Politecnico di Milano
Scuola di Ingegneria Industriale e dell'Informazione
Corso di Laurea Magistrale in Ingegneria Aeronautica

# CFD analysis of the influence of the Mori-Torbole tunnel on lake Garda

| | |
|---|---|
| **Tesi di** | Marco Gambarini (mat. 913569) |
| **Relatore** | Prof. Edie Miglio |

**Anno Accademico 2019 – 2020**

## Abstract

The discharge of water from the Adige-Garda tunnel into lake Garda was simulated using a 3D hydrodynamic code. A set of global simulations were run to assess tracer dispersion from the inflow and the corresponding thermal effects. The obtained results are in good agreement with SAR images. An analytical model for the propagation speed of an intrusion is presented. Its predictions are compared with simulations of the region near the inflow point on a fine grid, which enabled a closer analysis of the flow features. Local simulations were also run considering river Sarca's inflow to assess its impact on tracer dispersion from the Adige-Garda tunnel. The mathematical and numerical models used are described thoroughly and modelling hypotheses and issues are discussed together with possible further developments.

## Sommario

L'influsso di acqua dalla galleria Adige-Garda nel lago di Garda è stato simulato utilizzando un codice idrodinamico 3D. Attraverso simulazioni globali è stata analizzata la dispersione di tracciante dalla galleria, insieme ai corrispondenti effetti termici. I risultati mostrano un buon accordo con le immagini SAR disponibili. È stato inoltre introdotto un modello analitico per la velocità di propagazione di un'intrusione. I risultati del modello sono stati confrontati con quelli di simulazioni locali, effettuate nella regione attorno al punto di influsso su una griglia di calcolo fine; queste simulazioni permettono un'analisi più dettagliata della fisica del flusso. Nelle simulazioni locali è stato considerato anche l'influsso del fiume Sarca per verificarne l'impatto sulla dispersione di tracciante dalla galleria Adige-Garda. La tesi comprende una descrizione dettagliata dei modelli matematici e numerici utilizzati; vengono discusse le ipotesi e criticità modellistiche, insieme ai possibili sviluppi futuri.

# Preface

This thesis work stems from two interests I have had for a long time: fluid dynamics and environmental problems. It consists of numerical experiments on a lake, a system much different from the ones usually considered in aerodynamics; and since I am a student of aeronautical engineering, I had to change my point of view to try to understand what are the aspects worth focusing on and what to expect from the results. This is reflected in the structure and the content of the thesis: the first, and prominent, part is a review of models, methods and existing research, and a collection of relatively simple simulations that I ran to build confidence in my understanding of the physics and of the modelling choices. The physics differ from the typical problems of aerodynamics with regard to scales, lengths being larger and velocities lower, and the flow is governed by an interplay of atmospheric forcing, internal density stratification and geometry. As in aerodynamics, turbulence plays a major role and introduces large modelling difficulties and uncertainties. Turbulence in environmental flows is influenced by stratification and is typically patchy and intermittent; RANS models derived by slight modifications of the ones used in aerodynamics and applied on very coarse grids (compared to aerodynamic standards) have trouble reproducing this behaviour correctly and can sometimes yield results that are very hard to interpret. The choice of the model is critical, different models giving sometimes quite different results. The lack of field measurements as validation data means that at present the most reasonable solution is to resort to standard choices of the limnology/oceanography community. This can be frustrating and discouraging. What is the validity of the results? We should not expect extremely accurate quantitative results as the ones expected for lift and drag coefficients for an airfoil; we should rather accept uncertainty on the numerical results, but still expect to be able to gain a valuable qualitative and order-of-magnitude picture of the phenomena we are simulating. For instance, when simulating the thermal impact of inflowing water into the lake, we try to get an idea of which areas will be most affected, and to about which extent, rather than focusing on pointwise, instantaneous values. We will see that order-of-magnitude considerations on the velocity fields will allow to clarify what the main driving action is for the dispersion of inflowing water.

Due to the period in which this thesis was done, I spent almost all of the time working alone. Nonetheless, there are many people I want to acknowledge for their help. Prof. Edie Miglio proposed the problem and provided guidance throughout the work. Dr. Marina Amadori, Dr. Giuseppe Bilotta, Dr. Francesca de Santi, Prof. Marco Toffolon and Prof. Lorenzo Valdettaro gave helpful suggestions and observations; Francesca de Santi also provided and processed satellite data. Giuseppe Aloe and Luca Paglieri at MOX were very kind and helpful in giving support on the use of the cluster. Matteo Savatteri helped with software installation and Linux system administration advice. Many helpful comments and discussions were shared with colleagues, especially the people in BstTeam, and with users of the Telemac forum. Friends and family provided much needed distraction - even though I can be quite hard to distract.

# Contents

# List of Figures

# List of Tables

# Introduction

The aim of this work is to conduct numerical hydrodynamic simulations of lake Garda in order to evaluate the effects of wind and inflows on thermal and mixing properties, with particular emphasis on the discharge event from the Adige-Garda tunnel occurred in October 2018.

Lake Garda is a subalpine lake in northern Italy; it is the largest in Italy, with an area of 368 km$^2$, and one of the deepest, with maximum depth of 350 m [49]. It has one main tributary, river Sarca, which enters the lake at its northernmost extremity, and one outflow, river Mincio, exiting at the southeast. A tunnel was built in mid 20th century (started in 1939, completed in 1959) to discharge into the lake water from river Adige, which runs in a valley parallel to the longitudinal direction of the lake, to mitigate floods. The lake can be divided into two main parts based on the bottom morphology: the northern trunk is elongated, with steep sides and a flat bottom, similarly to other subalpine lakes such as lakes Como and Maggiore; the greatest depths are reached here. In contrast, the southeastearn basin is shallow. The two are separated by a submerged ridge than runs from the Sirmione peninsula to punta san Vigilio (see the map in Figure 1.2).

Simulations are run using the numerical hydrodynamic code Telemac3D[1], solving the Reynolds-averaged Navier-Stokes equations discretized with the finite element method.

Chapter 1 is devoted to giving a general introduction on limnology, the problems it deals with and the ones among them that are relevant for lake Garda. A review of research papers on the lake follows.

Chapter 2 presents the governing equations of the problem, the hypotheses they are based on and some possible modelling choices. A mathematical model for intrusions is also derived.

Chapter 3 shows how the equations presented in chapter 2 are solved, by discretizing in space (building a mesh), and in time, with a proper treatment for each of the terms in the equations (using the fractional step method). The way each step is solved is then discussed by recalling the main ideas and properties of the finite element method and of multidimensional upwind residual distribution (MURD) schemes.

Chapter 4 is a summary of some simple but relevant test cases that were devised in order to experiment with the model setup and parameters, dealing with the most important physical phenomena and modelling difficulties one at a time before starting

[1] http://opentelemac.org/

the main simulations.

Chapter 5 describes the setup and results of discharge simulations. The effects of wind-induced circulations, stratification and interaction with the Sarca river inflow on the distribution of water discharged by the spillway are explored first on the entire lake, and then at a local level using a finer mesh.

Chapter 6 contains a summary of the main results. The uncertainties and some possible developments of the present analysis are discussed.

# Chapter 1

# Review

## 1.1 Limnology

Limnology, the study of lakes, is motivated by biological interest and industrial necessity. Understanding how life in lakes is affected by climate, polluted inflows and circulation patterns helps in preserving it and planning corrective actions when needed. The study of lake fluid dynamics is commonly called physical limnology [26] and it is of support to chemical and biological studies; as a matter of fact, the coupling between fluid dynamics and biology is mainly one-way: the first influences the second. For example, hydrodynamics determine the motion of plankton, organisms that are unable to move autonomously and thus drift transported by currents (plankton literally means ≪wanderer≫, from the greek) and that are the base of the food chain for fish. In particular, phytoplankta depend on light to realize photosyntesis, and light is available only close to the surface: so, vertical flows determine how much exposure to light such organisms can get, how much nutrients are available in the different areas, and ultimately their population dynamics. This means that a knowledge of flow patterns and of the way they are influenced by external agents, mainly wind, air temperature and inflows, is important not only to determine the fate of pollutants released in the lake in the context of water quality studies, but also to understand how nutrients are distributed in the lake, which affects the health of its population. In particular, in lake hydrodynamics mixing properties are of primary importance and they are typically strongly influenced by stratification.

Stratification is the vertical variation of density and temperature in a fluid domain. Stratifications can be classified as stable or unstable, depending on the stability of the equilibrium of particles in the domain. If a particle displaced vertically from its equilibrium position tends to return to it, the stratification is stable; if the particle trajectory diverges, the stratification is unstable; if there is no restoring action, the stratification is neutral. In a stable stratification, vertical motions are inhibited. A simple stability condition is that density must decrease from the bottom to the free surface; for water above 4°C, this implies that temperature must increase. During summer, a typical stratification is as depicted in Figure 1.1. Heat is mainly exchanged by conduction/convection

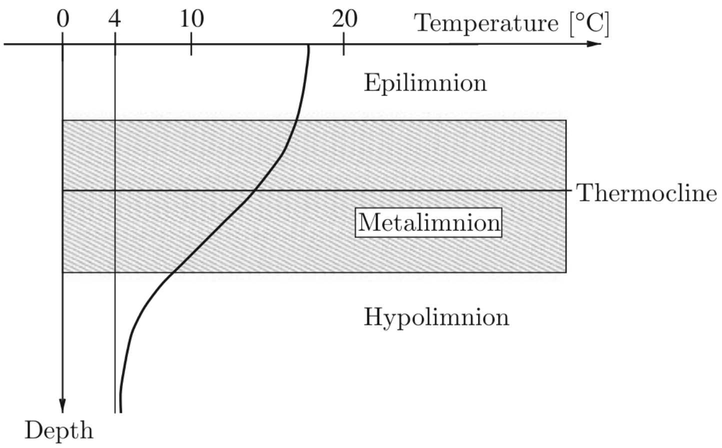


Figure 1.1: A typical summer stratification (from [26])

with the air above and by radiation. In summer, the upper layers get warmer than the deep ones, which remain close to constant temperature for the whole year. This is a stable stratification. Some terminology: the surface of maximum vertical temperature gradient is called thermocline and it separates the hypolimnion (deep lake) from the epilimnion (surface layer); the layer around the thermocline is called metalimnion. During autumn and winter, cold air makes the surface temperature decrease, so that the stratification either becomes neutral or even unstable if temperatures are low enough.

Mixing requires energy, and kinetic energy is provided to the water in the lake by wind: when wind blows over the surface, a shear layer is created in the lake. In a shear layer, turbulence appears[1]; turbulence is affected by the stratification: as will be discussed in section 2.5, in a stable stratification turbulent fluctuations are damped. So, the degree of mixing in the lake depends on the interplay between wind, that provides kinetic energy which drives turbulent mixing, and heat exchange, than can create stable or unstable stratifications. Wind also induces internal waves, that can have amplitudes much larger than surface waves, reaching tens of meters in lakes and even more in the ocean. Internal waves interact with each other and with boundaries in complex, and not yet completely understood, ways; it is believed that a large part in turbulent dissipation and mixing is played by waves breaking over complex topographies, in particular regions of steep bottom [29], [60].

Moreover, when a wind-generated current reaches the coast, water experiences downwelling, that is, it sinks along the sloping boundary; the depth to which it can sink depends on buoyancy forces: if the stratification is stable, water coming from the surface is less dense then the one it is surrounded by while it descends, so it cannot sink too deep. On the other hand, if the stratification is neutral or unstable, water can sink to the bottom. Once it stops sinking, it starts spreading horizontally; a numerical

[1] The turbulent kinetic energy production rate is proportional to the mean flow velocity gradient, i.e. the mean shear.

experiment showing this effect is presented in section 4.2. This, again, is an effect of global circulation and convective mixing, and it is the same effect experienced by water inflowing from tributaries, which can reach different depths depending on its temperature. An additional effect is the differential cooling of water in lakes experienced during autumn and winter: shallow areas are cooled more quickly and effectively than deep ones. Water from such areas, then, will start sinking, as described above.

In addition to the direct mixing effect induced by the wind and to the influence of topography, water can penetrate to depth in presence of hydrological fronts [26, p. 271], i.e. lines along which surface velocities converge and thus, for mass conservation, water must move downwards. They can be generated, again, by the effect of inflows, topography, or wind patterns.

Mixing is such an important phenomenon in lakes (for example, for oxygenation and nutrient transport) that a classification of lakes based on it exists:

- Holomictic lakes mix completely (down to the bottom). If they mix once a year, they are called monomictic; if they mix many times a year, they are called polymictic; if they mix once every several years, they are called oligomictic.

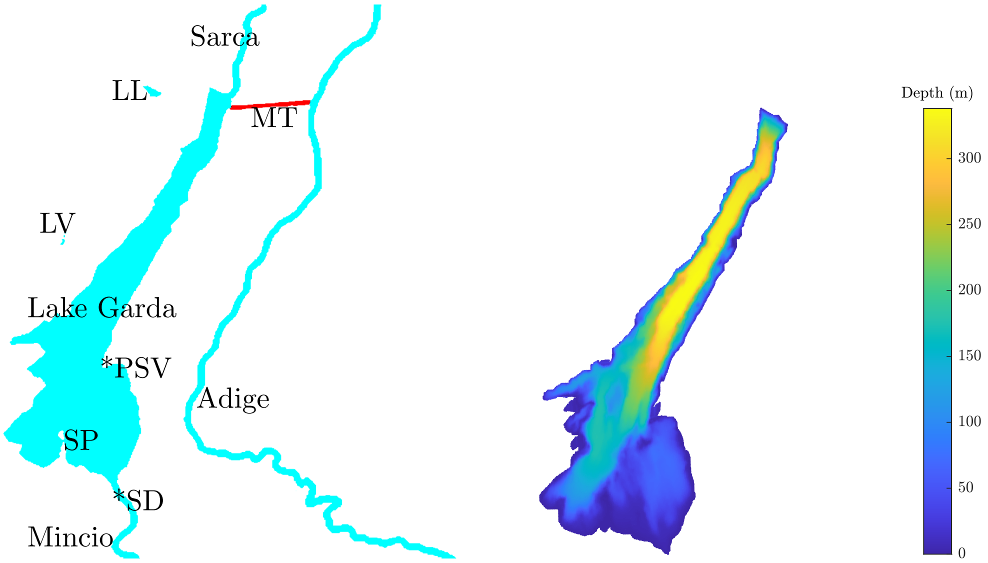
- Meromictic lakes mix only down to a limited depth.

- Amictic lakes never mix (this is very rare, and it happens when there is no wind forcing at all, for example for lakes buried below ice in Antarctica).

The mixing character of a lake is influenced by latitude, elevation, depth and exposition to wind.

The water budget in a lake is a critical issue for agriculture and energy production: the outflow of a lake is often used for irrigation of large regions; hydroelectrical plants may exist both along the inflows (e.g. high elevation artificial lakes) and along the outflows. Moreover, thermoelectrical and industrial plants may need substantial amounts of water for cooling. On the one hand, the presence of dams allows controlling the water level in the lake and the outflow rate; on the other hand, dependable energy production requires steady or at least predictable flow. The necessities of dependable outflow rates and keeping the lake water level within certain limits for ecological reasons, to allow navigation and to avoid flooding lakeside towns, may be in conflict, requiring careful planning especially in view of climate change, with severe events of flooding and drought becoming more frequent.

These matters are all present in the case of lake Garda: the level of the lake is controlled by a dam situated in Salionze, on the outflowing river Mincio (see Figure 1.2). Levels and outflow rates are prescribed for different periods of the year [1]. Also the inflow is partially controlled because of the presence of hydroelectrical plants on lakes Ledro, Valvestino and Molveno, the latter of which flows into river Sarca. An additional, sporadic inflow is the Mori-Torbole tunnel [14], built between 1939 and 1959 to discharge water from river Adige to lake Garda, in order to prevent floods along the river during intense rains; its discharge point is shown in Figure 1.3. It has since been used for this purpose in 1960, 1965, 1966, 1976, 1980, 1981, 1983, 2000, 2002 and



(a) (b)

Figure 1.2: (a) lake Garda with main inflows and outflows. LL: lake Ledro, MT: Mori-Torbole tunnel, LV: lake Valvestino, SD: Salionze dam, PSV: punta san Vigilio, SP: Sirmione peninsula. Base map: Google Maps; edited by the author. (b) available bathymetry, depths in meters; the bathymetry extends slightly beyond the boundary of the lake, in order to enable flooding simulations

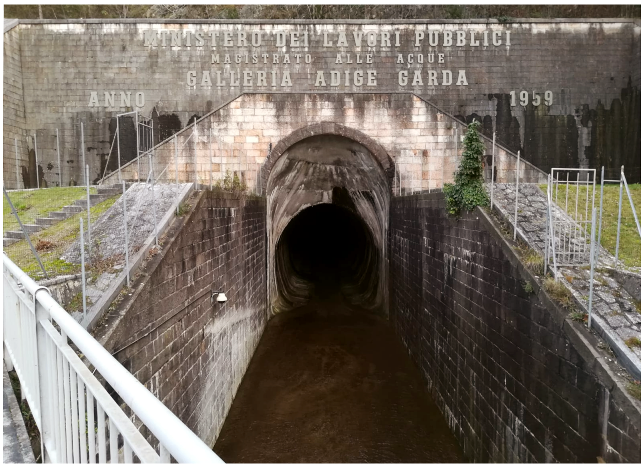


Figure 1.3: Discharge point of the Mori-Torbole tunnel. Picture taken by the author

2018 [3], in addition to sporadic maintenance operations. Each utilization of the tunnel is met with criticism because of the large mass of cold, silty water discharged into the lake.

### 1.1.1 CFD models of lakes

Different models are available for numerical simulations of lakes, with different levels of detail and complexity depending on the application.

One-dimensional (vertical) models are implemented in weather forecast codes to account for the role of lakes as heat reservoirs. This action is important both on timescales of the order of a day, generating breezes, and of the order of seasons, accumulating heat during the summer by forming a strong stable stratification and releasing it during autumn. A comparison of some models is presented in [54]. 1D models have also been used to forecast the impact of climate change on the mixing properties of lakes, as done in the paper [21] for lake Maggiore.

Two-dimensional models are typically based on the shallow water equations and can be used for shallow basins with limited depth variations.

Three-dimensional models, based on the incompressible Navier-Stokes equations and using different modelling choices, are used for shorter simulations (with few exceptions such as [12]) to identify currents and assess the impact of possible sources of pollution, usually by performing scenario simulations.

## 1.2 Existing research on lake Garda

Research studies about the hydrothermodynamics of lake Garda can be divided into two categories: long-term ones, spanning multiple years and examining global patterns, and studies on scenarios, analyzing data from measurements and simulations over few days or weeks with a focus on local features. Some of these studies are interdisciplinary, focusing especially on biological processes that are influenced by fluid mechanics and thermodynamics.

### 1.2.1 Long-term, statistical studies

In the 1990s, a series of measurements of temperature, pH, oxygen concentration and other quantities of biological interest was carried out; the results are analyzed in [47]. The data reveal the oligomictic nature of the lake. Typically, oligomictic lakes undergo a complete mixing event only during the coldest winters: the study confirms this behaviour for lake Garda, as shown in Figure 1.4. When a mixing event is not complete, its extent can be quantified by the mixing depth, defined as the lowest bound of the region where transported scalars are almost uniform, a sign that in such region mixing occurred effectively; below it, the concentration of scalars starts changing more rapidly. The mean winter air temperature correlates very well with water temperature and mixing depth: the winters of 1990/1991, 1998/1999 and 1999/2000 were the coldest of the time series, and they correspond to complete early spring mixing events. Between two full

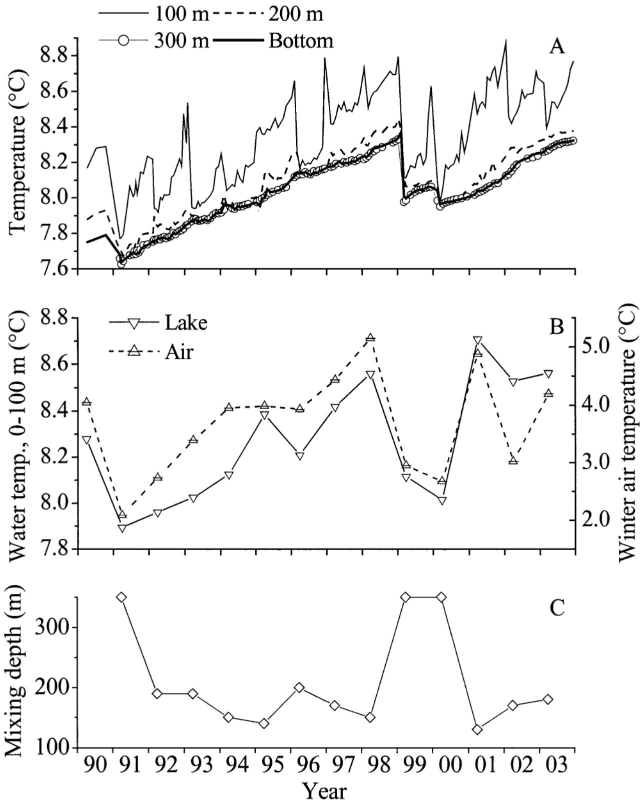


Figure 1.4: (A) local water temperature at various depths; (B) depth-averaged water temperature v. air temperature; (C) mixing depth (from [47])

circulations, the temperature in the deep lake keeps increasing gradually because of slow thermal diffusion from the upper layers, even if the atmospheric temperature shows no distinct upward trend. It is clear from the plots that the temperature history shows less and less fluctuations as depth increases.

A general trend of increase in water temperature was shown in [49] when considering datasets from 1970 to 2009. A linear regression suggests a temperature increase of 0.015°C/year; from the LOWESS (locally weighted scatterplot smoothing) regression, the presence of cycles with period greater than about 20 years seems to emerge. Depending on which statistical test is considered, though, the significance of the upward trend changes. The Mann-Kendall test, specifically devised to check whether a global trend (increasing or decreasing) is present in a time series, is used in the paper. This is done by estimating the probability that the data come from a trend-less time series, that is saying that the observed trend is actually determined by noise. If this probability is low enough, the null hypothesis (no trend) can be rejected. The result shows good significance, with $p < 0.01$, but if the trend-free pre-whitening (TFPW) procedure is

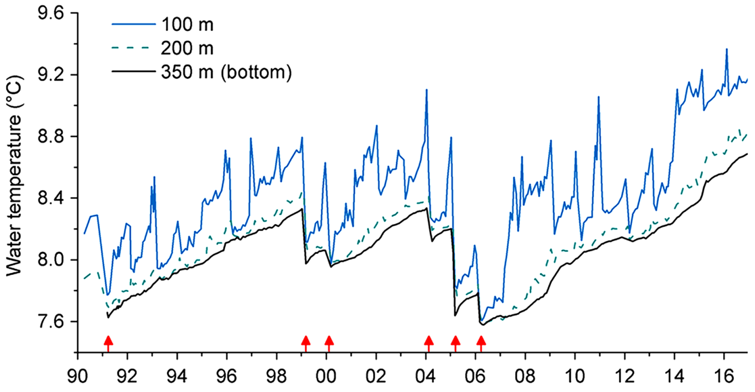


Figure 1.5: Time history of temperature at different depths in the lake. Red arrows mark complete mixing events. From [48]

applied before it to remove the effect of correlation, the result is $p < 0.10$, corresponding to a lower significance. In other words, to examine climate change trends it is necessary to consider data sets that are longer than the above mentioned cycles, otherwise it is hard to distinguish the upward phase of a cycle from long-term growth.

Updated data reported in paper [48], however, leave little doubt as to whether a general temperature increase is happening, as is clear from Figure 1.5. Moreover, it is shown that in the period 2006-2017 there were no full mixing events in the lake. The phenomena analyzed in this and in the previously described works are of biological interest because they determine to which extent algal nutrients are recycled from the deep layers to the surface on a given year, impacting the ecosystem. The focus is in particular on cyanobacteria, a family of phytoplankton, many species of which are toxigenic; the toxic substances they produce compromise balneability in the lake and potability of its water. Cyanobacterial blooms are favoured by warm water and availability of nutrients, mainly nitrogen and phosphorus. Climate change then has two orders of effects on the occurrence of such blooms: a direct influence, the water temperature increase, and an indirect one, the modification of the frequency of full mixing events. As a matter of fact, the highest concentration of phosphorus is typically found in the deepest layers; so, when a full mixing episode happens, water close to the surface, where phytoplankta mostly develop thanks to the presence of light, is enriched in nutrients.

The study [23], spanning about a year, helps clarify how mixing events are triggered and which basic phenomena are involved. A vertical line was moored in the deepest region of the lake, with an acoustic doppler velocity sensor at mid-depth and a chain of temperature sensors. The obtained vertical temperature profiles, which may contain overturns (areas of locally unstable stratification), are reordered to obtain monotonic, stable profiles. From the comparison between measured and reordered profiles, the displacements needed to obtain the latter from the former are calculated, and from these turbulent dissipation and diffusivity are estimated using semi-empirical laws based on the work of Thorpe and Osborn [42] (see section 2.5 and [15]). Wind and temperature data from the weather stations of Limone del Garda and Toscolano-Maderno were also considered to check correlation of the measurements with the atmospheric conditions.

The main observations are the following:

- Wind generates circulations, so, while surface currents are directly influenced by it, currents at mid-depth show a negative correlation. Wind also generates internal waves, which are observed throughout the year, and whose mutual non-linear interactions trigger intense local patches of turbulent dissipation and mixing.

- Water heated during the summer initially remains in the surface layers, confined by the strong stable stratification. During the autumn, northerly winds induce circulations and internal waves. The warm water from above is able to penetrate to greater depth; mixing is enhanced where internal waves steepen and break. The result of this action is the temperature profile in black in Figure 1.6(a): in subsequent stages, the water column is homogeneized down to limited depths, leaving a profile that is characterized by many steps and generally less stable than the one typical of summer.

- Following several days of very low air temperature, a cold current is observed. Its origin could not be determined in this study, because of the setup: it may have come from the surface or from the lake sides. The cold current generates turbulence by shear and because of the presence of unstably stratified regions. The largest dissipation rates of the year are observed in this situation, and the temperature in the deepest layers decreases by 0.1°C in just about one day, as is clear from Figure 1.6(b). This is the same order of magnitude of the temperature variations shown in Figure 1.4 for late winter deep mixing events. The result is the almost homogeneous temperature profile reported in blue in Figure 1.6(a).

### 1.2.2 Short-term, analyses and simulations

Until recent years, there have been very few 3D hydrodynamic studies of lake Garda; they are all scenario simulations.

The paper [35] focused on a specific problem related to the sewage collector system shared by some towns on the lake. After intense rainfall, there have been events of sewage ending up in the lake, deteriorating the water quality in some areas to the point of compromising balneability. Following a spin-up period of 24 h, a 48 h simulation was done with a constant discharge of polluted water from the towns of Garda, Desenzano, Sirmione and Riva del Garda. Even though extensive verification against measurements was not possible, the study highlighted areas of the lake where circulations develop, trapping pollutants for a longer period of time and thus with a stronger negative effect on water quality. It was also observed that under strong winds, that is with an intense power input feeding turbulent mixing, pollutants released from a deep underwater pipeline were able to emerge to the surface, diffusing through the thermocline.

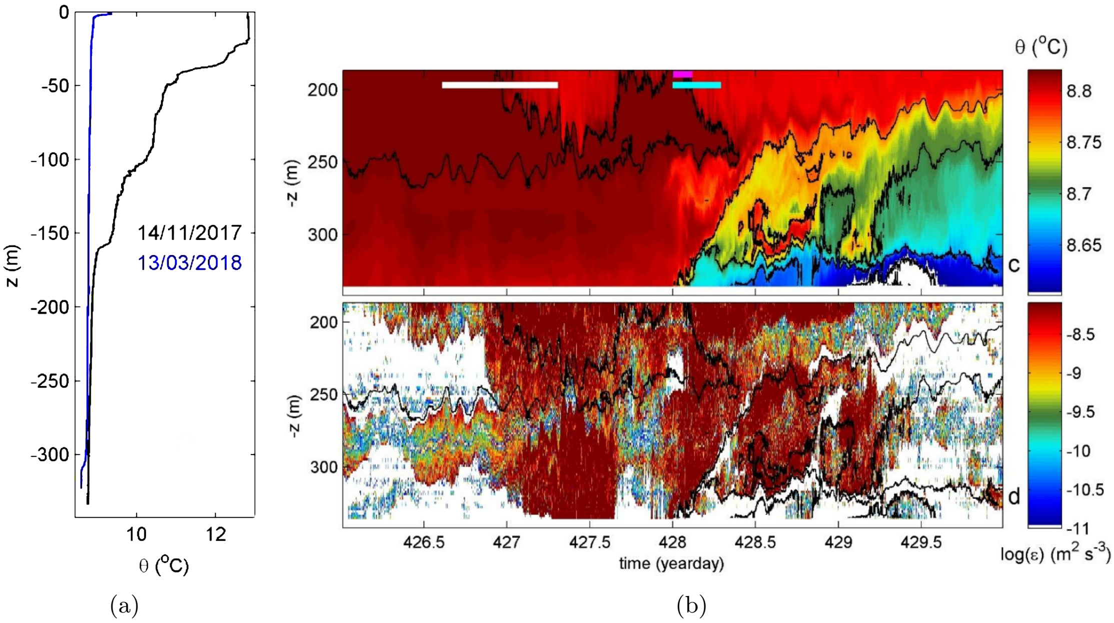


Figure 1.6: Autumn and late winter temperature profiles (a); time histories of temperature and dissipation during a late-winter mixing event (b) (from [23])

In [6], the effects of wind forcing, Earth rotation and stratification are simulated for typical winter and summer daily scenarios. The model is one-way coupled: the results of meteorological simulations are used as boundary conditions for the hydrodynamic simulations. The lake is modelled as an adiabatic system, i.e. no heat transfer is allowed at the free surface and on the bottom; inflow and outflow are neglected. Vertical temperature profiles are obtained from measurements, and they show marked seasonal differences: in summer, the surface temperature is about 24°C, while in winter it is about 9°C; below about 50 m depth, temperature remains close to 8°C all year round. The stratification is thus much stronger and more stable in summer, leading to reduced vertical transport of momentum. Two main scenarios are analyzed, averaging the results over a day:

- On winter days with Föhn winds blowing from the north-east to the south-west along the lake's longitudinal axis, a circulation appears on the vertical longitudinal plane at the northern trunk centerline: on the surface, currents are driven in a direction that is slightly deviated to the west with respect to the one of the wind due to the Coriolis effect, and in deeper layers water moves in the opposite direction. A secondary circulation develops due to the deviation caused by the Coriolis force: downwelling is generated at the western shore, and upwelling at the eastern. Two additional vertical circulations are formed in the southern basin. No gyres appear on the surface in the averaged velocity field.

- On typical summer days, and on winter days with local breezes, winds are more

> variable in space and time than during Föhn events. In particular, breezes are more intense and more clearly alternating on summer days than on winter days. In both cases, a complex surface flow pattern develops, with many gyres in different regions of the lake. The computation of particle path lines shows two interesting phenomena: particles released in the littoral area tend to travel farther than the ones released far from the shore, because the latter get caught up in gyres; time-averaged depths reached by particles are much larger in the winter scenarios, consistently with the fact that the stronger summer stratification inhibits vertical momentum transport.

The effect of Earth's rotation was investigated in [43], considering in particular its influence on ventilation (deep mixing) processes. This study refers to year 2017. Vertical temperature profiles were measured at different stations along a transverse sections of the northern lake. Hydrodynamic simulations were performed on periods of 4-15 days around strong wind events, distributing a tracer in the hypolymnion at the initial time and computing the fraction of tracer ending up in the epilimnion at the end of the simulation. It was shown that the secondary, transverse circulations induced by the wind and the Coriolis effect were able to ventilate about 4% of the hypolimnetic volume in four days of sustained wind. The presence of the secondary circulation is also indicated by field measurements: temperatures in the western part of the considered cross sections were higher than in the eastern part, consistently with downwelling at the western shore and upwelling at the eastern. It is claimed that the presented mixing events were triggered just by wind and Coriolis force, and not by the cooling of surface water as most often happens, because at the time of the events, early to mid-spring, the heat flux was positive, entering the lake.

In [37] the focus is on the effects of inflows and outflows, in particolar in the event of a release of water from river Adige through the Mori-Torbole tunnel. The modelling strategy is different from the previously cited paper: instead of employing a weather simulation code, data about wind, precipitations and flowrates are gathered from weather stations; then, precipitations are averaged, the wind field is recovered by inverse-distance interpolation, flowrates are directly input where measurements are available and obtained by sampling normal distributions of assigned mean and standard deviation in all other cases. Simulations are carried out using data from May and June 2017, thus mainly under the effect of the daily cyclic pattern of Peler and Ora del Garda winds: we should expect results that are reasonably close to the ones obtained in the summer scenario of the previous study. It is not straightforward to make this comparison, due to the way results are presented: daily averages in [6], instantaneous values in [37]; what can be noted is that surface velocities are at least of the same order of magnitude, of about 0.1 m/s or less, and that in both cases in the northernmost part of the lake, where the Mori-Torbole tunnel discharges, particle paths are directed from the eastern shore to the north, on average. Thus the effect of water flowing from the tunnel is first felt on the northern shore, in the area of Torbole and Riva del Garda. An observation of this study is that while the surface flow is mainly affected by the wind, on the bottom the velocity is mostly south oriented. The authors interpret this as an effect of the main (weak) flow from the Sarca inlet to the Mincio outlet. This effect was not observed in [6], because inflows were not considered: it would be interesting to understand if and how the main Sarca-Mincio flow interacts with the observed circulation patterns.

