

A conservative semi-Lagrangian finite volume method for convection-diffusion problems on unstructured grids

Ilham Asmouh^{1,*} Mofdi El-Amrani^{2,†} Mohammed Seaid^{3,‡} Najj Yebari^{4,§}

¹ *International Water Research Institute, University Mohammed VI Polytechnic, Benguerir, Morocco*

² *Dpto. Matemáticas Aplicada, Universidad Rey Juan Carlos, 28933 Móstoles-Madrid, Spain*

³ *Department of Engineering, University of Durham, South Road, Durham DH1 3LE, UK*

⁴ *Department of Mathematics, Faculty of Sciences, University Abdelmalek Essaadi, Tetouan, Morocco*

Abstract

A conservative semi-Lagrangian finite volume method is presented for the numerical solution of convection-diffusion problems on unstructured grids. The new method consists of combining the modified method of characteristics with a cell-centered finite volume discretization in a fractional-step manner where the convection part and the diffusion part are treated separately. The implementation of the proposed semi-Lagrangian finite volume method differs from its Eulerian counterpart in the fact that the present method is applied at each time step along the characteristic curves rather than in the time direction. To ensure conservation of mass at each time step, we adopt the adjusted advection techniques for unstructured triangular grids. The focus is on constructing efficient solvers with large stability regions and fully conservative to solve convection-dominated flow problems. We verify the performance of our semi-Lagrangian finite volume method for a class of advection-diffusion equations with known analytical solutions. We also present numerical results for a transport problem in the Mediterranean sea.

Keywords. Finite volume method; Modified method of characteristics; Cell-centered discretization; Convection-diffusion equations; Unstructured grids.

1 Introduction

Many physical applications in fluid mechanics, air pollution, atmospheric dynamics, ocean circulation, transport in porous media or meteorology have been modeled using a class of convection-diffusion-reaction equations as

$$\begin{aligned} \frac{\partial c}{\partial t} + \mathbf{v}(t, \mathbf{x}) \cdot \nabla c - \nu \Delta c &= f(t, \mathbf{x}), & (t, \mathbf{x}) \in (0, T] \times \Omega, \\ c(0, \mathbf{x}) &= c_0(\mathbf{x}), & \mathbf{x} \in \Omega, \end{aligned} \tag{1}$$

where Ω is an open bounded subdomain in \mathbb{R}^2 with smooth boundary Γ and $[0, T]$ is a time interval. Here, $c(t, \mathbf{x})$ denotes the concentration of some species, $\mathbf{v}(t, \mathbf{x})$ the velocity field, ν the diffusion coefficient, $f(t, \mathbf{x})$ the reaction term, and $c_0(\mathbf{x})$ is fixed initial condition. We assume that equations (1) are equipped with well defined boundary conditions depending on the problem under study. We also assume that \mathbf{v} is divergence-free

$$\nabla \cdot \mathbf{v} = 0. \tag{2}$$

*ilham.asmouh@um6p.ma

†mofdi.elamrani@urjc.es

‡m.seaid@durham.ac.uk

§yebarinaji@gmail.com

Solving numerically the equations (1) is still a challenging task in the case of convection-dominated problems, particularly when certain nondimensional parameters governing the dynamics of these problems reach high values. As example of these parameters, we mention the Reynolds number for the Navier-Stokes equations and the Peclet number for the convection-diffusion equations. At high Peclet numbers, this convective term is a source of computational difficulties and oscillations, see for example [29]. It is also well known that the solutions of equations (1) present steep fronts and even shock discontinuities, which need to be resolved accurately in applications and often cause severe numerical challenges. Lagrangian methods are among the numerical techniques widely used to handle these difficulties generated from the presence of convective terms in the governing equations. The main shortcoming for these methods is the grid distortion drawback specially when characteristic curves are required to be computed for time-dependent velocity fields. To overcome this disadvantage, the so-called semi-Lagrangian methods are considered in the literature. In contrast to the traditional Lagrangian methods, which follow the flow particles forward in time, the semi-Lagrangian methods trace backward the position at current time level of particles that will reach the points of a fixed mesh at next time level. The semi-Lagrangian methods have been extensively used in weather prediction for their power to describe the motion of a particle fluid, see for instance [26, 27, 30]. Allowing for large timesteps to be used in the simulations while guarantying stability and accuracy make this class of methods a very attractive choice in engineering computations. However, the drawback of these methods is the failure to conserve the mass during the simulation.

Several contributions have been published in the literature to develop fully conservative semi-Lagrangian methods. For example, authors in [32] proposed remapping the Lagrangian volume procedure using cubic-interpolated propagation method. However, the mass is introduced as an additional variable and used in a correction step. A conservative semi-Lagrangian method is introduced in [7] for the Vlasov-Poisson system in collision-less plasma applications. The procedure to guarantee conservation in this approach is based on dimensional splitting using a fifth-difference Hermite weighted interpolation with WENO limiters to control oscillations. However, this method would be computationally very demanding for realistic applications. Another procedure based on the adjoint property is proposed in [8]. This method follows ideas of support operators introduced in [33] where the adjoint property between continuity equation and advection equation is used together with column-balance property to enforce conservation on an arbitrary advection scheme. For convection-dominated problems on unstructured meshes, this method may become unstable. In electromagnetic field, authors in [20] studied a time-splitting Fourier spectral method for solving the semi-classical Schrödinger equation. The focus in their approach is on the non-uniform fast Fourier transform algorithm to interpolate Fourier series in the convection step but the conservation is achieved using high-order interpolation technique and considering periodic boundary conditions only.

Coupling the semi-Lagrangian method with finite volume discretizations has also been investigated in recent years. A conservative semi-Lagrangian finite volume method using spectral element dynamics for multi-tracer applications has been proposed in [17]. Similar ideas were proposed in [5] for the same field of applications with the difference in enforcing conservation via a problem of optimization instead. However, the extension of these techniques to unstructured meshes is not straightforward. Reconstruction of mass-conservative semi-Lagrangian method without relying on splitting operators, authors in [6] implemented a discontinuous Galerkin method. The main idea in their implementation was to reconstruct quadratic curves to approximate sides of upstream cells and to preserve positivity of the computed solution, a high-order bound-preserving filter is employed. However, the reconstruction of numerical fluxes in these methods requires solution of Riemann problems at the control volumes that may become complicated for convection problems for highly deformational fields. Following similar ideas as those reported in [20], a class of semi-Lagrangian finite volume method with arbitrary order of accuracy is proposed [13]. In this approach, spectral methods of periodic type are used for spatial discretization and high-order schemes are used for time integration. It should be stressed that most of previous methods are applied on regular Cartesian meshes and their extension to general complex geometries may be challenging in terms of computational cost and difficult in implementation. In addition, these methods have been studied for pure convection problems and their extension to convection-diffusion problems may limit their efficiency.

The objective of the current work is to devise a numerical approach able to accurately approximate solution to convection-diffusion problems on unstructured grids. The aim is to develop a fast and accurate family of semi-Lagrangian finite volume methods that satisfy the mass-conservation at each time step during the simulation process. Our main goal is to present a class of numerical methods that are simple, easy to implement, fully conservative, and accurately solve convection-dominated problems without relying on highly demanding solvers. The proposed semi-Lagrangian finite volume algorithm belongs to the class of methods that employ fractional steps where the convection part and the diffusion part are treated separately. To ensure conservation of mass at each time step, we implement the adjusted advection techniques for unstructured triangular grids. It should be mentioned that semi-Lagrangian finite element methods with adjusted advection have been proposed in [10] for solving transport problems on structured grids. The main features of the semi-Lagrangian finite volume method proposed in this study are on one hand, the capability to satisfy the conservation property resulting in numerical solutions free from spurious oscillations, and on the other hand, the achievement of strong stability and high accuracy for numerical solutions containing steep gradients. These features are verified using several test examples of the two-dimensional convection-diffusion problems (1) including a transport problem of pollution in the Mediterranean sea. Results presented in this paper show high resolution of the proposed semi-Lagrangian finite volume method and permit the straightforward application of the method to more complex, physically based convection-diffusion problems. The present study represents a step towards the implementation of a fully conservative semi-Lagrangian finite volume method for the numerical solution of transport and dispersion of pollutants on large sea-surface regions.

This paper is organized as follows. Formulation of the semi-Lagrangian finite volume method for convection problems is presented in section 2. We propose several procedures for the interpolation step of the Lagrangian stage and we also consider the adjusted advection procedure to ensure conservation in the method. Section 3 is devoted to the implementation of the method for convection-diffusion problems. We implement a second-order implicit scheme for the time integration of semi-discrete equations. In section 4, we examine the numerical performance of the proposed method using various test examples of convection-diffusion problems. The obtained results demonstrate that our semi-Lagrangian finite volume method preserves the expected conservation as well as the accuracy. Concluding remarks are summarized in section 5.

2 Conservative semi-Lagrangian finite volume method

To explain the steps involved in the semi-Lagrangian finite volume method we first consider the homogeneous advective part of the problem (1) reformulated using the total derivative as

$$\begin{aligned} \frac{Dc}{Dt} := \frac{\partial c}{\partial t} + \mathbf{v}(t, \mathbf{x}) \cdot \nabla c &= 0, & (t, \mathbf{x}) \in (0, T) \times \Omega, \\ c(0, \mathbf{x}) &= c_0(\mathbf{x}), & \mathbf{x} \in \Omega. \end{aligned} \tag{3}$$

Notice that $\frac{Dc}{Dt}$ measures the rate of change of the solution c following the trajectories of the flow particles. The main idea of the semi-Lagrangian method is to impose a regular grid at the new time level and to backtrack the flow trajectories to the previous time level. At the old time level, the quantities that are needed are evaluated by interpolation from their known values on a regular grid. In this section we formulate the fundamental steps of the semi-Lagrangian finite volume method we consider in the current work.

2.1 Calculation of departure points

Let us divide the time interval into sub-intervals $[t_n, t_{n+1}]$ with uniform length Δt and discretize the spatial domain $\bar{\Omega} = \Omega \cup \Gamma$ in conforming triangular elements \mathcal{T}_i as $\bar{\Omega} = \cup_{i=1}^{Ne} \mathcal{T}_i$, with Ne is the total number of control volumes. Each triangle represents a control volume and the variables are located at

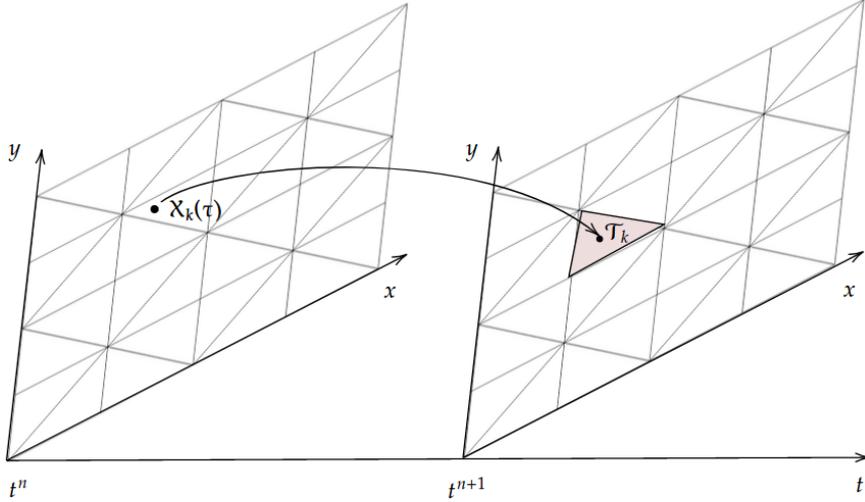


Figure 1: A schematic diagram showing the main quantities used in calculation of the departure points.

the geometric centers of the cells. We use the notation w^n to denote the value of a generic function w at time t_n and w_i^n to denote the average value of the solution w in the cell \mathcal{T}_i at time t_n ,

$$w_i^n = \frac{1}{|\mathcal{T}_i|} \int_{\mathcal{T}_i} w(t_n, \mathbf{x}) d\mathbf{x},$$

where $|\mathcal{T}_i|$ is the area of the control volume \mathcal{T}_i . Following for example [27], the characteristic curves of the equation (3) are solutions of the following initial-value problem

$$\begin{aligned} \frac{d\boldsymbol{\mathcal{X}}_i(t)}{dt} &= \mathbf{v}_i(t, \boldsymbol{\mathcal{X}}_i(t)), & t \in [t_n, t_{n+1}], \\ \boldsymbol{\mathcal{X}}_i(t_{n+1}) &= \mathbf{x}_i. \end{aligned} \quad (4)$$

Here, $\boldsymbol{\mathcal{X}}_i(t) = (X_i(t), Y_i(t))^T$ is the departure point at time t of a particle that will arrive at $\mathbf{x}_i = (x_i, y_i)^T$ the center of the control volume \mathcal{T}_i . Note that the semi-Lagrangian finite volume method does not follow the flow particles forward in time, as a Lagrangian method does, instead it traces backwards the position at time t_n of particles that will reach the points of a fixed mesh at time t_{n+1} , see Figure 1 for an illustration. Therefore, the semi-Lagrangian finite volume method avoids the grid distortion difficulties that the conventional Lagrangian schemes have. Hence, the solutions of (4) can be expressed as

$$\boldsymbol{\mathcal{X}}_i(t_n) = \mathbf{x}_i - \int_{t_n}^{t_{n+1}} \mathbf{v}_i(t, \boldsymbol{\mathcal{X}}(t)) dt. \quad (5)$$

