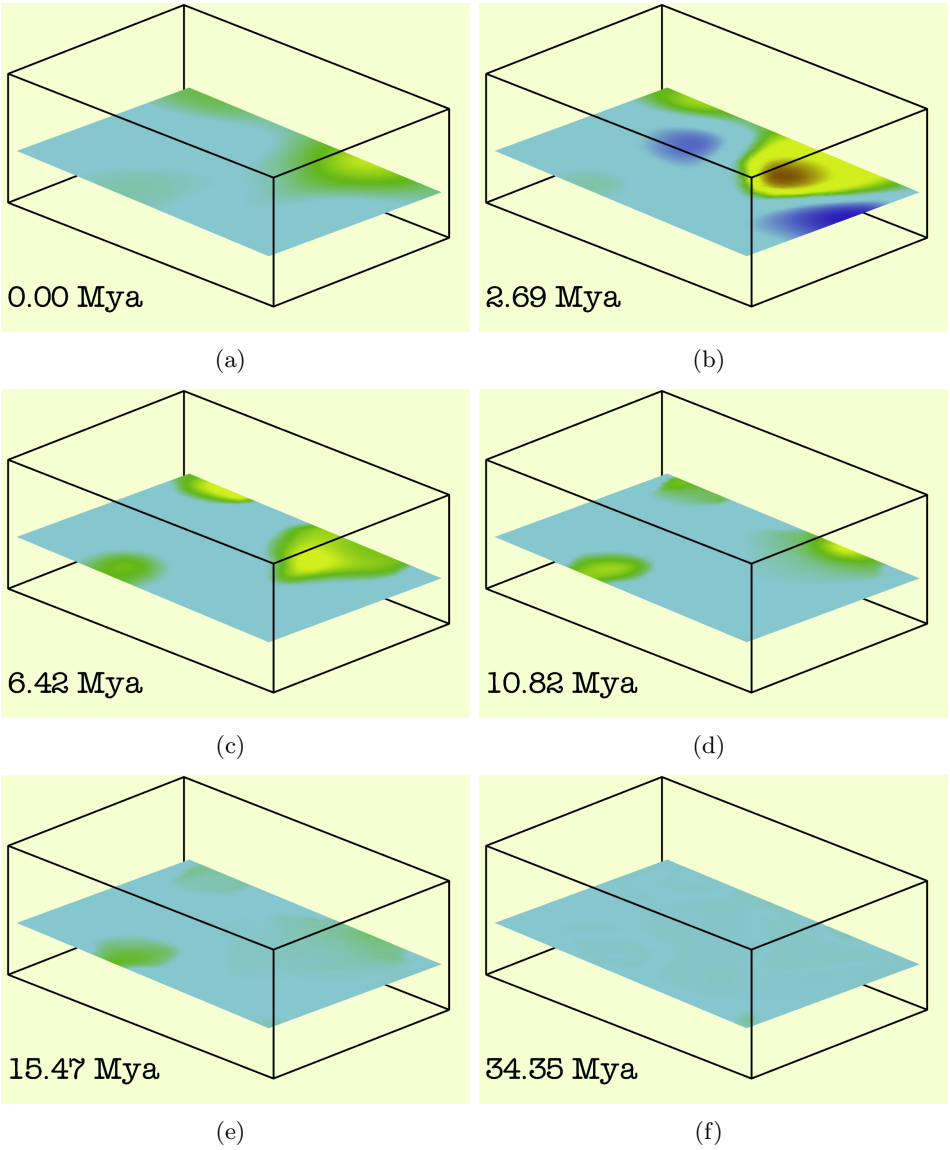
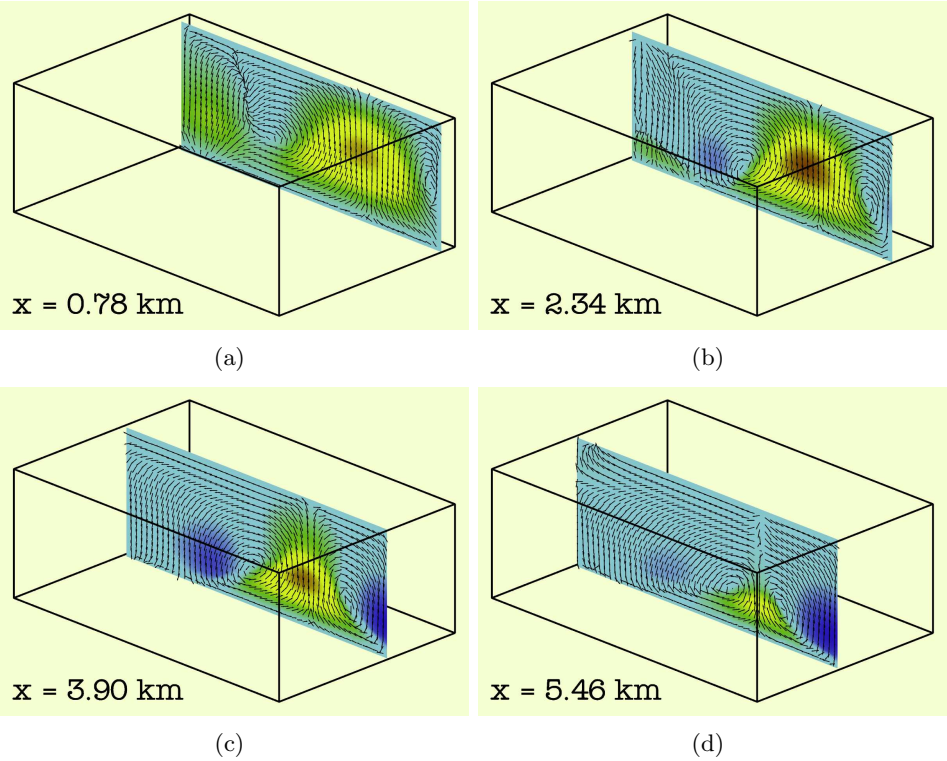