The different modelling choices in the cited articles are compared in Table 1.1.

| | Lovato et al., 2012 [35] | Amadori et al., 2018 [6] | Marzocchi et al., 2020 [37] |
|---|---|---|---|
| Model | TRIM3D (FD), hydrostatic | DELFT3D (FD), hydrostatic | TELEMAC3D (FEM), non-hydrostatic |
| Vertical layers | 12 | 61 | N.A. |
| Spin-up time | 24 h | 48 h | 1 month |
| Wind | modelled | from weather simulations | interpolated |
| Inflows/outflows | yes | no | yes |
| Stratification | yes | yes | N.A. |
| Turbulence model | N.A. | vertical: $k-\epsilon$, horizontal: constant viscosity | $k-\epsilon$ |

Table 1.1: Modelling choices in presents hydrodynamic simulations. FD: finite difference method, FEM: finite element method, N.A.: not specified in the paper

Up to now, no long-term hydrodynamic simulations of lake Garda have been done. One was done for lake Maggiore, and it is described in the consecutive studies [12] and [7]. They start from the consideration that the classical definition of residence time as the ratio of total basin volume $V$ to yearly volume rate $Q$ through it

$$\tau = \frac{V}{Q} \text{ (years)} \tag{1.1}$$

is not a very significant quantity. This is because only the mixed layer at the surface actually participates in water renewal, while the hypolimnion can remain isolated for long periods of time, as already discussed in section 1.2.1. Also, the topography of the bottom and wind patterns can generate currents such that some regions of the lake may mix with the rest of the water mass only to a very limited extent. So, to obtain a better understanding of lake dynamics it is more appropriate to think of residence time as a local quantity, rather than a global one. In [7], particles were released at the mouths of tributaries (tributary markers) and at different positions in the lake (environmental markers) and they were tracked for 4 years, a time close to the global residence time defined by eq. (1.1) and estimated to be 4.1 years for lake Maggiore [49]. The following conclusions were drawn:

- Environmental markers starting at depths of 100 m or less exited the lake in times between 1 and 4-5 years.
- Tributary markers exited the lake between 250 and 1000 days from release, depending on the distance between the mouth of the tributary and the outflow section. Also, the initial dynamics of such markers depend on the tributary temperature: markers from colder inflows reached greater depths.
- For the layers at depth greater than 100 m, the residence time is estimated to be of the order of decades.

The much longer residence time of lake Garda, estimated to be 26.6 years [49], would make such a study very complicated and computationally expensive. However, studies on single portions of the lake over shorter time scales may be useful to characterize some patterns; for example in relation to discharge events from the Mori-Torbole tunnel.

### 1.2.3 Studies on release from the Adige-Garda tunnel

During the progress of the work on this thesis, an article on the effects of an opening of the Adige-Garda tunnel was published [22]. This study focused on a routine maintenance opening of the spillway, which happened in early March 2020. Combining satellite image analysis of suspended particle matter concentration and numerical simulation, it was found that water from the tunnel first plunges down to some depth due to its density relative to the stratification, showing no immediate effect on the surface, and then emerges close to the eastern shore after several hours.

# Chapter 2

# Mathematical model

In this chapter, the incompressible Navier-Stokes equations are specialized for free-surface flow in a rotating frame, with buoyancy effects and averaged turbulent fluctuations. The aim is to provide an understanding of the models used in the project, starting from basic fluid dynamics and scaling arguments. Possible simplifying choices are presented and discussed.

## 2.1 Navier-Stokes equations

The general incompressible Navier-Stokes system is

$$\begin{cases} \nabla \cdot \mathbf{u} = 0 \\ \rho \dfrac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \mathbf{f} + \mu \nabla^2 \mathbf{u} \end{cases} \tag{2.1}$$

on a 3-dimensional domain. $\mathbf{u}$ is the velocity vector (m/s), $\rho$ is the density (kg/m$^3$), $p$ is the pressure (Pa), $\mathbf{f}$ is the volume force vector (N/m$^3$) and $\mu$ is the dynamic viscosity (Pa·s). In the following sections, volume forces and boundary conditions will be specified, and the relative importance of the terms in the equations will be discussed.

## 2.2 Rotating frame

For domains that are small compared to the Earth radius, equations (2.1) can be written on a local tangent plane in a rotating reference frame (this is typically called the f-plane approximation). To account for rotation, the following non-inertial volume force needs to be added:

$$\mathbf{f}_{\mathrm{R}} = -\rho[\mathbf{\Omega} \times (\mathbf{\Omega} \times \mathbf{r}) + 2\mathbf{\Omega} \times \mathbf{u}], \tag{2.2}$$

where $\mathbf{\Omega}$ is the angular velocity vector of Earth. The first term is the centrifugal force (see Figure 2.1): it results in a small perturbation to gravity. Its magnitude per unit mass is $\Omega^2 R \sin\theta \approx 2.4 \cdot 10^{-2}$ m/s$^2$, with $\Omega \approx 7.3 \cdot 10^{-5}$ rad/s, $R \approx 6.4 \cdot 10^3$ m and latitude $\theta \approx 45°$ for lake Garda. This value amounts to a very small fraction of the

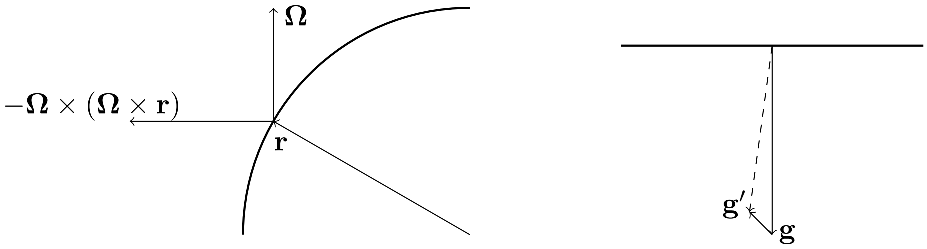

$\boldsymbol{\Omega}$
$-\boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{r})$
$\mathbf{r}$
$\mathbf{g}'$
$\mathbf{g}$

Figure 2.1: Centrifugal force per unit mass and composition with gravity

magnitude of gravity, and the angle between $\mathbf{g}$ and $\mathbf{g}'$ is about 0.1°: this effect can be considered negligible.

The second term is the Coriolis force: it acts on moving particles only, and it tends to deviate their trajectories to the right in the northern emisphere and to the left in the southern. Taking the $x$ axis positive towards the east, the $y$ axis positive towards the north and the $z$ axis positive upwards:

$$-2\boldsymbol{\Omega} \times \mathbf{u} = -2\Omega \begin{pmatrix} \cos\theta\, w - \sin\theta\, v \\ \sin\theta\, u \\ -\cos\theta\, u \end{pmatrix}. \tag{2.3}$$

This contribution is not negligible for lake Garda: its magnitude relative to the advection term is given by the inverse of the Rossby number

$$\mathrm{Ro} = \frac{U}{fL}, \tag{2.4}$$

where $f = 2\Omega \sin\theta$ is the Coriolis coefficient and $U$, $L$ are some suitably defined velocity and length scales of the problem. A related quantity is the Rossby radius $L_{\mathrm{Ro}} = U/f$. This is the radius of the trajectory of a body subject to the Coriolis force alone, on the horizontal plane. Horizontal circulations governed by the Coriolis force can develop only if the geometry and the scale of the domain allow it.[1]. For lake Garda, typical speeds are of order $10^{-1}$ m/s (or less) and the Coriolis coefficient is of order $10^{-4}$, so the Rossby radius is of the order of $10^3$ m (or less), allowing in principle the development of horizontal inertial circulations. As observed in previously cited studies [37] and [6], even when full circulations do not develop, the Coriolis effect markedly deviates wind-driven currents to the right.

## 2.3 Incompressibility and buoyancy

Variations of density in a fluid can be caused by different effects: temperature, pressure, salinity and presence of sediments. Even though variations are present, in many cases

[1] The Rossby number for flow in a sink, for example, is much too high for the Coriolis force to be effective: thus, it is not true that water going down the drain rotates in opposite directions in the two hemispheres because of this effect.

the incompressible form of the continuity equation $\nabla \cdot \mathbf{u} = 0$ can still be used as a very good approximation [8, p.167]. The general (compressible) form of the mass conservation equation is

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u} = 0. \tag{2.5}$$

The first term can be rewritten by using a thermodynamic relation $p = p(\rho, s)$, where $s$ is the entropy per unit volume. Then, by differentiating:

$$\frac{Dp}{Dt} = \left(\frac{\partial p}{\partial \rho}\right)_s \frac{D\rho}{Dt} + \left(\frac{\partial p}{\partial s}\right)_\rho \frac{Ds}{Dt} = c^2 \frac{D\rho}{Dt} + \left(\frac{\partial p}{\partial s}\right)_\rho \frac{Ds}{Dt}, \tag{2.6}$$

where $c$ is the speed of sound. Equation (2.5) becomes

$$\nabla \cdot \mathbf{u} = \frac{1}{\rho c^2} \left(\frac{\partial p}{\partial s}\right)_\rho \frac{Ds}{Dt} - \frac{1}{\rho c^2} \frac{Dp}{Dt}. \tag{2.7}$$

We will now estimate the ratio of the orders of magnitude of the right hand side terms to the left hand side, the latter being of order $U/L$. $U$ and $L$ are, again, reference values of velocity and length that are relevant to the problem; unless otherwise specified, we will take $U \sim 10^{-1}$ m/s, coherent with the velocities found in previous studies on the lake, and $L \sim 10^2$ m as an order of magnitude of the average depth. For the entropy part, a balance can be written [8, p. 156]:

$$T\frac{Ds}{Dt} = \Upsilon + \frac{1}{\rho}\nabla \cdot \mathbf{q}, \tag{2.8}$$

which translates the fact that entropy is produced by viscous dissipation $\Upsilon$ and exchanged by heat transfer. We also have, by thermodynamic relations,

$$\frac{1}{\rho c^2} \left(\frac{\partial p}{\partial s}\right)_\rho = \frac{\beta_T T}{c_P}, \tag{2.9}$$

where $\beta_T$ is the coefficient of thermal expansion at constant pressure:

$$\beta_T = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p. \tag{2.10}$$

Then the first term on the right hand side of (2.7) can be shown to be formed by two parts:

- A part due to viscous dissipation, whose ratio to $\nabla \cdot \mathbf{u}$ is of order

  $$\frac{\beta_T \mu U}{c_P \rho L} \sim 10^{-6}$$

  by using the thermodynamic properties of water.

- A part due to thermal effects, whose ratio to $\nabla \cdot \mathbf{u}$ is of order
$$\frac{\beta_T \kappa \Delta T}{UL} \sim 10^{-10},$$
where $\kappa$ is the thermal diffusivity $\kappa = k/(\rho c_p)$, $k$ the thermal conductivity and $\Delta T \sim 10\,\mathrm{K}$ is the order of magnitude of expected temperature variations.

For the isoentropic part, the convective derivative of pressure can be written exploiting the momentum conservation equation. Taking the scalar product of such equation by $\mathbf{u}$ and using the definition of convective derivative for $p$, we have

$$\frac{Dp}{Dt} = \frac{\partial p}{\partial t} - \frac{\rho}{2}\frac{Du^2}{Dt} + \mathbf{f} \cdot \mathbf{u} + \mu \nabla^2 \mathbf{u} \cdot \mathbf{u}. \tag{2.11}$$

- The first two terms, unsteadiness and kinetic energy transport, are negligible when the Mach number
$$\mathrm{Ma} = \frac{U}{c}, \tag{2.12}$$
ratio of typical flow speed to the speed of sound in the fluid, is much less than one, which is certainly the case since the speed of sound in water is $c \approx 1500$ m/s in standard conditions.

- The term due to volume forces is mainly due to gravity and its magnitude relative to the divergence of velocity is
$$\frac{gH}{c^2} \sim 10^{-3}$$
where $H$ is the maximum depth, about 300 m. The ratio can be recast as
$$\frac{gH}{c^2} = \frac{gH}{U^2}\frac{U^2}{c^2} = \frac{\mathrm{Ma}^2}{\mathrm{Fr}^2}, \quad \mathrm{Fr} = \frac{U}{\sqrt{gH}}, \tag{2.13}$$
where Fr is the Froude number, an important quantity in hydraulics: in the shallow water approximation, $\sqrt{gH}$ is the speed of gravity waves. In shallow water flow Fr plays the same role as Ma in compressible gas flow: both are ratios of flow speed to the speed of propagation of information; hydraulic jumps are the analogous phenomenon to shock waves.

- The viscous term is of order
$$\frac{\mathrm{Ma}^2}{\mathrm{Re}}$$
relative to the divergence term. Re is the Reynolds number
$$\mathrm{Re} = \frac{UL}{\nu}. \tag{2.14}$$
With $U \sim 10^{-1}$ m/s and $c \approx 1500$ m/s as above, $\mathrm{Ma} \sim 10^{-4}$. Concerning Re, for an order of magnitude estimate we may take $L$ as the average depth: $L \sim 10^2$ m, $U \sim 10^{-1}$ m/s, $\nu \sim 10^{-6}$ m$^2$/s, so that $\mathrm{Re} \sim 10^7$. The ratio $\mathrm{Ma}^2/\mathrm{Re}$ is thus certainly negligible.

The evolution of density, which as just seen will not influence the mass conservation equation in our case, can be taken into account using a linearized equation of state [59, p. 73]:

$$\rho = \rho_0 + \Delta\rho = \rho_0 \left[ 1 + \frac{1}{\rho_0 c^2}(p - p_0) - \beta_T (T - T_0) + \beta_C (C - C_0) + \dots \right], \tag{2.15}$$

where $C$ is salinity (concentration). The order of magnitude of the density variation due to hydrostatic pressure can be estimated, for the lake, by considering $\Delta p_h \approx \rho_0 g H$, which leads to $\Delta\rho_p/\rho_0 = gH/c^2 \sim 10^{-3}$. This effect can thus be considered negligible. On the other hand, the effect of temperature will depend strongly on local gradients that may develop for instance close to inflow boundaries or at the free surface, so it will need to be considered; moreover, this is a driving effect for intrusions (see section 2.9), an important feature for the numerical simulations considered in this thesis. More general, non-linear laws can be used for the temperature dependence, since water density reaches a maximum at about 4°C. As a first approximation, we may neglect the action of salinity: as reported in [23], it accounts for about 15% of the total density variations in lake Garda, the primary effect being the thermal one.

Having defined $\rho = \rho_0 + \Delta\rho$ as above, the Boussinesq approximation can be applied to the momentum equation: density variations are neglected everywhere, except where they are multiplied by the gravitational field. The resulting equation, with explicit volume forces including the Coriolis contribution, is

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho_0}\nabla p - \frac{\rho_0 + \Delta\rho}{\rho_0} g\mathbf{k} - 2\boldsymbol{\Omega} \times \mathbf{u} + \nu \nabla^2 \mathbf{u}, \tag{2.16}$$

where $\mathbf{k}$ is the vertical upward unit vector.

## 2.4 Energy equation

For an incompressible flow with constant density, the energy equation is uncoupled from the mass and momentum conservation equations. In our case we do allow density variations, which, as described above, are mainly caused by temperature variations, so the energy equation needs to be taken into account. When the effect of viscous dissipation is neglected, and when the height scales of interest are much smaller than $c_p/(\beta_T g) \sim 10^6$ m [59, p. 29], the balance of internal energy for an incompressible flow can be written as:

$$\rho c_P \frac{DT}{Dt} - \nabla \cdot (k \nabla T) = q_s. \tag{2.17}$$

This is an advection-diffusion equation, with a distributed source term $q_s$. It has the same form as any evolution equation for a tracer in the fluid, e.g. salinity or in general concentration of some advected scalar.

Boundary conditions must be specified to solve the equation. For temperature, the heat flux at the free surface can be imposed as:

$$-k\nabla T \cdot \mathbf{n} = A(T - T_{\text{air}}) \tag{2.18}$$

Conduction in the fluid matches convection, radiation and latent heat of evaporation; the latter phenomena are modelled through coefficient $A$, which depends on water temperature at the surface $T$ and wind velocity $u_{\text{wind}}$. This is a major simplification of actually quite complex phenomena.

## 2.5 Turbulence

The flow field in a lake is typically characterized by length and velocity scales such that turbulence can arise. Turbulence influences effective transport properties greatly, thus it is important to take it into account properly. The main feature of turbulence is the presence of structures (often called eddies) of different lengthscales interacting with each other. Physically, the interactions can be interpreted in terms of vortex dynamics; mathematically, they are explained by the presence of the non-linear advection term in the equations. In this process, kinetic energy provided to the flow at large scale (e.g. by wind forcing) is transferred to smaller structures and dissipated at small scales: this is known as the inertial cascade. Large eddies can have dimensions of the order of the length scale of the domain ($L$) and are influenced by the domain shape, so they can be highly anisotropic. In the cascade process, anisotropy is reduced, and if the Reynolds number (defined in eq. (2.14)) is large enough, then small scales can be considered isotropic [46]. By dimensional analysis, in the hypothesis of homogeneous isotropic turbulence, it can be shown that the sizes of large ($L$) and small, dissipation structures ($\eta$) satisfy

$$\eta \sim \mathrm{Re}^{-3/4} L \tag{2.19}$$

For Lake Garda, Re was already estimated to be of order $10^7$, so $\eta \sim 10^{-3}$ m. It is clear that not only it would be computationally unfeasible to resolve such small structures; it would also be quite useless, for our purpose, to obtain information about the flow with such precision.

The most common procedure is then to use the unsteady Reynolds-averaged equations (URANS) in place of the Navier-Stokes equations. The flow variables are decomposed into a mean part $\overline{\mathbf{u}}$ and a zero-mean fluctuating part $\mathbf{u}'$ (and analogously for pressure and tracers), and then averaged. For the unsteady form, the interpretation of the averaging process is quite tricky, because in general there might not be a clear separation of temporal scales between mean flow and turbulence. As observed in [27], such a separation exists for atmosphere dynamics: the typical power spectrum of wind shows a gap between about 10 minutes and 2 hours[2], so that URANS averaging can be interpreted, for instance, as a running average with a width of about one hour; but no such separation exists in general for hydrodynamic systems. In this case URANS equations are best viewed as the result of ensemble averaging, assuming that turbulent fluctuations are a random perturbation to a repeatable mean field that is the solution of the equations.

[2] This is known as the van der Hoven spectral gap.

The result is a system of equations that is not closed: the Reynolds stresses, terms of the form

$$\mathbf{R} = -\rho\overline{\mathbf{u}' \otimes \mathbf{u}'} \tag{2.20}$$

appear; these terms cannot be written in general as functions of the flow variables or their derivatives. A strategy for closing the system is based on the Boussinesq hypothesis [44, p. 93]: the deviatoric part of the Reynolds stresses is approximated as

$$\mathbf{R} - \frac{1}{3}\mathrm{Tr}(\mathbf{R})\mathbf{I} \; = \rho\nu_T(\nabla\mathbf{u} + \nabla\mathbf{u}^T) = 2\rho\nu_T\mathbf{D}, \tag{2.21}$$

where $\nu_T$ is a turbulent viscosity, to be specified or calculated in each point according to some model (to be discussed later), and $\mathbf{D}$ is the symmetric part of the velocity gradient. This closure is analogous to the Newtonian constitutive law for viscous stresses, but it has no general justification, and as a matter of fact it is often not verified, even in simple flows. The reason it is not justified is that the physical processes that determine Reynolds stresses, that are the presence and evolution of turbulent structures, are not analogous to the ones that determine viscous stresses, that are related to intermolecular interactions [44, p. 359]. In particular:

- in a continuum, molecular phenomena happen at length and time scales that are much smaller than the ones of the flow. Then the hypothesis of local equilibrium, needed to assume that viscosity in a point depends only on local quantities, holds. In contrast, there is no separation of scales between turbulence and the mean flow. Large turbulent eddies may have sizes of the order of the scales of the mean flow. Turbulence is thus not local, so it is not true in general that $\nu_t$ should depend on local quantities only;

- the anisotropy of viscous stresses is very small, in the sense that the ratio of anisotropic to isotropic molecular stress $\tau/p$ is typically a very small quantity, so it is reasonable to assume that it depends linearly on the velocity gradients. On the other hand, the ratio of anisotropic to isotropic Reynolds stresses is often not negligible, so the assumption of a linear law is questionable.

Despite the fact that it is hardly justifiable, this model is the most commonly adopted and works acceptably in many cases. For stratified flow, however, usually at least anisotropy is assumed, defining two different values of viscosity for the horizontal and vertical directions. In equation (2.21) the scalar $\nu_T$ is then substituted by a diagonal tensor

$$\boldsymbol{\nu}_T = \begin{bmatrix} \nu_{T,h} & 0 & 0 \\ 0 & \nu_{T,h} & 0 \\ 0 & 0 & \nu_{T,v} \end{bmatrix} \tag{2.22}$$

where $\nu_{T,h}$ is the turbulent viscosity in the horizontal direction and $\nu_{T,v}$ the one in the vertical direction.

The trace of $\mathbf{R}$ is connected to the kinetic energy of the fluctuations, known as the turbulent kinetic energy $k$:

$$\mathrm{Tr}(\mathbf{R}) = \rho(\overline{u'^2} + \overline{v'^2} + \overline{w'^2}) = 2k, \tag{2.23}$$

so that the averaged momentum equation becomes

$$\frac{\partial \overline{\mathbf{u}}}{\partial t} + (\overline{\mathbf{u}} \cdot \nabla)\overline{\mathbf{u}} = -\frac{1}{\rho_0}\nabla\left(\overline{p} + \frac{2}{3}k\right) - \frac{\rho}{\rho_0}g\mathbf{k} - 2\boldsymbol{\Omega} \times \overline{\mathbf{u}} + (\nu + \nu_T)\Delta\overline{\mathbf{u}}. \tag{2.24}$$

The same averaging process must be applied to the advection-diffusion equations for the tracers, that are in general in the form

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T - \nabla \cdot (\lambda \nabla T) = q. \tag{2.25}$$

$\mathbf{u}$ is decomposed as above, $T$ is decomposed as $\overline{T} + T'$; then the equation is averaged. Fluctuations have zero average by definition, so the result is

$$\frac{\partial \overline{T}}{\partial t} + \overline{\mathbf{u}} \cdot \nabla \overline{T} + \nabla \cdot \overline{\mathbf{u}'T'} - \nabla \cdot (\lambda \nabla \overline{T}) = q. \tag{2.26}$$

This equation is not closed due to the presence of the flux vector $\overline{\mathbf{u}'T'}$, derived from the advection term. The idea is that the effect of turbulence on tracers is to enhance their effective diffusion by advection due to fluctuations; in the gradient-diffusion hypothesis, the scalar flux is modelled as

$$\overline{\mathbf{u}'T'} = -\lambda_T \nabla \overline{T}, \tag{2.27}$$

and the equation becomes

$$\frac{\partial \overline{T}}{\partial t} + \overline{\mathbf{u}} \cdot \nabla \overline{T} - \nabla \cdot [(\lambda + \lambda_T)\nabla \overline{T}] = q, \tag{2.28}$$

where $\lambda_T$ is the turbulent diffusion coefficient, which is usually taken to be proportional to turbulent viscosity: $\lambda_T = \nu_T/\sigma$, with $\sigma$ the Prandtl number, usually taken as a constant[3]. Just as the Boussinesq hypothesis, the gradient-diffusion hypothesis is not true in general, but it is the most common and leads to an equation that has the same form as the original one.

Turbulence in flows with density stratification is an active research field, with large uncertaintites and controversies, as reported in the recent review [13]. Some understanding of its basic features can be gained by inspecting the turbulent kinetic energy equation as in the paper by Osborn [42]. Such equation is derived by multiplying the URANS momentum equation (without the Boussinesq approximation) by $\mathbf{u}'$ and averaging. In the steady state case, the leading terms in the result satisfy

$$-\overline{u_i'u_j'}\frac{\partial \overline{u_i}}{\partial x_j} = \epsilon + \frac{\overline{w'\rho'}g}{\rho_0}. \tag{2.29}$$

The term on the left hand side is the production of turbulent kinetic energy: it shows that a non-zero mean shear generates turbulence. In neutrally or stably stratified conditions, as we shall see, this is the only source of turbulent kinetic energy, so in such

[3] In Telemac, the constant is by default 1 [2].

a condition if the flow were uniform turbulence would be damped. The first term on the right hand side is loss by dissipation; the last term is the work of buoyancy forces. Defining the Richardson number Ri as the ratio of loss to buoyancy to production

$$\mathrm{Ri} = \frac{g}{\rho_0} \frac{\overline{w'\rho'}}{\overline{u_i' u_j'} \dfrac{\partial \overline{u}_i}{\partial x_j}}, \tag{2.30}$$

it is clear that no steady state is possible for $\mathrm{Ri} > 1$, since the viscous dissipation $\epsilon$ is always positive. This means that buoyancy forces can dampen turbulent fluctuations; in other words, part of the turbulent kinetic energy can be converted into gravitational potential energy. This happens in presence of a stable stratification, while in an unstable one potential energy is converted into kinetic energy. The reason lies in the global action of turbulence, that is of mixing, homogenizing: it tends to smooth out temperature profiles [26, p. 274]. A stable temperature profile has lower density water above higher density water, so the center of gravity of water in a basin with such stratification generally lies below[4] the half-depth, which would be its position if the temperature profile were homogeneous. To homogenize the profile, the total potential energy of the water mass must increase: the required energy comes from the turbulent kinetic energy. The physical picture is that turbulent eddies will, on average, carry denser water upwards and less dense water downwards. Conversely, an unstably stratified water mass has higher center of gravity than an homogeneous one, so potential energy is converted into turbulent kinetic energy, producing intense dissipation rates like the ones observed in [23].

Going back to the equations, using the gradient diffusion hypothesis on $\overline{w'\rho'}$ and the Boussinesq hypothesis on $\overline{u_i' u_j'}$, if the Prandtl number is taken as $\sim 1$, the Richardson number can be approximated as

$$\mathrm{Ri} = -\frac{g}{\rho_0} \frac{\dfrac{\partial \overline{\rho}}{\partial z}}{D_{ij} D_{ij}}, \tag{2.31}$$

which shows that Ri is positive when the background (mean) density profile is decreasing upwards, that is, in presence of stable stratification. In this last expression, the buoyancy frequency or Brunt-Väisälä frequency, $N$, can be recognized:

$$N^2 = -\frac{g}{\rho_0} \frac{\partial \overline{\rho}}{\partial z}. \tag{2.32}$$

In a stable stratification, $N$ is the frequency of vertical oscillation of particles displaced from their equilibrium position. This quantity is important because it is an upper bound to the frequencies of internal waves. In an unstable stratification, the trajectory of a particle displaced from its equilibrium position would diverge, and $N$ is undefined.

[4] This is true in most cases, though for particular situations the effect of thermal expansion may produce a different result.

Osborn's paper has been very influential in experimental studies [15]: it provides a simple way to estimate diffusivity from dissipation measurements. It starts from noticing that experimental results indicate the value 0.15 as a maximum Richardson number for steady state to be possible. From this, by defining the vertical diffusivity $K_\rho$ and rearranging eq. (2.30), we have

$$K_\rho = -\frac{\overline{w'\rho'}}{\frac{\partial \overline{\rho}}{\partial z}} = \frac{\mathrm{Ri}}{1-\mathrm{Ri}}\frac{\epsilon}{N^2} \leq 0.2\,\frac{\epsilon}{N^2}. \tag{2.33}$$

$\epsilon$ and $N$ can be estimated by measuring the thermal profile: the latter from its definition, the former using empirical relations. It has become common in experimental studies to replace the last inequality with an equality. This approach has been debated because it relies on the hypothesis of steady state conditions, which very seldom occur in practice: turbulence in environmental flows is mostly patchy and intermittent, as opposed to uniform and constant. Moreover, the model does not tend to the molecular diffusion coefficient as a limit for vanishing turbulent dissipation. Direct numerical simulations (DNS) also reported in [29] show, indeed, that using Osborn's parametrization gives a very good approximation of $K_\rho$ for intermediate turbulent intensity, but it underestimates it for vanishing intensity, and overestimates it for high intensity. A discussion is ongoing [13] about the way patches of turbulence can be created by different model instabilities, and about possible ways of distinguishing stirring from actual mixing. Stirring is the production of small structures (or correspondingly high gradients) in the flow, that is a non-viscous phenomenon and is in general reversible, while mixing is the consequent viscous homogeneization, that is irreversible.

A remark about notation: from now on, where there is no ambiguity the overline will be omitted when writing average (mean flow) quantities.

### 2.5.1 Turbulence models

The possible modelling choices for the turbulent viscosity $\nu_t$ are now discussed.

The simplest choice is taking a constant value, which is usually determined by calibration with measured data. This is sometimes done for large-scale oceanic problems and, when anisotropy due to stratification is concerned, usually a constant value is taken for the horizontal viscosity $\nu_{T,h}$ while a variable value is calculated for the vertical viscosity $\nu_{T,v}$, with one of the models described in the following.

A common algebraic model is Prandtl's mixing length, based on writing the turbulent viscosity, that has dimensions $L^2/T$, as the product $\nu_T = L_m u^*$. $L_m$ is interpreted as the mixing length, that is the length along which eddies act in mixing momentum; $u^*$ is a characteristic eddy velocity. In plane channel flow, with wall-normal coordinate $z$, the model specification is

$$L_m(z) = \kappa z, \quad u^* = L_m \left|\frac{\partial \overline{u}}{\partial z}\right| \tag{2.34}$$

where $\kappa$ is the von Karman constant, normally taken in the interval 0.38-0.41 depending on authors[5], so that the full expression is

$$\nu_T = L_m^2 \left| \frac{\partial \overline{u}}{\partial z} \right| . \tag{2.35}$$

Some corrections and extensions from channel flow to general 3D flows are due:

- the expression for $L_m$ holds in the logarithmic part of the boundary layer (described in section 2.7). When wind is acting on the free surface, a boundary layer appears at the surface, in addition to the one at the bottom [56]. To model this situation properly, a logarithmic velocity profile should appear in both boundary layers; so, close to the surface, the mixing length should behave as

  $$L_m(z) \approx \kappa(s - z), \tag{2.36}$$

  where $s$ is the local elevation of the free surface, and close to the bottom it should behave as

  $$L_m(z) \approx \kappa z. \tag{2.37}$$

  In Telemac, two laws that produce a behaviour of this kind are implemented [2, p. 33]: Quetin's law

  $$L_m(z) = \frac{1}{\dfrac{1}{\kappa z} + \dfrac{1}{0.65(s - z)}} \tag{2.38}$$

  and Tsanis's law:

  $$L_m(z) = \begin{cases} \kappa z, & z \leq 0.2\, s \\ 0.2\, \kappa z, & 0.2\, s \leq z \leq 0.8\, s \\ \kappa(s - z), & z \geq 0.8\, s; \end{cases} \tag{2.39}$$

- $z$ needs to be substituted with the distance of each point from the wall, which may not be horizontal in the general case. In place of $|\partial \overline{u}/\partial y|$, $\sqrt{2D_{ij}D_{ij}}$ is used;

- in presence of stratification, that as discussed before is a situation in which vertical motions can be damped, the vertical viscosity needs to be corrected. We call the vertical viscosity $\nu_{T,v}$ as in equation (2.22) and define it by correction of the base turbulent viscosity $\nu_T$ as follows:

  $$\nu_{T,v} = f(\mathrm{Ri})\, \nu_{T,h} = f(\mathrm{Ri})\, \nu_T, \tag{2.40}$$

  where $f(\mathrm{Ri})$ is a damping function that tends to zero for large positive Ri (strong stable stratification, almost no vertical momentum flux), is equal to 1 for $\mathrm{Ri} = 0$

[5] There is no general agreement on the value of $\kappa$: as a matter of fact there is not even agreement on whether it is a universal constant or a flow-dependent value [36].

> (neutral stratification) and takes large values for negative Ri (unstable stratification, enhanced vertical mixing). For example, the Munk-Anderson damping function implemented in Telemac[6] is of the form
>
> $$f(\mathrm{Ri}) = (1 + a\,\mathrm{Ri})^b. \tag{2.41}$$
>
> As shown in [24], in spite of being just a simple empirical law, this formulation is able to predict the formation of the thermocline under the combined effect of wind stress and heat input. Essentially, when the heat flux at the surface is greater than a certain threshold, heat cannot be uniformly distributed in the water column; this is ultimately due to the presence of a maximum diffusivity value $\mu_{max}$ for temperature, which occurs at a critical Richardson number Ri*. A Poisson equation for Ri can be derived, and the corresponding diffusion coefficient is negative for Ri>Ri*, implying that a peak forms: this is the thermocline.

The mixing length model relies on the hypothesis of local equilibrium [46]: the turbulent viscosity is calculated simply from the local gradient of the mean velocity, and effects of unsteadiness and advection of turbulent structures by the mean flow are neglected. This means, for instance, that turbulent structures generated by a grid and advected downstream are not taken into account.