To evaluate the integral in (5), we use a second-order extrapolation based on the mid-point rule investigated in [31] among others. Thus, we use $\boldsymbol{\delta}_i$ to denote the displacement between a mesh point \mathbf{x}_i on the new level and the departure point $\boldsymbol{\mathcal{X}}_i(t_n)$ of the trajectory to this point on the previous time level *i.e.*

$$\boldsymbol{\delta}_i = \mathbf{x}_i - \boldsymbol{\mathcal{X}}_i(t_n). \quad (6)$$

Applying the mid-point rule to approximate the integral in (5) yields

$$\boldsymbol{\delta}_i = \Delta t \mathbf{v}_i \left(t_{n+\frac{1}{2}}, \boldsymbol{\mathcal{X}}_i(t_{n+\frac{1}{2}}) \right).$$

Using the second-order extrapolation

$$\mathbf{v}_i \left(t_{n+\frac{1}{2}}, \boldsymbol{\mathcal{X}}_i(t_{n+\frac{1}{2}}) \right) = \frac{3}{2} \mathbf{v}_i \left(t_n, \boldsymbol{\mathcal{X}}_i(t_{n+\frac{1}{2}}) \right) - \frac{1}{2} \mathbf{v}_i \left(t_{n-1}, \boldsymbol{\mathcal{X}}_i(t_{n+\frac{1}{2}}) \right),$$

and the second-order approximation

$$\mathcal{X}_i(t_{n+\frac{1}{2}}) = \mathbf{x}_i - \frac{1}{2}\boldsymbol{\delta}_i,$$

we obtain the following implicit formula for $\boldsymbol{\delta}_i$

$$\boldsymbol{\delta}_i = \Delta t \left(\frac{3}{2}\mathbf{v}_i \left(t_n, \mathbf{x}_i - \frac{1}{2}\boldsymbol{\delta}_i \right) - \frac{1}{2}\mathbf{v}_h \left(t_{n-1}, \mathbf{x}_i - \frac{1}{2}\boldsymbol{\delta}_i \right) \right).$$

To compute $\boldsymbol{\delta}_i$ we consider the following successive iteration procedure

$$\begin{aligned} \boldsymbol{\delta}_i^{(0)} &= \Delta t \left(\frac{3}{2}\mathbf{v}_i(t_n, \mathbf{x}_i) - \frac{1}{2}\mathbf{v}_h(t_{n-1}, \mathbf{x}_i) \right), \\ \boldsymbol{\delta}_i^{(k)} &= \Delta t \left(\frac{3}{2}\mathbf{v}_i \left(t_n, \mathbf{x}_i - \frac{1}{2}\boldsymbol{\delta}_i^{(k-1)} \right) - \frac{1}{2}\mathbf{v}_i \left(t_{n-1}, \mathbf{x}_i - \frac{1}{2}\boldsymbol{\delta}_i^{(k-1)} \right) \right), \quad k = 1, 2, \dots \end{aligned} \quad (7)$$

The iterations (7) are terminated when the following criteria

$$\frac{\|\boldsymbol{\delta}^{(k)} - \boldsymbol{\delta}^{(k-1)}\|}{\|\boldsymbol{\delta}^{(k-1)}\|} < \varepsilon, \quad (8)$$

is satisfied for the Euclidean norm $\|\cdot\|$ and a given tolerance ε . In our computational test examples, the iterations in (7) were continued until the trajectory changed by less than $\varepsilon = 10^{-7}$. Once an approximation of the displacement $\boldsymbol{\delta}_i$ is achieved in (7), the characteristic curves are obtained for each control volume from (6) as

$$\mathcal{X}_i(t_n) = \mathbf{x}_i - \boldsymbol{\delta}_i. \quad (9)$$

The departure points $\mathcal{X}_i(t_n)$ do not coincide with the spatial position of a gridpoint in general. In order to find the host control volume which such point belongs, we apply the search-locate algorithm proposed in [1] for triangular elements in unstructured grids.

2.2 Interpolation procedures

Once the characteristics feet $\mathcal{X}_i(t_n)$ are calculated in (9), the solution of (3) in the control volume \mathcal{T}_i and at instant t_{n+1} is defined as

$$c_i^{n+1} = c(t_n, \mathcal{X}_i(t_n)). \quad (10)$$

Note that, since the departure point $\mathcal{X}_i(t_n)$ would not lie on a mesh point, the solution $c(t_n, \mathcal{X}_i(t_n))$ at the characteristic feet is obtained by interpolation from known values at the control volume $\widehat{\mathcal{T}}_i$ where $\mathcal{X}_i(t)$ belongs and its neighbors. In the present study, the following interpolation procedures are considered.

Inverse Distance Weighted (IDW) scheme. This technique is a simple method widely used to interpolate a field with a known scattered set of points, see for example [2, 19]. Thus, the solution (10) is obtained using the IDW interpolation as

$$c(t_n, \mathcal{X}_i(t_n)) = \sum_{j=1}^M \frac{\omega_{ij}}{\omega} c_j^n, \quad (11)$$

where the weight ω_{ij} are calculated using the distances between the departure point and the cell centers of the host control volume $\widehat{\mathcal{T}}_i$ and its neighbors as shown in Figure 2. In (11), c_j^n are known solutions at time t_n on the control volumes \mathcal{T}_j . Hence,

$$\omega_{ij} = \frac{1}{d_{ij}^2}, \quad \omega = \sum_{j=1}^M \omega_j,$$

where $d_{ij} = \|\boldsymbol{\mathcal{X}}_i(t_n) - \mathbf{x}_j\|$ is the Euclidean distance between the departure point $\boldsymbol{\mathcal{X}}_i(t_n)$ and the cell center \mathbf{x}_j of the control volume \mathcal{T}_j given by

$$\|\boldsymbol{\mathcal{X}}_i(t_n) - \mathbf{x}_j\| = \sqrt{(X_i(t_n) - x_j)^2 + (Y_i(t_n) - y_j)^2}. \quad (12)$$

In the present work, the total number M is either 2, 3 or 4 depending on how many neighbors the host control volume $\widehat{\mathcal{T}}_i$ may have.

Least Squares (LS) scheme. This method is known as one of the most accurate approaches to solve a problem of finding the best polynomial approximation to the input data, compare for example [24, 3]. Let \mathcal{I}_i^n be a given set of indices of control volumes surrounding the host control volume $\widehat{\mathcal{T}}_i$ where $\boldsymbol{\mathcal{X}}_i(t_n)$ resides at time $t = t_n$, and \mathbf{c}^n the vector of solutions c_j^n at points \mathbf{x}_j with $j \in \mathcal{I}_i^n$. Using the LS interpolation, the solution (10) is evaluated as

$$c(t_n, \boldsymbol{\mathcal{X}}_i(t_n)) = \sum_{k=0}^M C_k^n \Psi_k(\boldsymbol{\mathcal{X}}_i(t_n)), \quad (13)$$

where C_k^n are the fitting coefficients, $\psi_k(x, y)$ are the polynomial basis functions and M is the total number of fitting data assumed to be less than the dimension of the set \mathcal{I}_i^n . To solve this problem, the least squares method uses a linear regression to compute C_k^n based on the merit function F defined as

$$F^2 = \sum_{j \in \mathcal{I}_i^n} \left(c_j^n - \sum_{k=0}^M C_k^n \psi_k(\mathbf{x}_j) \right)^2. \quad (14)$$

The LS interpolation estimates the vector $\mathbf{C}^n = (C_0^n, C_1^n, \dots, C_M^n)^\top$ as the best fit to a given data set, if \mathbf{C}^n minimizes the functional (14). Thus, the fitting coefficients C_k^n can be obtained by solving the $M + 1$ normal equations

$$\frac{\partial F^2}{\partial C_k^n} = 0, \quad k = 0, 1, \dots, M.$$

Hence, using the definition (14), the normal equations are given by

$$\sum_{j \in \mathcal{I}_i^n} \left(c_j^n - \sum_{l=0}^M C_l^n \psi_l(\mathbf{x}_j) \right) \psi_k(\mathbf{x}_j) = 0, \quad k = 0, 1, \dots, M, \quad (15)$$

which can be reformulated for each departure point $\boldsymbol{\mathcal{X}}_i(t_n)$ as a linear system of the form

$$\mathbf{D}\mathbf{C}^n = \mathbf{b}, \quad (16)$$

where $\mathbf{D} = \mathbf{A}^\top \mathbf{A}$ with \mathbf{A} is the matrix with entries $\psi_l(\mathbf{x}_j)$, $1 \leq l \leq M$, $j \in \mathcal{I}_i^n$, and the right-hand side vector $\mathbf{b} = \mathbf{A}^\top \mathbf{f}$ with \mathbf{f} is the vector with entries c_j^n , $j \in \mathcal{I}_i^n$. For instance, a linear approximation of the function $c^n(x, y)$ is defined as

$$c^n(x, y) = C_0^n + C_1^n (x - X_i) + C_2^n (y - Y_i). \quad (17)$$

To calculate the coefficients C_0^n , C_1^n and C_2^n , we first evaluate (17) at the departure point $\boldsymbol{\mathcal{X}}_i(t_n) = (X_i(t_n), Y_i(t_n))^\top$ to obtain C_0^n , then we solve the linear system (16) to obtain C_1^n and C_2^n . In this case, the inverse matrix \mathbf{D}^{-1} is given by

$$\mathbf{D}^{-1} = \frac{1}{\Delta_i} \begin{pmatrix} \sum_{j \in \mathcal{I}_i^n} (y_j - Y_i)^2 & - \sum_{j \in \mathcal{I}_i^n} (x_j - X_i)(y_j - Y_i) \\ - \sum_{j \in \mathcal{I}_i^n} (x_j - X_i)(y_j - Y_i) & \sum_{j \in \mathcal{I}_i^n} (x_j - X_i)^2 \end{pmatrix},$$

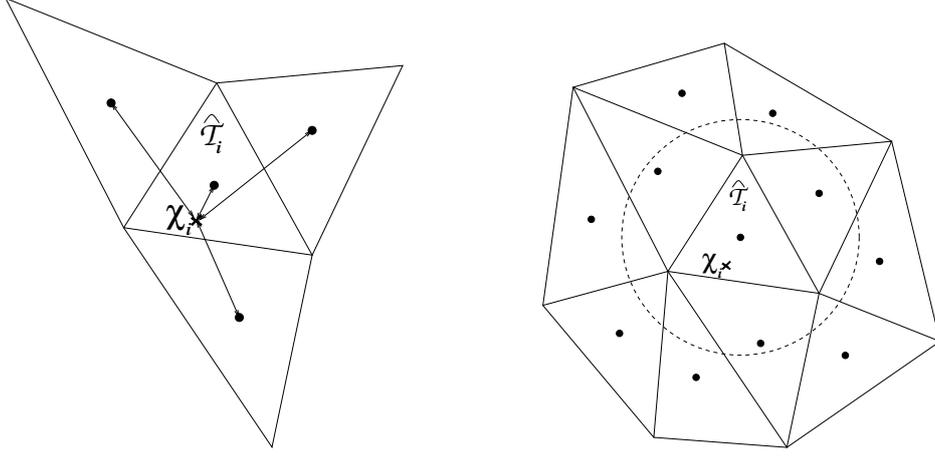


Figure 2: The control volume $\widehat{\mathcal{T}}_i$ where the departure point $\mathbf{x}_i(t_n)$ belongs and its neighboring elements used for IDW interpolation (left plot) and the set \mathcal{I}_i^n of points inside the dashed circle used for LS and TPS interpolations (right plot).

where

$$\Delta_i = \left(\sum_{j \in \mathcal{I}_i^n} (x_j - X_i)^2 \right) \left(\sum_{j \in \mathcal{I}_i^n} (y_j - Y_i)^2 \right) - \left(\sum_{j \in \mathcal{I}_i^n} (x_j - X_i)(y_j - Y_i) \right)^2.$$

Note that the formulation of a quadratic approximation in (17) can be obtained following similar steps as in the linear case. In our simulations, the set \mathcal{I}_i^n is formed by the neighboring cells that have a common edge or vertex with the control volume $\widehat{\mathcal{T}}_i$, see Figure 2 for an illustration.