Figure 11: Two isosurfaces showing the distribution of the vertical stress component $\bar{\bar{\sigma}}_{zz}$. (Red: 2.0 MPa, Blue: -2.0 Mpa)



(a) (b) (c) (d) (e) (f)

Figure 12: A series of slabs of vertical velocity V_3, at $x_3 = 2.18$ km from the bottom of the basin: the most part of the development of the diapirs happens in the first 11 Mya, while in the last 24 Mya the flows become gradually stationary. (Red: 1.64 km/Mya, Cyan: 0.0 km/Mya, Blue: -0.82 km/Mya)



(a) (b) (c) (d)

Figure 13: A series of slabs of vertical velocity V_3 and stream-lines at $t = 2.69$ Mya: two main vortices are visible near the biggest diapir. (Red: 1.64 km/Mya, Cyan: 0.0 km/Mya, Blue: -0.82 km/Mya)

such as non Newtonian rheologies.

Acknowledgments

The authors wish to acknowledge the support of the Italian MIUR (grant PRIN 2007) and G. Scrofani and P. Ruffo for having supported such an interesting research project. They also wish to thank L. Formaggia, E. Miglio and F. Nobile from MOX for their precious advices and suggestions.

References

[1] P. Batten, C. Lamberta and D.M. Causon. Positively conservative high resolution convection scheme for unstructured elements. Int. J. Numer. Methods Eng, pp. 1821–1838, vol. 39, 1996.

[2] S. Cloetingh and Y. Podladchikov. Perspectives in tectonic modeling. Tectonophysics, pp. 169–173, vol. 320, 2000.

[3] B. Cockburn and C.W. Shu. TVB Runge Kutta local projection discontinuous Galerkin finite element method for conservation laws 2: general framework. Math of Comp, pp. 411–435, vol. 52, 1989.

[4] R.G. Duran and M.A. Muschietti. The Korn inequality for Jones domains. Electron. J. of Differential equations, pp. 1–10, vol. 127, 2004.

[5] C. Elman, D.J. Silvester and A.J. Wathen. Finite Elements and Fast Iterative Solvers. Oxford University Press, 2005.

[6] A. Ern and J.L. Guermond. Theory and Practice of Finite Elements. Springer, 2000.

[7] L. Formaggia and A. Villa. Implicit tracking for multi-fluid simulations. MOX Report No. 11, Politecnico di Milano, 2009. website: `http://mox.polimi.it/it/progetti/pubblicazioni/quad.php?en=en`.

[8] P. Fullsack. An arbitrary lagrangian eulerian formulation for creeping flows and its application in tectonic models. Geophys J. Int. pp. 1–23, vol. 120, 1995.

[9] B.J.P. Kaus and Y. Podladchikov. Forward and reverse modeling of the three-dimensional viscous Rayleigh–Taylor instability. Geophys Res. Lett, pp. 1095–1098, vol. 28, 2001.

[10] R.J. LeVeque. Numerical Methods for Conservation Laws. Birkhauser, 1999.

[11] P. Massimi. Perspective in computational salt tectonics. Ph.D. Thesis, Politecnico di Milano, 2006.

[12] P. Massimi, A. Quarteroni, F. Saleri and G. Scrofani. Modeling of salt tectonics. Comp. Meth. Appl. Mech. Eng., pp. 281–293, vol. 197, 2007.

[13] P. Massimi, A. Quarteroni and G. Scrofani. An adaptive finite element method for modeling salt diapirism. M3AS vol. 16, no. 4, 2006.

[14] L. Moresi, F. Dufour and H.B. Muhlhaus. A Lagrangian integration point finite element method for large deformation modeling of viscoelastic geomaterials. Journal of Computational Physics, pp. 476–497, vol. 184, 2003.