A different approach to the mixing length idea is Smagorinsky's model: it considers as mixing length some measure of grid element size. The rationale behind this is that using larger elements means being unable to resolve larger eddies; thus the coarser the grid, the larger the value of turbulent viscosity that needs to be considered to model the flow features that cannot be represented on the grid. This model was first used by Smagorinsky for numerical simulation of atmosphere dynamics [52, p. 105].

Better accuracy can be achieved with 2-equation models, in which additional PDEs are added to the system to compute the evolution of turbulent quantities from which turbulent viscosity is obtained. In this case, since essentially advection-diffusion equations are solved, non-local effects can be taken into account to some extent. The most popular model is the $k - \epsilon$, where the computed quantities are turbulent kinetic energy $k$ as defined in equation (2.23) and its dissipation

$$\epsilon = 2\nu \overline{\frac{\partial u_i'}{\partial x_j}\frac{\partial u_i'}{\partial x_j}}. \tag{2.42}$$

This is actually what is called pseudo-dissipation, but the two can be shown to be practically equivalent [44, p.132]. Dimensional arguments yield

$$\nu_t = C_\mu \frac{k^2}{\epsilon}, \tag{2.43}$$

where $C_\mu$ is one of many model coefficients, which are tuned by requiring that the model behaves well on basic test cases (turbulence downstream of a grid, that is a good

[6] http://docs.opentelemac.org/dricv_8f_source.html

experimentally feasible approximation of homogeneous isotropic turbulence, and shear flow). Equations for the exact evolution of $k$ and $\epsilon$ can be obtained from the Navier-Stokes equations, but they are not closed, because they contain higher order moments of velocity; the equation for dissipation is particularly troublesome, because it contains terms that would be very hard to interpret and model individually. The choice is then to close the exact equation for $k$ using the gradient diffusion hypothesis and to rewrite completely the equation for $\epsilon$, in such a way that it takes the same form as the one for $k$, with additional model coefficients to be tuned. The result are two additional advection-diffusion equations with production and destruction terms: $k$ and $\epsilon$ behave as active tracers, meaning that they influence the flow and are influenced by it. The buoyancy term is already embedded in the $k - \epsilon$ equations, so no correction in the form of damping functions is needed.

## 2.6 Free surface

In the case of free surface flow, the surface elevation becomes an additional unknown: the domain changes in time, and its evolution is coupled with the evolution of other flow variables. At first glance, for the case of a whole lake it may seem like an unnecessary complication to bother calculating the motion of the free surface: there are no tidal phenomena to be described, and the wavelengths of most waves are much smaller than any reasonable grid size, so there is no possibility, nor necessity, of resolving them. Actually, though, the free surface displacement, and in particular its inclination, is important for the development of wind-induced currents: wind blowing over the surface will tend to accumulate surface water downwind; but if the basin is closed, the total mass flux on any cross section must be zero in steady conditions. What happens is that, under the action of wind, the water level lowers upwind and rises downwind. This tilting of the surface creates an hydrostatic pressure gradient: at any fixed elevation, hydrostatic pressure is greater upwind than downwind. The effect of the wind shear stress is damped with depth by viscosity, so that below a certain depth the hydrostatic pressure gradient dominates on it and water moves in the direction opposite to that at the surface, making a closed circulation possible. A detailed description of this phenomenon and analytical solutions are developed in [26].

Since a new unknown appears, an additional equation should be written. There are many possible choices; the one implemented in Telemac is presented here [30]. This formulation is based on integrating the differential form of the continuity equation from the bottom $z = B(x, y)$ to the free surface $z = S(x, y, t)$. This guarantees that mass conservation is enforced. It should be noted that no time dependence is present in the equation for the bottom surface, meaning that it is assumed that any variation due to sediment transport happens on time scales much longer than the ones of the flow. Another important point is that, since the free surface is described by a single-valued function, the model is not capable of describing some flow features such as breaking waves: see Figure 2.2 for an example.

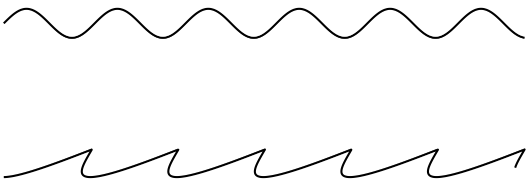


Figure 2.2: Surface shapes that can (above) and cannot (below) be represented by a single-valued function $S(x, y, t)$

The starting point is the integration of the continuity equation:

$$\int_{B(x,y)}^{S(x,y,t)} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) dz = 0. \tag{2.44}$$

Integrals and derivatives can be interchanged by using the Leibniz theorem for the $u$ and $v$ terms, and the fundamental theorem of calculus for the $w$ term. The result is

$$\begin{aligned} \frac{\partial}{\partial x} \int_{B(x,y)}^{S(x,y,t)} u \; dz + \frac{\partial}{\partial y} \int_{B(x,y)}^{S(x,y,t)} v \; dz & \\ + u_b \frac{\partial B}{\partial x} + v_b \frac{\partial B}{\partial y} - w_b & \\ - \left( u_s \frac{\partial S}{\partial x} + v_s \frac{\partial S}{\partial y} - w_s \right) &= 0, \end{aligned} \tag{2.45}$$

where $\mathbf{u}_s = (u_s, v_s, w_s)$ is the velocity at the surface and $\mathbf{u}_b = (u_b, v_b, w_b)$ is the one at the bottom.

On the bottom, the impermeability condition has to be enforced:

$$\mathbf{u}_b \cdot \mathbf{n} = 0, \tag{2.46}$$

where $\mathbf{n}$ can be defined as the vector product of two independent vectors tangent to the surface $z = B(x, y)$:

$$\mathbf{n} = \left( 1, 0, \frac{\partial B}{\partial x} \right) \times \left( 1, \frac{\partial B}{\partial y}, 0 \right) = \left( -\frac{\partial B}{\partial x}, -\frac{\partial B}{\partial y}, 1 \right). \tag{2.47}$$

The condition can then be written as

$$u_b \frac{\partial B}{\partial x} + v_b \frac{\partial B}{\partial y} - w_b = 0, \tag{2.48}$$

thus eliminating the second row of equation (2.45).

On the free surface, an analogous condition has to be enforced, taking into account that the boundary is moving with a velocity $\mathbf{v}_s$:

$$\mathbf{u}_s \cdot \mathbf{n} = \mathbf{v}_s \cdot \mathbf{n}. \tag{2.49}$$

**n** can be defined as the vector product of two independent vectors tangent to the surface $z = S(x, y, t)$:

$$\mathbf{n} = \left(1, 0, \frac{\partial S}{\partial x}\right) \times \left(1, \frac{\partial S}{\partial y}, 0\right) = \left(-\frac{\partial S}{\partial x}, -\frac{\partial S}{\partial y}, 1\right), \tag{2.50}$$

and

$$\mathbf{v}_s = \left(0, 0, \frac{\partial S}{\partial t}\right). \tag{2.51}$$

The result is

$$-u_s \frac{\partial S}{\partial x} - v_s \frac{\partial S}{\partial y} + w_s = \frac{\partial S}{\partial t}, \tag{2.52}$$

which substituted in equation (2.45), whose second row is zero because of the bottom boundary condition, yields

$$\frac{\partial S}{\partial t} + \frac{\partial}{\partial x} \int_{B(x,y)}^{S(x,y,t)} u \, dz + \frac{\partial}{\partial y} \int_{B(x,y)}^{S(x,y,t)} v \, dz = 0. \tag{2.53}$$

The same result could be obtained by a volume balance on a region $\Omega$: the time variation of water volume is

$$\int_{\Omega} \frac{\partial S}{\partial t} \, dA, \tag{2.54}$$

that must be balanced by a boundary flow

$$\oint_{\partial\Omega} \mathbf{U} \cdot \mathbf{n} \, dl \tag{2.55}$$

with

$$\mathbf{U} = \int_{B}^{S} \mathbf{u} \, dz. \tag{2.56}$$

By using the divergence theorem, the equation can be put into differential form as

$$\frac{\partial S}{\partial t} + \nabla \cdot \mathbf{U} = 0, \tag{2.57}$$

which is the form obtained above.

## 2.7 Boundary conditions

Impermeability kinematic conditions are enforced at the bottom and at the free surface in the forms of equations (2.48) and (2.52). In addition, the wind stress should be imposed at the surface, and the no-slip condition at the bottom. However, the no-slip condition implies a correct value of the bottom shear stress only if the first node away from the boundary is inside the viscous sublayer of the turbulent boundary layer. To understand why it is so, the main results for turbulent boundary layers are recalled.

The mean velocity profile close to a wall and its relation with wall shear stress in a turbulent flow can be determined by dimensional analysis [16, p. 137]. The idea is

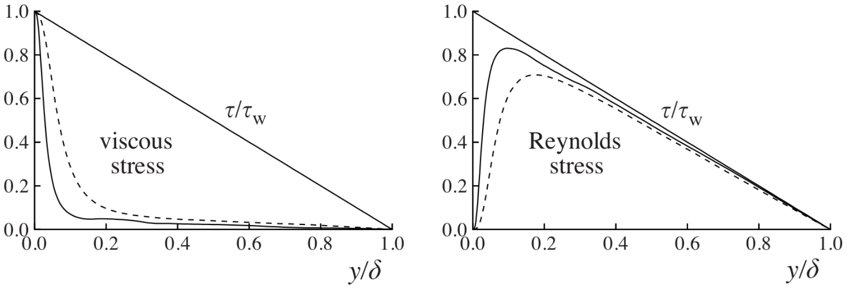


Figure 2.3: Viscous, turbulent and combined stress for channel flow, channel width $\delta$; dashed line, Re=5,600; solid line, Re=17,500; from [44]

that the flow close to the wall is not influenced by the outer flow, so that the outer velocity and length scales do not appear in the expression of the near-wall velocity profile: in their place, the wall scaling velocity $u_\tau = \sqrt{\tau_w/\rho}$ and length $\delta_\nu = \nu/u_\tau$ must appear. By considering proper limits of the general expression, it can be proved that the profile is linear close to the wall (viscous layer) and logarithmic in an intermediate layer (inertial, or logarithmic layer). To obtain a correct representatiom of the wall shear stress automatically in the momentum equation by imposing the no-slip condition, the first node away from the wall should then be inside the viscous sublayer. There are two observations worth pointing out. The first is that the obtained velocity profile can be considered universal when expressed in wall units: velocity scaled with $u_\tau$ and non-dimensional distance from the wall $y^+ = y/\delta_\nu$. The second is that the ratio of the outer lengthscale to the wall lengthscale $L/\delta_\nu$ grows with Re: at high Re, it becomes computationally expensive to resolve the velocity profile, because it would require very small elements in the boundary layer. But, since the near-wall velocity profile can be considered universal, it can be modeled instead of calculated, so that the first grid point away from the wall can be in the logarithmic layer, at maybe $y^+ \approx 50$, instead of at about $y^+ = 1$; and the stress can be directly imposed, instead of using the no-slip condition. This is known as the wall function or high Reynolds approach.

Following the above discussion, the shear vector $\boldsymbol{\tau}$ is imposed on both the bottom and the free surface:

$$\rho(\nu + \nu_T)\frac{\partial \mathbf{u}}{\partial \mathbf{n}} = \boldsymbol{\tau}, \tag{2.58}$$

meaning that the stress on the boundary must match the total (viscous and turbulent) stress in the fluid. The reason why this is a good approximation can be seen in Figure 2.3: the total stress is linear, so, if the first grid points are close enough to the wall, the calculated total stress is approximately equal to the viscous stress at the wall. The figure refers to channel flow, but this behaviour is characteristic of boundary layers in general. On the bottom, the condition is determined by the features of the turbulent boundary layer, influenced by the wall roughness; on the surface, it describes the effect of wind.

We start dealing with the bottom. In this case, the wall function approach leads to the adoption of an iterative scheme, since the velocity profile is given in terms of the wall scales, which in turn depend on the solution through $\tau_w$. In Telemac3D, the model mean flow profile is given by the Reichard law [2, p. 38]

$$\frac{u(y)}{u_\tau} = \frac{1}{\kappa}\log(1+\kappa y^+) + 7.8\left(1 - e^{-y^+/11}\right) - \frac{y^+}{11}e^{-0.33y^+}. \tag{2.59}$$

In this equation, $u(y)$ is known from the solution, $y$ depends on the grid and $\nu$ is known: the only unknown is $u_\tau$, which appears on both sides and can be determined iteratively; then the wall stress is recovered as

$$\tau_w = \rho u_\tau^2. \tag{2.60}$$

What has been discussed up to now holds for hydraulically smooth flow, meaning that the length scale of the roughness elements is much smaller than the viscous layer thickness, so that the effect of roughness can be neglected. When this is not true, and in particular when the roughness scale $k_s$ is large in wall units ($k_s^+ \gg 1$), the Reynolds number relative to the roughness elements $k_s u_\tau/\nu$ becomes large: in the interaction with the wall, inertial effects dominate over viscous effects, so the dependence on $\nu$ is lost and $\delta_\nu$ is no more a relevant quantity. The only relevant length scale is then $k_s$, and the corresponding law for the mean flow in the hydraulically rough case becomes

$$\frac{u(y)}{u_\tau} = \frac{1}{\kappa}\log\left(\frac{y}{k_s}\right) + 8.5\ . \tag{2.61}$$

In this case there is no need of using an iterative scheme: $u_\tau$ can be obtained simply by inverting the equation. In alternative, a classical choice is to use empirical friction laws based on the depth-averaged velocity components. The stress is written as

$$\boldsymbol{\tau} = -\frac{1}{2}\rho C_f \left\|\mathbf{u}_{2D}\right\| \mathbf{u}_{2D}, \tag{2.62}$$

where $\mathbf{u}_{2D}$ is the depth-averaged velocity and $C_f$ depends on depth and roughness size through empirical laws, different choices of which are possible (e.g. Chezy and Manning laws, see [2, p. 39]).

The wind-surface interaction is quite a complex phenomenon [53, p. 157]. Let us consider wind blowing over an initially flat surface: because of the presence of turbulent eddies in the wind, different portions of the surface will be subjected to higher or lower pressure. This will cause the formation of irregularly shaped waves: the surface will tend to move downwards in the high pressure regions, and upwards in the low pressure ones. Now the wind is no more blowing over a flat surface, but over a surface with a distinct roughness, which is determined by the wind itself; and the presence of roughness will modify the airflow, and in particular the shear stress exchanged at the air-water interface. It would be complicated to model such phenomena in detail, so empirical models are usually employed. In general, the stress is written as

$$\boldsymbol{\tau} = \rho_{\text{air}} c_d u_{\text{wind}} \mathbf{u}_{\text{wind}}, \tag{2.63}$$

where the coefficient $c_d$ is provided by empirical laws as function of $u_{\text{wind}}$; a number of them are available, in the form of piecewise constant, piecewise linear, power-law, exponential functions, and more. A selection of models is reported in Figure 2.4 (data from [11]).

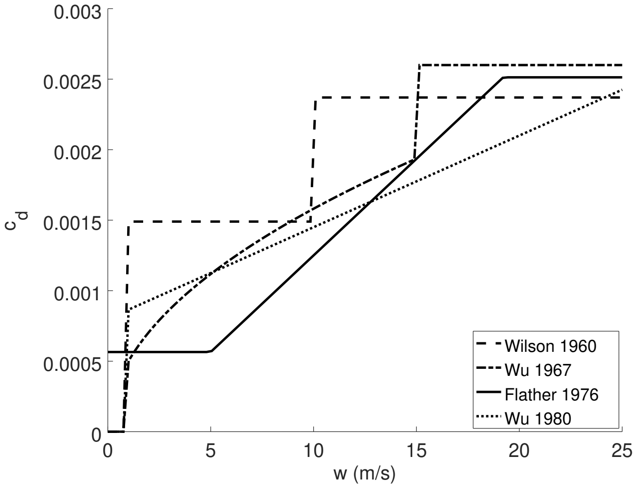


Figure 2.4: Some empirical laws for wind coefficient $c_d$

The law implemented in Telemac3D is [2, p. 12]

$$c_d = \begin{cases} 0.565 \cdot 10^{-3} & u_{\text{wind}} \leq 5\text{m/s} \\ (-0.12 + 0.137 u_{\text{wind}}) \cdot 10^{-3} & 5\text{m/s} \leq u_{\text{wind}} \leq 19.22\text{m/s} \\ 2.513 \cdot 10^{-3} & u_{\text{wind}} \geq 19.22\text{m/s} \end{cases} \tag{2.64}$$

## 2.8 Hydrostatic approximation

A common approximation adopted in free-surface hydrodynamics studies is to substitute the vertical component of the momentum equation with an hydrostatic equilibrium equation:

$$\frac{\partial p}{\partial z} = -\rho g. \tag{2.65}$$

To understand in which cases this is reasonable, the non-dimensional form of the vertical momentum equation can be examined [30, p. 19]:

$$\begin{aligned} &\frac{H}{L}\frac{W}{U}\left(\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y}\right) + \frac{W^2}{U^2} w \frac{\partial w}{\partial z} \\ &= -\frac{\partial p}{\partial z} - \frac{gH}{U^2}\frac{\rho}{\rho_0} + \frac{f_H H}{U} u + \frac{\nu_H}{LU}\frac{H}{L}\frac{W}{U}\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2}\right) + \frac{\nu_V}{LU}\frac{\partial^2 w}{\partial z^2}, \end{aligned}$$

where $L$ and $H$ are the horizontal and vertical lengthscales, $U$ and $W$ are the horizontal and vertical velocity scales, $f_H$ is the horizontal Coriolis coefficient and $\nu_H$ and $\nu_V$ are the horizontal and vertical effective kinematic viscosities (molecular and turbulent). For the hydrostatic approximation to be acceptable, the coefficients of all terms except the ones in equation (2.65) must be negligible, say at most of order $10^{-1}$. While this is true for the vertical Coriolis term, that can be considered a small perturbation to gravity, and for the viscous terms, which are scaled by Reynolds number that are assumed to be large, it is hardly justifiable for the advection terms in the case of lake Garda: the northern part of the lake is narrow, about 3.5 km wide on average, and deep, reaching about 350 m. We are evidently at the limit of applicability of the hydrostatic approximation, and maybe beyond it if we consider local regions of steep bottom where both ratios $H/L$ and $W/H$ cannot be considered negligible: as shown in Figure 2.5, in the northern part of the lake the bed descends very steeply close to the coastline and then becomes almost flat near the center.

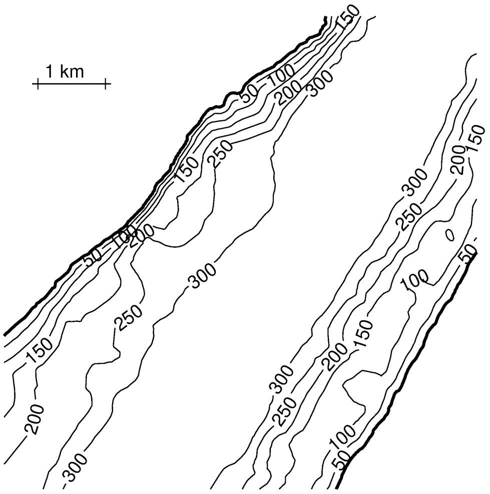


Figure 2.5: A portion of bathymetry in the northern lake

In addition employing the approximation would imply losing the ability to simulate buoyancy effects correctly. This is shown for instance in the lock exchange test case presented in [30, p.143]. This case consists of two volumes of fluid of different density initially separated by a vertical wall; the wall is then removed. This can be considered a simple model of local phenomena arising when cold water enters the lake and starts sinking. The result is that the shape of the interface between the two fluids changes when using the hydrostatic model instead of the non-hydrostatic one, and, more importantly for tracer transport, vertical velocities differ greatly, as shown in Figure 2.6. For all these reasons, for lake Garda the hydrostatic model does not seem to be a good approximation, especially when concerned with vertical circulations.

Even though the hydrostatic approximation cannot be used, it can still be convenient to split pressure into an hydrostatic part and a dynamic part. The former is due

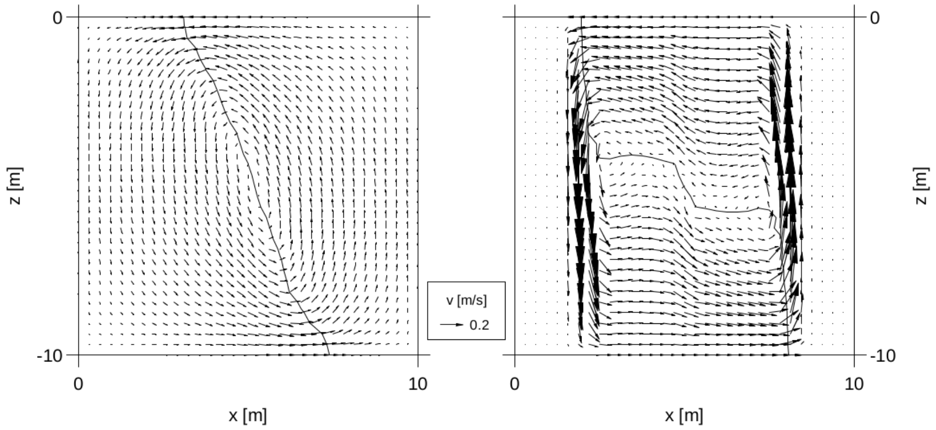


Figure 2.6: Lock exchange test case: velocity vectors for the non-hydrostatic (left) and hydrostatic (right) model; the solid line is the interface. From [30]

to the weight of the column of fluid above a point (possibly taking into account also atmospheric pressure), and it is the solution of equation (2.65); the latter can be interpreted mathematically as the lagrangian multiplier needed to enforce incompressibility. We can write

$$p = p_{\text{hyd}} + p_{\text{dyn}} \tag{2.66}$$

and

$$\begin{aligned} p_{\text{hyd}}(x,y,z,t) &= p_{\text{atm}}(x,y,t) + \int_z^{S(x,y)} \rho g \, d\zeta \\ &= p_{\text{atm}}(x,y,t) + \rho_0 g(S(x,y) - z) + \int_z^{S(x,y)} \Delta\rho \, g \, d\zeta. \end{aligned} \tag{2.67}$$

Then, the hydrostatic pressure gradient terms can be treated as source terms in the momentum equations, being split from the dynamic pressure, which remains an unknown. This splitting is exploited in the numerical scheme discussed in the next chapter; moreover, the treatment of horizontal hydrostatic pressure gradients on the mesh is not trivial: it can lead to dramatic errors if not handled properly, as will be discussed in section 4.3.

## 2.9 Analytical treatment of intrusions

When water from a river or a channel enters a lake or the sea, it propagates as a current. After a possible initial jet-like behaviour, in which the flow is driven by inertia and influenced by viscous or turbulent effects depending on the regime, the current's behaviour will be mostly driven by hydrostatic pressure differences due to the density difference between the inflowing water and the ambient.

The aim of this section is to present an analytical model for currents in stratified flow. This presentation is based on the 1968 paper by Benjamin [9] and the 2009 book by Ungarish [58], but the derivation proposed here differs from these references: we consider the limit of depth much larger than current thickness. To be able to compute such limit, an assumption about the downstream velocity profile has to be changed with respect to the one adopted in the references above: this point will be clarified later.

Currents that are characterized by a density difference with the ambient are classified into gravity currents and intrusions:

- Gravity currents have a density that is greater than the largest one in the ambient or less than the smallest one, so they propagate along the bottom or along the surface;
- Intrusions have an intermediate density, so they propagate at intermediate depths, without interfering with the bottom or the surface.

The scenario simulated in this thesis is classifiable as an intrusion. The results of this chapter are specialized to intrusions. The driving force in both cases is given by the hydrostatic pressure difference between the inside and the outside of the current.

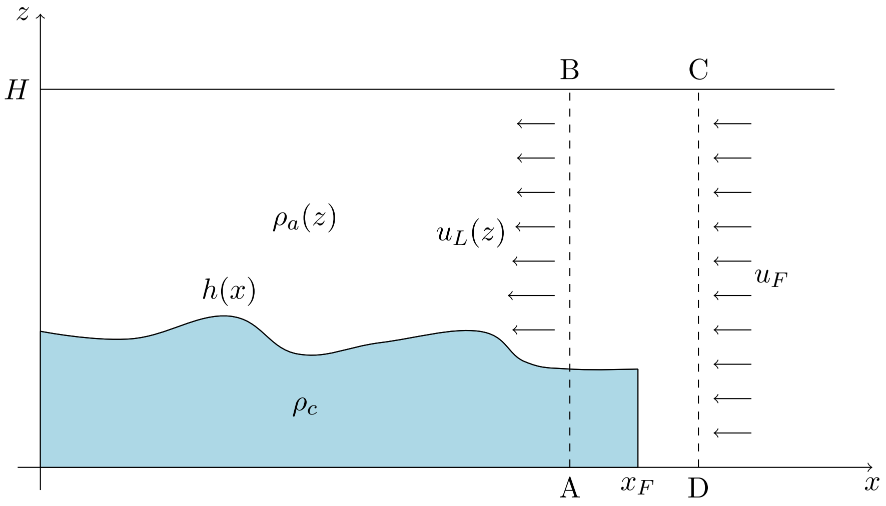


Figure 2.7: Half-domain schematic for the derivation. The reference system is attached to the front.

Consider an intrusion propagating in a stratified ambient. Figure 2.7 shows the half domain, assuming geometrical symmetry around $z = 0$. For the derivation, consider a reference frame attached to the front. In this reference frame, the ambient fluid travels towards the front, whose position is $x_F$, with velocity $u_F$. Just downstream of $x_F$, the outer velocity has a profile $u_L(z)$, in general not uniform. The intrusion half-thickness is $h(x)$ and the half-depth of the domain is $H$. Density inside the intrusion is constant and equal to $\rho_c$, while the density profile in the ambient is $\rho_a(z)$. We make the following assumptions:

- The depth is much greater than the intrusion thickness: $H \gg h$.
- The Boussinesq approximation is valid: density variations about a reference value are neglibile except in the buoyancy (gravity) term.
- The ambient density is assumed equal to $\rho_c$ on the plane $z = 0$: $\rho_a(0) = \rho_c$; this is taken as reference density for the Boussinesq approximation.
- The hydrostatic approximation is valid.
- Outside the boundary of the front, identified by the line $h(x)$ and the vertical discontinuity at $x_F$, viscous effects are neglibile, i.e. the flow field evolves according to the Euler equations (with the aforementioned Boussinesq and hydrostatic approximations). This is analogous to the approach used in aerodynamics of subdividing the flow around an airfoil into the boundary layer, where viscous effects are important, and the outer flow, where they are negligible and the Euler equations are an appropriate model.

The velocity of the front with respect to the ambient, $u_F$, can be computed as function of the half-thickness of the intrusion at the front, $h_F = h(x_F)$, by considering the hydrostatic balance and integral balances of mass and x-momentum over the control volume ABCD in Figure 2.7, in the limits $|x_D - x_A| \to 0$ and $H \gg h$.

Mass conservation requires

$$u_F H = \int_h^H u_L(z)\, dz. \tag{2.68}$$

Since $H \gg h$, most of the ambient above the intrusion will be almost unperturbed by the obstacle below, so we will take

$$u_L(z) \simeq u_F. \tag{2.69}$$

In the original work by Benjamin, $u_L$ was taken as uniform and equal to $u_F H/(H - h)$. This a reasonable approximation for finite $H$, but not for $H \gg h$, and it can be verified that in this case it leads to unphysical results.

The general integral momentum balance reads

$$\int_V \frac{\partial}{\partial t}(\rho_c \mathbf{u})\, dV + \int_{\partial V} \rho_c \mathbf{u}\, \mathbf{u} \cdot \mathbf{n}\, dS = \int_V \rho(x, z, t)\mathbf{g}\, dV - \int_{\partial V} p\mathbf{n}\, dS. \tag{2.70}$$

For vanishing volume $V$, the volume integrals vanish, while the surface integrals remain finite. In the horizontal direction, for the control volume ABCD, the balance reduces to

$$\int_{AB} (\rho_c u^2 + p)\, dS = \int_{CD} (\rho_c u^2 + p)\, dS, \tag{2.71}$$

while, in the vertical direction,

$$\frac{\partial p}{\partial z} = -\rho(x, z, t) g. \tag{2.72}$$

The density stratification is approximated by a linear function around $z = 0$:

$$\rho(z) \simeq \rho_c + \alpha z, \tag{2.73}$$

where $\rho(0) = \rho_c$, as assumed, and $\alpha < 0$ for a stable stratification. The evaluation of the pressure integrals in equation (2.71) is simplified by observing that the pressure fields are the same on sections AB and CD for $z > h_N$: we have

$$\int_{AB} p\, dS - \int_{CD} p\, dS = \int_0^H (p_{AB}(z) - p_{CD}(z))\, dz = \int_0^{h_N} (p_{AB}(z) - p_{CD}(z))\, dz \tag{2.74}$$

The pressure profiles are computed in $[0, h_F]$ from eq. (2.72):

$$\begin{aligned} p_{AB}(z) =& \rho_c g(h_F - z) + p_a(h_F) \\ p_{CD}(z) =& \rho_c g(h_F - z) + \frac{\alpha}{2} g(h_F^2 - z^2) + p_a(h_F) \end{aligned} \tag{2.75}$$

It follows that

$$\int_{AB} p\, dS - \int_{CD} p\, dS = \int_0^{h_F} -\frac{\alpha}{2} g(h_F^2 - z^2)\, dz = -\frac{\alpha}{3} g h_F^3. \tag{2.76}$$

Notice that this approach did not require any assumption on the ambient density profile outside the interval $[-h_F, h_F]$: it only required enough smoothness to approximate it with a linear function in such interval. The momentum flux integrals in equation (2.71) can be computed, remembering that the reference system is attached to the front and using equation (2.69), as

$$\int_{AB} \rho_c u^2\, dS - \int_{CD} \rho_c u^2\, dS = \rho_c \int_0^H (u_{AB}^2(z) - u_{CD}^2(z))\, dz = -\rho_c u_F^2 h_F. \tag{2.77}$$

The momentum balance requires the sum of the last two expressions to be zero, so an expression relating $u_N$ and $h_N$ is obtained:

$$u_F = \sqrt{\frac{1}{3} g' h_F}, \tag{2.78}$$

where $g'$ is called reduced gravity and defined by

$$g' = -\frac{\alpha h_F}{\rho_c} g \simeq -\frac{\rho_a(h_F) - \rho_a(0)}{\rho_a(0)} g. \tag{2.79}$$

As pointed out by Ungarish, eq. (2.78) is a single equation in two unknowns. For the purposes of this work, the equation will suffice for comparison with the numerical results, but for prediction purposes it is incomplete. For a complete description, [58] suggests using relations of the form $u_F = u_F(h_F)$, like the one derived here, as boundary conditions for a system of shallow water equations obtained by averaging the Navier-Stokes continuity and momentum equations over the thickness of the intrusion. The result is an hyperbolic system which can be solved exactly using the method of characteristics in simple cases such as the dam break problem, or solved approximately for more general cases using numerical methods (e.g. finite volumes, smoothed particle hydrodynamics).

# Chapter 3

# Numerical model

In Telemac3D, the model equations are solved using the finite element method (FEM). The mesh is formed by stacking layers of 2D, unstructured meshes, and moves according to the free surface evolution. Equations are discretized using a fractional step method.

## 3.1 Mesh

In Telemac3D, the 3D mesh is made of layers of prisms (see Figure 3.1), built from the bottom to the free surface by replicating a 2D unstructured triangulation of the domain. All lateral faces of prisms are vertical. Thus the mesh can be considered horizontally unstructured and vertically structured. A vertical section of the lake mesh is shown in Figure 3.2.