Thin-Plate Spline (TPS) scheme. Interpolation with so-called thin-plate splines (also known as surface splines, D^m -splines or poly-harmonic splines) is a classical topic in spline theory, see for instance [25, 15]. Using the set \mathcal{I}_i^n with dimension N_i introduced above and shown in Figure 2, the solution (10) is calculated using the TPS interpolation as

$$c(t_n, \mathbf{x}_i(t_n)) = \sum_{j \in \mathcal{I}_i^n} \lambda_j \|\mathbf{x}_i(t_n) - \mathbf{x}_j\| \log(\|\mathbf{x}_i(t_n) - \mathbf{x}_j\|) + \alpha_0 + \alpha_1 x_i + \alpha_2 y_i, \quad (18)$$

where the coefficients $(\lambda_j)_{j \in \mathcal{I}_i^n}$, α_0 , α_1 and α_2 are obtained by solving the linear system

$$\begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1N_i} & 1 & x_1 & y_1 \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2N_i} & 1 & x_2 & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \phi_{N_i 1} & \phi_{N_i 2} & \cdots & \phi_{N_i N_i} & 1 & x_{N_i} & y_{N_i} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_{N_i} \\ \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{N_i} \end{pmatrix}, \quad (19)$$

where ϕ_{ij} are the radial basis functions defined as

$$\phi_{ij} = \|\mathbf{x}_i(t_n) - \mathbf{x}_j\| \log(\|\mathbf{x}_i(t_n) - \mathbf{x}_j\|).$$

By introducing the following matrices

$$\boldsymbol{\phi} = \begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1N_i} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2N_i} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N_i1} & \phi_{N_i2} & \cdots & \phi_{N_iN_i} \end{pmatrix}, \quad \mathbf{P} = \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ \vdots & \vdots & \vdots \\ 1 & x_{N_i} & y_{N_i} \end{pmatrix},$$

and the following vectors

$$\boldsymbol{\lambda} = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_{N_i} \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{N_i} \end{pmatrix}, \quad \boldsymbol{\alpha} = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{pmatrix},$$

the linear system (19) can be formulated in a compact form as

$$\begin{pmatrix} \boldsymbol{\phi} & \mathbf{P} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{pmatrix} = \mathbf{C}. \quad (20)$$

Note that the system (20) is formed with N_i equations for $N_i + 3$ unknowns. Therefore, a non-singular system can be constructed by adding the constraints

$$\sum_{j=1}^{N_i} c_j = 0, \quad \sum_{j=1}^{N_i} c_j x_j = 0, \quad \sum_{j=1}^{N_i} c_j y_j = 0.$$

Hence, the coefficients $(\lambda_j)_{j \in \mathcal{I}_i^n}$, α_0 , α_1 and α_2 in (18) are determined by solving the block linear system

$$\begin{pmatrix} \boldsymbol{\phi} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{pmatrix} = \begin{pmatrix} \mathbf{C} \\ \mathbf{0} \end{pmatrix}. \quad (21)$$

Note that other radial basis functions can also be used in (18) without major conceptual modifications.

2.3 Advection adjusted algorithm

For the convection problem (3), the divergence-free condition (2) implies that

$$\int_{\Omega} \mathbf{v} \cdot \nabla c \, d\mathbf{x} = \oint_{\Gamma} c \mathbf{v} \cdot \mathbf{n} \, ds, \quad (22)$$

where \mathbf{n} is the unit outward normal to the boundary Γ . Assuming that at any time t , the amount of mass entering and exiting the domain is the same, then the net flux through the boundary Γ vanishes and the right-term in (22) is zero. Consequently, integrating (3) over the domain Ω leads to the conservation property

$$\int_{\Omega} c(t, \mathbf{x}) \, d\mathbf{x} = \int_{\Omega} c_0(\mathbf{x}) \, d\mathbf{x}, \quad \forall t \in [0, T], \quad (23)$$

In general, a semi-Lagrangian method does not preserve the condition (23) and further reconstructions are needed to convert the method to mass conservative. In the present study, we propose an algorithm for the finite volume discretization in unstructured meshes based on the adjusted advection techniques. Thus, given the solution $c(t_n, \mathbf{x})$ at current time t_n , the solution at the next time level $c(t_{n+1}, \mathbf{x})$ is obtained as follows:

For each control volume \mathcal{T}_i in the computational mesh, compute the departure point $\mathbf{x}_i(t_n)$ using the iterative procedure (5)-(7), identify the control volume $\widehat{\mathcal{T}}_i$ where such a point is located, and compute the approximation $\widehat{c}(t_{n+1}, \mathbf{x}_i) = c(t_n, \mathbf{x}_i(t_n))$ employing an interpolation procedure from section 2.2. Evaluate the conservation integrals $C(t_n)$ and $\widehat{C}(t_{n+1})$ as

$$C^n = \int_{\Omega} c(t_n, \mathbf{x}) d\mathbf{x}, \quad \widehat{C}^{n+1} = \int_{\Omega} \widehat{c}(t_{n+1}, \mathbf{x}) d\mathbf{x}.$$

Perturb the characteristic curves according to

$$\mathbf{x}_i^+(t_n) = \mathbf{x}_i(t_n) + h\boldsymbol{\delta}_i \quad \text{and} \quad \mathbf{x}_i^-(t_n) = \mathbf{x}_i(t_n) - h\boldsymbol{\delta}_i,$$

where $\boldsymbol{\delta}_i$ is the displacement obtained from the iterations (7) and h denotes the radius of the circle circumscribed by the host triangle $\widehat{\mathcal{T}}_i$. Then, define the solution $\widetilde{c}(t_{n+1}, \mathbf{x}_i)$ by

$$\widetilde{c}(t_{n+1}, \mathbf{x}_i) = \begin{cases} \max\left(c(t_n, \mathbf{x}_i^+(t_n)), c(t_n, \mathbf{x}_i^-(t_n))\right), & \text{if } C^n > \widehat{C}^{n+1}, \\ \min\left(c(t_n, \mathbf{x}_i^+(t_n)), c(t_n, \mathbf{x}_i^-(t_n))\right), & \text{if } C^n \leq \widehat{C}^{n+1}, \end{cases}$$

and calculate the corresponding conservation integral

$$\widetilde{C}^{n+1} = \int_{\Omega} \widetilde{c}(t_{n+1}, \mathbf{x}) d\mathbf{x}.$$

If $C^n \neq \widehat{C}^{n+1}$, we compute the limiter function $\theta(t_n)$ such that

$$\theta\widehat{C}^{n+1} + (1 - \theta)\widetilde{C}^{n+1} = C^n.$$

Update the new solution $c(t_{n+1}, \mathbf{x})$ using a limiter procedure as

$$c(t_{n+1}, \mathbf{x}) = \begin{cases} \theta\widehat{c}(t_n, \mathbf{x}) + (1 - \theta)\widetilde{c}(t_{n+1}, \mathbf{x}), & \text{if } \widetilde{C}^{n+1} \neq \widehat{C}^{n+1}, \\ \widehat{c}(t_n, \mathbf{x}), & \text{if } \widetilde{C}^{n+1} = \widehat{C}^{n+1}. \end{cases} \quad (24)$$

Note that it is easy to verify that

$$C^{n+1} = \int_{\Omega} c(t_{n+1}, \mathbf{x}) d\mathbf{x} = \theta\widehat{C}^{n+1} + (1 - \theta)\widetilde{C}^{n+1} = C^n,$$

and therefore, the proposed advection adjusted solution (24) conserves the mass at each time step. Similar approaches have been investigated in [9, 11] for semi-Lagrangian methods on structured meshes. It should be stressed that this advection adjusted method which requires additional computational work can be easily implemented in an existing conventional semi-Lagrangian code.

3 Implementation for convection-diffusion problems

We consider the convection-diffusion problem (1) reformulated using the total derivative as

$$\frac{Dc}{Dt} - \nabla \cdot (\nu \nabla c) = f(t, \mathbf{x}). \quad (25)$$

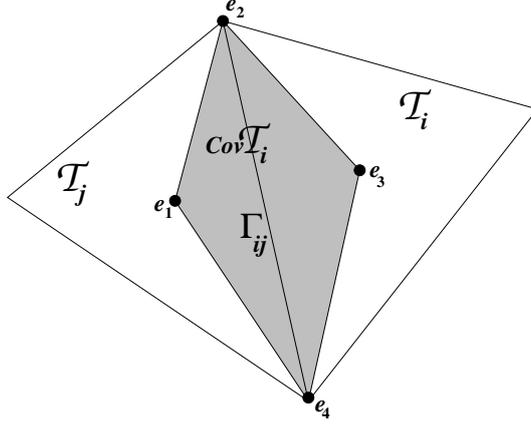


Figure 3: An illustration of the co-volume used in the discretization of diffusion terms.

Hence, using the control volume depicted in Figure 3, a finite volume discretization of (25) yields

$$\frac{Dc_i}{Dt} = \frac{1}{|\mathcal{T}_i|} \sum_{j \in N(i)} \oint_{\Gamma_{ij}} \nu \nabla c \cdot \mathbf{n} \, d\sigma + \frac{1}{|\mathcal{T}_i|} \int_{\mathcal{T}_i} \mathbf{f}(t, \mathbf{x}) \, d\mathbf{x}, \quad (26)$$

where $N(i)$ is the set of neighboring triangles of the cell \mathcal{T}_i , $\mathbf{n} = (n_x, n_y)^T$ denotes the unit outward normal to the surface of control volume \mathcal{T}_i , and $|\mathcal{T}_i|$ denotes the area of \mathcal{T}_i . For a scalar diffusion coefficient,

$$\nu \nabla c \cdot \mathbf{n} = \nu \frac{\partial c}{\partial x} n_x + \nu \frac{\partial c}{\partial y} n_y.$$

To discretize the diffusion fluxes in (26) we consider a Green-Gauss diamond reconstruction as discussed in [4] among others. This method is second-order accurate and it can be applied on general unstructured grids with large deformation since there is no serious restrictions on the angles of control volumes. Hence, a co-volume, $cov\mathcal{T}$, is first constructed by connecting the barycenters of the control volumes that share the edge Γ_{ij} and its endpoints as shown in Figure 3. Then, in the x -direction, the diffusion fluxes are evaluated at an inner edge Γ_{ij} as

$$\oint_{\Gamma_{ij}} \nu \frac{\partial c}{\partial x} n_x \, d\sigma = \frac{\bar{\nu}}{|cov\mathcal{T}|} \sum_{\epsilon \in \partial T} \frac{c_{e_1} + c_{e_3}}{2} \int_{\epsilon} n_{x\epsilon} \, d\sigma, \quad (27)$$

where e_1 and e_3 are the nodes of the edge ϵ on the surface $\partial cov\mathcal{T}$, c_{e_1} and c_{e_3} are the values of the concentration c at node e_1 and e_3 , respectively. Since this method is cell centered, all unknowns should be expressed in term of control volumes centroids. For this reason, the solutions at endpoints of Γ_{ij} are approximated by interpolation from the values on the cells sharing the same vertex e_k . Notice that for a non-constant diffusion, the diffusion coefficient $\bar{\nu}$ in (27) is defined by

$$\bar{\nu} = \frac{\nu_{e_1} + \nu_{e_2} + \nu_{e_3} + \nu_{e_4}}{4},$$

with ν_{e_k} , $k = 1, \dots, 4$, are values of the diffusion coefficient ν at the co-volume nodes e_k . The semi-discrete equations (26) can be reformulated in a compact form as a system of ordinary differential equations

$$\frac{D\mathbf{c}}{Dt} = [\mathbf{S}] \mathbf{c} + \mathbf{f}(t), \quad (28)$$

where \mathbf{c} is solution vectors with entries c_i , $[\mathbf{S}]$ is the stiffness matrix the entries of which are defined above, and $\mathbf{f}(t)$ the force vector with entries

$$f_i(t) = \frac{1}{|\mathcal{T}_i|} \int_{\mathcal{T}_i} f(t, \mathbf{x}) \, d\mathbf{x}.$$

For the time discretization, we integrate the system (28) along the characteristics using a second-order implicit time stepping scheme as

$$\frac{\mathbf{c}^{n+1} - \hat{\mathbf{c}}^n}{\Delta t} = \frac{1}{2} [\mathbf{S}] \mathbf{c}^{n+1} + \frac{1}{2} [\mathbf{S}] \hat{\mathbf{c}}^n + \frac{1}{2} \mathbf{f}^{n+1} + \frac{1}{2} \mathbf{f}^n, \quad (29)$$

where $\hat{\mathbf{c}}^n$ is the solution vector with entries $c(t_n, \mathcal{X}_i(t_n))$ obtained in (24) using the adjusted advection algorithm. It is worth remarking that to update the solution \mathbf{c}^{n+1} in (29), a linear system of algebraic equation has to be solved each time step. To perform this step in our semi-Lagrangian finite volume method, we use the conjugate gradient solver with incomplete Cholesky decomposition. In addition, all stopping criteria for iterative solvers were set to 10^{-7} , which is small enough to guarantee that the algorithm truncation error dominates the total numerical error.

4 Numerical results and examples

A number of numerical examples are selected to illustrate the accuracy of the new semi-Lagrangian finite volume method introduced in the above sections. For some of these test examples the analytical solution is known, so that we can evaluate the error function \mathbf{e} at time t_n as

$$e_i^n = c_i^n - c_{\text{exact}}(t_n, \mathbf{x}_i), \quad (30)$$

where $c_{\text{exact}}(t_n, \mathbf{x}_i)$ and c_i^n are the exact and numerical solutions, respectively, at time t_n in the control volume with center \mathbf{x}_i . We also define the CFL number associated with equation (1) as follows

$$\text{CFL}_x = \max_{x,y} |u| \frac{\Delta t}{h}, \quad \text{CFL}_y = \max_{x,y} |v| \frac{\Delta t}{h}, \quad \text{CFL} = \sqrt{\text{CFL}_x^2 + \text{CFL}_y^2}. \quad (31)$$

It should be stressed that, in all our computations the resulting linear systems of algebraic equations were solved using the preconditioned conjugate gradient and a tolerance of $10^{-7}h$ to stop the iterations.