[15] C. O'Neill, L. Moresi, D. Muller, R. Albert and F. Dufour. Ellipsis 3D. A particle-in-cell finite-element hybrid code for modelling mantle convection and lithosperic deformation. Computers and Geosciences, pp. 1769–1779, 2006.

[16] S. Osher and R.P. Fedkiw. Level Set Methods and Dynamic Implicit Surfaces. Springer, 2003.

[17] D. Peric and A.J.L. Crook. Computational strategies for predictive geology with reference to salt tectonics. Comput. Methods Appl. Mech. Engrg, pp. 5195–5222, vol. 193, 2004.

[18] A. Quarteroni and A. Valli. Numerical Approximation of Partial Differential Equations Second Edition. Springer–Verlag, 1997.

[19] S.J. Ruuth. A diffusion-generated approach to multiphase motion. J Comp. Phys. pp. 166–192, vol. 145, 1998.

[20] A. Scotti and A. Villa. Predictive numerical models of basin evolution and petroleum generation. To appear on AIMI III.

[21] G. Scrofani. Numerical Basins Modeling and Tectonics. Ph.D. Thesis, Politecnico di Milano, 2007. website: `http://mox.polimi.it/it/progetti/pubblicazioni/tesi/scrofani.pdf`

[22] A. Sethian. Level Set Methods and Fast Marching Methods. Cambridge University Press, 1999.

[23] SOPALE: a finite-element computer code for the computation of visco-plastic creeping flows with applications to tectonics. website: `http://geodynamics.oceanography.dal.ca/sopaledoc.html`

[24] L. Tan and N. Zabaras. A level set simulation of dendritic solidification of multi-component alloy. J Comp Phys, pp. 9–40, vol. 221, 2007.

[25] D.L. Turcotte and G. Schubert. Geodynamics, Cambridge University Press, 2001.

[26] A.T. Ismail-Zadeh, I.A. Tsepelev, C. Talbot and P. Oster. A numerical method and parallel algorithm for three-dimensional modeling of salt diapirism. Problems in Dynamics and Seismicity of the Earth, pp. 62–76, vol. 31, GEOS, Moscow, 2000.

[27] A.T. Ismail-Zadeh, C.J. Talbot and Yu.A. Volozh. Dynamic restoration of profiles across diapiric salt structures: numerical approach and its applications. Tectonophysics, pp. 21–36, vol. 337, 2001.

[28] A.T. Ismail-Zadeh, I.A. Tsepelev, C. Talbot and A.I. Korotkii. Three-dimensional forward and backward modelling of diapirism: Numerical approach and its applicability to the evolution of salt structures in the Pricaspian basin. Tectonophysics, pp. 81–103, vol. 387, 2004.

[29] S. Zaleski and P. Julien. Numerical simulation of Rayleigh–Taylor instability for single and multiple salt diapirs. Tecnophysics, pp. 55–69, vol. 206, 1991.

[30] S. Zlotnik and P. Diez. Hierarchical X-FEM for n-phase flow ($n > 2$). Computer Methods in Applied Mechanics and Engineering, pp. 30–32, vol. 198, 2009.

MOX Technical Reports, last issues

Dipartimento di Matematica "F. Brioschi",
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

26/2009 M. Longoni, A.C.I. Malossi, A. Villa:
A robust and efficient conservative technique for simulating three-dimensional sedimentary basins dynamics

25/2009 P.E. Farrell, S. Micheletti, S. Perotto:
An anisotropic Zienkiewicz-Zhu a posteriori error estimator for 3D applications

24/2009 F. Di Maio, P. Secchi, S. Vantini, E. Zio:
Optimized Fuzzy C-Means Clustering and Functional Principal Components for Post-Processing Dynamic Scenarios in the Reliability Analysis of a Nuclear System

23/2009 L. Gerardo Giorda, F. Nobile, C. Vergara:
Analysis and optimization of Robin-Robin partitioned procedures in Fluid-Structure Interaction Problems

22/2009 L. Formaggia, S. Minisini, P. Zunino:
Modeling erosion controlled drug release and transport phenomena in the arterial tissue

21/2009 L. Bonaventura, S. Castruccio, L.M. Sangalli:
A Bayesian approach to geostatistical interpolation with flexible variogram models

20/2009 N.Accoto, T. Rydén, P. Secchi:
Bayesian hidden Markov models for performance-based regulation of continuity of electricity supply

19/2009 L. Formaggia, A.Mola, N. Parolini, M. Pischiutta:
A three-dimensional model for the dynamics and hydrodynamics of rowing boats

18/2009 J. Silva Soares, P. Zunino:
A mathematical model for water uptake, degradation, erosion, and drug release from degradable polydisperse polymeric networks

17/2009 M. Longoni, C. Magistroni, P. Ruffo, G. Scrofani:
3D Inverse and Direct Structural Modeling Workflow