### 3.1.1 Sigma transform

Because the free surface moves, all the equations described in the previous chapter need to be solved in a variable domain. Accounting for this is not trivial when using a mesh-based method, like the finite element method used in Telemac3D. The adopted solution

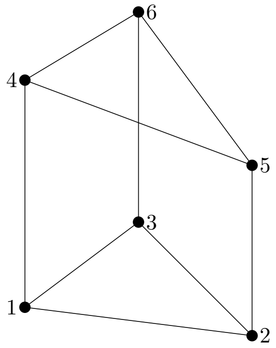


Figure 3.1: A prism element

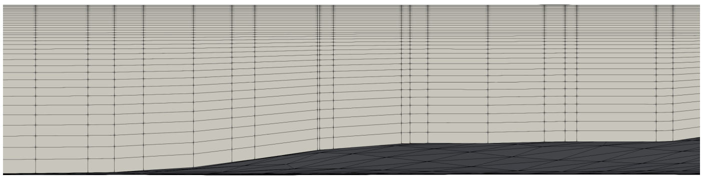


Figure 3.2: Vertical section of the lake mesh, 36 planes (35 layers)

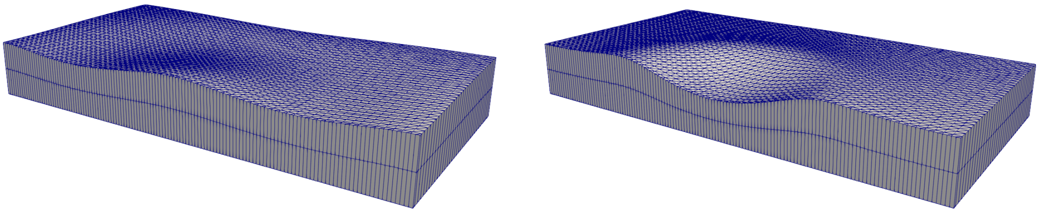


Figure 3.3: A Telemac3D mesh at two different timesteps of test case `gouttedo` (gaussian initial surface)

is to let the mesh points move vertically (so that prism lateral faces remain vertical) to adjust to the surface (see Figure 3.3) and to define a new vertical coordinate $z^*$ in such a way that the points at the bottom are at $z^* = 0$ and the ones at the free surface at $z^* = 1$.[1]

This requires rewriting the equations in a new set of coordinates [25, p. 18]:

$$\begin{cases} x^* = x \\ y^* = y \\ z^* = \dfrac{z - B(x,y,t)}{S(x,y,t) - B(x,y,t)} = \dfrac{z - B(x,y,t)}{h(x,y,t)} \\ t^* = t \end{cases} \tag{3.1}$$

Now, to transform the equations we need the expression of partial derivatives for a given quantity $f(x,y,z,t)$ in terms of derivatives in the system $(x^*, y^*, z^*, t^*)$. We write $f(x,y,z,t) = f^*(x^*, y^*, z^*, t^*)$: in this case it pays to be a little pedantic and to point out that, even though the physical quantity $f$ is always the same, its functional expression

[1] Other vertical transforms are possible: for example, every vertical layer of the mesh could be stretched to a specific extent. The one presented here is the most simple case.

changes if coordinates change. By using the chain rule we can write

$$\begin{aligned}
\frac{\partial f}{\partial x} &= \frac{\partial f^*}{\partial x^*}\frac{\partial x^*}{\partial x} + \frac{\partial f^*}{\partial y^*}\frac{\partial y^*}{\partial x} + \frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial x} + \frac{\partial f^*}{\partial t^*}\frac{\partial t^*}{\partial x} \\
&= \frac{\partial f^*}{\partial x^*} + \frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial x} \\
\frac{\partial f}{\partial y} &= \frac{\partial f^*}{\partial y^*} + \frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial y} \\
\frac{\partial f}{\partial z} &= \frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial z} \\
\frac{\partial f}{\partial t} &= \frac{\partial f^*}{\partial t^*} + \frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial t}.
\end{aligned} \tag{3.2}$$

The last step is to find the transformation of velocities. Velocities are changed in the new system because the $z^*$ axis gets stretched as the free surface deforms. There is no stretching along $x$ and $y$, so

$$u^* = u, \;\; v^* = v. \tag{3.3}$$

To find the transformation of $w$, we follow the path of a particle whose coordinates are $(x_p(t),\, y_p(t),\, z_p(t))$:

$$\begin{aligned}
w^* &= \frac{dz_p^*}{dt} \\
&= \lim_{\Delta t \to 0} \frac{z^*[x_p(t+\Delta t), y_p(t+\Delta t), z_p(t+\Delta t), t+\Delta t] - z^*[x_p(t), y_p(t), z_p(t), t]}{\Delta t} \\
&= \frac{\partial z^*}{\partial x}u + \frac{\partial z^*}{\partial y}v + \frac{\partial z^*}{\partial z}w + \frac{\partial z^*}{\partial t}.
\end{aligned} \tag{3.4}$$

We now have all the elements to understand how the terms in the equations are transformed. As an example, we transform the material derivative: first we substitute equations (3.2)

$$\begin{aligned}
\frac{Df}{Dt} &= \frac{\partial f}{\partial t} + u\frac{\partial f}{\partial x} + v\frac{\partial f}{\partial y} + w\frac{\partial f}{\partial z} \\
&= \frac{\partial f^*}{\partial t^*} + \frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial t} + u\left(\frac{\partial f^*}{\partial x^*} + \frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial x}\right) + v\left(\frac{\partial f^*}{\partial y^*} + \frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial y}\right) + w\frac{\partial f^*}{\partial z^*}\frac{\partial z^*}{\partial z}.
\end{aligned} \tag{3.5}$$

Then we recognize the expression (3.4) of $w^*$ multiplied by $\partial f^*/\partial z^*$ in the last result, so we can write

$$\frac{Df}{Dt} = \frac{\partial f^*}{\partial t^*} + u^*\frac{\partial f^*}{\partial x^*} + v^*\frac{\partial f^*}{\partial y^*} + w^*\frac{\partial f^*}{\partial z^*} \tag{3.6}$$

The transformations of other terms in the equations are more elaborate, but they still rely on the expressions derived here.

Using the described mesh and the sigma trasform is advantageous because it simplifies the application of the impermeability boundary condition at the bottom and the kinematic condition at the surface: they both become [30, p. 83]

$$w^* = 0 \tag{3.7}$$

This can be verified by taking partial derivatives of $z^*$ from eqs. (3.1), substituting them in (3.4) and evaluating the result on the bottom and on the surface. Some computations for matrix elements of the FEM are also simplified. Some disadvantages are the difficulty of treating diffusion terms and possible inaccuracies in representing the effects of density stratification [18]. The issues that emerge while using the sigma transform are presented in section 4.3, along with the results of some test simulations.

## 3.2 Fractional step method

The Navier-Stokes equation contain different operators, with different mathematical properties and physical meanings: advection, diffusion and mass conservation. It would be desirable to deal with each one of them separately, dividing the original complex problem into several simpler ones and solving each with a suitable technique. This is the idea behind the fractional step method. In the following, function $f$ can be velocity or a transported tracer.

The first step is the advection one, treating the hyperbolic part of the equations:

$$\frac{f^a - f^n}{\Delta t} + (\mathbf{u} \cdot \nabla) f = 0. \tag{3.8}$$

Owing to its hyperbolic nature, this step can be solved using the method of characteristics, which in this case means following streamlines. At every node of the discretization we need to calculate $f^a$, the value obtained from $f^n$ through a time-step $\Delta t$ under the action of advection alone. The value of $f^a$ at a node in position $\mathbf{r}$ is then the same value that $f$ had at time $t^n$ in (approximate) position

$$\mathbf{r}' = \mathbf{r} - \mathbf{u}(\mathbf{r}, t^n)\, \Delta t. \tag{3.9}$$

In general, $\mathbf{r}'$ will not be the exact position of a mesh node: then $f^a(\mathbf{r}) = f^n(\mathbf{r}')$ will need to be calculated by interpolation. This step can be implemented in the sigma-transformed domain, because the advection term has the same form in the sigma domain as in the physical domain (see equation (3.6)).

Next, the diffusion step is performed:

$$\frac{f^d - f^a}{\Delta t} - \nabla \cdot (\nu \nabla f) = S_f. \tag{3.10}$$

If $f = \mathbf{u}$, $\nu$ has the meaning of kinematic viscosity and $S_f$ are volume forces, including hydrostatic pressure gradients as discussed in section 2.8. If $f$ is a tracer, $\nu$ is a diffusivity and $S_f$ is a generic source term. The equation to be solved is parabolic and it can be discretized using the finite element method. Alternatively, this step can be solved together with the advection step as an advection-diffusion equation using the FEM, with the SUPG formulation for stability, or with a multidimensional upwind residual distribution method (MURD); both will be described in following sections.

At this point, the time stepping for tracer equations is complete. For the Navier-Stokes equations, we are still missing the (dynamic) pressure gradient in the momentum equation and the enforcement of the solenoidality of the velocity field:

$$\begin{cases} \dfrac{\mathbf{u}^{n+1} - \mathbf{u}^d}{\Delta t} = -\dfrac{1}{\rho_0} \nabla p^{n+1} \\ \nabla \cdot \mathbf{u}^{n+1} = 0 \end{cases} \tag{3.11}$$

These corrections are done in the pressure-continuity step. By taking the divergence of the first equation and using the second to eliminate $\nabla \cdot \mathbf{u}^{n+1}$, the Poisson equation for pressure is obtained:

$$\nabla^2 p^{n+1} = \frac{\rho_0}{\Delta t} \nabla \cdot \mathbf{u}^d. \tag{3.12}$$

(notice that the right hand side is known from the diffusion step). This is an elliptic equation, and it can be solved using the FEM. Once the pressure has been calculated, the velocity can be corrected (or ≪projected≫):

$$\mathbf{u}^{n+1} = \mathbf{u}^d - \frac{\Delta t}{\rho_0} \nabla p^{n+1}. \tag{3.13}$$

This last step is omitted when solving the equations under the hypothesis of hydrostatic pressure, leading to a substantial decrease in the computational time; however, as already discussed in section 2.8, the hydrostatic model is not suitable for all purposes.

The one described above is a quite general procedure for the solution of the Navier-Stokes equations together with equations for tracers in a 3D domain. In the case of a free surface problem, at each time step the surface position and, accordingly, the mesh, must evolve. In Telemac3D, this is done in the so-called wave equation or propagation step: equation (2.57) is solved together with a depth-averaged momentum equation (see [2, p. 78]). The propagation step and the consequent mesh update are performed between the diffusion step and the pressure-continuity step. A non-trivial modelling choice that was not discussed here are boundary conditions to be imposed at each step: see [25], [30] for details.

## 3.3 Finite element method

Given a problem in strong (differential) form

$$\mathcal{L}u - f = 0 \quad \text{on } \Omega, \tag{3.14}$$

the finite element method (FEM) searches for a solution in the form of a linear combination of known shape functions, where the unknowns are the coefficients of the combination and correspond to the values at the nodes. Since the shape functions can be chosen arbitrarily, in general the exact solution cannot be obtained; but a good approximation of it is produced by solving a weak discretized form of the problem.

The problem in weak form corresponding to equation (3.14) is derived by multiplying the equation $\mathcal{L}u = f$ by a test function $v$ and integrating over the domain $\Omega$:

$$\text{find } u \in U : \quad \int_\Omega \mathcal{L}u\, v\, d\Omega = \int_\Omega f\, v\, d\Omega \quad \forall v \in V, \tag{3.15}$$

where $U$, $V$ are appropriate infinite-dimensional function spaces, to be defined specifically for each problem. The key of the above statement is that it must hold for any function in $V$. In the FEM, the infinite-dimensional spaces $U$ and $V$ are replaced by finite-dimensional spaces $U_h$ and $V_h$, of dimensions $N_h$; in particular, in the standard Galerkin formulation, $U_h$ and $V_h$ are chosen to be the same space. We call $\varphi_i(\mathbf{x})$, $i = 1 \dots N_h$, the basis function of space $U_h$. This means that any function $u_h$ in $U_h$ is a linear combination of the functions $\varphi_i$

$$u_h(\mathbf{r}) = \sum_{i=1}^{N_h} u_i \varphi_i(\mathbf{r}), \tag{3.16}$$

where the coefficients $u_i$ are the values of $u$ at the nodes of the mesh, and the same holds for any function $v_h$ in $V_h$:

$$v_h(\mathbf{r}) = \sum_{i=1}^{N_h} v_i \varphi_i(\mathbf{r}). \tag{3.17}$$

The discretized weak form is then

$$\text{find } u_h \in U_h : \quad \int_\Omega \mathcal{L}u_h\, v_h\, d\Omega = \int_\Omega f\, v_h\, d\Omega \quad \forall v_h \in U_h. \tag{3.18}$$

Each function $v_h$ is identified by the coefficients $v_i$ in eq. (3.17). An arbitrary choice of $v_h$ then translates to an arbitrary choice of coefficients: so, by linearity, the only way the integral can be zero for any $v_h$ is that it is zero for any $\varphi_i$. The problem can thus be written as

$$\text{find } u_h \in U_h : \quad \int_\Omega \mathcal{L}u_h\, \varphi_i\, d\Omega = \int_\Omega f\, \varphi_i\, d\Omega \quad \forall i = 1 \dots N_h. \tag{3.19}$$

This is a system of $N_h$ equations.

The shape functions $\varphi_i$ have compact support: the support $T_i$ of $\varphi_i$ is the union of all elements sharing node $i$, and the value of $\varphi_i$ is 1 in node $i$, 0 in all other nodes. This particular choice implies that each of the $N_h$ equations in problem statement (3.19) involves only a small subset of coefficients $u_i$. If operator $\mathcal{L}$ is linear, the solution of the problem is found by solving a linear system with a sparse matrix; in the general case, the system may be non-linear.

For a prism element (Figure 3.1) the basis functions are defined on a reference element, one in which coordinates are $P1 = (1, 0, -1)$, $P2 = (0, 1, -1)$, $P3 = (0, 0, -1)$,

$P4 = (1, 0, 1)$, $P5 = (0, 1, 1)$, $P6 = (0, 0, 1)$. For node 1, the corresponding function must be zero on the plane $x' = 0$ and on the plane $z' = 1$, so

$$\varphi_1 = -\frac{1}{2}x'\,(z' - 1); \tag{3.20}$$

$\phi_3$ must be zero on planes $y' = 1 - x'$ and $z' = 1$, so

$$\varphi_3 = -\frac{1}{2}(x' + y' - 1)(z' - 1). \tag{3.21}$$

The other basis functions are built by analogous considerations:

$$\begin{aligned} \varphi_2 &= -\frac{1}{2}y'\,(z' - 1) \qquad & \varphi_4 &= \frac{1}{2}x'\,(z' + 1) \\ \varphi_5 &= \frac{1}{2}y'\,(z' + 1) \qquad & \varphi_6 &= \frac{1}{2}(x' + y' - 1)(z' + 1). \end{aligned} \tag{3.22}$$

All the basis functions $\varphi_i$ can be split into two factors: one depending on the horizontal coordinates $x'$ and $y'$, $\varphi_i^H$, and one depending on the vertical coordinate $z'$, $\varphi_i^V$:

$$\varphi_i(x', y', z') = \varphi_i^H(x', y')\,\varphi_i^V(z') \tag{3.23}$$

This observation will be useful in section 3.5.2.3 when building 3D MURD schemes.

The transformation from reference coordinates $(x', y', z')$ to physical coordinates $(x, y, z)$ is defined as

$$x(x') = \sum_{i=1}^{6} x_i\,\varphi_i(x') \tag{3.24}$$

and analogously for other components. When such transformation is given as a linear combination of the shape functions, like in this case, elements are called iso-parametric.

### 3.3.1 FEM formulation of the pressure Poisson equation

To describe how the weak formulation of a problem turns into a linear system, we consider the Poisson pressure equation (3.12), which can be written as

$$\nabla^2 p = R. \tag{3.25}$$

The formulation in (3.19) becomes

$$\text{find } p: \quad \int_\Omega \nabla^2 p\,\varphi_i\,d\Omega = \int_\Omega R\,\varphi_i d\Omega \quad \forall i = 1 \ldots N_h \tag{3.26}$$

and, integrating by parts the first term,

$$\text{find } p: \quad -\int_\Omega \nabla p \cdot \nabla\varphi_i\,d\Omega + \int_{\partial\Omega} \nabla p \cdot \mathbf{n}\,\varphi_i\,d\partial\Omega = \int_\Omega R\,\varphi_i d\Omega \quad \forall i = 1 \ldots N_h, \tag{3.27}$$

where $\mathbf{n}$ is the outer normal at the boundary. The solution $p$ is written as linear combination of the shape functions:

$$p(\mathbf{r}) = \sum_j p_j \varphi_j(\mathbf{r}) \tag{3.28}$$

and the surface flux, which is imposed as a Neumann boundary condition, is approximated in Telemac as [30, p. 89]

$$\nabla p \cdot \mathbf{n} = c_1 p + c_0, \tag{3.29}$$

where $c_0$, $c_1$ are functions of the position along the boundary. Substituting eq. (3.29) and (3.28) into (3.27)

$$-\sum_j p_j \int_\Omega \nabla\varphi_j \cdot \nabla\varphi_i \, d\Omega + \sum_j p_j \int_{\partial\Omega} c_1 \varphi_j \varphi_i \, d\partial\Omega + \int_{\partial\Omega} c_0 \varphi_i \, d\partial\Omega = \int_\Omega R \, \varphi_i \, d\Omega \tag{3.30}$$

The unknowns are now the $N_h$ nodal values of pressure $p_j$, and $N_h$ equations have to be solved (one for each value of $i$). The values of $p_j$ can be gathered into column vector $\mathbf{p}$; then matrices $\mathbf{D}$ and $\mathbf{C}_1$ can be defined: their generic elements are

$$D_{ij} = -\int_\Omega \nabla\varphi_j \cdot \nabla\varphi_i \, d\Omega, \quad C_{1,ij} = \int_{\partial\Omega} c_1 \varphi_j \varphi_i \, d\partial\Omega \tag{3.31}$$

and vectors $\mathbf{r}$ and $\mathbf{c}_0$, with generic elements

$$r_i = \int_\Omega R \, \varphi_i \, d\Omega, \quad c_{0,i} = \int_{\partial\Omega} c_0 \varphi_i \, d\partial\Omega. \tag{3.32}$$

The problem has then been transformed into a linear system:

$$\mathbf{A}\mathbf{p} = \mathbf{b} \tag{3.33}$$

with $\mathbf{A} = \mathbf{D} + \mathbf{C}_1$, $\mathbf{b} = \mathbf{r} + \mathbf{c}_0$. Dirichlet boundary conditions are enforced by modifying matrix $\mathbf{A}$ and vector $\mathbf{b}$: if the pressure value in node $k$ has to be imposed, then the $k$-th row of $\mathbf{A}$ becomes all zeroes except for the term on the diagonal, equal to 1, and the $k$-th element of $\mathbf{b}$ is substituted with the value of pressure to be imposed. Notice that $\mathbf{A}$ is symmetric by definition.

## 3.4 Krylov methods for linear systems

The linear system matrices resulting from the finite element method are usually sparse, i.e. most of their elements are zero, because of the mesh connectivity. Very efficient iterative schemes are available for solving linear systems with sparse matrices: some of them are the conjugate gradient method, for symmetric positive definite matrices, and the GMRES (generalized minimal residual) for more generic matrices. They both belong to the class of Krylov subspace methods (see [28] for an introduction). It is important to have at least an idea of how they work, because in some instances their parameters need to be tweaked in order to obtain the desired accuracy. Krylov methods search for the solution of system $Ax = b$ in the Krylov subspace of order $n$

$$\mathcal{K}_n(A, b) = \mathrm{span}(b, Ab, A^2b, \ldots A^{n-1}b) \tag{3.34}$$

The following theorems hold:

**Theorem 1.** *Consider a matrix $A$ and its Jordan canonical form $C$. The polynomial*

$$q(t) = \prod_{j=1}^{N_{eig}} (t - \lambda_j)^{m_j}$$

*is the monic[2] polynomial of minimum order such that $q(A) = 0$ and it is called* minimal polynomial of A*. $N_{eig}$ is the number of distinct eigenvalues of $A$ and $m_j$ is the maximum order among the ones of all Jordan blocks associated to eigenvalue $\lambda_j$ in $C$.*

**Theorem 2.** *Consider the linear system $Ax = b$. Suppose $A$ is non-singular. If $m$ is the degree of the minimal polynomial of $A$, then*

$$x \in \mathcal{K}_m(A, b)$$

For more details about canonical forms and the minimal polynomial see [50]. The degree of the minimal polynomial depends on the eigenstructure of $A$: in particular, it depends on the number of eigenvalues and on the maximum sizes of their associated Jordan blocks. If the minimal polynomial of A has low order $m$, then the solution can be searched for in a subspace of dimension $m \ll n$ ($A \in \mathbb{R}^{n \times n}$), thus potentially with few iterations: this is the strength of Krylov methods. Often, a matrix arising from discretization may not have a favourable structure in such sense; then preconditioning can be used: the original system is premultiplied by preconditioner $P$

$$PAx = Pb \tag{3.35}$$

with $P$ chosen such that matrix $PA$ is as close as possible to having the desired eigenstructure.

To sum up, for the practical purposes of code users, if linear system solvers do not converge or have trouble converging within the prescribed tolerance, the parameters to be modified are maximum number of iterations, dimension of the Krylov subspace and choice of preconditioner.

## 3.5 Solution of the advection-diffusion step

The main issue in the numerical solution of an advection-diffusion problem with the finite element method is the appearance of non-physical oscillations. Two strategies will be presented here: the SUPG formulation and MURD schemes; in both cases, the idea is to evolve the solution using information from upwind.

[2] A monic polynomial has coefficient 1 for the term of highest order.

### 3.5.1 FEM-SUPG formulation

For 1D problems, non-physical oscillations appear in the FEM solution when the Pèclet number

$$\mathrm{Pe} = \frac{uh}{2\nu}, \tag{3.36}$$

where $h$ is a characteristic dimension of the grid, is greater than 1. This can be considered a sort of Reynolds number, and what happens is that when advection dominates over diffusion, or when the mesh element size is large, spurious numerical oscillations appear. A possible, but costly, solution could be to use a more refined mesh; another one is the SUPG method [45, p. 321], that is the Streamline-Upwind Petrov-Galerkin method. Streamline-upwind means that the physics of the problem can be exploited to obtain a more fitting approximation of derivatives, where information coming from upwind is weighted more than information coming from downwind, and Petrov-Galerkin refers to the fact that this weighting is obtained by using different functions as shape functions and test functions (in contrast to the above cited standard Galerkin formulation). The modification to the test functions is

$$\varphi_i^{\mathrm{SUPG}} = \varphi_i + \alpha \frac{\mathbf{u}}{\|\mathbf{u}\|} \cdot \nabla \varphi_i \tag{3.37}$$

and the result is that the upstream part is weighted more than the downstream.

### 3.5.2 MURD schemes

To present MURD (Multidimensional Upwind Residual Distribution) schemes, we start by writing a generic conservation law in conservation form:

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{f}(u) = q(x,t), \tag{3.38}$$

where $\mathbf{f}(u)$ is called flux. Neglecting for now the source term $q$ (its effect will be discussed later) and defining the advection speed $\boldsymbol{\lambda}(u) = \mathbf{f}'(u)$, the equation can be rewritten in quasi-linear form as

$$\frac{\partial u}{\partial t} + \boldsymbol{\lambda}(u) \cdot \nabla u = 0 \tag{3.39}$$

The residual $\phi^T$ on an element $T$ of the discretization, that is the total flux balance on $T$, is defined as

$$\phi^T = \int_T \nabla \cdot \mathbf{f} \, dT = \int_T \boldsymbol{\lambda} \cdot \nabla u \, dT \tag{3.40}$$

MURD schemes [19] have a philosophy that is hybrid between finite elements and finite volumes. The motivation for building them is to obtain methods that apply proper upwinding while adding just the needed amount of numerical dissipation.

The finite element method, as already described, is based on a discretization of the weak form of the equations. The solution is computed at each node of the discretization, and (in the standard implementation) it is interpolated as a continuous function on the

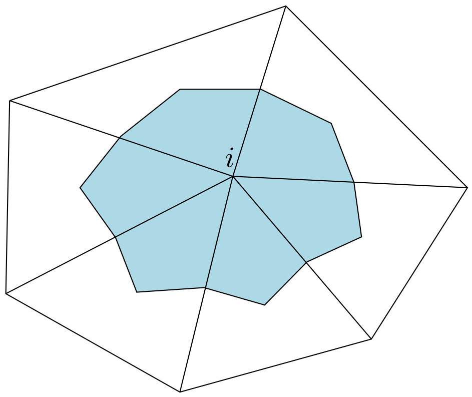


Figure 3.4: Triangles sharing node $i$ and dual cell $S_i$ (blue region)

domain. This is an intrinsically multidimensional method. A drawback is that the dissipation, added through the term $\alpha$ in eq. (3.37) to obtain numerical stability, must be tuned properly.

The finite volume (FV) method is based on a discretization of the integral form of the equations. The solution is computed cell by cell by evaluating a flux balance, and it is considered discontinuous at the interfaces between cells: the cell-average solution in cell T is computed as

$$u_T^{n+1} = u_T^n - \frac{\Delta t}{S_T}\phi^T, \tag{3.41}$$

where $S_T$ is the total area (or volume, in 3D) of cell $T$. The FV method, being based on the integral form, can be made to ensure conservation in a natural way. Extending finite volume schemes to multiple dimensions without introducting spurious directional biases is not straightforward, though.

Residual-distributing schemes combine the FEM piecewise linear approximation with the flux balance typical of FV methods. As the name of the scheme suggests, the residual on a cell is distributed among the cell nodes: we will now see how. For simplicity, an inconsistent formulation, equivalent to mass lumping, is presented here. Mass-lumping consists in approximating matrix $\mathbf{M}$ in definitions (3.31) by a diagonal matrix in which every element on the diagonal is the sum of all elements of the corresponding row. The evolution in time is then

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{S_i}\sum_T \beta_i^T \phi^T, \tag{3.42}$$

where $S_i$ is the area of the dual cell corresponding to node $i$, that is one third of the sum of the areas of all triangles meeting in node $i$ (see Figure 3.4). Coefficients $\beta_i^T$ determine how the distribution is done, or how much of the residual in cell $T$ has to be attributed to node $i$: the idea is that at every point the solution must evolve according to data coming from upwind. The sum of all coefficients $\beta$ for a triangle must be one for consistency.

The actual form of residual $\phi^T$ will now be derived for 2D elements (triangles); then, two desirable properties of methods will be introduced together with the conditions for obtaining them and the resulting choices for schemes; lastly, the extension of such schemes to 3D will be briefly described. With linear shape functions on triangles, deriving eq. (3.16) we have

$$\nabla u = \sum_{i=1}^{3} u_i \, \nabla \varphi_i. \tag{3.43}$$

Shape function $\varphi_i$ is equal to one in node $i$ and is zero an all of the opposite edge. The direction of the gradient of a function is the one of maximum growth, and its magnitude is the corresponding directional derivative. Then $\nabla \varphi_i$ must be perpendicular to the edge opposite node $i$, and its magnitude must be the inverse of the distance between the cited node and edge: this distance is the height of the triangle with respect to base $\mathbf{l}_{jk}$ (for example node 3 and edge vector $\mathbf{l}_{12}$ in Figure 3.5), equal to twice the area $S_T$ divided by the norm of $\mathbf{l}_{jk}$. By defining the unit vector $\hat{\mathbf{z}}$, normal to the plane of the triangle and pointing out of the page, and vectors $\mathbf{n}_i = \hat{\mathbf{z}} \times \mathbf{l}_{jk}$, normal to the edge opposite node $i$, we have

$$\nabla \varphi_i = \frac{\mathbf{n}_i}{2\, S_T}. \tag{3.44}$$

The gradient of $u$ becomes

$$\nabla u = \frac{1}{2\, S_T} \sum_{i=1}^{3} u_i \mathbf{n}_i, \tag{3.45}$$

that is constant on $T$. We can thus take the gradient out of the integral in equation (3.40) and consider the average of $\boldsymbol{\lambda}$ over $T$, that we will call $\overline{\boldsymbol{\lambda}^T}$. The resulting expression for the residual is

$$\phi_T = S_T \, \overline{\boldsymbol{\lambda}^T} \cdot \nabla u = \frac{1}{2} \sum_{i=1}^{3} u_i \, \overline{\boldsymbol{\lambda}^T} \cdot \mathbf{n}_i = \sum_{i=1}^{3} u_i \, k_i, \tag{3.46}$$

with

$$k_i = \frac{1}{2} \overline{\boldsymbol{\lambda}^T} \cdot \mathbf{n}_i. \tag{3.47}$$

Figure 3.5 shows that $k_i$ determines whether a node is an upstream or a downstream one: nodes 1 and 2 are downstream, and $k_1 > 0$, $k_2 > 0$; node 3 is upstream, and $k_3 < 0$. To obtain an upwind scheme, the residual on the cell should be distributed to the downstream nodes, so that the solution in those points evolves based on information coming from upstream.

In addition to being truly upwind, there are two additional good properties that a scheme can have: positivity, related to stability, and linearity preservation, related to accuracy.

A scheme is called positive if the result of time-stepping on a node can be expressed as

$$u_i^{n+1} = \sum_k c_k u_k^n \tag{3.48}$$

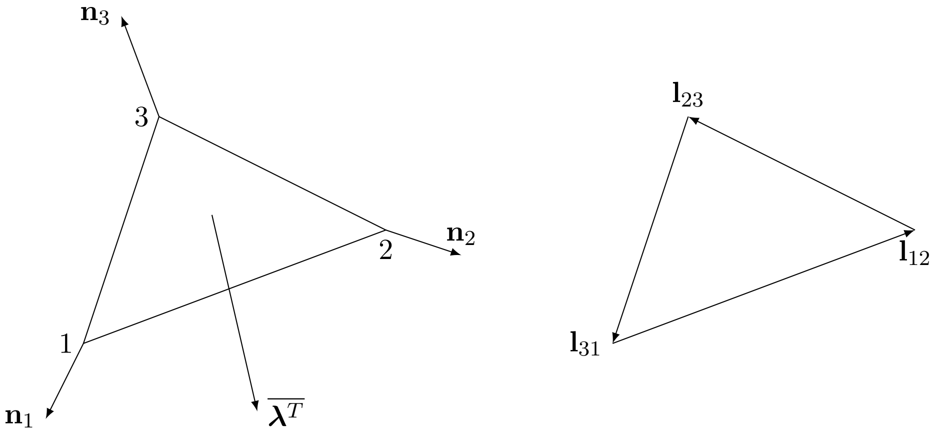


Figure 3.5: Normals (left) and edge vectors $\mathbf{l}_{ij}$ (right) on a 2D element. Vectors are scaled by a factor 3 for drawing convenience: actual length is as much as the opposite edge's

with non-negative coefficients $c_k$. This ensures that no new maxima or minima will appear in the solution, which means that non-physical oscillations will not be generated by the numerical scheme. Here we are considering an homogeneous equation: for a general equation with non-zero source term, the source term may generate physical oscillations that the scheme will be able to capture. The point is avoiding spurious oscillations. By putting together eqs. (3.42) and (3.46), the time step can be recast in the above form:

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{S_i} \sum_T \beta_i^T \sum_{j=1}^{3} u_j^n k_j \tag{3.49}$$

that is, as a linear combination of the values in the stencil at the previous time step. An easy to check, sufficient condition for positivity is local positivity, where the summation is restricted to the nodes of a single triangle, and the obtained coefficients are all required to be positive:

$$\begin{aligned} u_i^{n+1,T} &= u_i^n - \frac{\Delta t}{S_i}\beta_i^T (k_i u_i^n + k_j u_j^n + k_k u_k^n) \\ &= \left(1 - \frac{\Delta t}{S_i}\beta_i^T k_i\right) u_i^n + \left(-\frac{\Delta t}{S_i}\beta_i^T k_j\right) u_j^n + \left(-\frac{\Delta t}{S_i}\beta_i^T k_k\right) u_k^n \\ &= c_i u_i^n + c_j u_j^n + c_k u_k^n \end{aligned} \tag{3.50}$$

The condition is fulfilled by careful choices of time-step and of coefficient $\beta_i^T$. For example, consider node 3 in Figure 3.5. As already observed, $k_3 < 0$, while $k_1$ and $k_2$ are positive. Then the only way of obtaining positivity is to choose $\beta_3^T = 0$: this is a reasonable choice, because node 3 is upwind with respect to $\overline{\boldsymbol{\lambda}^T}$, so no information will reach node 3 from the inside of this triangle during the time step. If the direction of

$\overline{\boldsymbol{\lambda}^T}$ were the opposite, then we would have $k_1 < 0$, $k_2 < 0$ and $k_3 > 0$: node 3 would be a downstream one, and the only condition required for positivity would be on the time step.

A scheme is linearity-preserving if, given an exact steady state solution that is linear, it is able to obtain it exactly. A linearity-preserving scheme allows to reach second order accuracy. The condition to be fulfilled to obtain this property is

$$\beta_i^T \phi^T \to 0 \; \forall i \quad \text{as} \quad \phi^T \to 0, \tag{3.51}$$

which requires coefficients $\beta_i^T$ to be bounded by a finite constant:

$$\exists C : \quad \max_T \max_{j \in T} |\beta_j^T| < C. \tag{3.52}$$

Both properties are very desirable, but the following theorem from Godunov [34, p. 174] holds:

**Theorem 3.** *A linear scheme cannot be both positive and linearity-preserving.*

A scheme is linear if the quantity $\beta_i^T \phi^T$ depends linearly on the values $u_j^n$ in eq. (3.42).

#### 3.5.2.1 MURD-N scheme, 2D

We start by noting that since the vector identity $\mathbf{l}_{12} + \mathbf{l}_{23} + \mathbf{l}_{31} = 0$ holds for the edges of a triangle, by linearity of the vector product and of the scalar product by a constant we also have

$$\sum_{i=1}^{3} k_i = 0. \tag{3.53}$$

Then, by exploiting $k_3 = -k_1 - k_2$:

$$\begin{aligned} \phi^T &= k_1 u_1 + k_2 u_2 + k_3 u_3 \\ &= k_1(u_1 - u_3) + k_2(u_2 - u_3) = -k_1(u_3 - u_1) - k_2(u_3 - u_2) = \dots \\ &= \sum_{i=1}^{3} \sum_{j=1}^{3} \gamma_{ij}(u_i - u_j) \end{aligned} \tag{3.54}$$

where the equivalent, alternative expressions on the second line can be used (selected by switches in the code) in such a way that coefficients $\gamma_{ij}$ are all non-negative. In the MURD-N scheme the choice is to split the residual by taking only the part containing the contributions of node $i$ in the last expression:

$$\begin{aligned} \beta_i^T &= \frac{\sum_{j=1}^{3} \gamma_{ij}(u_i - u_j)}{\phi^T}, \\ \beta_i^T \phi^T &= \sum_{j=1}^{3} \gamma_{ij}(u_i - u_j). \end{aligned} \tag{3.55}$$

The second expression is the one actually used in codes: it saves operations and avoids possible divisions by zero; the first is useful to understand what happens to coefficients $\beta_i^T$.