4.1 Advection-diffusion of a Gaussian pulse

To ascertain the performance of the semi-Lagrangian finite volume method we consider the advection-diffusion of a rotating Gaussian pulse investigated in [11, 12] among others. The governing equation is of the form (1) with $\mathbf{v} = (-\omega y, \omega x)^T$ and $\omega = 4$ and $f = 0$. Initial and boundary conditions are taken from the analytical solution

$$c_{\text{exact}}(t, x, y) = \frac{100}{1 + \frac{2\nu t}{\sigma^2}} \exp\left(-\frac{(\bar{x} - x_0)^2 + (\bar{y} - y_0)^2}{2(\sigma^2 + 2\nu t)}\right),$$

where $\bar{x} = x \cos(\omega t) + y \sin(\omega t)$, $\bar{y} = -x \sin(\omega t) + y \cos(\omega t)$, $x_0 = -0.25$, $y_0 = 0$ and $\sigma^2 = 0.002$. The computational domain is $\Omega = [-0.5, 0.5] \times [-0.5, 0.5]$ covered by different uniform meshes and the time period required for one complete rotation is $\frac{\pi}{2}$. From (31), the CFL number associated to this test example is $\omega \frac{\sqrt{2}}{2} \frac{\Delta t}{h}$ and it is set to $3\pi\sqrt{2}$ in our simulations.

In the first run of this example we consider the problem of pure advection corresponding to $\nu = 0$. Table 1 presents a quantitative comparison of the results obtained using different interpolation procedures on different structured meshes after 1, 2 and 5 revolutions. We report the L^1 -errors, relative mass (Mass), maximum (max) values of the computed solutions, and CPU times given in seconds. We also present a comparison between conventional approach (without using adjusted advection procedure) and the conservative approach proposed in this study. For the considered meshes and numbers of revolutions, the conservative approach preserves the mass conservation at an additional cost referring to the CPU times. Failure of mass conservation is clear in the results obtained using the conventional approach whereas, the relative mass remains fixed to unity in the conservative approach. The numerical diffusion

Table 1: Results for advection of the Gaussian pulse test after 1, 2 and 5 revolutions using different meshes and interpolation procedures. The analytical maximum is 1 and the CPU times are given in seconds.

Results for the IDW interpolation									
# Rev	Mesh	Conventional approach				Conservative approach			
		max c	L^1 -Error	Mass	CPU	max c	L^1 -Error	Mass	CPU
1	32 × 32	0.6930	1.048E-02	1.0327	0.04	0.6851	1.035E-02	1.0000	0.05
	64 × 64	0.8073	6.949E-03	0.9979	0.22	0.8077	6.965E-03	1.0000	0.26
	128 × 128	0.8912	5.606E-03	1.0222	1.42	0.8876	5.522E-03	1.0000	1.69
2	32 × 32	0.5416	1.755E-02	1.0456	0.07	0.5282	1.723E-02	1.0000	0.09
	64 × 64	0.6786	1.223E-02	0.9960	0.35	0.6790	1.228E-02	1.0000	0.42
	128 × 128	0.8065	1.062E-02	1.0414	2.45	0.7986	1.039E-02	1.0000	2.89
5	32 × 32	0.3188	3.090E-02	1.0351	0.16	0.3068	3.052E-02	1.0000	0.19
	64 × 64	0.4525	2.423E-02	0.9866	0.62	0.4547	2.446E-02	1.0000	0.72
	128 × 128	0.6319	2.343E-02	1.0859	5.44	0.6135	2.258E-02	1.0000	6.25

Results for the LS interpolation									
# Rev	Mesh	Conventional approach				Conservative approach			
		max c	L^1 -Error	Mass	CPU	max c	L^1 -Error	Mass	CPU
1	32 × 32	0.8163	4.71740E-03	0.9895	0.12	0.8178	4.81230E-03	1.0000	0.15
	64 × 64	0.8923	2.45723E-03	1.0000	2.20	0.8925	2.48778E-03	1.0000	2.75
	128 × 128	0.9466	1.23609E-03	0.9994	66	0.9464	1.26036E-03	1.0000	81
2	32 × 32	0.6830	8.54007E-03	0.9844	0.22	0.6870	8.76210E-03	1.0000	0.27
	64 × 64	0.8135	4.66360E-03	1.0001	4.38	0.8138	4.75328E-03	1.0000	5.41
	128 × 128	0.8999	2.43363E-03	0.9979	132	0.8990	2.50876E-03	1.0000	160
5	32 × 32	0.4682	1.68701E-02	0.9698	0.64	0.4708	1.76955E-02	1.0000	0.78
	64 × 64	0.6431	1.03115E-02	1.0030	10.85	0.6390	1.10730E-02	1.0000	12.91
	128 × 128	0.7838	1.43739E-03	0.7164	330	0.8001	1.85016E-03	1.0000	402

Results for the TPS interpolation									
# Rev	Mesh	Conventional approach				Conservative approach			
		max c	L^1 -Error	Mass	CPU	max c	L^1 -Error	Mass	CPU
1	32 × 32	0.9874	1.848E-04	0.9982	0.30	0.9711	1.119E-03	1.0000	0.37
	64 × 64	0.9887	7.246E-05	0.9997	3.29	0.9929	5.298E-04	1.0000	3.91
	128 × 128	0.9993	4.062E-05	1.0002	73	0.9975	2.951E-04	1.0000	91
2	32 × 32	0.9822	3.854E-04	0.9944	0.62	0.9626	2.168E-03	1.0000	0.77
	64 × 64	0.9909	1.377E-04	0.9997	6.58	0.9865	1.094E-03	1.0000	7.91
	128 × 128	0.9999	7.796E-05	0.9996	145	0.9947	6.148E-04	1.0000	185
5	32 × 32	0.9759	3.854E-04	0.9980	1.38	0.9104	7.844E-04	1.0000	1.89
	64 × 64	0.9810	3.311E-04	0.9994	17	0.9655	3.310E-04	1.0000	19.85
	128 × 128	0.9996	1.965E-04	0.9985	365	0.9856	1.965E-04	1.0000	454

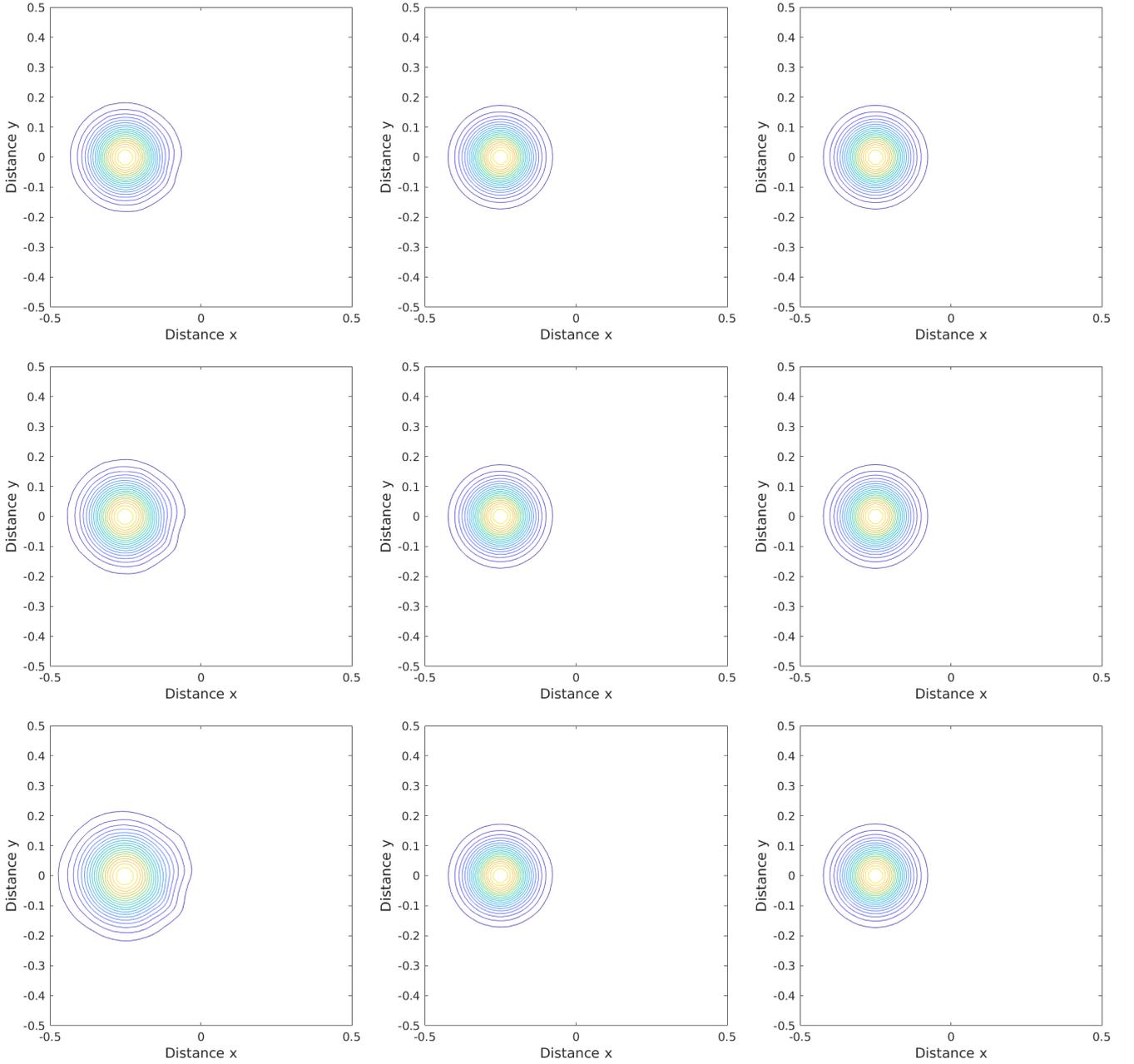


Figure 4: Numerical solutions obtained using LS (first column), TPS (second column) and exact solution (third column) for $\nu = 0$ on a mesh with 64×64 control volumes after one rotation (first row) and 2 rotations (second row) and 5 rotations (third row).

is also more visible in the results obtained using the conventional approach than those obtained using the conservative approach, compare the maximum values of the computed solutions in Table 1. Note that in all considered cases the CPU time in the conservative approach is about 1.25 times larger than that in the conventional approach. In terms of accuracy, Table 1 demonstrates that the TPS interpolation is the more accurate than the IDW and LS interpolation procedures. In addition, the IDW procedure requires less CPU times than the other procedures but the results obtained using the IDW procedure are the worst. It is to be remarked that, the semi-Lagrangian finite volume method is typically built to solve this class of convection-dominated problems using CFL numbers four to five times larger than its Eulerian counterparts.

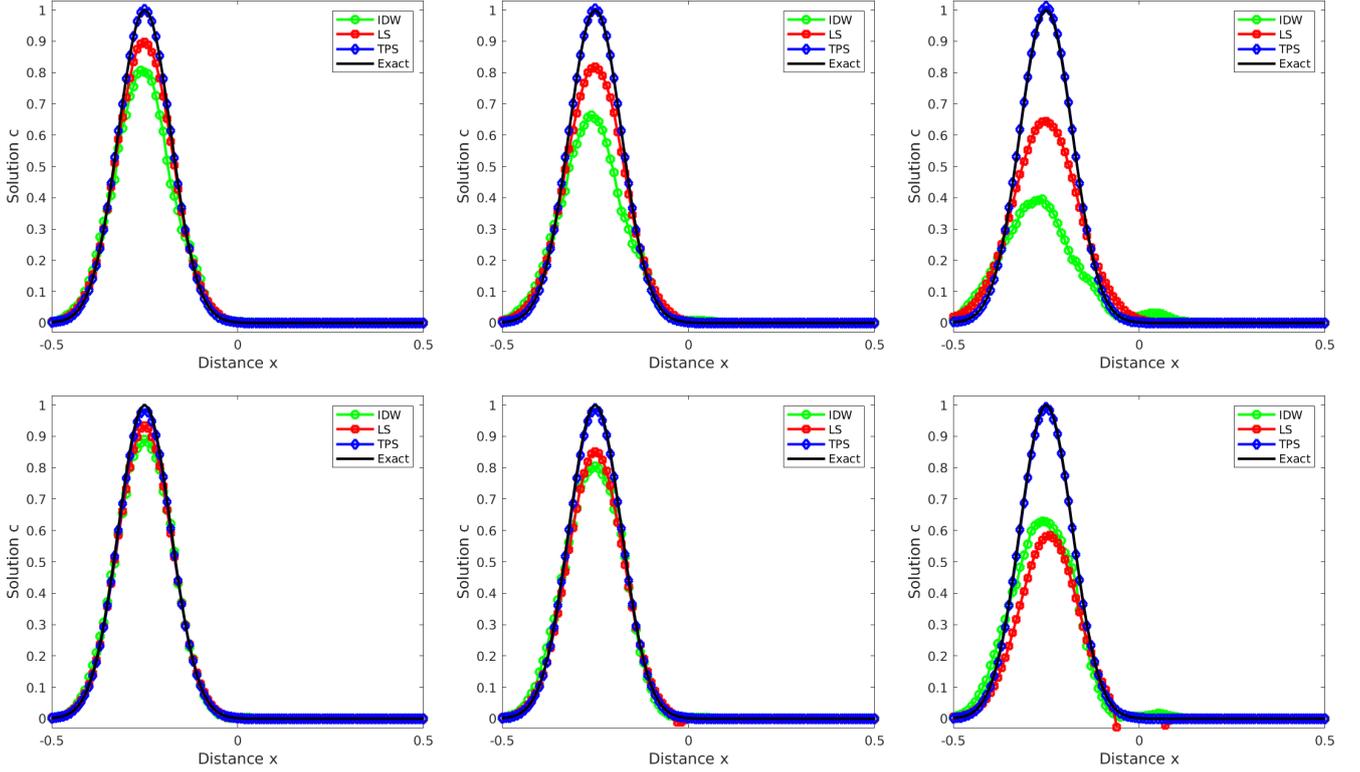


Figure 5: Cross-sections of the results in Figure 4 at $y = 0$ after one revolution (first column), 2 revolutions (second column) and 5 revolutions (third column) using a mesh with 64×64 control volumes (first row) and 128×128 control volumes (second row).

In Figure 4 we display 20 equi-distributed contourlines of the solutions obtained by the conservative approach using LS and TPS interpolation procedures after 1, 2 and 5 revolutions on the mesh with 64×64 control volumes. For comparison, we have also included the exact solutions in this figure. Note that the results obtained using the IDW interpolation are not included in this figure as they exhibit nonphysical oscillations and substantially greater distortion, specially at the feet of the Gaussian pulse where the gradients are sharper. The corresponding one-dimensional cross-sections at $y = 0$ are illustrated in Figure 5 using two unstructured meshes with 64×64 and 128×128 control volumes. It is evident that, after one revolution on the fine mesh of 128×128 control volumes, the considered interpolation procedures give roughly similar results with some small differences on the maximum value of the numerical solutions. However, by increasing the number of revolutions to 5, the results obtained using the TPS procedure are more accurate than those obtained using the IDW and LS procedures.

Next we include a physical diffusion in this problem by solving the advection-diffusion equations (1) with diffusion coefficients $\nu = 10^{-5}$, 10^{-4} and 10^{-3} . For these coefficients, a quantitative comparison of the results computed using conventional and conservative approaches not reported here for brevity, yields similar conclusions as those drawn from the results summarized in Table 1. Here, we illustrate only the one-dimensional cross-sections at $y = 0$ for a comparison reason. Figure 6 depicts the obtained results for the considered diffusion coefficients using two structured meshes with 64×64 and 128×128 control volumes. It is clear that numerical diffusion is very pronounced in the results computed using the IDW and LS interpolations compared to the TPS procedure. However, by increasing the physical diffusion or refining the mesh, the considered interpolation procedures produce the same results. Again the TPS interpolation performs best for this test example of linear advection-diffusion problems.

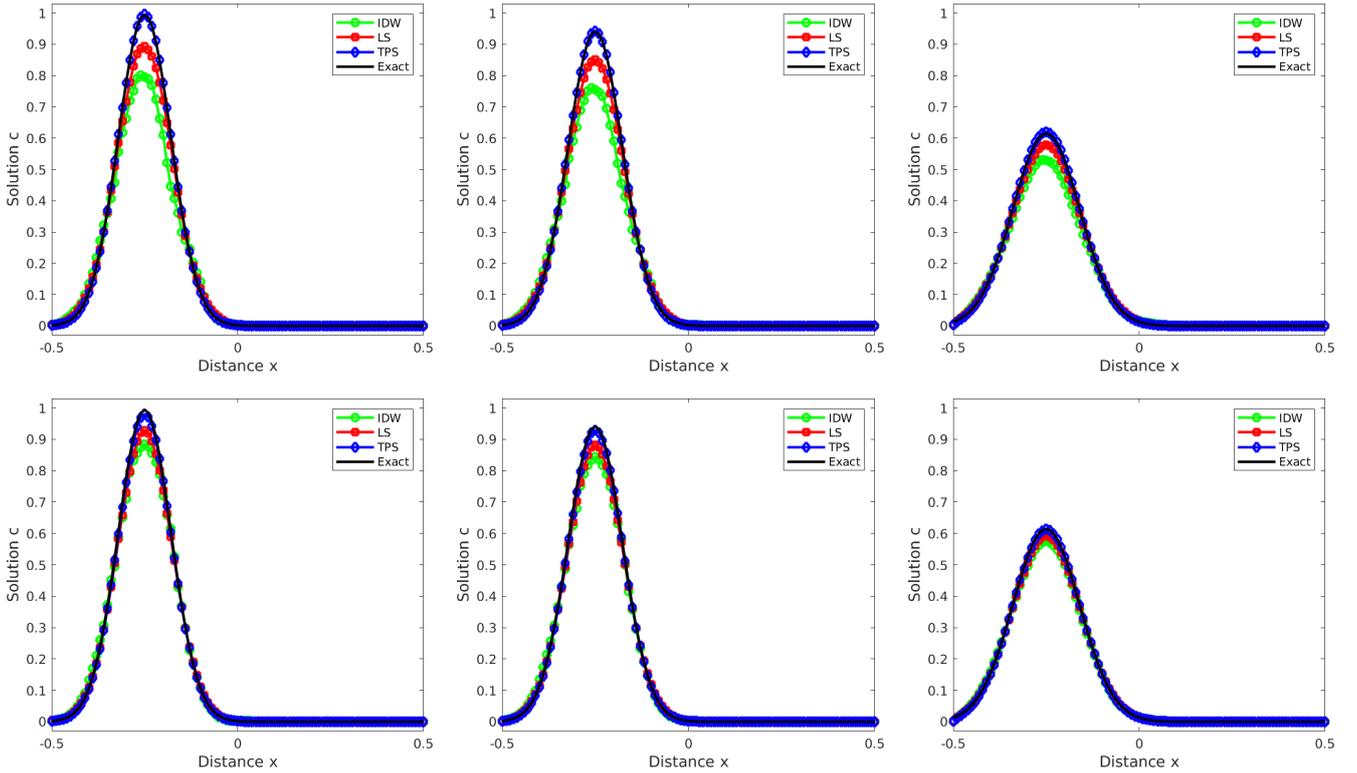


Figure 6: Cross-sections of the results for the advection-diffusion problem with $\nu = 10^{-5}$ (first column), $\nu = 10^{-4}$ (second column) and $\nu = 10^{-3}$ (third column) using a mesh with 64×64 control volumes (first row) and 128×128 control volumes (second row).

4.2 A deformational flow problem

In this example we consider the well-established problem of the deformational flow example widely been served as a prototype to examine the performance of advection schemes and semi-Lagrangian methods, see for example [14, 23]. Here, we solve the linear advection equation (1) in the spatial domain $\Omega = [-4, 4] \times [-4, 4]$ equipped with a highly deformational flow field and the initial condition for the scalar field

$$c(0, x, y) = -\tanh\left(\frac{y - y_0}{\eta}\right), \quad (32)$$

where η is the width of the front zone. The velocity field is a steady circular vortex with tangential velocity depending on the radius of the vortex as

$$v_t(r) = v_0 \operatorname{sech}^2(r) \tanh(r), \quad (33)$$

where v_0 is such that the maximum value of v_t never exceeds unity. The analytical solution of the considered problem is defined by

$$c_{\text{exact}}(t, x, y) = -\tanh\left(\frac{y - y_0}{\eta} \cos(\omega t) - \frac{x - x_0}{\eta} \sin(\omega t)\right), \quad (34)$$

with (x_0, y_0) is the center of the vortex and $\omega = \frac{v_t}{r}$ is its angular velocity. In our simulations, the vortex is centered at the origin of the computational domain, $v_0 = 2.58$, $\eta = 0.05$ and numerical results are presented at time $t = 4$. Note that the considered value of η in the initial data corresponds to a steep hyperbolic tangent profile that results in a highly deformational solution as the time progresses.

In Figure 7 we display snapshots of the computed solutions obtained using the LS and TPS procedures with $\text{CFL} = 3$ at time $t = 4$ on two unstructured meshes with 9908 and 33078 control volumes.

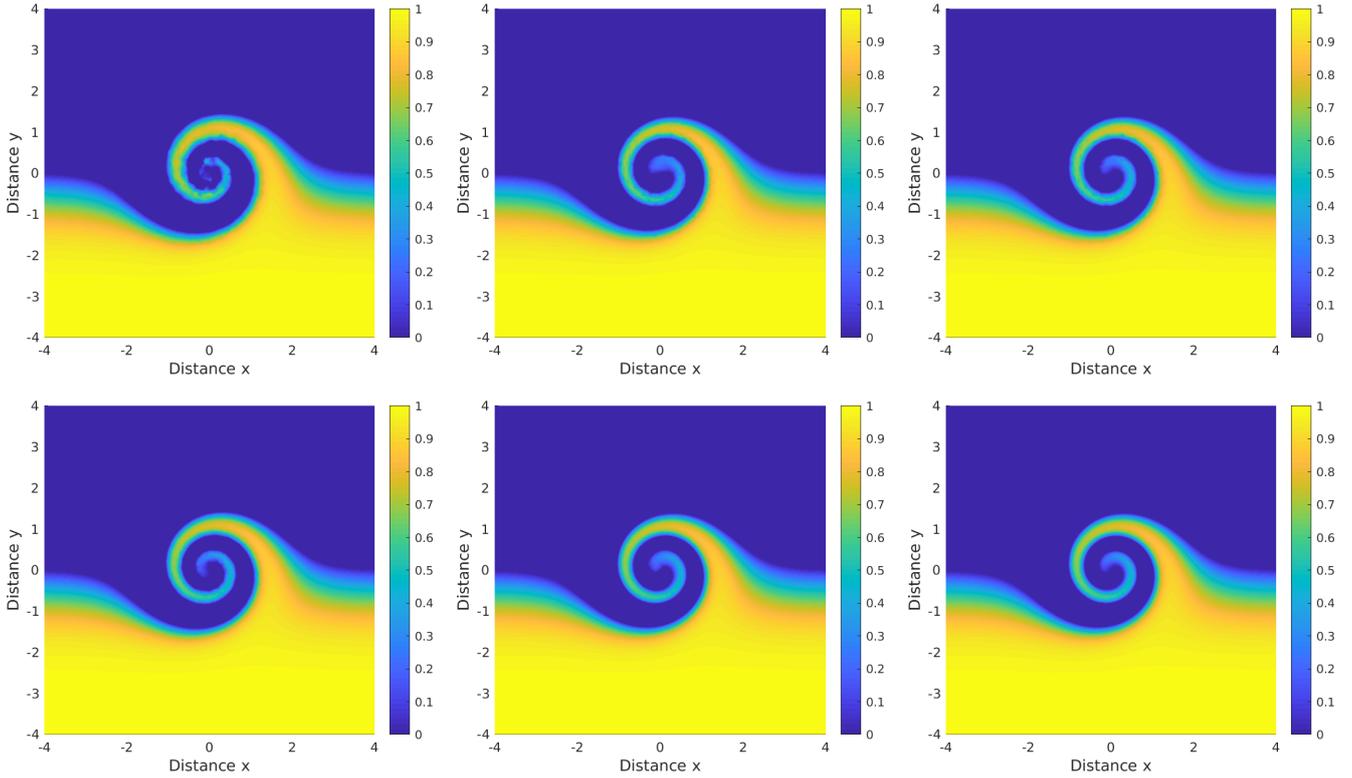


Figure 7: Results for the deformational flow example at time $t = 4$ using the LS procedure (first column), TPS procedure (second column) and exact solution (third column) on a mesh with 9908 control volumes (first row) and 33078 control volumes (second row).

For comparison reasons, we have also included the exact solution in Figure 7. Notice that the results obtained using the IDW procedure are not included in this figure because these results exhibit non-physical oscillations. As in the previous example, we display in Figure 8 the corresponding one-dimensional cross-sections at $x = 0$ and $y = 0$ of the computed results. As expected, the IDW and LS procedures exhibit substantially greater numerical dissipation, specially at the center of the spatial domain where the gradient is steep. From the same figures we observe an absence of this numerical diffusion in the results obtained using the TPS procedure. It is clear that the numerical results obtained by the LS procedure are more diffusive than those computed using the TPS procedure and the IDW procedure is the most diffusive. Compare the good agreement between the results obtained using the TPS procedure and the analytical solutions even when using the coarse mesh with 9908 control volumes.