To help clarify how the scheme works, we can consider an example based again on Figure 3.5. Since $k_1 > 0$, $k_2 > 0$ and $k_3 < 0$, the correct form of equation (3.54) in this case is

$$\phi^T = k_1(u_1 - u_3) + k_2(u_2 - u_3) \tag{3.56}$$

or, collecting coefficients $\gamma_{ij}$ in a matrix $\boldsymbol{\gamma}$,

$$\boldsymbol{\gamma} = \begin{bmatrix} 0 & 0 & k_1 \\ 0 & 0 & k_2 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.57}$$

all the elements of matrix $\boldsymbol{\gamma}$ being non-negative. We can now apply equations (3.55):

$$\begin{aligned} \beta_1^T \phi^T &= k_1(u_1 - u_3) \\ \beta_2^T \phi^T &= k_2(u_2 - u_3) \\ \beta_3^T \phi^T &= 0 \end{aligned} \tag{3.58}$$

confirming that $\beta_3^T$ must be equal to zero for proper upwinding. The contribution of triangle $T$ to the mass-lumped evolution of $u_1$ is then (3.50)

$$\begin{aligned} u_1^{n+1,T} &= u_1^n - \frac{\Delta t}{S_i} k_1(u_1^n - u_3^n) \\ &= \left(1 - \frac{\Delta t}{S_i} k_1\right) u_1^n + \frac{\Delta t}{S_i} k_1 \, u_3^n, \end{aligned} \tag{3.59}$$

so that the scheme is locally positive under a condition on the time step; the same could be verified for the evolution of $u_2$ and (trivially) $u_3$.

The scheme is then conditionally positive [25, p. 189]. Being linear and positive, though, it cannot be linearity-preserving, owing to Godunov's theorem. This can be understood from the first equation in (3.55): even when $\phi^T$ is zero, the single terms in the summation on index $i$ that form $\phi^T$ can be non-zero, leading to infinite values of $\beta_i^T$. Thus such coefficients are not bounded.

The N scheme is known as optimum first order scheme, being the least diffusive first order scheme [2, p. 114].

#### 3.5.2.2 MURD-PSI scheme, 2D

The MURD-PSI scheme [2, p. 118] is a modification of the MURD-N scheme done in order to make it linearity-preserving. It consists in applying a limiter to coefficients $\beta_i^T$:

$$\beta_i^{T,PSI} = \max(\min(\beta_i^{T,N}, 1), 0) \tag{3.60}$$

This way the coefficients are bounded, and the scheme is no more linear.

#### 3.5.2.3 Extension to 3D

The extension to 3D of the schemes described above is discussed in [2, p. 116]. Essentially, when dealing with prisms instead of triangles, the calculation of the gradient changes, and the possibility of splitting shape functions into horizontal and vertical parts (see eq. (3.23)) is exploited:

$$\begin{aligned}\phi^P &= \int_P \boldsymbol{\lambda} \cdot \nabla u \, dP = \int_P \left( \lambda_x \frac{\partial u}{\partial x} + \lambda_y \frac{\partial u}{\partial y} + \lambda_z \frac{\partial u}{\partial z} \right) dP \\ &= \sum_{i=1}^{6} \frac{\partial \varphi_i^H}{\partial x} u_i \int_P \lambda_x \varphi_i^V \, dP + \sum_{i=1}^{6} \frac{\partial \varphi_i^H}{\partial y} u_i \int_P \lambda_y \varphi_i^V \, dP + \sum_{i=1}^{6} \frac{\partial \varphi_i^V}{\partial z} u_i \int_P \lambda_z \varphi_i^H \, dP \\ &= \sum_{i=1}^{6} u_i (k_i^H + k_i^V),\end{aligned} \tag{3.61}$$

with

$$k_i^H = \frac{\partial \varphi_i^H}{\partial x} \int_P \lambda_x \varphi_i^V \, dP + \frac{\partial \varphi_i^H}{\partial y} \int_P \lambda_y \varphi_i^V \, dP, \quad k_i^V = \frac{\partial \varphi_i^V}{\partial z} \int_P \lambda_z \varphi_i^H \, dP$$

The expression is then recast, as in the 2D case, in the form

$$\phi^P = \sum_{i=1}^{6} \sum_{j=1}^{6} \gamma_{ij} (u_i - u_j) \tag{3.62}$$

where coefficients $\gamma_{ij}$ are chosen from the values $k_i^H + k_i^V$ in such a way that positivity is guaranteed. This way, the 3D N scheme is obtained; then, the PSI scheme follows by limiting coefficients $\beta_i^P$.

# Chapter 4

# Preliminary tests

In this chapter, the basic setup of simulations is described. Some simple test cases are run to verify expected qualitative behaviours and to make specific modelling choices in view of the subsequent steps.

## 4.1 Setup

Meshes are made using the software BlueKenue[1]. Simulations are run in Telemac 3D[2]; in most cases, some small modifications are needed to adapt the standard Fortran subroutines of the code to the case at hand, especially for defining the mesh vertical layers and for setting up initial conditions on temperature profiles and particles. ParaView[3] is used for post-processing and visualization of data. The conversion of solution files to .vtk files is done using a Python script[4]; files from simulations with particles are converted using a script written by the author[5]. Unless otherwise specified, the MURD-PSI scheme is used for advection-diffusion (as recommended in the Telemac documentation) and the $\kappa - \epsilon$ model for turbulence.

## 4.2 Effects of stratification on wind-induced mixing

In this test case, a constant and uniform wind blows over a rectangular closed domain of uniform depth. An initial thermal stratification is imposed: stable (lower temperatures on the bottom) in the first simulation and unstable in the second; see Figure 4.1. They are idealized, piecewise linear functions: the stable one models a typical summer situation, with small temperature gradients in the lower layers and a quite sharp variation close to the surface; the unstable one is chosen with same magnitudes of temperature

[1] https://nrc.canada.ca/en/research-development/products-services/software-applications/blue-kenuetm-software-tool-hydraulic-modellers

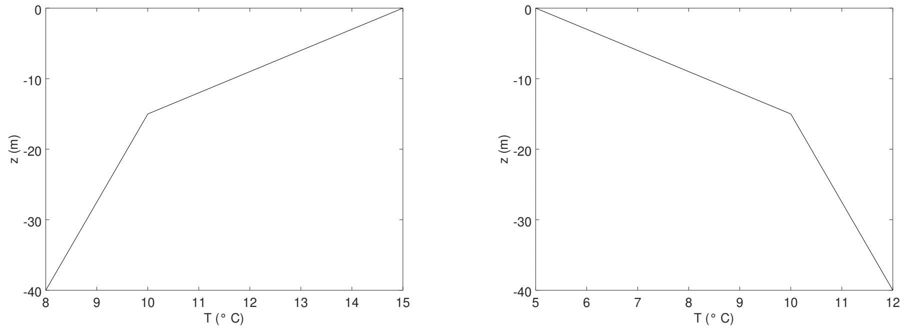
[2] http://opentelemac.org/

[3] https://www.paraview.org/

[4] `sel2vtk.py` and `sel2vtk_bin.py` by Pat Prodanovic: https://github.com/pprodano/pputils

[5] `tecplot2csv.py`: https://github.com/marcogambarini/particles-telemac

variation, for comparison. The $k-\epsilon$ turbulence model is employed. Zero velocity is imposed at time zero. It is verified that the action of wind is to induce mixing, by introducing kinetic energy in the domain, and that a stable thermal stratification inhibits vertical mass transport, while an unstable one promotes it. Particle paths are followed from the initial distribution shown in Figure 4.2(a).



(a) Stable stratification

(b) Unstable stratification

Figure 4.1: Initial temperature profiles

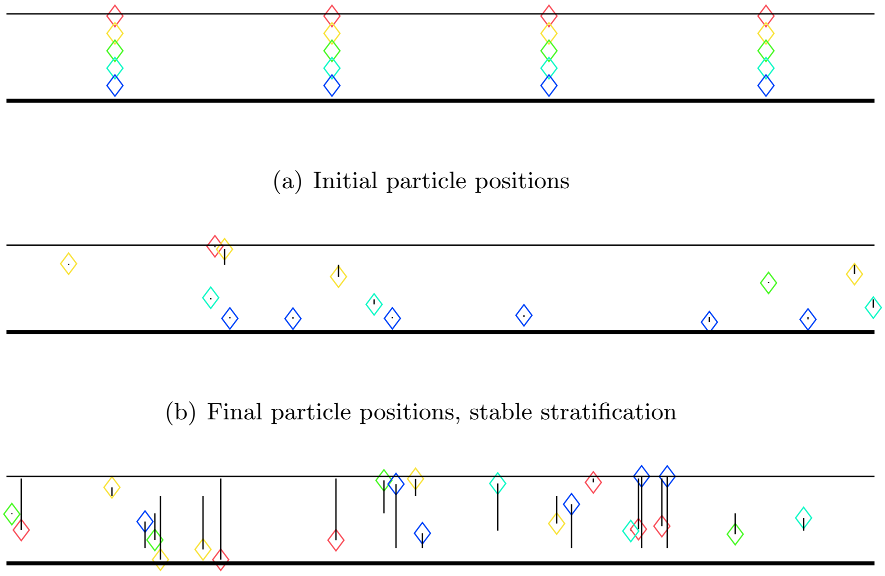
The dimensions of the domain are 1000 m × 200 m, with depth 40 m; the typical edge length of triangles is 20 m. Wind blows along the $x$ direction at 10 m/s. A total of 50 particles are introduced in the domain. 20 vertical surfaces (19 layers) are used, positioned by a quadratic law so that layers on the bottom are thicker than at the surface, in order to be able to better represent the wind interaction. This implies a slight modification to the sigma transform as described in section 3.1: surfaces are not equally spaced, but they are placed at constant sigma coordinates. In this case all surfaces are (initially)[6] horizontal planes because the bottom is flat; the more general case of variable bathymetry will be treated in the following test cases. 3600 time steps of 45 seconds each are performed, for a total simulated time of 45 hours. This time period was chosen as long enough to contain the time needed to reach a steady state wind-induced circulation. In numerical experiments on the stable configuration, the time history of

$$\int_{Syz,u>0} u\,dS \tag{4.1}$$

was observed to tend to a steady state after about one day. $Syz$ is a slice of the domain in the $y-z$ plane, and, as reported in the equation, only positive velocities are considered: the value can be interpreted as a measure of internal mass transport when circulations generate, and this definition is analogous to the one of lateral transport defined in [6]. The idea is thus to make the simulation at least long enough to have fully developed circulations. The computational time is about 6 minutes on a single core.

[6] After some time, the surfaces are subjected to a slight tilting because of wind set-up.

In the stable case, an almost closed clockwise circulation appears in the layers close to the surface, while in the lower layers particles move more erratically and at lower velocity. Particles released on the surface are dragged to the right by the wind, then experience downwelling at the right wall and descend to a limited depth due to buoyancy, being warmer and less dense than the surrounding fluid; they then start travelling towards the left until they either go back to the surface or reach the left wall. As seen in Figure 4.2(b), particles starting close to the surface (in red) and particles starting close to the bottom (in blue) do not mix: there is some mixing in the upper layer, but no particle travels from the surface to the bottom, nor vice versa. This is clear from the vertical lines marking the total vertical displacement, which are all very short. The final temperature profile, shown in Figure 4.3, is much smoother than the initial one, showing that some mixing has happened.



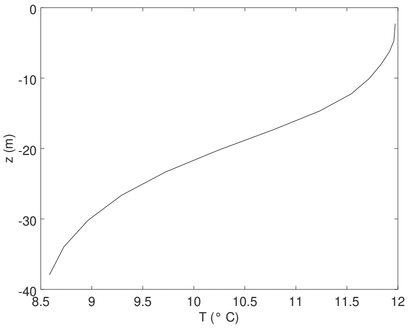
(a) Initial particle positions

(b) Final particle positions, stable stratification

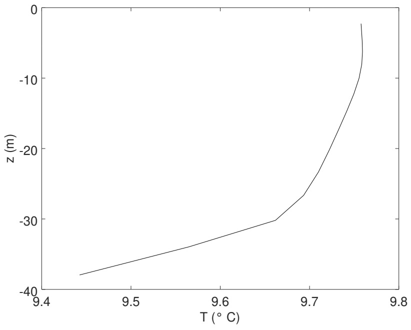
(c) Final particle positions, unstable stratification

Figure 4.2: Particle positions, projected onto the $x - z$ plane (portions of domain). In (b) and (c), lines represent the vertical components of particle displacements; the horizontal components are not drawn for clarity of visualization

In the unstable case, a full overturn happens: particles travel in the whole domain and through the whole depth, as seen in Figure 4.2(c), where vertical displacements are much larger than in the previous case. After about 2 hours, the initially unstable stratification becomes a neutral one and then a stable one, until the final temperature profile is reached (Figure 4.3). The overturning in the basin is quite fast compared to the timescale of the simulation. To better investigate this an additional, more detailed



(a) Stable initial stratification



(b) Unstable initial stratification

Figure 4.3: Final temperature profiles

simulation, was carried out for the first 2.5 hours with a time step of 5 seconds. Also in this case, particles on the surface are trasported by the wind and reach the right boundary, but they are now able to immediately reach the bottom: because of the temperature profile, they are denser than the surrounding water. A pattern of vertical circulations appears in the domain (pathlines shown in Figure 4.4), and the temperature profile is correspondingly overturned. Note that, for analyzing transport patterns in unsteady flows, pathlines, i.e. particle trajectories, are usually more significant that streamlines, that are integral curves of the instantaneous velocity field; this is the reason why only the former are shown here.

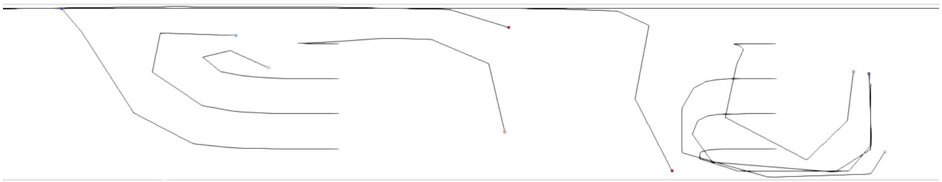


Figure 4.4: Pathlines from t=0 to t=1.2 h, unstable stratification; central portion of the domain. Particles are drawn at their final position

As a comparison, this test case was also run using the hydrostatic model, all other parameters being exactly the same. While qualitatively the same patterns are obtained, velocities close to the right wall, where downwelling takes place, are markedly different: in particular, the absolute value of the vertical component is larger, leading also to a larger circulation. A comparison is shown in Figure 4.5. The difference is due to the fact that the hydrostatic model does not take into account vertical accelerations: so, where vertical motions are important the results can be very different from the ones

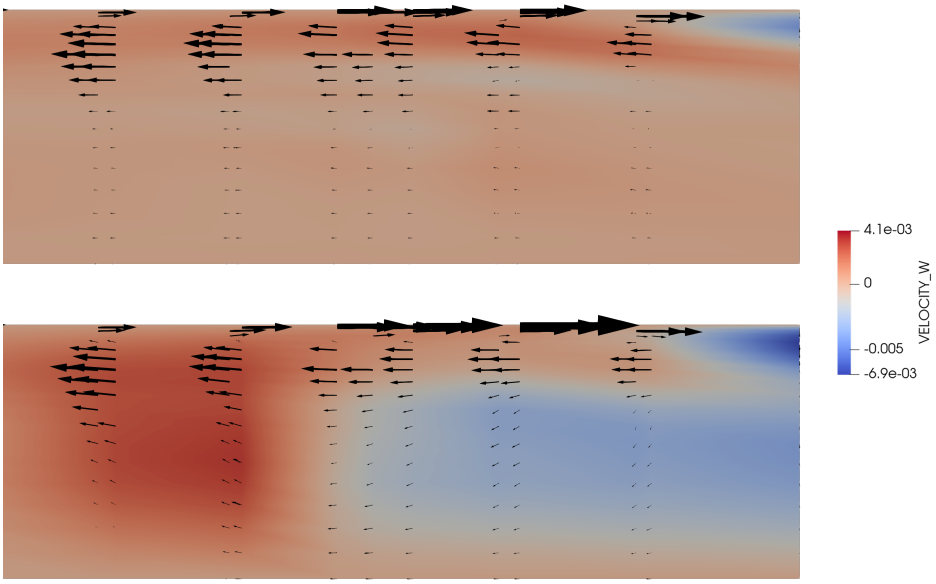


Figure 4.5: Colormap for the vertical component of velocity at the right (downwind) wall: non-hydrostatic (above) and hydrostatic (below) models

obtained using the non-hydrostatic model.

This test case proves the ability to simulate stratified flows obtaining results that are at least qualitatively reasonable with regards to mixing. Quantitative data can be extracted from the results of simulations: the extent of mixing can be quantified by analyzing statistically the behaviour of particles; as an alternative, a fictitious passive tracer can be introduced, as done in [40] and [43]. If the tracer is set to 1 in a region of the domain, e.g. the hypolimnion, and 0 elsewhere, then the time evolution of the tracer concentration is an indicator of how much hypolimnetic water mixes with epilimnetic water. Some relevant integral quantities are the first and second moment of the tracer distribution $c(x, y, z, t)$ along the vertical:

$$\begin{aligned} z_b &= \frac{\int_\Omega c\, z\, d\Omega}{\int_\Omega c\, d\Omega}, \\ d_z &= \sqrt{\frac{\int_\Omega c\,(z - z_b)^2\, d\Omega}{\int_\Omega c\, d\Omega}}, \end{aligned} \tag{4.2}$$

and the fraction of the total mass (or volume) of tracer that was advected through the thermocline, or some otherwise defined separation surface: this last indicator was used, for instance, in [43].

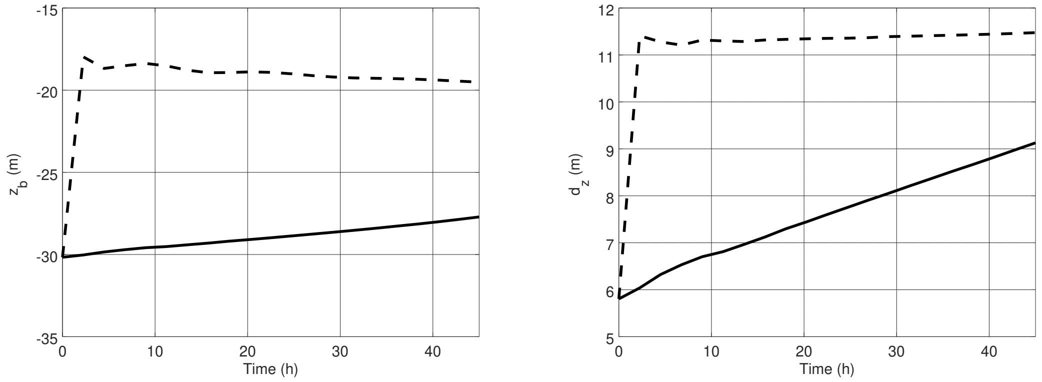


Figure 4.6: Evolution in time of first and second moment of the fictitious tracer distribution. Solid line: stable initial stratification; dashed line: unstable initial stratification

## 4.3 Mesh layers and hydrostatic inconsistency

Using a mesh coherent with the sigma transformation, i.e. made up of layers that adapt to the bottom and free surface shapes, can lead to an error in the computation of horizontal hydrostatic pressure gradients. Interpolation errors on the density field are amplified when the gradient is computed [18]: spurious gradients can appear even in presence of a density (or, correspondingly, temperature) profile that depends only on z. In this section, an explanation of this problem is presented along with some ways to mitigate it.

Hydrostatic pressure was derived in equation (2.67). Its gradients appear in the momentum equations; we will focus on the horizontal ones. Neglecting the gradient of atmospheric pressure and focusing on the $x$ component:

$$\frac{\partial p_{\text{hyd}}}{\partial x} = \rho_0 g \, \frac{\partial S}{\partial x} + \frac{\partial}{\partial x} \int_z^{S(x,y)} \Delta\rho \, g \, d\zeta. \tag{4.3}$$

The first term does not create particular problems; the second is rewritten using the Leibniz theorem:

$$\frac{\partial}{\partial x} \int_z^{S(x,y)} \Delta\rho \, g \, d\zeta = \int_z^{S(x,y)} \frac{\partial \Delta\rho}{\partial x} \, g \, d\zeta + \Delta\rho(x, y, S(x,y), t) \, g \frac{\partial S}{\partial x}. \tag{4.4}$$

In presence of a density field that varies only along $z$, the exact value of the first term in the last expression would be zero and, unless gradients of the free surface are present, there would be no horizontal pressure gradient at all. But when the first term is computed on the mesh, complications arise. The difficulty is not in the evaluation of the integral, that is straightforward once $\partial\Delta\rho/\partial x$ is known at each point because grid points are aligned vertically, but rather in the evaluation of the derivative. To understand this, we need to look at how derivatives are computed on finite elements.

A generic quantity $f$ has definite values $f_j$ at the mesh nodes, and its value in other points is obtained by interpolation using shape functions $\varphi_j$:

$$f(\mathbf{r}) = \sum_{j=1}^{N_p} f_j \, \varphi_j(\mathbf{r}), \tag{4.5}$$

where $\varphi_j$ is the shape function whose value is 1 in node $i$ and 0 in all other nodes. Its support is $T_i$, the set of all elements to which node $i$ belongs (see section 3.3). With elementwise linear shape functions, partial derivatives are not well defined at the nodes, because the same derivative may in general have different values depending on the element in which it is evaluated (in 1D this would simply mean that the left derivative does not in general coincide with the right derivative). Partial derivatives at node $i$ can then be estimated by weighting their values on the elements of $T_i$ [30, p. 91]:

$$\frac{\partial f}{\partial x}(\mathbf{r}_i) = \frac{\displaystyle\int_\Omega \frac{\partial f}{\partial x} \varphi_i \, d\Omega}{\int_\Omega \varphi_i d\Omega} = \frac{\displaystyle\sum_{T \in T_i} \int_T \frac{\partial f}{\partial x} \varphi_i \, d\Omega}{\int_\Omega \varphi_i d\Omega} \tag{4.6}$$

where the sum is over all elements $T$ in $T_i$. In each element, $f$ is differentiable: the partial derivative of eq. (4.5) can be computed as

$$\left.\frac{\partial f}{\partial x}\right|_T = \sum_{j=1}^{N_{p,T}} f_j \, \frac{\partial \varphi_j}{\partial x}, \tag{4.7}$$

where $N_{p,T}$ is the number of points of element $T$ (6 for a prism, 3 for a triangle). Then eq. (4.6) can be rewritten as

$$\frac{\partial f}{\partial x}(\mathbf{r}_i) = \frac{\displaystyle\sum_{T \in T_i} \sum_{j=1}^{N_{p,T}} f_j \int_T \frac{\partial \varphi_j}{\partial x} \varphi_i \, d\Omega}{\int_\Omega \varphi_i d\Omega} \tag{4.8}$$

The point is that the computational stencil for the derivative at a node B (see Figure 4.7, which depicts a side of the stencil in a 2D case) includes all, and only, the nodes of elements that B belongs to. The most natural way to calculate a derivative along the horizontal direction would be by finite differences: after obtaining the value of the function in B' by interpolation of values in E, F and possibly D, the value in B can be subtracted from it and the result divided by the distance BB'. What happens inside the FEM is quite similar to this, because the derivative in B is calculated as a linear combination of the values of the function in all nodes of elements B belongs to. For a function varying only along $z$, linear finite element approximations are exact only for linear functions [18]; for all other functions, spurious gradients appear. The error on calculating these gradients, anyway, goes to zero if the mesh is refined, so the method is still consistent, unless there are elements with at least one point on the lower plane

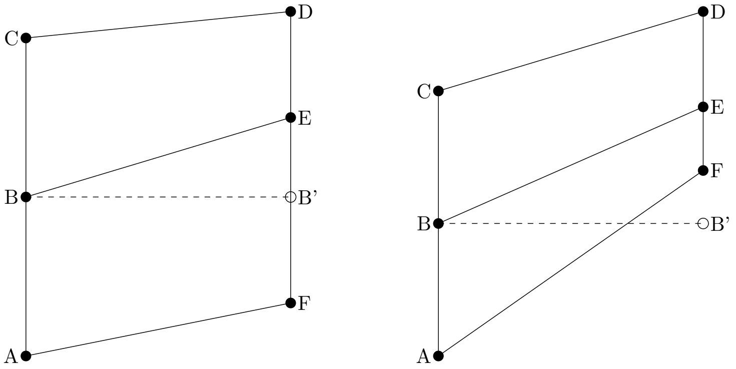


Figure 4.7: Part of the stencil for the computation of $\frac{\partial \Delta\rho}{\partial x}$ in point B: consistent (left) and inconsistent (right) elements; projected on plane xz

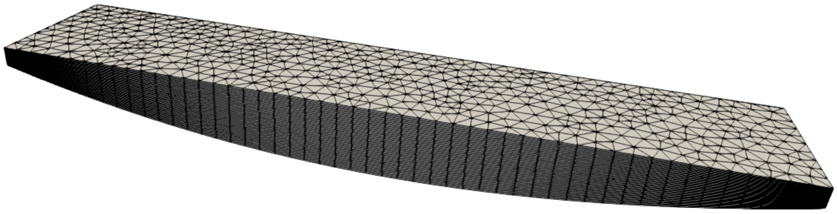


Figure 4.8: Geometry of the hydrodynamic inconsistency test case

at higher elevation than one on the higher plane[7] (see Figure 4.7 on the right). In this case, the value of the function in B' has to be obtained by extrapolation, as opposed to interpolation, and the method is inconsistent in the sense that the error does not decrease as the resolution is increased. In the opposite limit case in which mesh planes are horizontal, horizontal derivatives can potentially be calculated exactly, provided that the density field was interpolated properly.

Some numerical experiments in Telemac3D are presented in [18]. The geometry is a basin with a bottom elevation varying quadratically in the $x$ direction and constant in $y$, as in Figure 4.8. When a standard sigma mesh is used, the main results are the following:

- From a linear initial stratification, no spurious velocities appear and the density profile does not change.

[7] In the context of sigma meshes, the term ≪plane≫ is used loosely to refer to surfaces between layers, which in general are not planes.

- From a parabolic initial stratification, the density profile is unchanged, but spurious velocities of order $10^{-3}$ m/s appear.

- From a discontinuous initial stratification, the density profile is smeared and spurious velocities of order $10^{-1}$ m/s appear. Once the vertical profile becomes smoother, spurious velocity decrease to about $10^{-2}$ m/s.

When the mesh layers are modified such that a plane coincides exactly with the density discontinuity surface, and the sigma transform is used from the bottom to the plane and from the plane to the free surface separately, no spurious velocity appear and the density profile does not change.

The magnitudes of the spurious velocities appearing in the simulations of the article cited above are of the order of the flow velocities expected in the lake. This means that if no action is taken to mitigate this issue, results of simulations would be hard to interpret at best (because one would have to distinguish between physical and spurious velocities) and completely wrong at worst. Then, it was decided to perform some simulations on a similar geometry, with dimensions close to the characteristic ones of the northern trunk of lake Garda, and with a temperature/density profile that varies linearly over a distance equal to a mesh layer thickness, in order to quantify the extent of the problem for the case at hand, eventually extending the analysis to the entire lake geometry. The geometry is as depicted in Figure 4.8: a rectangle of dimensions 4000 m × 800 m and depth varying quadratically between 100 m and 350 m along the longest edge. This is a simplified model of the bathymetry reported in Figure 2.5. The initial condition is zero velocity with a sharp temperature variation from 8°C to 24°C between 30 m and 20 m depth; 100 time steps of 10 s each are simulated. With 35 layers (36 planes) and a standard sigma transform, the error, evaluated as the maximum (spurious) velocity, is $\|\mathbf{u}\|_{\max} = 0.025$ m/s. When an horizontal plane is positioned at depth 50 m with the sigma transform applied above and below it (5 layers above and 30 layers below, thus with 10 m average layer thickness), the error is of order $\|\mathbf{u}\|_{\max} \sim 10^{-9}$ m/s. The results are thus promising; but, when the same stratification is applied to the lake, the situation gets more complicated.

During an early test on the uniform grid of the lake with double sigma transform, it became clear that spurious velocities appear because of the presence of a large number of elements that do not satisfy the consistency condition (see Figure 4.9). It was then decided to try and build a new 2D grid to avoid the presence of inconsistent elements as much as possible. This implies that element sizes should decrease where the bottom is steep and that they may increase in regions of flat bottom[8]. The consistency conditions for two nodes of the same element separated by a distance $L$ on the horizontal and on two consecutive vertical levels $i$ and $i+1$ reads

$$z_i(x+L) < z_{i+1}(x) \tag{4.9}$$

[8] Of course making 2D elements smaller everywhere while keeping the same vertical discretization would solve the problem, but it would also increase the computational time in an unacceptable way.

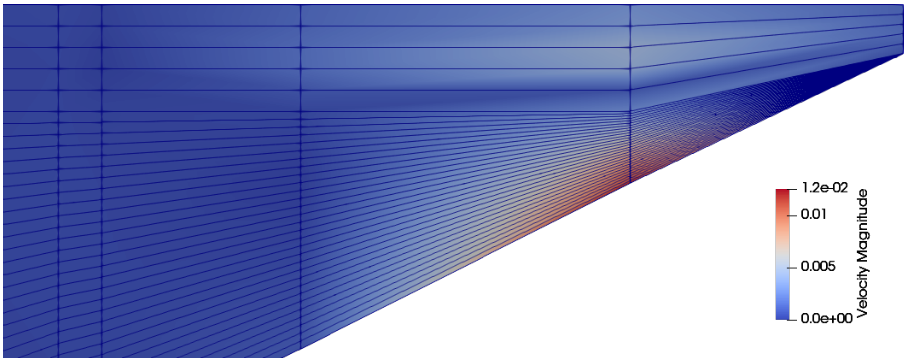


Figure 4.9: Inconsistent elements and spurious velocities, uniform grid, double sigma transform (early test)

For simplicity, we will now work on a vertical plane $x - z$. Using the same notation as in section 3.1, the physical elevation $z_i(x)$ of plane $i$ in position $x$ can be written in terms of the sigma elevation $z_i^*$, that is independent of $x$, as

$$z_i(x) = B(x)\,(1 - z_i^*) \tag{4.10}$$

Using a linear approximation for the bottom, we have

$$z_i(x+L) = B(x+L)(1 - z_i^*) \approx \left[B(x) + \frac{dB}{dx}\,L\right](1 - z_i^*) \tag{4.11}$$

Putting everything together and substituting the derivative along $x$ with the gradient, the condition becomes

$$L < \frac{|B|}{\|\nabla B\|}\,\frac{z_{i+1}^* - z_i^*}{1 - z_i^*} \tag{4.12}$$

For the layer at the free surface, $z_{i+1}^* = z_N^* = 1$ and the condition is $L < |B|/\|\nabla B\|$, always verified. For the layer at the bottom, $z_i^* = z_0^* = 0$ and we have the condition $L < z_1^*\,|B|/\|\nabla B\|$. For equally spaced vertical planes, $z_i^* = i/N$ and

$$L < \frac{|B|}{\|\nabla B\|}\,\frac{1}{N - i} \tag{4.13}$$

Thus we may accept inconsistent elements (or use a filter) close to the bottom, and choose $L$ such that the levels above a certain one are consistent. This is the choice that was made in the present case: $N - i = 15$ was considered; the target element edge length $L_{\text{target}}$ was chosen as

$$L_{\text{target}} = \min\left(\max\left(\frac{1}{15}\,\frac{|B|}{\|\nabla B\|}, 75\right), 300\right) \tag{4.14}$$

with lengths in meters. Lengths were limited between 75 m and 300 m to avoid increasing too much the number of elements. Calculations were performed by building

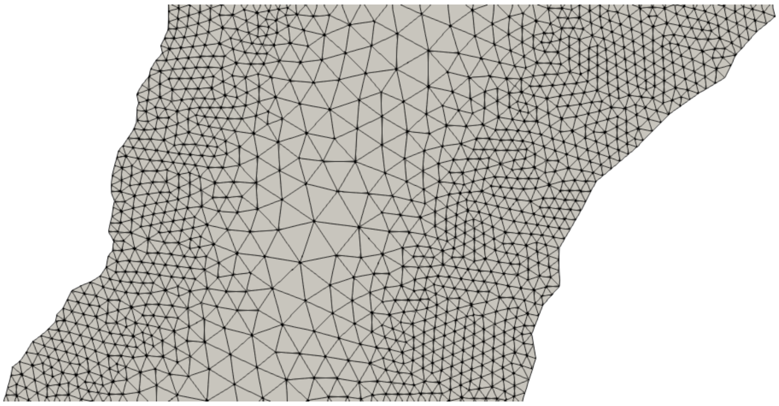


Figure 4.10: A portion of the mesh obtained by adaptation to the bathymetry

a Delaunay triangulation from the bathymetry sampling points and then computing the gradient on triangles; this was done in ParaView. The target edge length field was then fed into BlueKenue and a portion of the resulting mesh is reported in Figure 4.3. It consists of 84443 triangles. The uniform grid was built with typical edge length of 150 m and consists of 34947 triangles. With the adapted mesh, computational times will be significantly longer, so its adoption would have to be justified by a reasonable improvement in the solution. In both cases, the vertical discretization is the same as for the rectangular domain: 36 planes (35 layers) with either standard sigma trasnform or double sigma transform above and below a fixed plane at 50 m depth. The initial thermal stratification is also the same.