To further quantify the accuracy of the proposed semi-Lagrangian finite volume method for this deformational flow problem, we compare in Table 2 the conventional approach (without using adjusted advection procedure) and the conservative approach using the TPS interpolation. Here, we present the L^1 -errors, relative mass (Mass), and CPU times given in seconds for different unstructured meshes and values of CFL. The clear indication from Table 2 is that the L^1 -errors decay as the number of control volumes increases for both conventional and conservative methods. Moreover, increasing the CFL number results in a decrease of the L^1 -errors. The TPS interpolation performs very satisfactorily for this highly deformational flow problem. An examination of the relative mass in Table 2 reveals that, the proposed semi-Lagrangian finite volume method is fully conservative on all considered meshes and used values of CFL in the simulations. In addition, the CPU time in Table 2 confirms that, on the coarse meshes, there is no noticeable differences between the computational cost required for both methods. In all the results presented in Table 2, the CPU time needed for the conservative approach is about two times more than that needed for the conventional approach. It should be pointed out that the performance of the

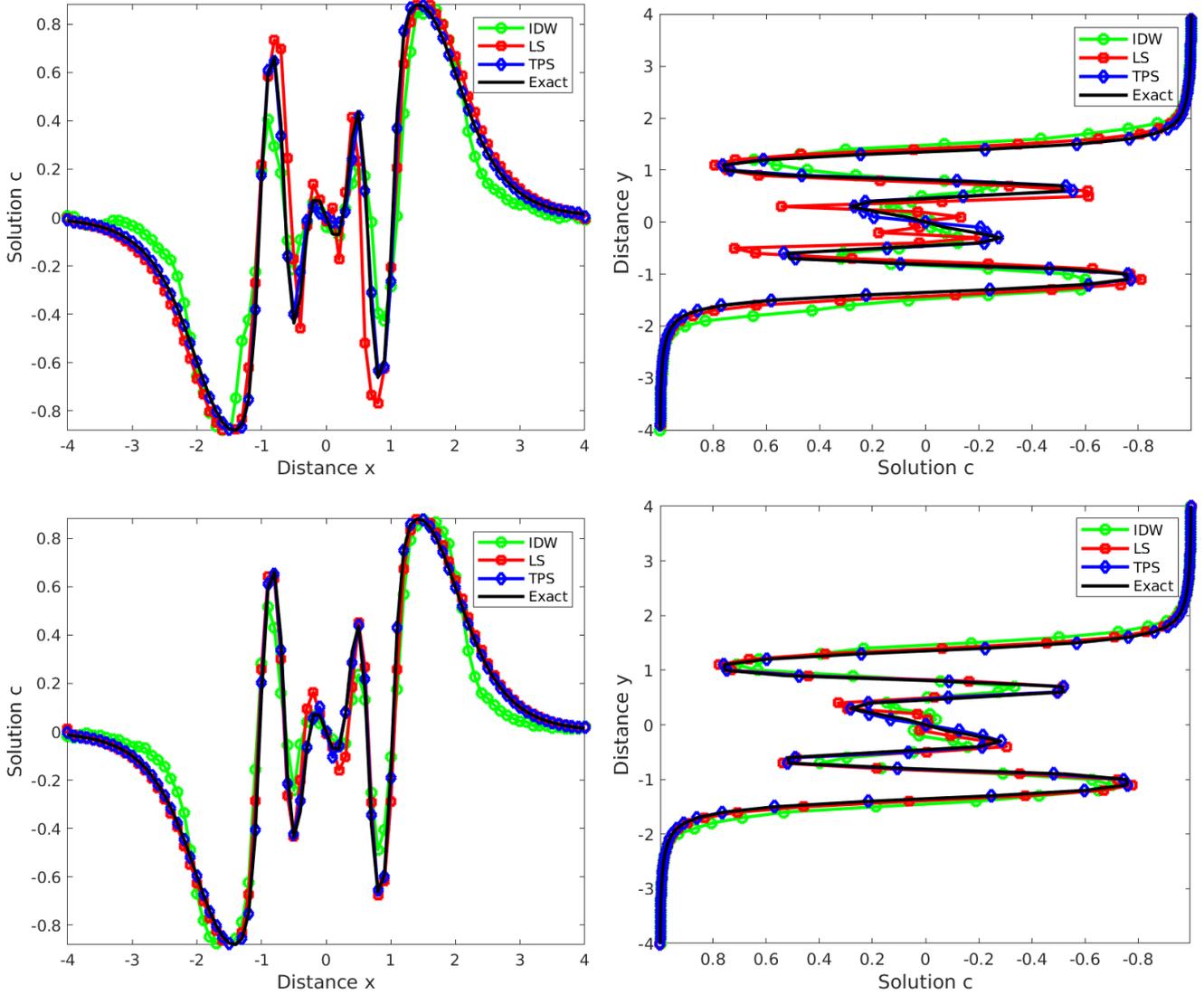


Figure 8: Cross-sections of the results in Figure 7 at $y = 0$ (first column) and at $x = 0$ (second column) using a mesh with 9908 control volumes (first row) and 33078 control volumes (second row).

proposed method is very attractive since the computed solutions remain stable and conservative even when coarse meshes are used without requiring Riemann solvers or complicated techniques to reconstruct the numerical fluxes.

4.3 Moving fronts problem

We consider the problem of moving fronts modeled by the equations (1) equipped with a velocity field varying in time. Initially, two separate fronts travel along the main diagonal of the computational domain at different speeds and eventually coalesce into one front for longer time. This problem has been previously solved in a squared domain in [22] using a moving finite element method and in [16] using a family of finite element alternating-direction methods combined with a modified method of characteristics. In the present study, we compute the solution of this problem in a circular domain centered at $(0.5, 0.5)$ and with radius 0.5. Thus, we solve the equations (1) with $f = 0$ and the velocity field given by

$$u(t, x, y) = \frac{-0.1e^{-A(t,x)} + 0.5e^{-B(t,x)} + e^{-C(t,x)}}{e^{-A(t,x)} + e^{-B(t,x)} + e^{-C(t,x)}}, \quad v(t, x, y) = \frac{-0.1e^{-A(t,y)} + 0.5e^{-B(t,y)} + e^{-C(t,y)}}{e^{-A(t,y)} + e^{-B(t,y)} + e^{-C(t,y)}},$$

Table 2: Error-norms and computational times for the deformational flow problem at $t = 4$ using different meshes and different CFL numbers. CPU time is listed in seconds.

CFL	# of control volumes	Conventional approach			Conservative approach		
		L^1 -Error	Mass	CPU	L^1 -Error	Mass	CPU
3	1541	2.32818E-02	0.9974	0.98	2.33210E-02	1.0000	1.21
	3068	1.74345E-02	1.0010	2.76	1.74259E-02	1.0000	3.33
	15452	7.69612E-03	1.0024	60	7.69413E-03	1.0000	75
6	1541	1.91841E-02	0.9988	0.52	1.92648E-02	1.0000	0.63
	3068	1.55190E-02	1.0013	1.55	1.54372E-02	1.0000	1.90
	15452	4.75112E-03	1.0020	28.38	4.74362E-03	1.0000	34.1
10	1541	1.63514E-02	0.9989	0.34	1.63169E-02	1.0000	0.41
	3068	8.60391E-03	0.9997	0.88	8.58088E-03	1.0000	0.97
	15452	2.84319E-03	1.0009	15.62	2.82657E-03	1.0000	18.95

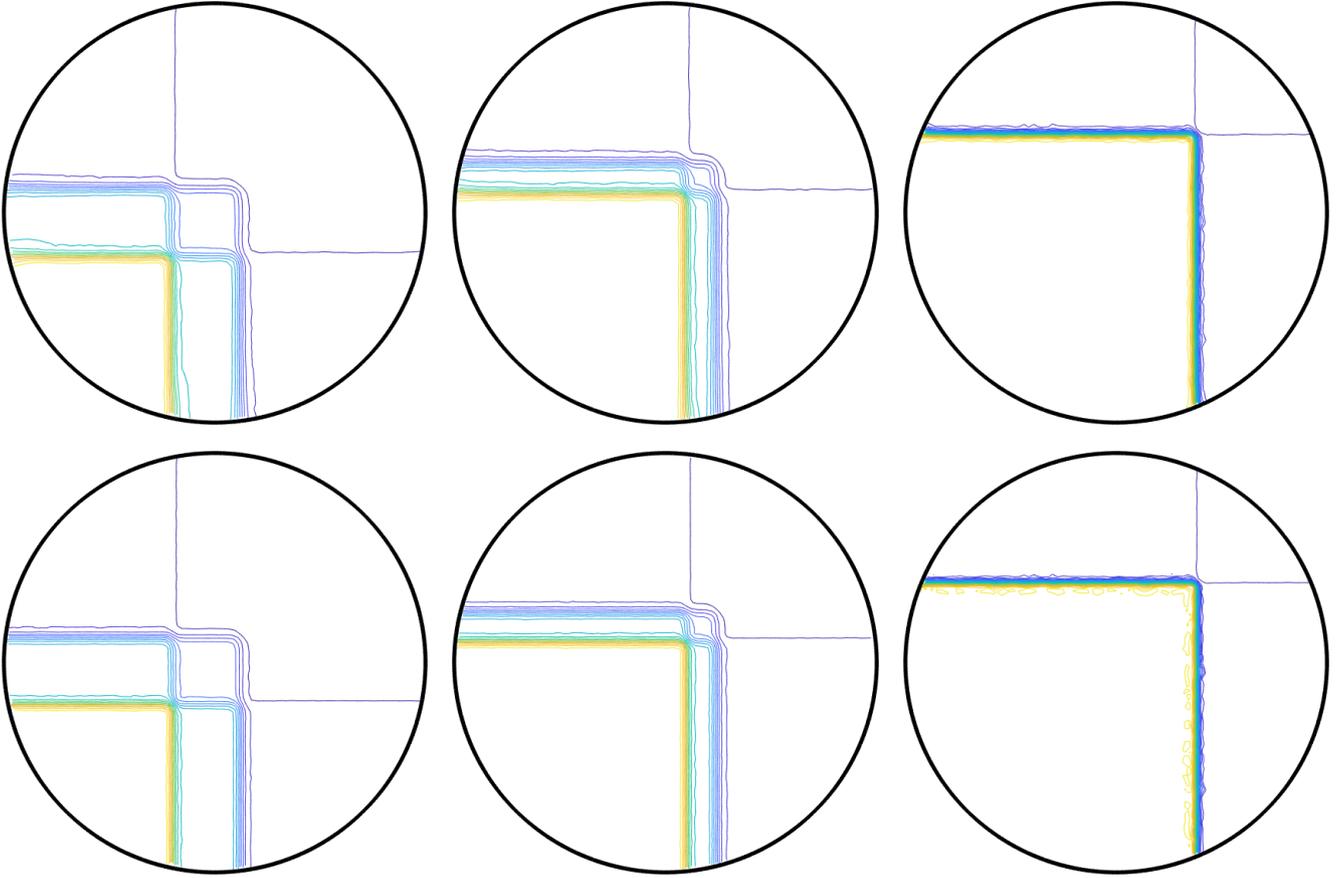


Figure 9: Results for the moving fronts problem using $\nu = 1.5 \times 10^{-3}$ at time $t = 0.2$ (first column), $t = 0.4$ (second column) and $t = 0.6$ (third column) using the TPS procedure (first row) and exact solution (second row) on a mesh with 9670 control volumes and CFL = 5.

where

$$A(t, z) = \frac{0.05}{\nu}(z - 0.5 + 4.95t), \quad B(t, z) = \frac{0.25}{\nu}(z - 0.5 + 0.75t), \quad C(t, z) = \frac{0.50}{\nu}(z - 0.375),$$

with $z = x$ or y . Initial and boundary conditions are defined by the following analytical solution

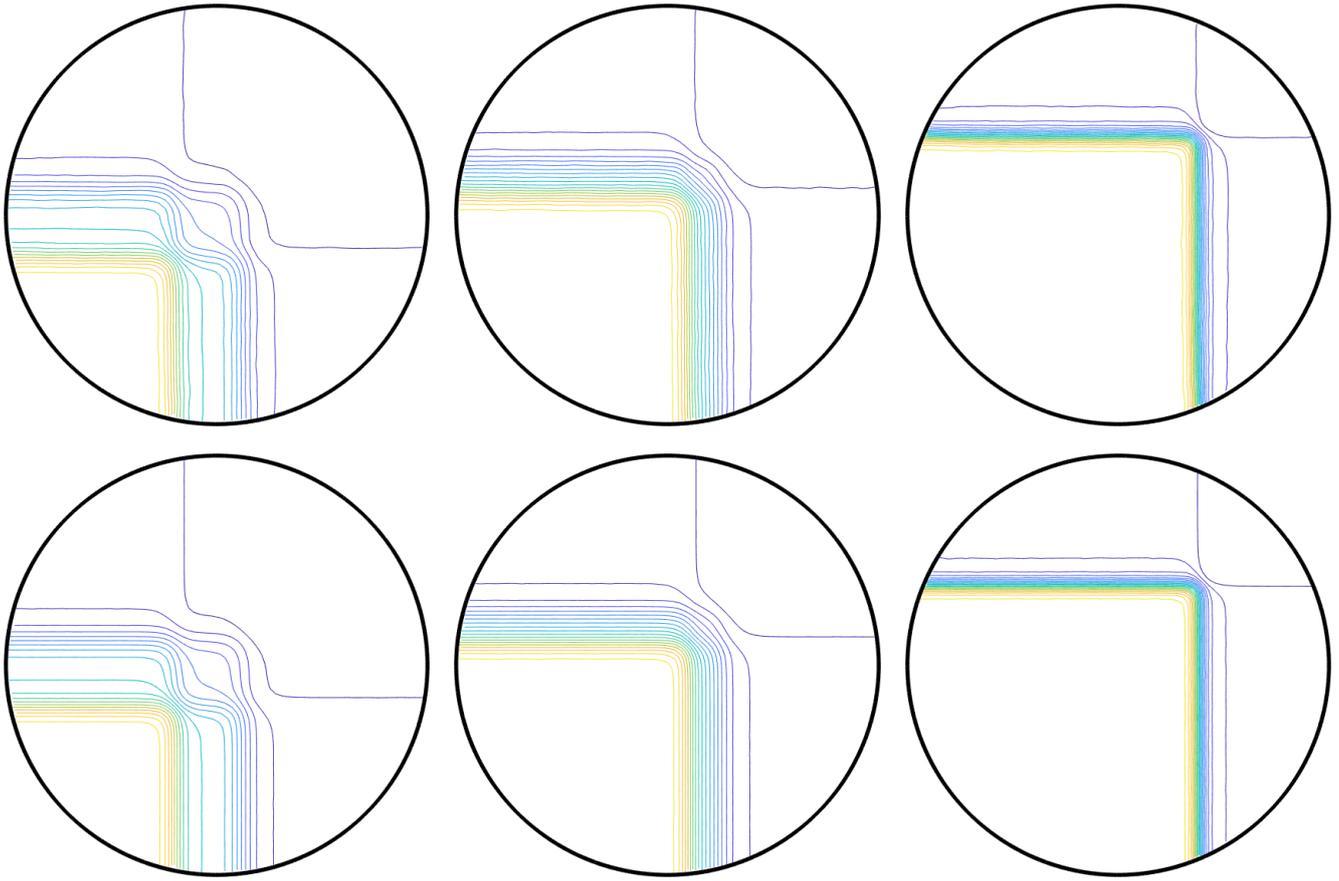


Figure 10: Results for the moving fronts problem using $\nu = 4 \times 10^{-3}$ at time $t = 0.2$ (first column), $t = 0.4$ (second column) and $t = 0.6$ (third column) using the TPS procedure (first row) and exact solution (second row) on a mesh with 9670 control volumes and CFL = 5.

$$c(t, x, y) = u(t, x, y)v(t, x, y).$$

The computational domain is discretized into an unstructured triangular mesh with 9670 control volumes and two diffusion coefficients $\nu = 1.5 \times 10^{-3}$ and 4×10^{-3} are used in the simulations. The CFL is fixed to 5 and numerical results are presented at three different instants namely $t = 0.2, 0.4$ and 0.6 . In Figure 9 we present 20 equi-distributed contourlines of the solutions obtained by the conservative approach using TPS interpolation procedure and $\nu = 1.5 \times 10^{-3}$ at the three selected times. Exact solution is also included in this figure for comparisons. Those results obtained using $\nu = 4 \times 10^{-3}$ are depicted in Figure 10. It is clear that, by decreasing the values of ν the convective term becomes dominant and steep internal layers are formed near the vicinity of front lines in the computational domain. For both values of ν , the internal layers are wide and diffuse in the flow domain and as time increases, the moving fronts merge in a single front which moves along the main diagonal of the circular domain. It is apparent that the solution structures are in good agreement with the exact solutions presented in these figures. These results give a clear view of the overall transport pattern and the effect of the diffusion coefficient ν on the structure of moving fronts in the circular domain. It is worth remarking that the thinning of the internal layers with decreasing ν is evident from these plots and the rate of this thinning is slower for $\nu = 1.5 \times 10^{-3}$ than for $\nu = 4 \times 10^{-3}$. These features clearly demonstrate the high accuracy achieved by the proposed semi-Lagrangian finite volume method for solving moving fronts problems using large time steps. In addition, compared to the results published for example in [22, 16], it can be seen that our method resolves accurately the solution features and the moving fronts seem to be localized in the

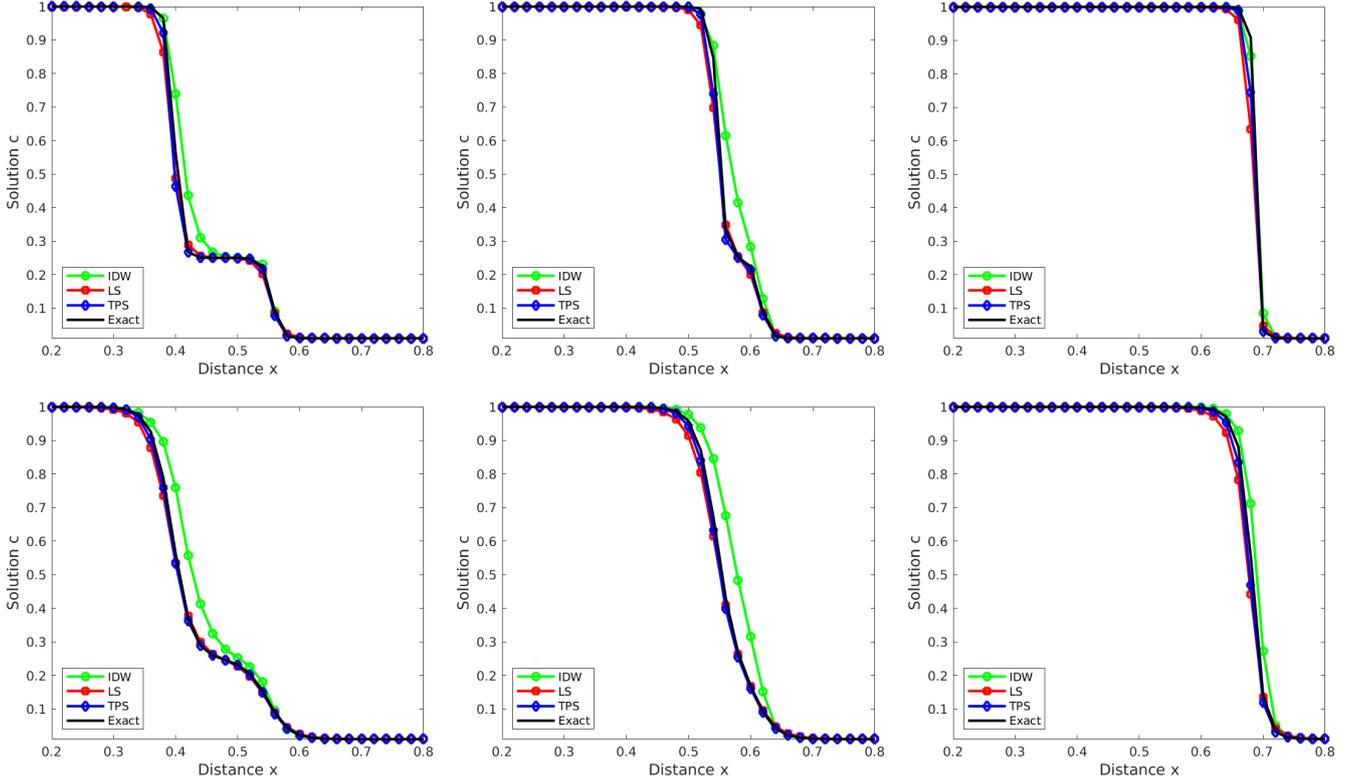


Figure 11: Cross-sections of the results for the moving fronts problem at the main diagonal $y = x$ using $\nu = 1.5 \times 10^{-3}$ (first row) and $\nu = 4 \times 10^{-3}$ (second row) at time $t = 0.2$ (first column), $t = 0.4$ (second column) and $t = 0.6$ (third column) using different interpolation procedures on a mesh with 9670 control volumes and CFL = 5.

correct place in the flow domain.

For visualizing the comparisons, we display in Figure 11 cross-sections of the results at the main diagonal $y = x$ using IDW, LS and TPS interpolation procedures. For large values of ν , it is clear that the LS and TPS procedures produce practically identical results on the mesh of 9670 control volumes. This can be attributed to the large physical diffusion presented in the problem. However, decreasing the value of ν the results computed by TPS procedure are more accurate than those computed by the LS procedure. Apparently, by using the TPS procedure, high resolution is achieved in those regions where the solution gradients are steep such as the moving fronts. Comparing the results obtained using the considered procedure, it is clear that the IDW procedure produces diffusive solutions resulting in smearing the shocks. On the other hand, this numerical diffusion has remarkably been reduced in the results computed using the TPS procedure. Needless to say that for convection-dominated cases, the semi-Lagrangian finite volume method does not diffuse the fronts or gives spurious oscillations near the steep gradients. To quantify the results for this test example, we present in Table 3 L^1 -errors and computational times for LS and TPS interpolation procedures at $t = 0.2$ and 0.6 using different meshes. In terms of the L^1 -errors, the TPS results are more accurate than the results obtained using the LS procedure for both diffusion coefficients considered. Concerning the computational times, Table 3 shows that the CPU time of TPS procedure is less than 1.3 times the CPU time of LS procedure.

4.4 Transport problem in the Mediterranean sea

The Mediterranean sea is considered to be an almost isolated water system as it is nearly completely surrounded by land; on the north by Europe, on the south by North Africa, and on the east by the

Table 3: Error-norms and computational times for the moving fronts problem at $t = 0.2$ and $t = 0.6$ using different meshes and interpolation procedures with $\nu = 1.5 \times 10^{-3}$ and $\nu = 4 \times 10^{-3}$.

		$t = 0.2$				$t = 0.6$			
		LS method		TPS method		LS method		TPS method	
ν	# elements	L^1 -error	CPU	L^1 -error	CPU	L^1 -error	CPU	L^1 -error	CPU
1.5×10^{-3}	1066	2.67697E-02	1.5	1.05318E-02	1.8	4.02518E-02	4.6	1.08510E-02	5.5
	1978	1.75511E-02	4	6.29145E-03	4.8	2.26506E-02	12.5	6.64615E-03	14.7
	3680	1.03368E-02	20	4.40345E-03	25	2.30111E-02	62	4.75241E-03	77
	7583	8.68046E-03	152	2.88446E-03	178	9.02707E-03	451	3.94406E-03	540
4×10^{-3}	1066	1.72880E-02	1.5	4.08353E-03	1.9	3.14676E-02	4.7	5.02832E-03	5.7
	1978	1.14393E-02	4	3.02945E-03	4.9	1.74601E-02	13	4.11802E-03	14.8
	3680	7.73985E-03	22	2.33330E-03	26	1.27555E-02	61	3.72815E-03	78
	7583	4.80700E-03	145	1.70628E-03	177	6.73888E-03	453	3.38959E-03	544

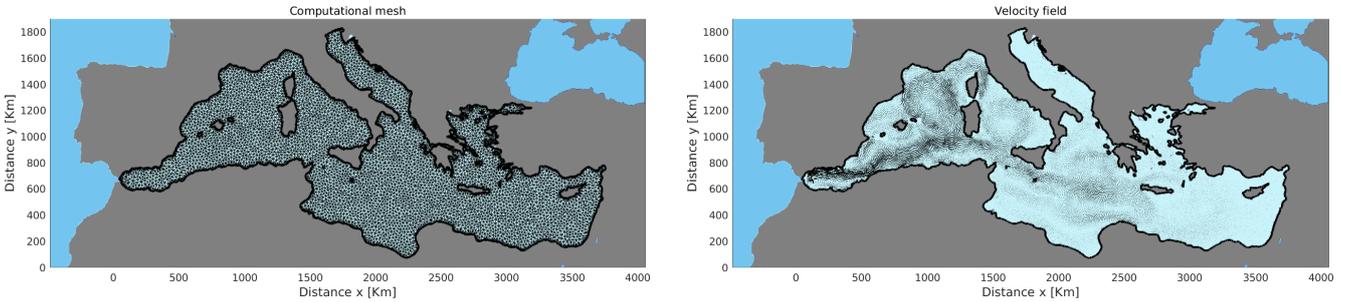


Figure 12: Computational mesh (left plot) and velocity field (right plot) used for the transport problem in the Mediterranean sea.

Middle east. The Mediterranean Sea has an average depth of 1500 m and its west-east length, from strait of Gibraltar to the Gulf of Iskenderun on the southwestern coast of Turkey, is about 4000 km . The Mediterranean climate is characterized by warm temperatures, winter-dominated rainfall, and dry summer, see for example [21] and further references are therein. Connected with the Atlantic Ocean, the Mediterranean sea exchanges water, salt, heat, and other properties with the north Atlantic Ocean. The Mediterranean sea is also known as one of the busiest shipping lanes in the world with 15 % of global maritime transport in terms of number of stopovers and 10 % in terms of deadweight tons. Usually these shipping activities are sources of sea pollution, see [18] among others. The transport and dispersion of pollutants closely depend on the circulation of sea which is forced by water flow exchange through various straits (Gibraltar, Otranto and Sicily), by wind stress, and by buoyancy forces at the surface due to fresh water and heat fluxes, see for instance [28]. In this test example, we consider a problem of pollutant transport released in the Mediterranean sea. The velocity fields in this test case is obtained from the incompressible steady-state Navier-Stokes model subject to given inflow conditions on the strait of Gibraltar. A pollutant concentration is initially released in the sea-surface and its transport is computed using the proposed method at different times. Our objective for this numerical example is twofold, on one hand to assess the capability of our semi-Lagrangian finite volume method to accurately handle complex geometry and on the other hand to develop robust numerical tools to efficiently study pollution transport in the Mediterranean sea. Hence, the problem statement consists of solving the equations (1) in the computational domain defined by the Mediterranean sea with $\nu = 100 \text{ m}^2/s$ and subject to a continuous release defined as

$$f(t, x, y) = 100 \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2}{40000}\right),$$