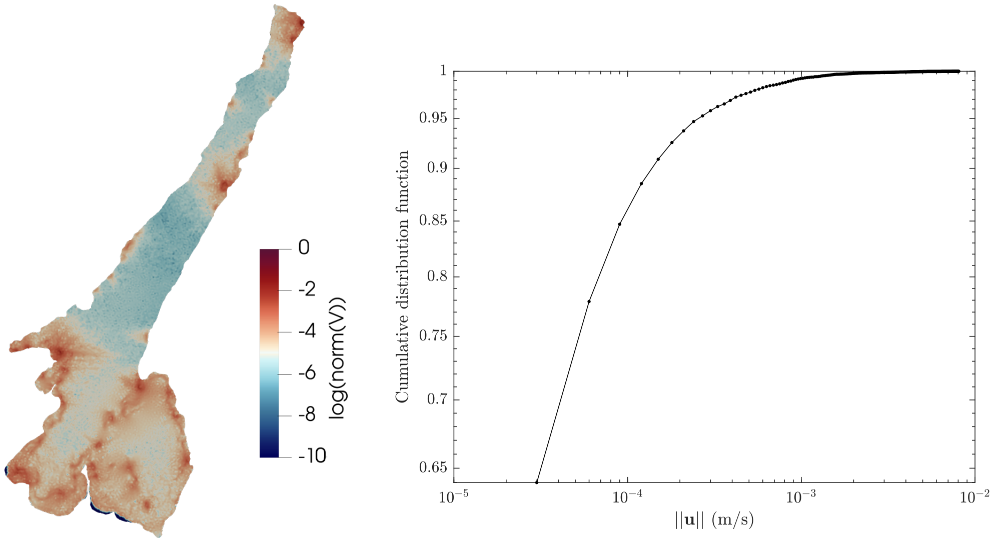
The first simulations were run for just 2 time steps of 10 s each to understand the initial evolution of spurious velocities. Two indicators are used: the maximum velocity magnitude and the integral average of velocity magnitude. The performance of Telemac3D's hydrostatic inconsistency filters is tested: filter 1 simply eliminates horizontal hydrostatic pressure gradients from inconsistent elements; filters 2 and 3 are more refined and are based on the idea of guessing the possibility that an horizontal surface of constant density crosses an element. They are described in more detail in [2, p. 135].

| | Uniform grid,<br>sigma transform | Uniform grid,<br>double sigma transform | Adapted grid,<br>double sigma transform |
|---|---|---|---|
| No filter | $2.5 \cdot 10^{-2}$ | $1.9 \cdot 10^{-2}$ | $1.7 \cdot 10^{-2}$ |
| Filter 1 | $1.3 \cdot 10^{-3}$ | $2.2 \cdot 10^{-3}$ | $4.2 \cdot 10^{-3}$ |
| Filter 2 | $6.1 \cdot 10^{-4}$ | $7.5 \cdot 10^{-4}$ | $1.5 \cdot 10^{-3}$ |
| Filter 3 | $6.0 \cdot 10^{-4}$ | $1.1 \cdot 10^{-3}$ | $2.5 \cdot 10^{-3}$ |

Table 4.1: $\|\mathbf{u}\|_{\max}$ (m/s) with different choices of mesh and filters

| | Uniform grid,<br>sigma transform | Uniform grid,<br>double sigma transform | Adapted grid,<br>double sigma transform |
|---|---|---|---|
| No filter | $2.0 \cdot 10^{-4}$ | $5.4 \cdot 10^{-5}$ | $4.9 \cdot 10^{-5}$ |
| Filter 1 | $8.2 \cdot 10^{-5}$ | $4.1 \cdot 10^{-5}$ | $5.2 \cdot 10^{-5}$ |
| Filter 2 | $1.3 \cdot 10^{-5}$ | $1.1 \cdot 10^{-6}$ | $1.3 \cdot 10^{-6}$ |
| Filter 3 | $1.9 \cdot 10^{-5}$ | $1.2 \cdot 10^{-5}$ | $1.1 \cdot 10^{-5}$ |

Table 4.2: $\|\mathbf{u}\|_{\text{ave}}$ (m/s) with different choices of mesh and filters



(a) Colormap of $\log_{10}(\|\mathbf{u}\|)$

(b) Cumulative distribution function

Figure 4.11: Distribution of spurious velocities, 1000 s test

The results are shown in tables 4.1 and 4.2. When no filter is used, the improvement in using a double sigma transform or the adapted grid is visible, but not very significant. With filters, the results in terms of $\|\mathbf{u}\|_{\text{max}}$ are surprisingly better with the uniform grid and the standard sigma transform than in other cases, but $\|\mathbf{u}\|_{\text{ave}}$ tells a whole different story: using a double sigma transform either on the uniform or on the adapted grid is a better global choice, as expected. The best results are obtained with filter 2. The reason why the adapted grid does not perform better than the uniform grid may be that the refinement was not enough, that is, the lower bound of 75 m may be too high to actually reduce the number of inconsistent elements. Already with this bound, though, the number of elements doubles with respect to the uniform grid: this means that it would be very expensive to compute the solution on a grid that actually avoids inconsistent elements entirely. These results also show that it would be good to gain a better understanding of the different actions of the filters on different grids.

After verifying that the best results were obtained using the uniform grid and the double sigma transform, a longer test, 100 time steps of 10 s each, was performed with the same setup. The results are shown in Figure 4.11. The maximum velocity magnitude is 0.022 m/s, but the colormap and the cumulative distribution function of velocities show that spurious velocities appear in small patches, while in 95% of the nodes velocity magnitudes are below $2.2 \cdot 10^{-4}$ m/s. As a matter of fact, $\|\mathbf{u}\|_{\text{ave}} = 2.5 \cdot 10^{-5}$ m/s. This suggested trying to refine the mesh based on the error, rather than on the bathymetry gradients. Target element edge lengths were defined as

$$L_{\text{target}} = \min\left(\max\left(75 - (150 - 75)\log_{10}\left(\frac{\|\mathbf{u}\|}{10^{-3}}\right), 75\right), 150\right) \tag{4.15}$$

with lengths in meters and velocities in m/s. Essentially, the target size is kept at 150 m where the velocity is less than $10^{-4}$ m/s and gradually reduced for increasing velocity till the minimum size of 75 m is reached at $10^{-3}$ m/s. The resulting mesh has 35874 elements, a small increase over the uniform mesh, leading to only slightly more expensive computations. The results, however, are disappointing: on 100 time steps of 10 s each, as before, $\|\mathbf{u}\|_{\max} = 3.2 \cdot 10^{-5}$ m/s and $\|\mathbf{u}\|_{\text{ave}} = 5.3 \cdot 10^{-5}$ m/s. It was observed that the areas where the mesh was refined are no more critical with respect to spurious velocities, but additional patches appear elsewhere in the domain, and, in some instances, close to refined areas, which may indicate that the refinement procedure was not optimal.

Up to now, 5 layers were placed above the fixed plane and 30 below it, with two uniform sigma transforms: layers have all the same thickness of about 10 m in the deepest parts of the lake. This was done to check if an extension of the results obtained in the case of the rectangle applies. In practice, it would be more appropriate to place thinner layers close to the surface, in order to obtain a better description of wind influence. Two non-uniform sigma transforms are applied, above and below the fixed plane, with a quadratic distribution of layers such that, in the deepest regions, the bottom layer is about 30 m thick and the surface one is about 2 m thick. Now, if layers close to the bottom are thicker, there is a better chance of being able to fulfill the consistency condition (equation (4.12)). This is confirmed in simulations: using filter 2 on a 1000 s simulation, a maximum velocity $\|\mathbf{u}\|_{\max} = 7.0 \cdot 10^{-3}$ m/s and an average velocity magnitude $\|\mathbf{u}\|_{\text{ave}} = 1.8 \cdot 10^{-5}$ m/s are obtained. These are significantly better results than the ones obtained using an uniform vertical discretization.

The analyses presented in this section aimed at understanding hydrostatic inconsistencies and checking and adapting to the present case the results of paper [18]. While for a simple rectangular geometry with quadratically varying bottom the conclusions were the same, the situation became much more complicated on the complete mesh of the lake: it appears that there is no way of eliminating hydrostatic inconsistencies without refining the mesh to the point that computations become too expensive. The best that can be done, according to the results obtained here, is to keep using an uniform grid, to adopt a double sigma transform to better represent stratifications in the upper layers where their effects are more important, and to apply an appropriate filter. The solutions of subsequent simulations will need to be considered with care, especially when velocities are low, to understand whether the observed flow structures are physical or spurious: the average error on the domain will remain low, but patches of higher error have to be expected.

## 4.4 Inflow effects

The combined effect of inflow and stratification is now simulated: a pollutant is introduced in the inflow continosly for a period of time that is short compared to the global residence time. Inflowing water will sink more or less deep and disperse to different extents depending on its temperature in relation to the temperature stratification of the lake. Simulating such process helps in predicting how polluting substances from tributaries disperse in the lake, which regions (horizontally and vertically) are more affected by pollution, and how long it takes for the substances to leave the basin. The aim of this section is to obtain this kind of information for a simple test case, discussing general behaviours, definitions of relevant quantities and methods for calculating them.

The domain is shown in Figure 4.12: it is a rectangle with inflow close to the north-west corner and an open boundary with prescribed constant elevation on the east side. The bottom elevation varies linearly along the north-south direction, with minimum depth on the inflow side, and it is constant along the east-west direction. The 2D mesh is made up of 1000 elements, with 18 layers (19 planes). Following the results about hydrostatic inconsistency, a fixed elevation plane was positioned at 25 m depth; above it, a standard (uniformly spaced) sigma transform was adopted, and below it a non-uniform transform with thicker layers close to the bottom was used.

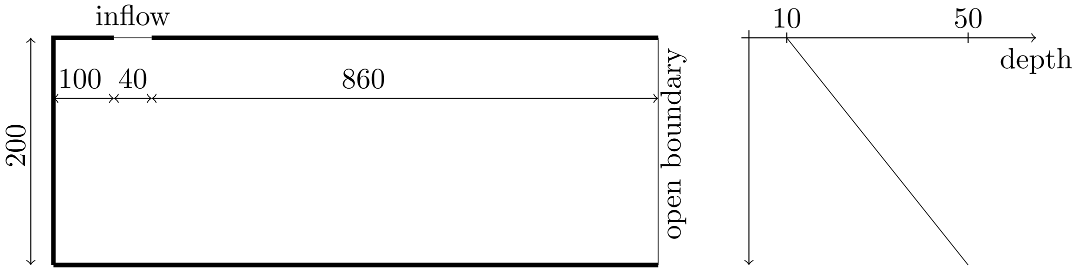


Figure 4.12: Geometry and depth for the inflow test case. Drawing not to scale for readability; all lengths in meters

The inflow mass rate is 100 m$^3$/s, constant in time. The fictitious tracer technique used in [40] and already mentioned in section 4.2 is employed: in the first part of the simulation, a tracer with unity concentration is introduced in the inflow. After some time, the tracer concentration is set to zero in the inflow, and the simulation is continued for 60000 seconds, which is the estimated global residence time according to eq. (1.1): since the full volume of the domain is $6 \cdot 10^6$ m$^3$, $\tau = 6 \cdot 10^{-6}/100$ s $= 60000$ s. The temperature profile was defined as piecewise linear: increasing from 8°C to 10°C between the bottom and 30 m depth, and from 10°C to 20°C between 30 m depth and the surface.

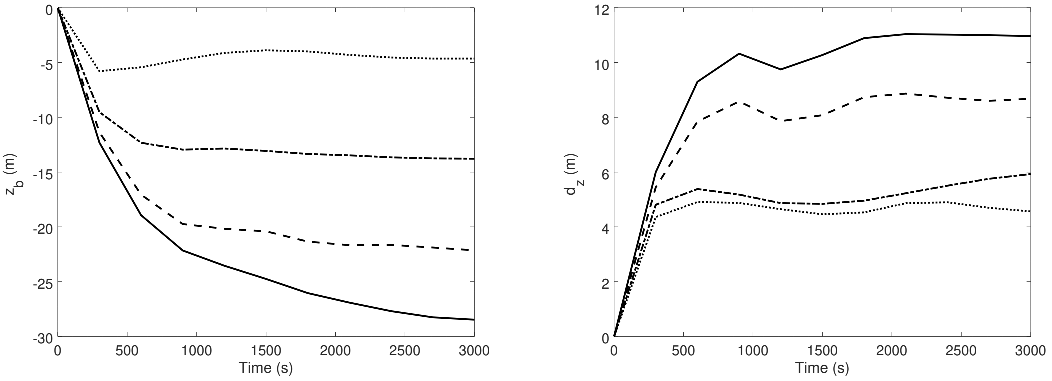


Figure 4.13: Evolution in time of first and second moment of the inflow fictitious tracer distribution (step 2). Solid line: inflow temperature 6°C; dashed line: 10°C; dash-dotted line: 15°C; dotted line: 22°C

A careful choice of parameters had to be made in order to have stability and conservation of tracers. It was observed that the results in terms of conservation were improved by adding a spin-up period with the inflow rate growing linearly from zero to the final, constant value. The setup is:

1. Spin-up: 45 steps of 10 s, inflow ramp in the first 300 s, then constant inflow, with zero concentration of fictitious tracer.

2. 200 steps of 15 s, constant inflow rate with unity concentration of tracer. Total mass of tracer at the end: 300000 kg (negligible outflow of tracer because of the significant distance between inflow and outflow sections).

3. 4000 steps of 15 s, constant inflow rate with zero tracer concentration.

Results in terms of the moments defined in eq. (4.2) computed for the inflow tracer are shown in Figure 4.13 for the first 3000 s after the spin-up (when the tracer is introduced in the inflow). As the inflow temperature increases, the average depth reached by inflowing water decreases, and so does the standard deviation of the concentration.

In the last step of the simulation, when no tracer is present in the inflow, the total mass of tracer in the domain decays as shown in Figure 4.14. After some time in which it remains constant, mass starts decreasing with a law that is close to an exponential, so that globally the ratio of mass at time $t$ to mass at time 0, $M_0$, can be approximated as

$$\frac{M}{M_0}(t) = \min\left(1, \exp\left(-\frac{t-t_0}{\tau}\right)\right) \tag{4.16}$$

This law was fit to the obtained time series by numerically minimizing the sum of square deviations; the results are shown in Table 4.3.

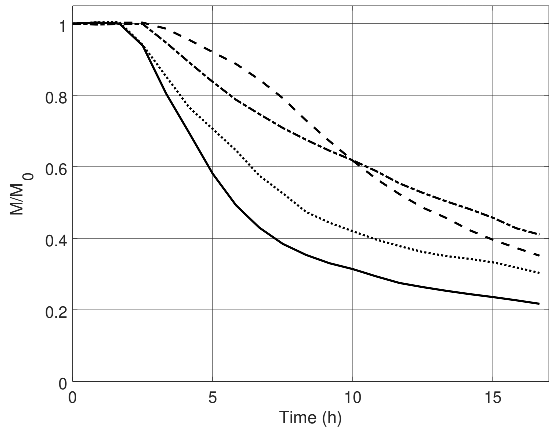


Figure 4.14: Time history of ratio of total tracer mass in the domain to initial tracer mass. Time from end of tracer release in the inflow (step 3). Solid line: inflow temperature 6°C; dashed line: 10°C; dash-dotted line: 15°C; dotted line: 22°C

| Inflow temperature (°C) | $\tau$ (h) | $t_0$ (h) |
|---|---|---|
| 6 | 7.67 | 1.41 |
| 10 | 11.49 | 4.46 |
| 15 | 15.94 | 2.27 |
| 22 | 10.82 | 1.37 |

Table 4.3: Parameters of function (4.16) fit to the time series

The appearance of an exponential decay is consistent with the continuously stirred tank reactor (CSTR) model [38], [40]. Consider a basin, or a portion of a basin (e.g. a bay, or a layer along the vertical, such as the epilimnion). Suppose that such domain is continuosly stirred, so that the concentration $C(t)$ of a tracer is always uniform in space: $C(t) = M(t)/V$, where $M(t)$ is the total mass of tracer in the domain. In the present case, stirring is operated by inflowing water, that displaces water that was already there in different manners at the different temperatures; also, behaving, at least initially, as a jet, it entrains fluid. In a more general case, wind would act as a stirring agent too, as already seen in section 4.2. Then, if there is no tracer in the inflow and there is neglibigle return flow (like in step 3 of the present simulation), the tracer can only exit the domain. This consideration and the hypothesis of uniform concentration imply that the time variation of total tracer mass is proportional to the outflow rate $Q$ (which, in steady state, will be equal to the inflow rate):

$$\dot{M}(t) = -Q\,C(t) = -\frac{Q}{V}M(t) = -\frac{1}{\tau}M(t) \tag{4.17}$$

where the definition of global residence time (eq. (1.1)) was used in the last equality.

Integration yields the exponential expression

$$\frac{M}{M_0}(t) = \exp\left(-\frac{t}{\tau}\right) \tag{4.18}$$

In light of this definition, $\tau$ can be interpreted as the e-folding time, that is the time taken by the tracer mass to reach about 36.8% of its initial value.

Two corrections are needed to apply the CSTR model to the present case. The first is that exponential decay does not start right after the tracer stops being introduced. This is because the first tracer particles take a finite time to reach the outflow section (or, equivalently, mixing is not instantaneous). The delay before exponential decay starts is represented by the term $t_0$ in equation (4.16). The second correction stems from the observation that the value of $\tau$ obtained by fitting the model to the data does not correspond to the theoretical value $\tau = V/Q = 16.7$ h: all estimated times are less than this. The reason is that not all of the water volume takes part in the mixing; in other words, the tracer mixes well only in a limited portion of the domain (and even there, mixing is partial): so the volume $V$ in the expression of $\tau$ should be more correctly called stirred volume, and in general it does not correspond to the total volume.

## 4.5 Turbulence model comparison

Some initial tests on the lake mesh revealed difficulties in the use of the $k-\epsilon$ model, which would show convergence problems and unrealistic values in some areas (in particular the shallow ones). The adoption of the mixing model was then considered, in view of it being simple while still being able to account for stratification[9]. The aim of this section is to compare the two on a very simple geometry, under the effect of wind.

A reference solution for turbulent mixing induced by surface stress in a stratified flow was elaborated in 1981 by Kundu [32] using similarity arguments, that is assuming that the solution can be expressed as

$$f(z,t) = F\left(\frac{z}{h(t)}\right), \tag{4.19}$$

$f$ being any flow variable in the solution (velocity, density, tracer concentration, averaged turbulent fluxes) and $h$ being the depth of the mixed layer at the surface.

For high-Reynolds, statistically one-dimensional flow with linear initial density stratification and zero initial velocity, the depth of the mixed layer at the surface is found to evolve as

$$h(t) = Cu_\tau\sqrt{\frac{t}{N_0}}, \tag{4.20}$$

where $u_\tau$ is the friction velocity at the surface, $N_0$ is the buoyancy frequency related to the initial density stratification and $C$ is a constant to be determined. The constant

[9] For stratified flows, the mixing length model is still a recommended option in Telemac3D, according to the user manual.

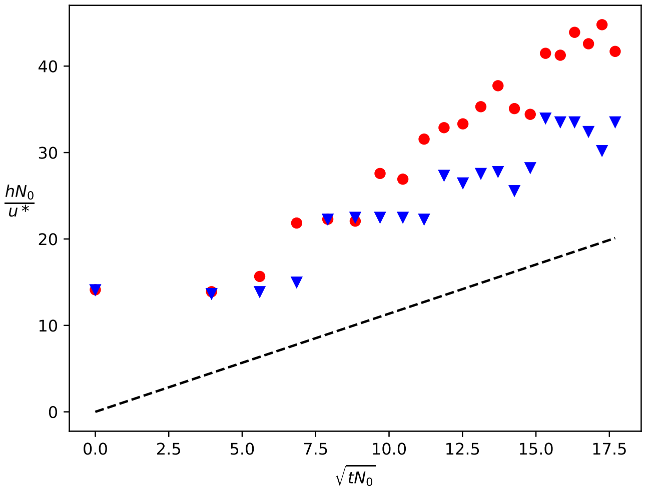
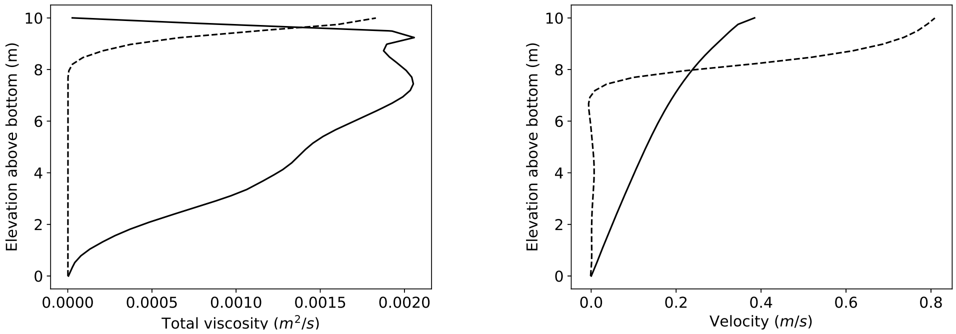


Figure 4.15: $k-\epsilon$ (red circles) and mixing length (blue triangles) numerical solutions obtained with Telemac v. reference solution (black dashed line)

was estimated to be 1.047 from 1D numerical simulations, and 1.136 from experimental data of Kato and Phillips (1969) [31]. The experiments were conducted in an annular tank filled with water, with a rotating screen driving the fluid.

To test the turbulence models available in Telemac against the law (4.20), simulations were performed on a rectangular domain of dimensions 1000 × 200 × 10 m with constant and uniform wind and an initially linear vertical density profile, with a variation of 12.5 kg/m$^3$ from the bottom to the surface. Zero friction was imposed at the walls and bottom to realize ideal conditions. Periodic boundary conditions were imposed on the boundaries perpendicular to the wind direction, while no-penetration conditions were set on the two remaining sides. This setup was devised in order to reproduce the ideal conditions sought by Kato and Phillips in their initial experiment. A passive tracer was positioned with initial concentration 1 in the surface layer and 0 below. The value of $h$ was obtained from the solution at each time by selecting the depth corresponding to the maximum derivative of passive tracer concentration. The vertical mesh was made of 40 layers placed uniformly, yielding an average thickness of about 0.25 m. 2000 steps of 5 seconds each were simulated. In the vertical direction, the $k-\epsilon$ and mixing length models were employed in the two tests, while in the horizontal direction a constant, uniform viscosity value was imposed. Results and comparison with the reference solution are reported in Figure 4.15.

It can be observed that the slope of the mixing length numerical solution is closer to the reference solution than the $k-\epsilon$ one. As a matter of fact, both models yield reasonable results with respect to tracer distribution, while results differ largely for velocity and turbulent viscosity (see Figure 4.16).



(a) Turbulent viscosity

(b) Velocity

Figure 4.16: Vertical profiles for the mixing length (solid line) and the $k - \epsilon$ (dashed line) model at the end of the simulation

In [57] it is argued that the behaviour of 2-equation turbulence models on problems such as this is largely determined by the value of the buoyancy constant $C_{\epsilon 3}$ in the equation for dissipation. The value used in Telemac, which is 1 for stable stratification and 0 for unstable stratification, is nowhere close to any of the proposed ones. This matter deserves further consideration, but such additional analysis was beyond the scope of this project.

## 4.6 Parallelization

To investigate the effect of using the parallel version of the code on the computational time, the last test of section 4.3 was shortened to 25 time steps of 10 s each and repeated with different numbers of cores. In Telemac, parallelization is handled using the message passing interface (MPI), which allows working on distributed memory clusters. Computations were run on a cluster on which the user is allowed to launch jobs on at most 2 nodes at a time. Each node has an Intel Xeon E5-2640 v4 CPU, with 20 physical cores and 2 threads per core. The results are reported in Table 4.4. In particular, speedup and efficiency are calculated as defined in [33, p. 20]:

$$S_p = \frac{T_1}{T_p}, \tag{4.21}$$

$$E_p = \frac{T_1}{p\,T_p} = \frac{S_p}{p}, \tag{4.22}$$

where $p$ is the number of cores used.

This indicates that, as usual when parallelizing computations, increasing the number of cores is beneficial only up to a certain point, as the time taken for communication becomes the limiting factor, especially when multiple nodes are involved. It should be

| Cores | Nodes | Walltime | Speedup | Efficiency |
|---|---|---|---|---|
| 1 | 1 | 5’53” | | |
| 2 | 1 | 3’9” | 1.87 | 0.93 |
| 4 | 1 | 1’39” | 3.57 | 0.89 |
| 8 | 1 | 58” | 6.09 | 0.76 |
| 16 | 1 | 45” | 7.84 | 0.49 |
| 20 | 1 | 44” | 8.02 | 0.40 |
| 32 | 2 | 45” | 7.84 | 0.25 |
| 40 | 2 | 44” | 8.02 | 0.20 |

Table 4.4: Speedup and efficiency for different numbers of cores. The walltime is the actual physical time, as opposed to CPU time, which is the cumulative working time of all cores

noted that the model computation used for this test is very simple, since it does not include inflows or wind forcing. It may be expected that by including those effects the workload of single cores will increase and the relative incidence of communication time will decrease, thus leading to better speed-ups for larger numbers of cores than observed here. Simulations with 32 and 40 cores were also run by using the corresponding number of virtual cores on a single node, and the same results regarding walltime as the ones reported in the table were obtained.

## 4.7 Remarks on the preliminary tests

In this section, some basic test cases investigating single phenomena that are relevant to the whole lake and the possible choices and issues in modeling them were discussed. It was shown that both pathlines and statistics based on fictitious tracers can be useful for characterizing mixing both qualitatively and quantitatively. The importance of building the mesh properly was underlined: in particular, it was found that the best choice for this case is a uniform 2D grid on the horizontal, and a vertical discretization with a fixed horizontal plane (its position depending on stratification) and non-uniform sigma transforms above and below it. This choice yields reasonably controllable spurious velocities and the ability to simulate wind interaction well, having very thin elements at the free surface. The inflow test case confirmed the validity of the common practice of increasing inflow rates gradually when they need to be introduced in the simulation. In the last section, the behaviours of two different turbulence models were compared to each other and to a reference solution, highlighting the need for further discussion.

# Chapter 5

# Numerical simulations

This chapter presents the simulations done to assess the impact of the 2018 discharge event of the Adige-Garda tunnel on the lake. First, computations on a rather coarse mesh of the whole lake were run; then, a more refined mesh was built for the northern part of the lake only. These simulations are used to better understand the flow field features close to the inflow point.

## 5.1 Geometry and external forcing data

The boundary shape for simulations on the full lake was obtained as a shapefile from the Regione Lombardia website[1]. In 2018, an high-resolution bathymetry survey was started, but the relative data were incomplete at the time of working on the computations for this thesis, so the bathymetry from the 1966 survey of the Italian Military Geographic Institute was used for simulations on the whole lake instead. The digital version of the 1966 bathymetry data set consists of about 170000 points, while the (incomplete) 2018 data consists of about 6000000 points. For the simulations on the northern part of the lake it was possible to use the new 2018 bathymetry, since it was complete in that area; the outline was drawn manually in BlueKenue before mesh generation.

Wind data was obtained by inverse distance interpolation (see subsection 5.1.1 for details) from values measured at the weather stations of Nago-Torbole (Meteotrentino[2]), Limone sul Garda, Toscolano Maderno, and Manerba del Garda (ARPA Lombardia[3]). These stations are the closest available to the lake, and are all situated within 30 m height above the lake surface. Thermal stratification data were provided by APPA Trento[4]; the approximate measuring point location is shown in Figure 5.1. Measurements are taken at irregular intervals, about once a month. For the present project, the profile measured on October 23, 2018 was used, taken as uniform over the lake.

[1] https://www.dati.lombardia.it/Territorio/Lago-10000-CT10/qm9t-uzst

[2] https://www.meteotrentino.it/

[3] https://www.arpalombardia.it/Pages/ARPA_Home_Page.aspx

[4] http://www.appa.provincia.tn.it/

The inflow temperature was taken at the closest available measuring point, which is the APPA station at canale Biffis, about 20 km downstream of the Mori-Torbole tunnel derivation from river Adige. Because of the lack of data in the time interval of interest (presumably due to the effects of the concomitant severe weather events), the time series was first reconstructed using an autoregressive model, then averaged over the inflow time, obtaining a value of 10.5 °C. Temperature change was considered the only effect on density variations. As already discussed in section 2.3, this is the main effect. The total inflown volume was about 17.500.000 m$^3$, starting gradually at about 22:30 on October 29, peaking at 350 m$^3$/s and decreasing until the end of the discharge at about 16:00 on October 30. The time history of inflow was approximated as piecewise linear according to the aforementioned known values. Inflow rate data for river Sarca were obtained from Ufficio Dighe - Protezione Civile Trento[5], while the corresponding temperatures were provided by APPA Trento (private communication).

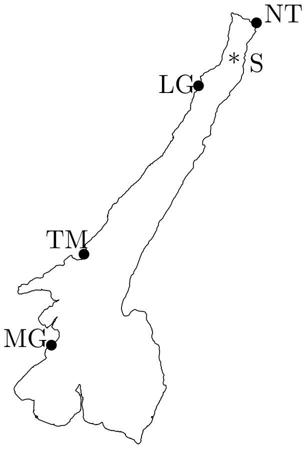


Figure 5.1: Position of weather stations. NT: Nago-Torbole; LG: Limone sul Garda; TM: Toscolano Maderno; MG: Manerba del Garda. S: thermal stratification measuring point

### 5.1.1 Wind data interpolation

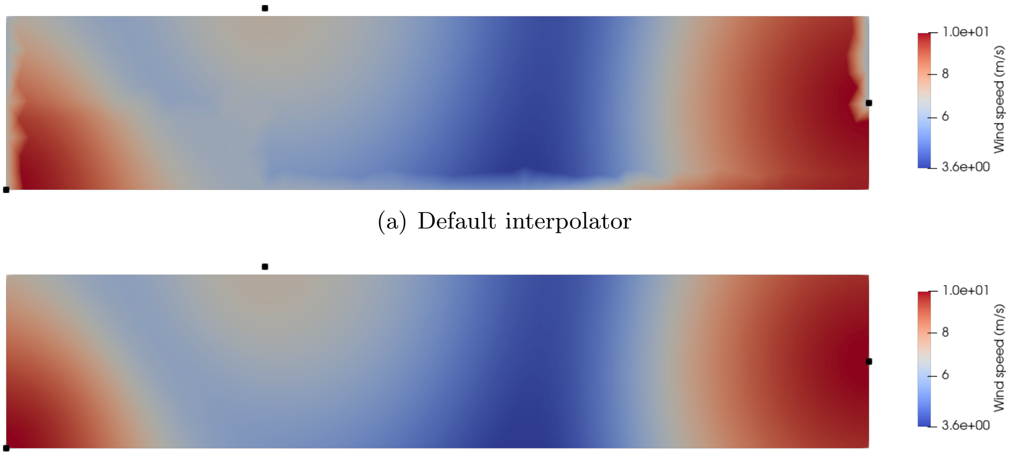
For wind variable in space and time, the Telemac3D user manual suggests the use of subroutines based on inverse distance interpolation. However, these have some issues:

- Wind data is updated from the atmospheric data file only if the time reached in the simulation is exactly equal to one of the times at which data is provided. Not only does this lead to a zero order (piecewise constant) interpolation in time; it

[5] https://www.floods.it/public/DatiStorici.php

also means that unfortunate combinations of time steps can lead to the wind data never being updated.

- The inverse distance interpolation subroutine divides the space around each mesh point into four quadrants and uses the nearest data point (weather station) in each quadrant for interpolation. This leads to artifacts in the wind field: areas of sharp variation along vertical or horizontal lines, as two adjacent mesh points may have different weather stations as closest one in each quadrant. Moreover, the interpolated value in points close to a station is not close to the value at the station. These problems are shown in Figure 5.2(a), where weather stations are marked by black squares.



(a) Default interpolator

(b) Modified interpolator

Figure 5.2: Wind speed norm from default and modified inverse distance interpolators. Black squares: weather stations

These issues have been solved by modifying functions `meteo.f` and `idwm_t2d.f`. In the former, a linear interpolation in time was implemented before calling the latter for interpolation in space. `idwm_t2d.f` was rewritten with the same method (inverse distance), but in a much simpler way, without using quadrants, and considering all weather stations for interpolation in each point. This is motivated by the small number of weather stations and by the fact that stations far from a given mesh point will be automatically assigned a very small weight relative to the closer ones. The result of these modification is reported in Figure 5.2(b). In most areas the result is the same, but the field is much smoother in the second case, especially close to stations.

The inverse distance algorithm with exponent $m$ is as follows:

$$f(\mathbf{x}) = \frac{\sum_{i=1}^{N} f_i \, w_i(\mathbf{x})}{\sum_{i=1}^{N} w_i(\mathbf{x})}, \tag{5.1}$$

where $N$ is the number of stations and the weights $w_i(\mathbf{x})$ are defined as

$$w_i(\mathbf{x}) = \frac{1}{\|\mathbf{x} - \mathbf{x}_i\|^m}. \tag{5.2}$$

$f_i$ is the known value at station $\mathbf{x}_i$. The method consists essentially in a weighted average of the known values, where the values at the closest stations are assigned greater weights. If implemented exactly this way, weights would be infinite at the stations, which is not acceptable; there are two ways to solve this issue:

1. limiting weights;
2. defining $f(\mathbf{x}) = f_i$ any time $\|\mathbf{x} - \mathbf{x}_i\| < d_{min}$, where $d_{min}$ is some minimum distance.

The second choice has been implemented, to keep the code structure similar to the original one. It was decided to use an exponent $m$=2; an higher exponent would yield a less smooth field, with values almost identical to the ones at the closest station and sharp variations between the areas of influence of the different stations. It can be proved [51] that using this method the partial derivatives of $f$ are zero at $\mathbf{x} = \mathbf{x}_i$, for each $i$. This is quite arbitrary and may not be desirable; corrections are possible, but were not explored in this context.

An alternative method was considered and implemented, based on radial basis functions (RBF) [55]. A radial function centered in $\mathbf{x}_i$ is one of the form

$$\phi_i(\mathbf{x}) = \phi(\|\mathbf{x} - \mathbf{x}_i\|). \tag{5.3}$$

For example, the inverse distance weights seen above are a possible choice of radial functions; but the RBF method works in a radically different way. The interpolated field is still written as a linear combination of the basis functions:

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i \, \phi_i(\mathbf{x}) = \phi(r), \tag{5.4}$$

but this time coefficients $\alpha_i$ have to be determined by imposing

$$f(\mathbf{x}_i) = f_i \quad \forall i. \tag{5.5}$$

This is done by solving a linear system of $N$ equations. For the case of interest, the number of stations is very low: less than 10 for the purposes of this project, and in general typically much less than the number of mesh nodes. A direct method is then a reasonable choice for solving the system; the Gauss elimination method was implemented from scratch in the Fortran routine, avoiding the use of external libraries which would have required recompiling the whole Telemac package. The basis function was chosen as

$$\phi(r) = \exp(-k\, r^2). \tag{5.6}$$

$k$ can be taken as the inverse of some measure of the typical distance between stations, squared. The choice of this parameter is critical because it determines the effective support of the basis function: if not chosen properly, it can lead to the interpolated value being zero or close to zero far from any station. This problem does not occur when using the inverse distance method, since in that case the value at each point is a weighted average of the known ones. On the other hand, the RBF method does not require a special treatment close to the stations. A balance of these consideration lead to the choice of using the inverse distance method, modified as described above.

Fortran source codes for both methods are available on GitHub[6].

### 5.1.2 Extreme value analysis on Sarca flowrate

After a short review of extreme value theory, the generalized extreme value distribution (GEVD) for the flow rates of river Sarca will be estimated, together with the expected return period of the peak value observed in this scenario. This review is based on [10] and [17].

Consider $n$ independent and identically distributed (*iid*) random variables $X_1$, $X_2$, ... , $X_n$ with cumulative distribution function $F(x)$:

$$F(x) = P(X < x), \tag{5.7}$$

where $P$ stands for probability. The variables represent, in our case, measures of flow rate at different times over a year. We look for the distribution of the maximum value of the variables, which will be the distribution of annual maxima:

$$G(x) = P(\max(X_1, X_2, \ldots, X_n) < x). \tag{5.8}$$

For the maximum value to be less than $x$, all variables must be less than $x$. Since independence is assumed, this can be expressed as

$$\begin{aligned} G(x) =& P[(X_1 < x) \cap (X_2 < x) \cap \cdots \cap (X_n < x)] \\ =& P(X_1 < x)\, P(X_2 < x)\, \ldots\, P(X_n < x) \quad . \\ =& [F(x)]^n \end{aligned} \tag{5.9}$$

For $n \to \infty$, and under suitable assumptions that will be introduced shortly, the distributions of maxima $G$ tend to a universal form for a very broad class of underlying functions $F$. This is analogous to what happens for the distribution of the sum of *iid* random variables, which tends to a Gaussian according to the central limit theorem. We define the right endpoint of the underlying distribution

$$x^* = \sup\{x : F(x) < 1\} \tag{5.10}$$

which can be either finite or infinity. If it is finite, then $G(x) = [F(x)]^n$ for $n \to \infty$ is a step in position $x^*$, while the corresponding probability distribution function is a Dirac

[6] https://github.com/marcogambarini/interpolation-telemac

delta centered in $x^*$. If it is infinity, then $G(x) \to 0$ for any real value of $x$. In both cases, the result is a degenerate distribution. It turns out that a non-degenerate distribution of universal form can be obtained by a linear change of variables (normalization). This result is formalized in the following theorem, known as the Fisher-Tippett-Gnedenko theorem:

**Theorem 4.** *Consider the iid random variables $X_1$, $X_2$, $\ldots$ , $X_n$ with cumulative distribution function $F(x)$. If there exist two sequences $a_n > 0$, $b_n$ such that*

$$\lim_{n\to\infty} P[\max(X_1, X_2, \ldots, X_n) < a_n x + b_n] = \lim_{n\to\infty} [F(a_n x + b_n)]^n = G(x) \tag{5.11}$$

*where $G(x)$ is a non-degenerate distribution function, then*

$$G(x) = \exp\left[ -\left( 1 + \gamma \frac{x-\mu}{\sigma} \right)^{-1/\gamma} \right] \tag{5.12}$$

*which is the GEVD (generalized extreme value distribution). $\mu$ is the location parameter, $\sigma$ is the scale parameter and $\gamma$ is the shape parameter.*

Equation (5.11) is a condition on the underlying function $F$. For our purposes, we are not directly interested in the implications of this condition, but rather in the fact that function $G(x)$ exists. As a matter of fact, it is possible to apply the so-called block maxima sampling, by fitting expression (5.12) directly on the sequence of maximum yearly flowrates extracted from the dataset. This was done in Matlab using the function `gevfit`[7], that returns a maximum likelihood estimate of the parameters $\gamma$, $\sigma$ and $\mu$.

An interesting information that can be extracted from the distribution after fitting is the return period $T_{x_0}$, which is the mean time between events corresponding to a value $X \geq x_0$. In other words, we can estimate how often we should expect an event at least as strong as a given one to happen. The probability $w(m)$ that the value is exceeded only at the $m$th extraction, after not happening in the $m-1$ extractions before that, is

$$w(m) = [F(x_0)]^{m-1}(1 - F(x_0)) \tag{5.13}$$

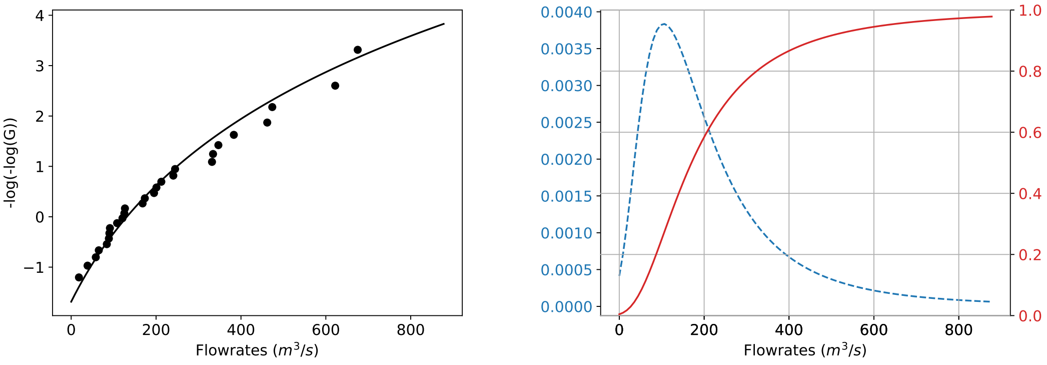
since $(1 - F(x_0))$ is the probability of exceeding $x_0$ and $F(x_0)$ is the probability of not exceeding it. The return period is the expected value of $m$:

$$T_{x_0} = E[m] = \sum_{m=1}^{\infty} m\, w(m) \tag{5.14}$$

It can be proved, using the expression for the sum of an infinite geometric progression, that

$$T_{x_0} = \frac{1}{1 - F(x_0)} \tag{5.15}$$

[7] https://mathworks.com/help/stats/gevfit.html



(a) (b)

Figure 5.3: Probability plot with sample data (circles) and fitted distribution (line) (a), pdf (blue dotted line, left vertical axis) and cdf (red solid line, right vertical axis) (b) for the GEVD fit

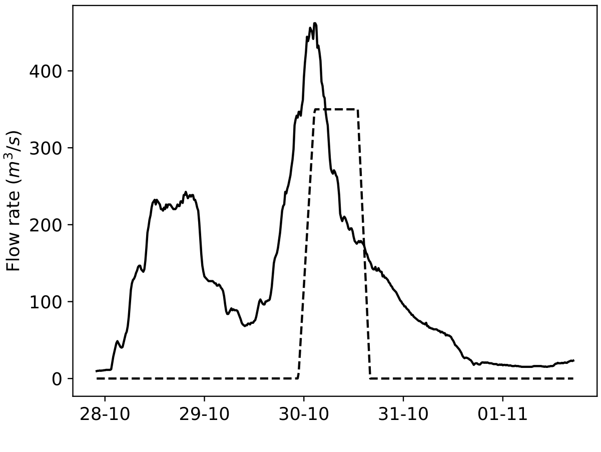


Figure 5.4: Flow rates for river Sarca (solid line) and Adige-Garda tunnel (dashed line), year 2018

For the present case, data were available from 1994 until 2020, so 27 samples could be extracted. The estimates for the parameters are $\hat{\gamma} = 0.3156$, $\hat{\sigma} = 100.33$ m$^3$/s, $\hat{\mu} = 131.21$ m$^3$/s. The probability plot, showing how well the distribution fits the data, and the probability and cumulative distribution functions are shown in Figure 5.3. The maximum value of flowrate in the scenario analyzed in this thesis was $Q_0 = 461.9$ m$^3$/s, recorded on October 30th 2018 at 02:30. The corresponding return period is estimated using equation (5.15) to be about 10 years. In the available time series, this value was exceeded on September 20th, 1999 (675.09 m$^3$/s), October 3rd, 2020 (622.07 m$^3$/s) and June 28th, 1997 (473.79 m$^3$/s). The Sarca river flowrate in the present scenario is then a very large value compared to the average, but not an historically exceptional event.

The temperature data for Sarca were quite sparse: among the available measurements, the closest in time to the tunnel opening were taken on October 8th and November 19th. The average value of the two was taken, equal to 11.6°C. Flowrate data for the Sarca river (measured) and the Adige-Garda tunnel (modelled) are shown in figure 5.4 for the timeframe of interest.

## 5.2 Global simulations

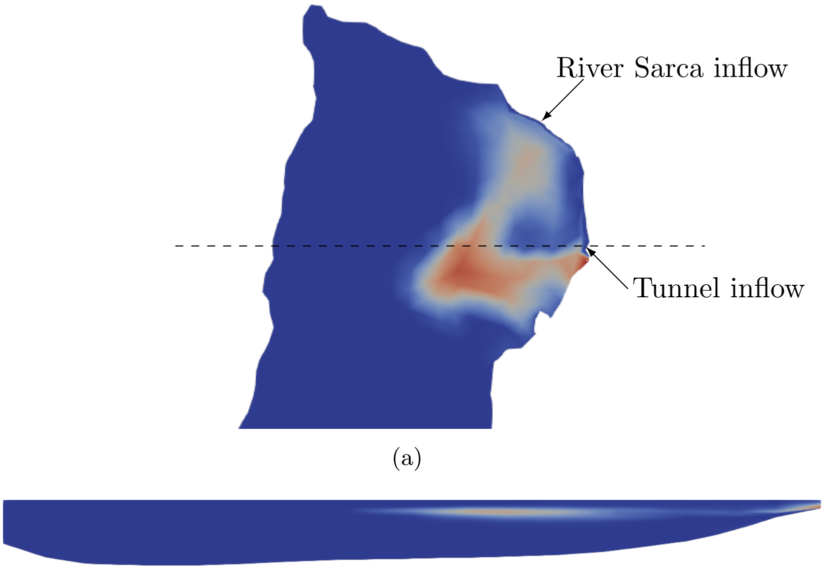
The mesh used for simulations on the entire lake was uniform horizontally, with typical edge length of 150 m, and was composed of 50 layers in the vertical direction (51 planes). The mixing length model was used in the vertical direction with Quetin's profile (equation (2.38)) and the Munk-Anderson damping function (equation (2.41)), while a constant viscosity was imposed in the horizontal one, with a value of $2 \times 10^{-2}$m$^2$/s, obtained by calibration in [4]. The Nikuradse model was used for bottom and lateral wall stresses. The simulation was divided into the following phases:

1. A 7-day spinup period, from 22:00, 22/10/2018 to 22:00, 29/10/2018, with wind forcing and no heat transport/diffusion (frozen stratification); 30 seconds time step.

2. A 19-hour simulation corresponding to inflow from the tunnel, with wind forcing and heat transport/diffusion; 10 seconds time step.

3. A 4-day simulation after the end of inflow from the tunnel, thus ending at 17:00, 3/11/2018, with wind forcing and heat transport/diffusion; 30 seconds time step.

During phases 2 and 3, a tracer was introduced in the inflow with unity concentration to track the evolution of incoming water. The inflow was distributed on 3 boundary elements for numerical reasons; this has an impact on the correct representation of inertia, as will be discussed later. Heat transfer at the lake surface was neglected; this has an impact on mixing phenomena, but it was not considered an issue because of the relatively short time frame of the simulation. Taking it into account would require using more advanced assimilation or simulation techniques for weather data, because radiation plays a major role, and the radiation heat flux is influenced by the surrounding orography and by cloud cover. Using a simple formula based on air temperature data



(a)

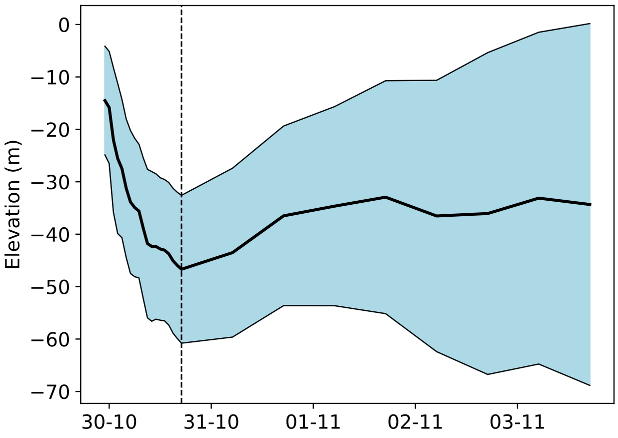
(b)

Figure 5.5: Inflow tracer distribution at 17:00, 30/10/2018: view (a) from above, slice at 44 m depth, and (b) from south, slice at the inflow point latitude. The dashed line corresponds to the section cutout. Figures out of scale for better visualization

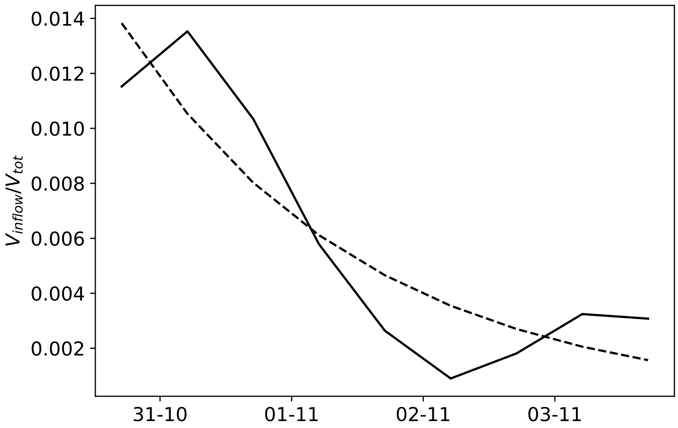
obtained by interpolation from values at few stations would probably be a quite limited improvement. Floating particles were released at the discharge point of the tunnel in order to generate pathlines. A first group was released during the inflow, proportionally to the flow rate, and another after the end of the inflow.

During the inflow time, water coming from the tunnel travelled horizontally behaving as a jet for a limited distance before starting to plunge to depth: this happens when buoyancy forces start dominating on inertia. In the following phase, water descended to a depth of about 44 m on average due to its density relative to the ambient one, and started propagating as an intrusion (see section 2.9). The intrusion rolled up horizontally (see Figure 5.5): this can be due to the Coriolis effect. This statement was confirmed by repeating phase 2 of the simulation without a spinup period and without the action of wind, but with the Coriolis force: the structure of the solution is qualitatively the same, confirming that in this phase non-inertial effects dominate on wind-induced currents and mixing. It can be noted that part of the inflown water travels towards the mouth of river Sarca. Such observation can lead to a reasonable suspect that the Sarca inflow may influence the dispersion of tracer from the tunnel: this will be investigated in the next section. Particles released during the inflow time remained close to the release point, suggesting the presence of an hydrologic front corresponding to the downwelling of inflowing water and the convergence of surface currents in front of the inflow section. This result is confirmed by videos shot during the opening[8] and is commonly observed

[8] https://www.youtube.com/watch?v=C2JNK83eldc



(a) Inflow tracer vertical distribution: first moment (thick solid line), first moment ± second moment (filled area); vertical dashed line: end of inflow time



(b) Ratio of water volume inflown from the Adige-Garda tunnel to total lake volume, computed for the region north of the section in Figure 5.5. Solid line: data from the simulation; dashed line: exponential fit

Figure 5.6: Inflow tracer statistics for the global simulation

in estuaries [41].

In the following days, due to wind-induced currents and turbulent mixing, the barycenter of the tracer distribution moved closer to the surface, while the second moment of the distribution increased as shown in Figure 5.6(a). The first and second moments are computed according to their definitions, reported in equation (4.2). Accordingly, the concentration of tracer increased at the surface, in particular at the northern shore (close to the town of Riva del Garda) and at the eastern (Veneto) shore. This is due to upwelling: as already observed in [6], the predominance of strong northern winds together with the Coriolis effect generates a secondary circulation with downwelling at the western shore and upwelling at the eastern; a similar result is reported in [22], relative to a different opening event of the tunnel. Floating particles released after the end of the inflow travelled about 3 km south along the eastern shore in 24 hours, suggesting that such area is the most influenced by possible presence of floating debris from the tunnel. As done in section 4.4, the flushing time of inflown water from any given section of interest can be estimated. For instance, Figure 5.6(b) reports the result obtained considering the portion of lake north of the tunnel inflow latitude (see Figure 5.5): the e-folding time is 44 hours, which can give an idea of the time scale of the dispersion.

Since a large part of the debate about the impact of water released from the tunnel is about thermal effects, an additional simulation was run without the inflow. A close-up analysis of absolute temperatures would be of limited validity because of the lack of surface heat exchange in the model; but since both runs were done with this hypothesis, a mutual comparison can provide insight. Figure 5.8 shows the surface temperature anomaly in the area interested by the presence of inflow tracer on November 3rd, 2018, four days after the discharge ended. The anomaly is computed as the temperature difference between the cases with and without inflow. The largest negative temperature anomaly is obtained in the northern part of the lake and it exceeds -2°C. More to the south, the western shore is almost unaffected, while on the eastern shore the temperature anomaly is still negative and generally less (in absolute value) than -1°C. A patch of positive temperature anomaly appears in the northern region, which can be attributed to water being displaced by the intrusion and the currents it generates. On the days before November 3rd, a small temperature anomaly emerged, mainly concentrated at the northernmost shores. It should be noted that the simulations showed temperature differences also farther from the inflow section, in the southern basin, which can be assumed to be due to convergence problems in the shallow part of the lake. This fact had been a concern from the start, and leaves doubts about the use of the sigma transform in a lake with such large depth variations, even when all the precautions described in section 4.3 are taken. However, the results in the northern trunk have the clear pattern described above.

This analysis focused on the surface temperature because this is the layer in which most biological phenomena happen, and since it is the most important for human activities. A detailed discussion of the biological effects of this temperature change is beyond the scope of this work, but the results presented here can be an important premise

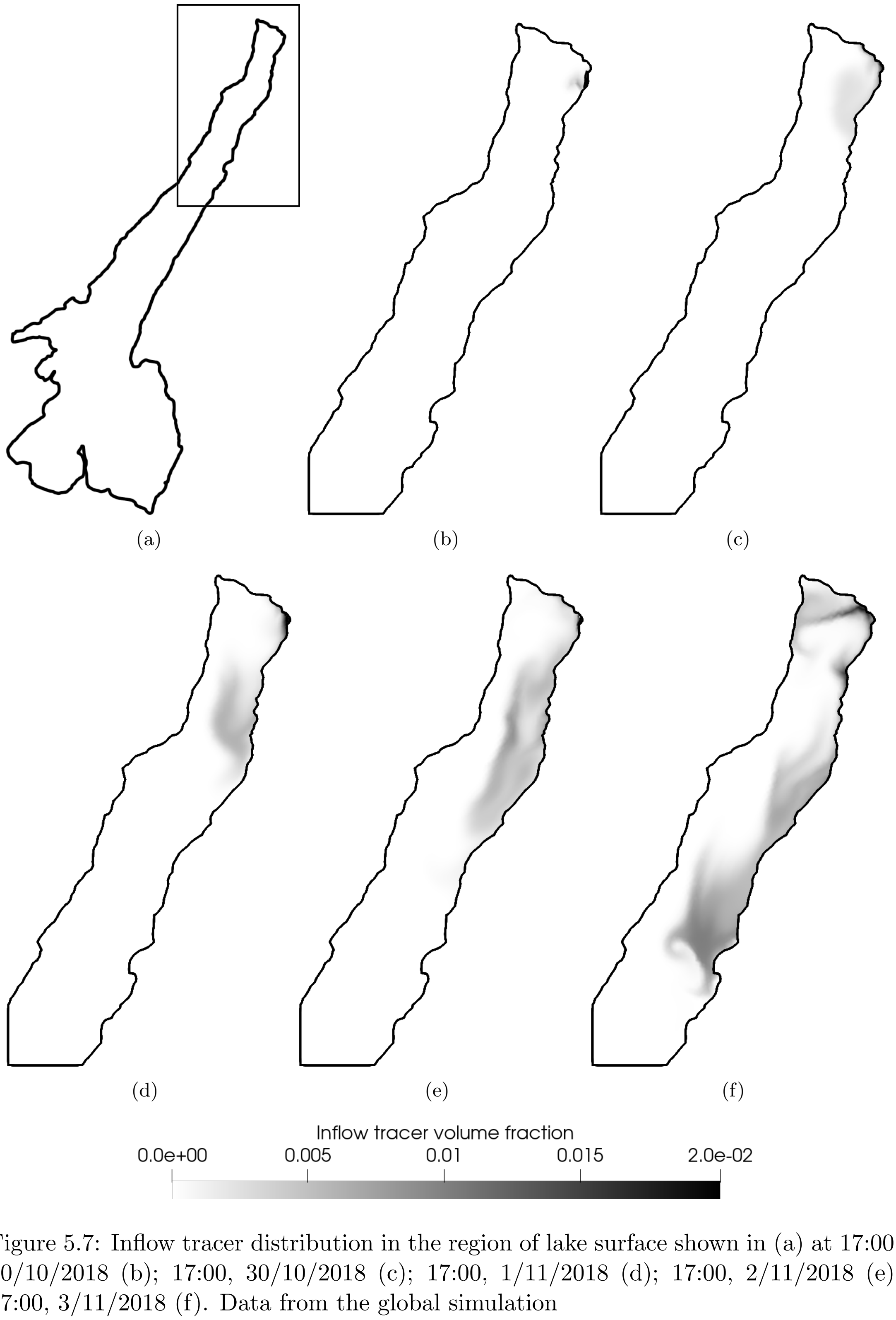
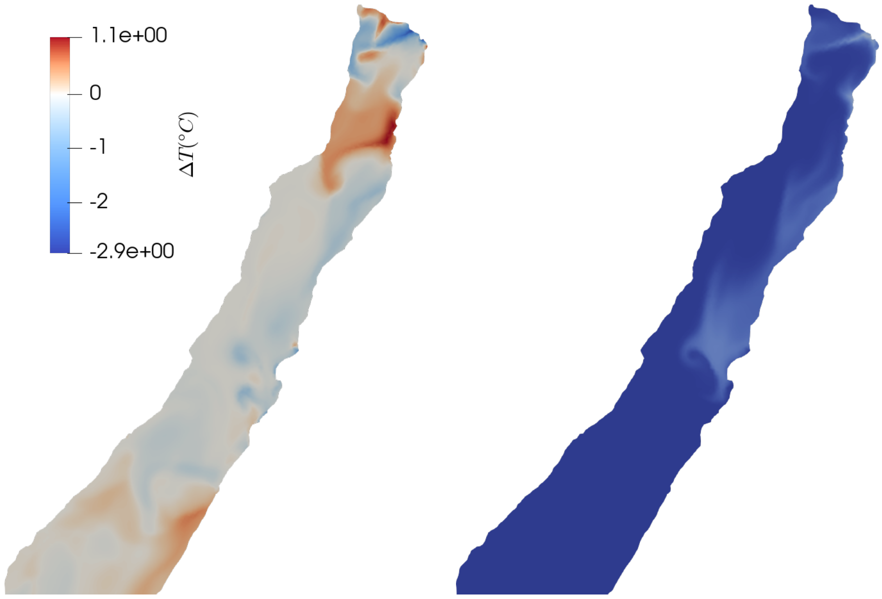


Figure 5.7: Inflow tracer distribution in the region of lake surface shown in (a) at 17:00, 30/10/2018 (b); 17:00, 30/10/2018 (c); 17:00, 1/11/2018 (d); 17:00, 2/11/2018 (e); 17:00, 3/11/2018 (f). Data from the global simulation



(a) Temperature difference between simulations with and without inflow

(b) Inflow tracer distribution

Figure 5.8: Surface maps at 17:00, 3/11/2018. Data from the global simulations

for hydrological analyses, as they provide a first assessment of the actual impact of an intense discharge event on the lake.

## 5.3 Local simulations

Simulations were repeated on a finer mesh of the northernmost region of the lake for better resolution of the flow structure. The southern boundary was chosen far enough from the tunnel inflow section that no amount of tracer reaches it in the first 24 hours after the tunnel opening, and the water level was imposed as equal to its reference value on this boundary; this choice is justified by the distance of the boundary from the area of interest. The target element edge length was set to 5 m in a region of radius $\sim 300$ m around the inflow point, then smoothly transitioning to a target edge length of 50 m in the rest of the domain. The much smaller element size with respect to the global simulation required reducing the time step, to ensure stability of the numerical schemes (see section 3.5.2). Adequate time steps had to be found by trial and error, and the final setup was as follows:

1. A 2-day spinup period, from 22:00, 27/10/2018 to 22:00, 29/10/2018, with wind forcing and no heat transport/diffusion (frozen stratification); 10 seconds time step.

2. A 19-hour simulation corresponding to inflow from the tunnel, with wind forcing

and heat transport/diffusion; 5 seconds time step for the first 5 hours, 2 seconds time step for the remaining time. The required reduction of the time step is due to the inflow rate increasing to its maximum value, leading to a velocity increase near the inflow section.

The same settings with respect to turbulence models were used as in the global simulations. In this case too, the inflow was distributed on multiple boundary elements; this time, the modelled inflow width amounted to about 50 m, as can be seen in Figure 5.9, thus still larger than the actual one, but much closer to it than in the global simulation. This is reflected, as already mentioned, on the modelled initial inertia of the inflow, which is slightly less than the one in the real conditions, hence the extent of the surface layer of tracer at the mouth of the tunnel is probably underestimated. Moreover, Figure 5.9 clearly shows a convergence of velocities at the edge of the area where the tracer concentration is high: this is where downwelling occurs.

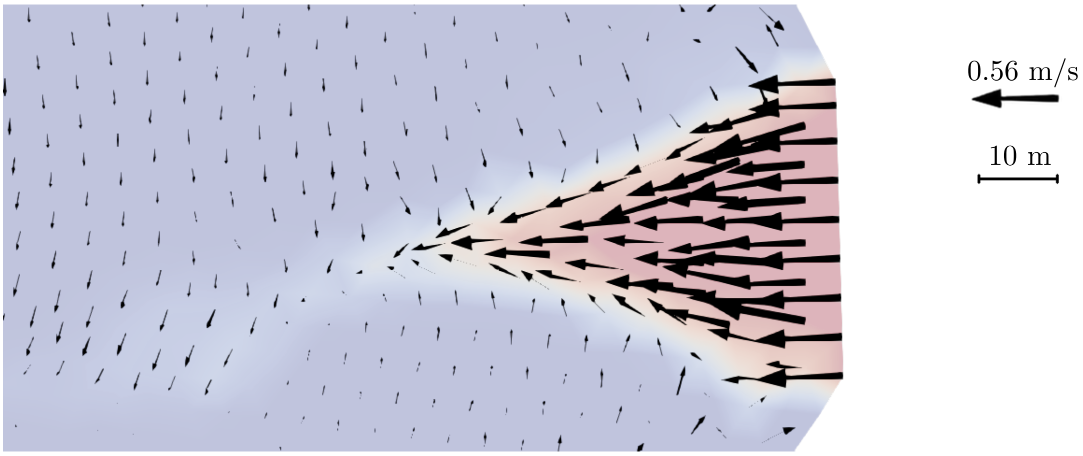


Figure 5.9: Surface velocity map at the tunnel inflow point at 03:00, 30/10/2018. Background colour: surface concentration of inflow tracer

Figure 5.11(a) shows flow velocities and tracer concentration on an horizontal slice at 49 m depth, while Figure 5.11(b) shows a vertical slice. Close to the inflow point, the flow behaves as a jet. The velocity gradually decreases as the jet section increases and some of the surrounding water is entrained, mixing with the inflowing water and thus changing its temperature. In the region far from the inflow point, the flow shows the typical behaviour of an intrusion. This was verified by comparing the velocity of the front relative to the current along the section cutout line calculated in the simulation with the one predicted by equation (2.79). The greatest difficulty in doing this comparison is in defining the boundary of the intrusion properly. The orders of magnitude match for any reasonable choice anyway. For example, subtracting velocity vectors from the numerical solution just upstream and downstream of the front along the cutout line (at point marked F in the horizontal section) and taking the norm, a value of 0.085 m/s is obtained; on the other hand, a theoretical estimate computed by sampling density and intrusion thickness in the points marked F1 and F2 in the vertical section yields a value of $u_F = 0.089$ m/s. This fact shows that the main driver of the tracer propagation is indeed the hydrostatic pressure gradient, as assumed in section 2.9, and not turbulent or molecular diffusion as one might have reasonably suspected.

The simulation was repeated including the inflow from river Sarca. Since the river had a very large flow rate in the days before the discharge from the Adige-Garda tunnel (see figure 5.4), it was important to simulate the effects of its inflow during the spinup phase, in order to obtain a realistic velocity field after the initialization. This required avoiding the frozen stratification assumption: heat advection-diffusion needed to be activated from the start so that the thermal effects of the inflow of relatively cold water from the river could be taken into account.

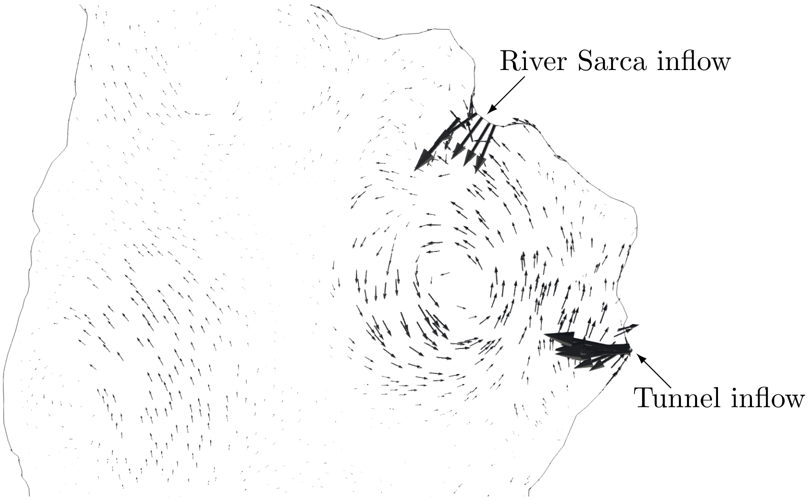
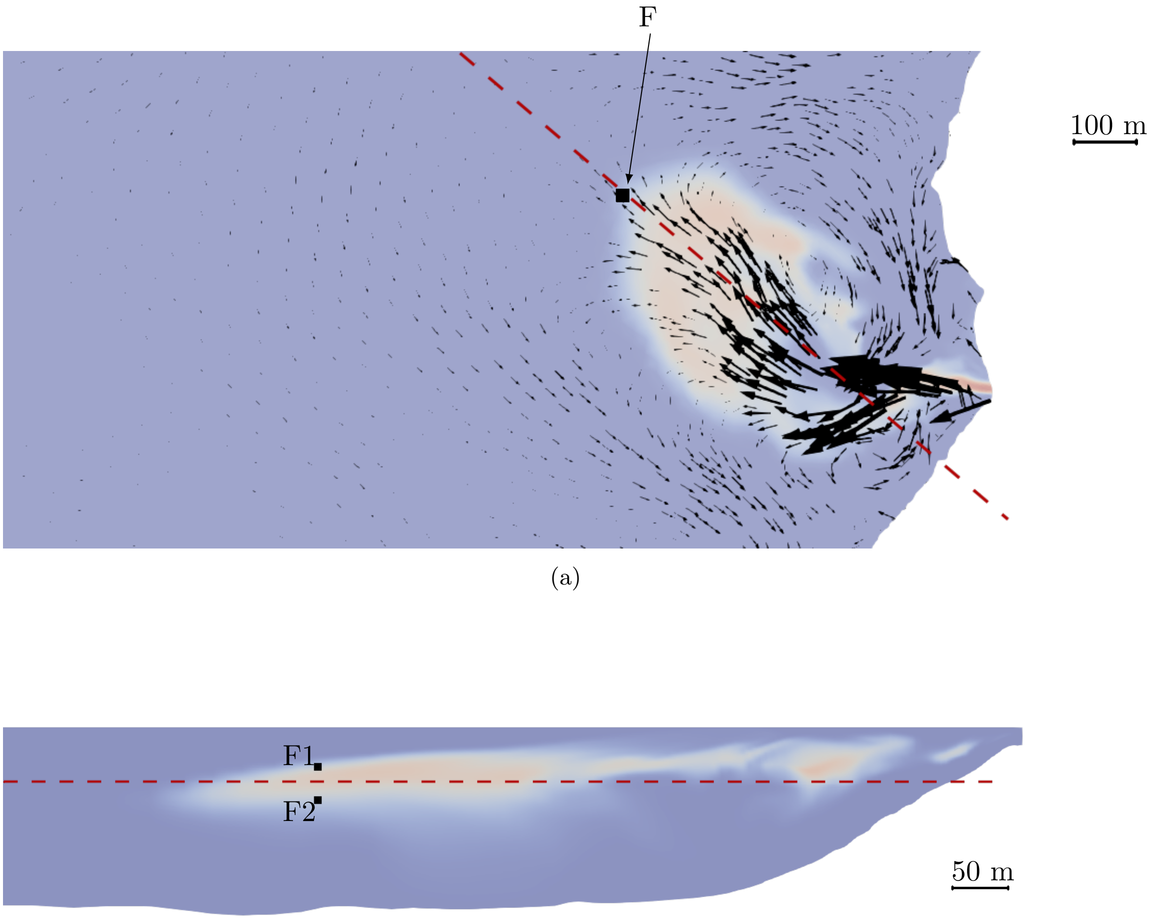


Figure 5.10: Surface velocity map with inflow from river Sarca at 06:00, 30/10/2018. A counterclockwise gyre is clearly visible

The inflow of river Sarca generated an intrusion, but the most evident feature of interest for the present study is that it created a surface counterclockwise gyre (see Figure 5.10) that was not apparent in the simulations without the Sarca inflow. It is interesting to notice that such flow feature had been observed in interviews with people who work or practice sports on the lake (see [5]), but in the rather different context of summer conditions, characterized by daily breezes and flowrates of river Sarca much smaller than the ones considered in this scenario. This current, having an impact also at depth, influenced the transport of the tracer from the Adige-Garda tunnel, leading to higher concentrations of such tracers close to the northern shore than compared to the case in which the effect of Sarca is not taken into account. While this effect cannot be expected to influence the global tracer dispersion pattern, it indicates that global simulations without the Sarca inflow may underestimate the tracer distribution along the northeastern shore.



(a)

(b)

Figure 5.11: Detail of the jet-intrusion dynamics close to the tunnel inflow; horizontal (a) and vertical (b) sections. Red dashed lines: projection of the vertical section on the horizontal section and vice versa. Background colour: concentration of inflow tracer. Data sampled at 03:00, 30/10/2018

## 5.4 Comparison with SAR data

In this section, the results of the global simulations are compared with data from satellite-mounted SAR. Because of cloud cover, optical images were not available for the interval of interest.

SAR is the acronym of Synthetic Aperture Radar. To understand how this remote sensing system works, we first recall how a radar works. The core of a radar system is a transmitting/receiving antenna, which emits coherent electromagnetic wave trains and listens for their echo. The echo can be produced by reflection of the wave on an object or surface; in general, a target. The delay and phase shift of the received pulse with respect to the sent pulse depend on the distance between the radar and the target. The ratio of the received pulse amplitude (backscatter) to the sent pulse amplitude depends on many properties:

- the target size and distance;

- the target electromagnetic properties, which determine the relative amplitude of the reflected and transmitted waves at its interface;

- the surface roughness of the target. Indeed, while the wave reflected by a smooth surface has maximum amplitude for a reflection angle equal to the incidence angle, and negligible amplitude in all other directions, a rough surface scatters incident waves in all directions randomly. A surface can have a smooth or rough behaviour depending on the ratio of its typical roughness scale $k$ to the incident electromagnetic wavelength $\lambda$: if $k$ is larger than, or comparable to, $\lambda$, we have a rough-surface behaviour.

Radar resolution can be improved by using wide-band signals, and 2D mapping of targets is made possible by the adoption of long arrays of antennas, instead of a single antenna. The SAR technology is a modification of the antenna-array setup which exploits the motion of the radar antenna over a target region. A single antenna travels along a trajectory, sending and receiving pulses at predetermined intervals. The antenna can be mounted on a rail, aircraft, or satellite.

Satellite-mounted SAR data can be used to estimate the wind field over a body of water by detecting the surface roughness with lengthscale comparable with the electromagnetic wavelength. Such lengthscale, indeed, depends on wind forces, and semi-empirical relationships have been formulated to retrieve wind speed from SAR [61]. SAR data can also be relevant to water-quality assessments [62], in particular for oil spill detection and turbidity estimation. Some advantages of SAR data over optical images are the possibility of observing at night and a less marked influence of weather.

While the use of SAR data for marine applications is quite common, it is much less widespread in limnology, so that not many references or semi-empirical models are available. However, it is still possible to compare the available images with the results of simulations in terms of surface tracer distribution. Since at the time of data acquisition the wind field over the lake was smooth, with a northern wind of moderate intensity at all stations, then its effect on backscatter can be considered approximately uniform; this means that the dominant cause of backscatter variation must be a change of the electromagnetic properties of the surface water. Thus, a change in backscatter could be attributed to the presence of turbidity or temperature variations and, ultimately, a possible signature of the presence of water from the Adige-Garda spillway.

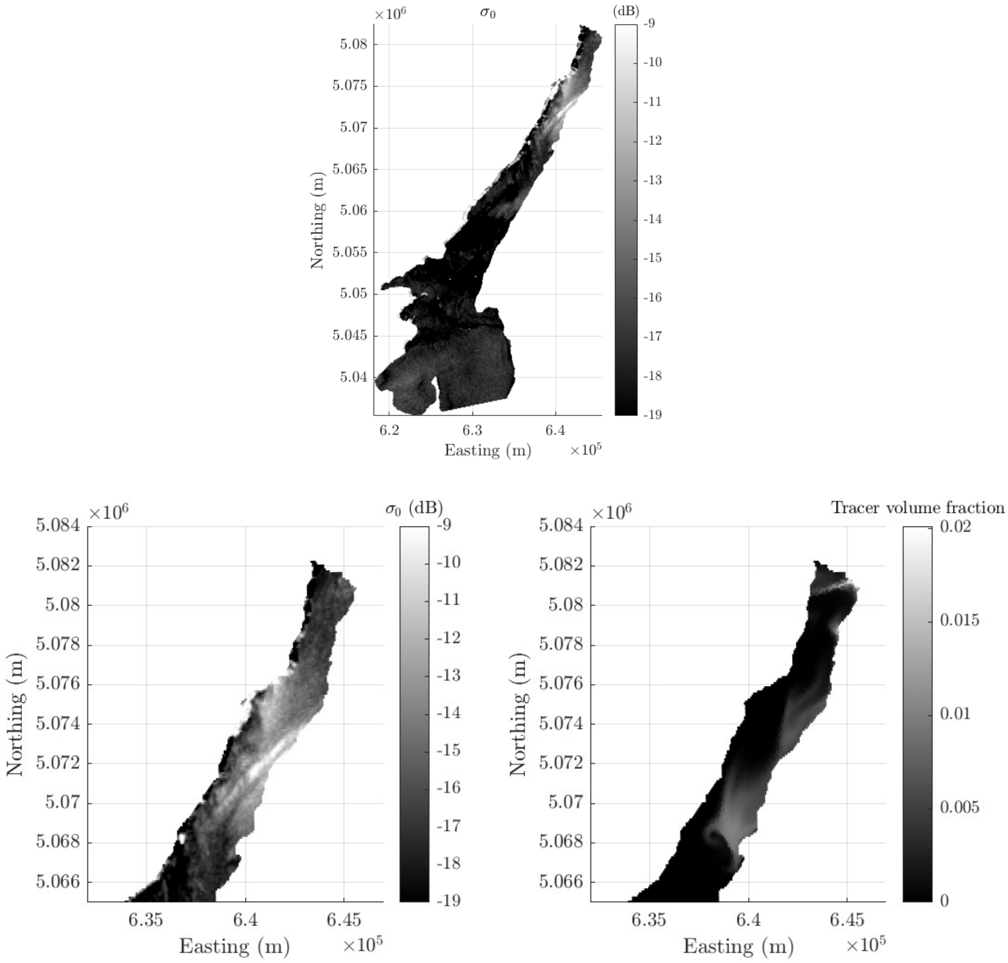


Figure 5.12: SAR backscatter map at 17:06, November 3rd 2018 (full map and detail of the northern region with comparison with numerical simulations)

SAR data are obtained from the Sentinel 1 mission[9]. For the present scenario, the most reliable image was the one taken at 17:06 on November 3rd, because it was the only available time with no precipitations. Precipitations influence the backscatter value, so in presence of rain the interpretation of the images becomes more critical. The backscatter map is reported in Figure 5.12. Comparing the data with the results of numerical simulation, we can see that the extension of the surface tracer concentration

[9] https://sentinel.esa.int/web/sentinel/missions/sentinel-1

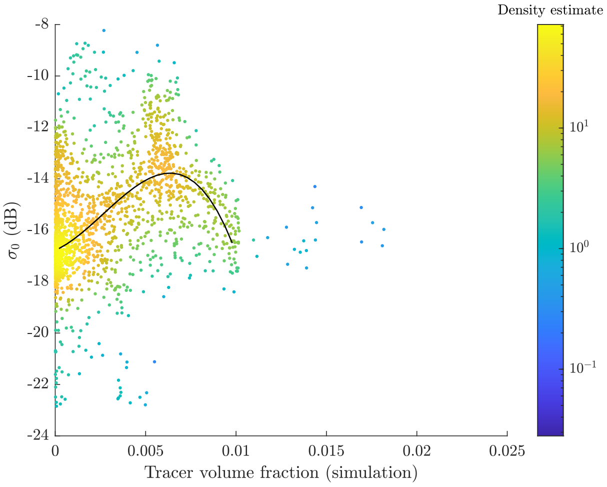


Figure 5.13: Scatterplot of SAR backscatter v. tracer concentration from the simulations. Black solid line: polinomial regression of order 3

along the lake's longitudinal axis in the northern region is reasonably consistent with the backscatter anomaly in the SAR image: this corroborates the results of the numerical simulations, showing that the spreading of the tracer was predicted correctly. The two distributions, though, do not match except for their similar spatial extent along the northern trunk. Moreover, the SAR image shows a high backscatter anomaly along the western shore, that is due to the terrain slope. This is an effect called foreshortening [20] [39]: slopes directly facing the radar (fore-slopes) appear more illuminated and smaller than their actual extent, while back-slopes appear dark and larger than they actually are. Once the foreshortening effect is removed, the presence of correlation between the observed SAR data and the results in terms of tracer concentration can be investigated objectively using a scatter plot, shown in Figure 5.13. The plot is obtained by first interpolating the SAR data over the grid of the numerical simulations and then using the Matlab function `scatter_kde`[10]. The colour of each point depends on the density of other points in its neighbourhood. For tracer volume fractions less than about 0.008, a correlation emerges; the correlation is lost for greater values of tracer volume fraction, also because of the low number of data points in such region.

We can thus conclude that, although there are some interpretation difficulties and the literature in this field is still quite scarse, the analysis of SAR images can be used to validate the results of simulations. More data would be needed to obtain statistics with better significance, but the other satellite images available for the present scenario could not be used because of the weather conditions.

[10] https://it.mathworks.com/matlabcentral/fileexchange/65728-scatter-plot-colored-by-kernel-density-estimate

# Chapter 6

# Conclusions

In this thesis, CFD simulations on lake Garda were carried out in order to understand the effects of a release of water from the Adige-Garda tunnel on the lake and to describe the relevant flow features. The main and most characterizing results obtained are summed up here, concerning both water quality properties and flow physics.

Since inflowing water is colder, and thus denser than the surface water (salinity being regarded as a secondary effect), it plunges to depth propagating first as a jet, and later as an intrusion. The main driver is thus the hydrostatic pressure gradient due to the density difference between the water discharged from the tunnel and the ambient stratification, rather than turbulent or viscous diffusion as might have been expected for such a scenario. This was confirmed by comparison between the numerical solution and an analytical model obtained by an extension of models available in the literature on intrusions. The Coriolis effect was also observed to have a relevant influence. On the surface, an hydrologic front forms close to the inflow point, meaning that surface velocities converge close to the lines of plunging of the inflowing water; the presence of a clear color demarcation on the surface is confirmed by eyewitnesses (videos taken during the event). The water from the tunnel starts slowly reemerging to the surface in significant quantities only hours after the opening event is over; this was observed also in a previous paper on the subject [22]. The eastern shore is the most affected by the arrival of the inflown water as the current travels southwards, thus affecting temperature and pollutant concentration in such area. This effect is consistent with the fact, established in previous research [43], that under prevalent northern winds a circulation develops on transverse planes due to the Coriolis effect, with downwelling at the western shore and upwelling at the eastern. The extent of the surface distribution of tracer at the end of the simulation is also in good qualitative agreement with satellite-mounted SAR images. It was possible to estimate the temperature anomaly distribution on the surface with respect to an idealized case in which the tunnel does not discharge water.

This work also highlighted many modelling criticalities and uncertainties. The lack of hydrodynamic measurements in the lake limits the possibility of calibrating the model properly or validating the results. In particular, this is reflected by the difficulty in setting the turbulence models. The model choice proved very doubtful in itself, since different models lead to quite different solutions even on very simple domains. Indeed, the research about the possible modelling choices, and even about the underlying basic physical phenomena of turbulence in stratified flows, still has many open questions. Concerning boundary conditions, imposing atmospheric forcing data correctly is not trivial, and a more refined choice would have been coupling the hydrodynamic simulations with weather simulation, but this was not possible in the time frame of a master's thesis. The most important possible improvements and developments are thus related to turbulence models and external forcing.

I am hopeful that the discussions reported here may be helpful for a future development of studies such as this, and that the results obtained in this project may be expanded further. Also, I hope that this thesis can be useful for other people with the same background as mine, aerodynamics, who are interested in working on this kind of problems.

# Appendice A

# Estratto in italiano

## A.1 Introduzione

Lo scopo di questo lavoro di tesi è valutare, tramite simulazioni idrodinamiche, gli effetti del vento e degli influssi sulle proprietà termiche e di miscelamento del lago di Garda, con particolare attenzione all'evento di apertura della galleria Adige-Garda avvenuto nel mese di ottobre 2018.

Il lago di Garda è il più grande lago italiano, con una superficie di 368 km$^2$, e uno dei più profondi, con una profondità massima di circa 350 m. Il suo immissario principale è il fiume Sarca, che entra nel lago a nord, e l'unico emissario è il fiume Mincio, che esce dal lago a sudest. A metà del XX secolo (i lavori di costruzione sono stati svolti tra il 1939 e il 1959) fu costruita una galleria in modo da poter scaricare nel lago l'acqua del fiume Adige in caso di piena. Il lago si può considerare diviso in due regioni in base alla batimetria: il ramo settentrionale, allungato e profondo, con fondale ripido vicino alla riva e pressoché piatto lungo l'asse longitudinale, simile ad altri laghi subalpini come il Lario e il Maggiore, e il bacino sudorientale, relativamente poco profondo. Le due regioni sono separate da una cresta sommersa che si estende dalla penisola di Sirmione alla punta san Vigilio (mostrate nella mappa in Figura 1.2).

Le simulazioni sono state effettuate utilizzando il codice idrodinamico Telemac3D[1] per la soluzione numerica delle equazioni di Navier-Stokes con media di Reynolds (RANS) discretizzate tramite il metodo degli elementi finiti.

Nel capitolo 1 viene data un'introduzione generale alla limnologia, ai problemi che affronta e in particolare, tra questi, ai più rilevanti per il Garda; segue una revisione degli articoli di ricerca a riguardo.

Nel capitolo 2 vengono presentate le equazioni di governo del problema, le ipotesi su cui sono basate e alcune possibili scelte modellistiche. Viene inoltre derivato un modello analitico per le intrusioni.

Il capitolo 3 mostra i metodi numerici per la soluzione delle equazioni descritte nel capitolo 2: la discretizzazione dello spazio (generazione della griglia di calcolo) e del tempo e il trattamento separato dei diversi termini delle equazioni (metodo a passi

[1] http://opentelemac.org/

frazionari). Vengono richiamate le principali proprietà del metodo degli elementi finiti e dei metodi a distribuzione del residuo (MURD).

Nel capitolo 4 vengono descritte alcune semplici, ma rilevanti, simulazioni preliminari, effettuate introducendo gradualmente fenomeni fisici e difficoltà modellistiche in modo da comprendere la fisica di base e poter scegliere in modo appropriato modelli e parametri.

Il capitolo 5 riporta i parametri e l'organizzazione delle simulazioni dell'apertura della galleria. Vengono investigati gli effetti delle circolazioni indotte dal vento, della stratificazione e dell'influsso del fiume Sarca sulla dispersione dell'acqua scaricata nel lago, prima a livello globale (intero lago) e poi a livello locale, su una griglia più fine.

Il capitolo 6 è un sommario dei principali risultati. Vengono discusse le incertezze e i possibili sviluppi dell'analisi presentata.

## A.2 Conclusioni

I principali risultati di questa tesi riguardano due aspetti principali: la fisica della corrente indotta dall'influsso dalla galleria e le sue conseguenze in termini di dispersione di eventuali sostanze inquinanti in soluzione.

L'acqua scaricata dalla galleria, essendo più fredda e quindi più densa rispetto all'acqua del lago alla superficie, tende a scendere in profondità. In seguito a un comportamento inizialmente simile a quello di un getto, la corrente assume le caratteristiche di un'intrusione, cioè una corrente governata principalmente dalla differenza di densità tra il suo interno e la stratificazione dell'ambiente. Questa affermazione è supportata dal confronto tra la soluzione numerica e un modello analitico sviluppato a partire da un'estensione di modelli disponibili in letteratura. Anche l'effetto di Coriolis si è dimostrato avere un'influenza rilevante. In superficie, si osserva la formazione di un fronte idrologico attorno al punto di scarico: le velocità superficiali convergono in corrispondenza delle linee lungo cui l'acqua in ingresso si inabissa; tali linee sono ben visibili a causa della differenza di turbidità, e quindi di colore, tra l'acqua dell'Adige e quella del lago, e la loro presenza è confermata da testimonianze video dell'evento. L'acqua immessa dalla galleria, evidenziata da uno scalare passivo nelle simulazioni, inizia a riemergere in quantità significative ore dopo la chiusura della galleria; questo effetto è stato osservato anche in un precedente studio del problema [22]. La riva est del lago risulta la più interessata dall'arrivo dell'acqua dalla galleria, con conseguenti effetti termici e di concentrazione di inquinanti. Questo effetto è consistente con il fatto, osservato in lavori di ricerca precedenti [43], che in condizioni di vento prevalente da nord si creino circolazioni su piani trasversali a causa della rotazione terrestre, con downwelling alla riva ovest e upwelling alla riva est. I risultati in termini di distribuzione di tracciante in superficie al termine della simulazione sono in buon accordo qualitativo con i dati SAR disponibili. È stato possibile stimare la distribuzione dell'anomalia di temperatura in superficie rispetto al caso ideale in cui la galleria non fosse stata aperta.

Questo lavoro di tesi ha evidenziato diverse criticità e incertezze modellistiche. La carenza di misure idrodinamiche nel lago limita la possibilità di calibrare adeguatamente il modello e validarne i risultati. In particolare, questo problema si riflette sulla difficoltà di impostazione dei modelli di turbolenza. La scelta dei modelli si rivela di per sè critica, dato che diversi modelli portano a risultati diversi anche su domini molto semplici. In effetti, la ricerca sulle scelte modellistiche e sulla fisica di base della turbolenza in ambienti stratificati presenta molte questioni aperte. Rispetto, inoltre, alle condizioni al contorno, imporre correttamente le forzanti atmosferiche non è banale, e una scelta più accurata sarebbe stata accoppiare le simulazioni idrodinamiche con simulazioni meteorologiche, ma questo non è stato possibile per motivi di tempo. Pertanto, i principali miglioramenti dell'analisi presentata riguardano i modelli di turbolenza e le forzanti atmosferiche.

Spero che i risultati e le discussioni riportate in questa tesi possano essere utili per lo sviluppo ulteriore di studi come questo, e che questo testo possa essere utile ad altri studenti di aerodinamica che, come me, siano interessati a problemi di fluidodinamica ambientale.

# Bibliography

[1] Informazione sui livelli del lago di Garda. https://www.comunitadelgarda.it/Informazione-sui-livelli/1132-1.html. Accessed: Oct 9, 2020.

[2] *TELEMAC-3D Theory Guide*. www.opentelemac.org, v8p1 edition, 2020.

[3] L'alluvione scampata grazie alla galleria Adige-Garda. A rischio l'ecosistema del lago, riversati 17 milioni di metri cubi di acqua e fango. *Il Dolomiti*, Oct 30, 2018.

[4] M. Amadori. *On the physical drivers of transport processes in Lake Garda: A combined analytical, numerical and observational investigation*. PhD thesis, Doctoral School in Civil, Environmental and Mechanical Engineering, University of Trento, 2020.

[5] M. Amadori, G. Morini, S. Piccolroaz, and M. Toffolon. Involving citizens in hydrodynamic research: A combined local knowledge - numerical experiment on lake Garda, Italy. *Science of The Total Environment*, 722:137720, 2020.

[6] M. Amadori, S. Piccolroaz, L. Giovannini, D. Zardi, and M. Toffolon. Wind variability and Earth's rotation as drivers of transport in a deep, elongated subalpine lake: The case of lake Garda. *Journal of limnology*, 77, 09 2018.

[7] W. Ambrosetti, L. Barbanti, A. Rolla, L. Castellano, and N. Sala. Hydraulic paths and estimation of the real residence time of the water in lago Maggiore (N. Italy): Application of massless markers transported in 3D motion fields. *Journal of limnology*, 71, 05 2012.

[8] G. K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge Mathematical Library. Cambridge University Press, 2000.

[9] T. B. Benjamin. Gravity currents and related phenomena. *Journal of Fluid Mechanics*, 31(2):209–248, 1968.

[10] S. Beretta. *Affidabilità delle costruzioni meccaniche: Strumenti e metodi per l'affidabilità di un progetto*. UNITEXT. Springer Milan, 2010.

[11] K. M. Bryant and M. Akbar. An exploration of wind stress calculation techniques in hurricane storm surge modeling. *Journal of Marine Science and Engineering*, 4(3):58, 2016.

[12] L. Castellano, W. Ambrosetti, L. Barbanti, and A. Rolla. The residence time of the water in lago Maggiore (N. Italy): first results from an Eulerian-Lagrangian approach. *Journal of Limnology*, 69:15–28, 2010.

[13] C. Caulfield. Layering, instabilities, and mixing in turbulent stratified flows. *Annual Review of Fluid Mechanics*, 53(1):113–145, 2021.

[14] V. Cristofori. La galleria Adige-Garda ed il lago di Loppio. https://web.archive.org/web/20191115150832/http://www.bacinimontani.provincia.tn.it/pdf/galleria_adige_garda_loppio.pdf. Accessed: Oct 9, 2020.

[15] T. Dauxois, T. Peacock, P. Bauer, C.-c. Caulfield, C. Cenedese, C. Gorlé, G. Haller, G. Ivey, P. Linden, E. Meiburg, N. Pinardi, A. Neves, N. Vriend, and A. Woods. Confronting grand challenges in environmental fluid dynamics, 11 2019.

[16] P. Davidson. *Turbulence: An Introduction for Scientists and Engineers*. OUP Oxford, 2004.

[17] L. de Haan and A. Ferreira. *Extreme Value Theory: An Introduction*. Springer Series in Operations Research and Financial Engineering. Springer New York, 2006.

[18] A. Decoene and J. Gerbeau. Sigma transformation and ALE formulation for three-dimensional free surface flows. *International Journal for Numerical Methods in Fluids*, 59:357 – 386, 02 2009.

[19] H. Deconinck, M. Ricchiuto, and K. Sermeus. Introduction to residual distribution schemes and comparison with stabilized finite elements. 01 2003.

[20] ESA (European Space Agency). Radar course notes. https://earth.esa.int/web/guest/missions/esa-operational-eo-missions/ers/instruments/sar/applications/radar-courses/content-3/-/asset_publisher/mQ9R7ZVkKg5P/content/radar-course-3-slant-range-ground-range. Accessed: Mar 22, 2020.

[21] A. Fenocchi, M. Rogora, S. Sibilla, M. Ciampittiello, and C. Dresti. Forecasting the evolution in the mixing regime of a deep subalpine lake under climate change scenarios through numerical modelling (lake Maggiore, Northern Italy/Southern Switzerland). *Climate Dynamics*, 51:3521–3536, 11 2018.

[22] N. Ghirardi, M. Amadori, G. Free, L. Giovannini, M. Toffolon, C. Giardino, and M. Bresciani. Using remote sensing and numerical modelling to quantify a turbidity discharge event in lake Garda: Turbidity discharge event in lake Garda from satellite and numerical modelling. *Journal of Limnology*, 10 2020.

[23] H. Haren, S. Piccolroaz, M. Amadori, M. Toffolon, and H. A. Dijkstra. Moored observations of turbulent mixing events in deep lake Garda (I). 2020.

[24] C. Hearn. On the Munk-Anderson equations and the formation of the thermocline. *Applied Mathematical Modelling*, 12(5):450 – 456, 1988.

[25] J.-M. Hervouet. *Hydrodynamics of Free Surface Flows.* John Wiley & Sons, Ltd, 2007.

[26] K. Hutter, W. Yongqi, and I. Chubarenko. *Physics of Lakes: Volume 1: Foundation of the Mathematical and Physical Background.* Advances in Geophysical and Environmental Mechanics and Mathematics. Springer Berlin Heidelberg, 2010.

[27] D. Imboden and A. Wuest. *Mixing Mechanisms in Lakes.* 01 1995.

[28] I. Ipsen and C. Meyer. The idea behind Krylov methods. *The American Mathematical Monthly*, 105:889–899, 12 1998.

[29] G. Ivey, K. Winters, and J. Koseff. Density stratification, turbulence, but how much mixing? *Annual Review of Fluid Mechanics*, 40(1):169–184, 2008.

[30] J. Jankowski. *A non-hydrostatic model for free surface flows.* PhD thesis, 01 1999.

[31] H. Kato and O. M. Phillips. On the penetration of a turbulent layer into stratified fluid. *Journal of Fluid Mechanics*, 37(4):643–655, 1969.

[32] P. K. Kundu. Self-similarity in stress-driven entrainment experiments. *Journal of Geophysical Research: Oceans*, 86(C3):1979–1988, 1981.

[33] S. Kurgalin and S. Borzunov. *A Practical Approach to High-Performance Computing.* Springer International Publishing, 2019.

[34] R. LeVeque. *Numerical Methods for Conservation Laws.* Lectures in Mathematics ETH Zürich, Department of Mathematics, Research Institute of Mathematics. Springer, 1992.

[35] T. Lovato and G. Pecenik. Three-dimensional modeling of pollutant dispersion in lake Garda (North Italy). In F. Jordán and S. E. Jørgensen, editors, *Models of the Ecological Hierarchy*, volume 25 of *Developments in Environmental Modelling*, pages 319 – 330. Elsevier, 2012.

[36] P. Luchini. Universality of the turbulent velocity profile. *Phys. Rev. Lett.*, 118:224501, Jun 2017.

[37] A. Marzocchi, M. Paolini, F. Pasquarelli, and P. Tesini. Numerical simulation of external water inflows into lake Garda.

[38] N. E. Monsen, J. E. Cloern, L. V. Lucas, and S. G. Monismith. A comment on the use of flushing time, residence time, and age as transport time scales. *Limnology and Oceanography*, 47(5):1545–1553, 2002.

[39] A. Monti Guarnieri. *Electromagnetic Imaging (lecture notes).* Politecnico di Milano, 2015.

[40] S. Morillo, J. Imberger, J. P. Antenucci, and D. Copetti. Using impellers to distribute local nutrient loadings in a stratified lake: Lake Como, Italy. *Journal of Hydraulic Engineering*, 135(7):564–574, 2009.

[41] J. O'Donnell. Surface fronts in estuaries: A review. *Estuaries and Coasts*, 16:12–39, 03 1993.

[42] T. R. Osborn. Estimates of the local rate of vertical diffusion from dissipation measurements. *Journal of Physical Oceanography*, 10(1):83–89, 01 1980.

[43] S. Piccolroaz, M. Amadori, M. Toffolon, and H. Dijkstra. Importance of planetary rotation for ventilation processes in deep elongated lakes: Evidence from lake Garda (Italy). *Scientific Reports*, 9:8290, 06 2019.

[44] S. Pope. *Turbulent Flows.* Cambridge University Press, 2000.

[45] A. Quarteroni. *Modellistica Numerica per Problemi Differenziali.* UNITEXT. Springer Milan, 2016.

[46] W. Rodi. *Turbulence Models and Their Application in Hydraulics.* IAHR Monographs. Taylor & Francis, 1993.

[47] N. Salmaso. Effects of climatic fluctuations and vertical mixing on the interannual trophic variability of lake Garda, Italy. *Limnology and Oceanography*, 50(2):553–565, 2005.

[48] N. Salmaso, A. Boscaini, C. Capelli, and L. Cerasino. Ongoing ecological shifts in a large lake are driven by climate change and eutrophication: evidences from a three-decade study in Lake Garda. *Hydrobiologia*, 824, 11 2017.

[49] N. Salmaso and R. Mosello. Limnological research in the deep southern subalpine lakes: Synthesis, directions and perspectives. *Advances in Oceanography and Limnology*, 1:29–66, 06 2010.

[50] E. Schlesinger. *Algebra lineare e geometria.* Zanichelli, 2011.

[51] D. Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM National Conference*, ACM '68, page 517–524, New York, NY, USA, 1968. Association for Computing Machinery.

[52] J. Smagorinsky. General Circulation Experiments with the Primitive Equations. *Monthly Weather Review*, 91(3):99, Jan 1963.

[53] R. Sorensen. *Basic Coastal Engineering.* Springer US, 1997.

[54] V. M. Stepanenko, A. Martynov, K. D. Jöhnk, Z. M. Subin, M. Perroud, X. Fang, F. Beyrich, D. Mironov, and S. Goyette. A one-dimensional model intercomparison study of thermal regime of a shallow, turbid midlatitude lake. *Geoscientific Model Development*, 6(4):1337–1352, 2013.

[55] W. D. Toit. Radial basis function interpolation. Master's thesis, 2008.

[56] I. K. Tsanis. Simulation of wind-induced water currents. *Journal of Hydraulic Engineering*, 115(8):1113–1134, 1989.

[57] L. Umlauf, H. Burchard, and C. Hutter. Extending the $k$–$\omega$ turbulence model towards oceanic applications. *Ocean Modelling*, 5:195–218, 12 2003.

[58] M. Ungarish. *An Introduction to Gravity Currents and Intrusions.* CRC Press, 2009.

[59] G. Vallis. *Atmospheric and Oceanic Fluid Dynamics: Fundamentals and Large-scale Circulation.* Cambridge University Press, 2006.

[60] H. van Haren, A. Cimatoribus, and L. Gostiaux. Where large deep-ocean waves break. *Geophysical Research Letters*, 42(7):2351–2357, Apr. 2015.

[61] X. Yang and X. Li. *Ocean Remote Sensing with Synthetic Aperture Radar.* 01 2018.

[62] Y. Zhang, J. Pulliainen, S. Koponen, and M. Hallikainen. Water quality retrievals from combined Landsat TM data and ERS-2 SAR data in the Gulf of Finland. *Geoscience and Remote Sensing, IEEE Transactions on*, 41:622 – 629, 04 2003.