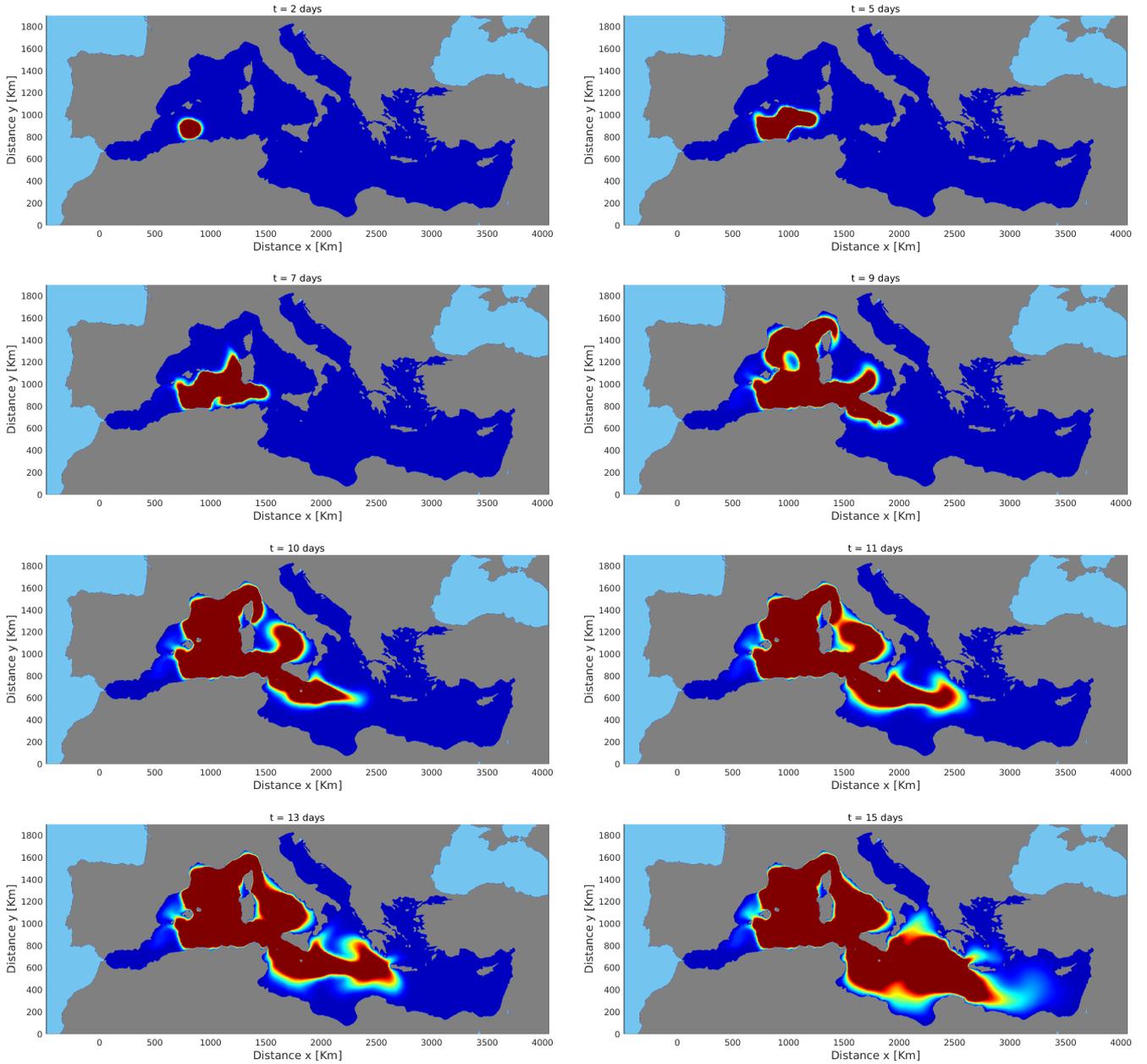


Figure 13: Results for pollutant transport in the the Mediterranean sea at eight different times $t = 2, 5, 7, 9, 10, 11, 13$ and 15 days using a mesh of 18827 control volumes and $CFL = 10$.

where $(x_0 = 900 \text{ km}, y_0 = 880 \text{ km})$ is the selected location for pollutant release. The computational domain is discretized in an unstructured triangular mesh with a total number of 18827 control volumes and the $CFL = 10$. In Figure 12 we present the computational mesh and the velocity field used in our simulations of the advection-diffusion problem (1). Note that, unlike the previous test examples where advection-diffusion problems are solved in small simple geometries, the considered transport problem is solved in a large domain with complex geometry. As a consequence, the later transport is more difficult to handle and the results shown here illustrate the robustness of the semi-Lagrangian finite volume method. Furthermore, the considered pollutant transport is a problem conservative in nature and therefore, good numerical accuracy is required in order to capture the different phenomena present in its evolving solution.

Figure 13 shows the concentration distribution at eight different times, namely $t = 2, 5, 7, 9, 10, 11, 13$ and 15 days. At earlier time of the simulation, the concentration front released in the sea starts to

develop and is transported later on by the flow at far east of the Mediterranean sea. The interaction between the pollution transport and the water flow is detected across the sea during the simulation time. It can be clearly seen that the complicated concentration is captured by the semi-Lagrangian finite volume method. We can see that the major recirculations are located near the Sicily narrow and Sypric basin. Inside these recirculations, there is a more complex transport pattern. The decrease and increase of the strengths of recirculations with time can be seen in Figure 13. Note that results from the proposed transport model should be compared with observations of real pollutant distributions on the sea-surface in the Mediterranean sea. However, there is no data available until now in the literature to carry out this work. Thus, at the moment we can only perform simulations and verify that results are plausible and consistent. In summary, the pollutant transport is captured accurately and the concentration front is resolved reasonably. It should be stressed that all these features have been achieved using time steps larger than those required for Eulerian-based methods in convection-dominated flows.

5 Conclusions

In this study we have developed a new conservative semi-Lagrangian finite volume method for solving convection-diffusion equations using the adjusted advection technique for unstructured triangular grids. This method exploits the interesting features offered by both techniques to construct a fully conservative algorithm for numerical treatment of convection-diffusion problems. The important advantage of the new method is that, the convective term that has to be treated carefully in most of Eulerian-based finite volume methods has been removed from the new method by using the semi-Lagrangian method to interpret the transport nature of the governing equations. A comparison to the conventional semi-Lagrangian finite volume method demonstrates the feasibility of the present adjusted advection approach to solve convection-dominated flow problems. A series of numerical examples including a transport problem in the Mediterranean sea were considered to test the accuracy of the proposed method. A comparison to the conventional semi-Lagrangian finite volume method were also performed in the present study. The obtained results using the present adjusted advection algorithm show good solution accuracy and less numerical dissipation compared to the results obtained using the conventional semi-Lagrangian method. It should be mentioned that, since the present method uses fractional steps to separately resolve the convection and the diffusion terms, the overall accuracy is first order and to increase its accuracy, high-order splitting operators should be employed. The future research should also be focused on the extension of these techniques to systems of nonlinear convection-dominated problems on unstructured meshes. The computational efficiency of the method can be further improved by advanced interpolation procedures and optimization of the code. A desirable study would also be a more thorough evaluation of the method accuracy in the semi-Lagrangian finite volume method for three-dimensional problems. The computational domains in these problems are more typical of realistic cases and is expected to be more interesting and serve as a better test of efficiency and accuracy.

Acknowledgment. The authors would like to thank anonymous referees for giving very helpful comments and suggestions that have greatly improved this paper.

References

- [1] A. Allievi and R. Bermejo. A generalized particle search-locate algorithm for arbitrary grids. *J. Comp. Physics*, 132:157–166, 1992.
- [2] O. Babak and C. Deutsch. Statistical approach to inverse distance interpolation. *Stochastic Environmental Research and Risk Assessment*, 23(5):543–553, 2009.
- [3] C. Bajaj, I. Ihm, and J. Warren. Higher-order interpolation and least-squares approximation using implicit algebraic surfaces. *ACM Transactions on Graphics*, 12(4):327–347, 1993.

- [4] F. Benkhaldoun, I. Elmahi, and M. Seaid. Well-balanced finite volume schemes for pollutant transport by shallow water equations on unstructured meshes. *Journal of computational physics*, 226(1):180–203, 2007.
- [5] P. Bochev, S. Moe, K. Peterson, and D. Ridzal. A conservative, optimization-based semi-Lagrangian spectral element method for passive tracer transport. *Coupled Problems 2015-Proceedings of the 6th International Conference on Coupled Problems in Science and Engineering*, 23–34, 2015.
- [6] X. Cai, W. Guo, and J.M. Qiu. A high order conservative semi-Lagrangian discontinuous Galerkin method for two-dimensional transport simulations. *Journal of Scientific Computing*, 73(2-3):514–542, 2017.
- [7] X. Cai, J. Qiu, and J-M. Qiu. A conservative semi-Lagrangian HWENO method for the Vlasov equation. *J. Computational Physics*, 323:95–114, 2016.
- [8] A. Cameron. A systematic method to enforce conservativity on semi-Lagrangian schemes. *International J. for Numerical Methods in Fluids*, 88(10-11):463–478, 2018.
- [9] J. Douglas, C. Huang, and F. Pereira. The modified method of characteristics with adjusted advection. *Numer. Math.*, 83:353–369, 1999.
- [10] J. Douglas and T.F. Russell. Numerical methods for convection dominated diffusion problems based on combining the method of characteristics with finite elements or finite differences. *SIAM J. Numer. Anal.*, 19:871–885, 1982.
- [11] M. El-Amrani and M. Seaid. Eulerian-Lagrangian time-stepping methods for convection-dominated problems. *International Journal of Computer Mathematics*, 85(3-4):421–439, 2008.
- [12] M. El-Amrani and M. Seaid. An L^2 -projection for the Galerkin-characteristic solution of incompressible flows. *SIAM Journal on Scientific Computing*, 33(6):3110–3131, 2011.
- [13] L. Fatone, D. Funaro, and G. Manzini. Arbitrary-order time-accurate semi-Lagrangian spectral approximations of the Vlasov-Poisson system. *J. of Computational Physics*, 384:349–375, 2019.
- [14] E.V. Hólm. A fully two-dimensional, non-oscillatory advection scheme for momentum and scalar transport equations. *Mon. Wea. Rev.*, 123:536–552, 1995.
- [15] W. Keller and A. Borkowski. Thin plate spline interpolation. *Journal of Geodesy*, 1–19, 2019.
- [16] S.V. Krishnamachari, L.J. Hayes, and T.F. Russell. A finite element alternating-direction method combined with a modified method of characteristics for convection-diffusion problems. *SIAM Journal on Numerical Analysis*, 26(6):1462–1473, 1989.
- [17] P. Lauritzen, M. Taylor, J. Overfelt, P. Ullrich, R. Nair, S. Goldhaber, and R. Kelly. CAM-SE–CSLAM: Consistent coupling of a conservative semi-Lagrangian finite-volume method with spectral element dynamics. *Monthly Weather Review*, 145(3):833–855, 2017.
- [18] P. Le Lourd. *Oil pollution in the Mediterranean Sea*, volume 6. Royal Swedish Academy of Sciences, 1977.
- [19] G. Lu and D. Wong. An adaptive inverse-distance weighting spatial interpolation technique. *Computers & Geosciences*, 34(9):1044–1055, 2008.
- [20] Z. Ma, Y. Zhang, and Z. Zhou. An improved semi-Lagrangian time splitting spectral method for the semi-classical Schrödinger equation with vector potentials using NUFFT. *Applied Numerical Mathematics*, 111:144–159, 2017.

- [21] A. Malagó, F. Bouraoui, B. Grizzetti, and A. De Roo. Modelling nutrient fluxes into the mediterranean sea. *Journal of Hydrology: Regional Studies*, 22:100592, 2019.
- [22] A. Mueller. Continuously deforming finite element methods for transport problems. *PhD dissertation, University of Texas at Austin*, 1984.
- [23] R. Nair, Côté J., and A. Staniforth. Monotonic cascade interpolation for Galerkin-characteristics advection. *Quart. J. Roy. Met. Soc.*, 125:197–212, 1999.
- [24] N. Petrovskaya. The accuracy of least-squares approximation on highly stretched meshes. *International Journal of Computational Methods*, 5(03):449–462, 2008.
- [25] M. Powell. The uniform convergence of thin plate spline interpolation in two dimensions. *Numerische Mathematik*, 68(1):107–128, 1994.
- [26] J. Pudykiewicz and A. Staniforth. Some properties and comparative performance of the semi-Lagrangian method of robert in the solution of advection-diffusion equation. *Atmos. Ocean.*, 22:283–308, 1984.
- [27] A. Robert. A stable numerical integration scheme for the primitive meteorological equations. *Atmos. Ocean*, 19:35–46, 1981.
- [28] A. Robinson, W. Leslie, A. Theocharis, and A. Lascaratos. Mediterranean sea circulation. *Ocean currents*, 1:19, 2001.
- [29] M. Seaid. Semi-Lagrangian integration schemes for viscous incompressible flows. *Comp. Methods in App. Math.*, 4:392–409, 2002.
- [30] A. Staniforth and Côté J. Semi-Lagrangian integration schemes for the atmospheric models: A review. *We. Rev.*, 119:2206–2223, 1991.
- [31] C. Temperton and A. Staniforth. An efficient two-time-level Galerkin-characteristics semi-implicit integration scheme. *Quart. J. Roy. Meteor. Soc.*, 113:1025–1039, 1987.
- [32] T. Yabe, R. Tanaka, T. Nakamura, and F. Xiao. An exactly conservative semi-Lagrangian scheme (CIP–CSL) in one dimension. *Monthly Weather Review*, 129(2):332–344, 2001.
- [33] Z. Zhang. Conservative finite-difference methods on general grids. *SIAM Review*, 39(2):367–369, 1997.



Citation on deposit: Asmouh, I., El-Amrani, M., Seaid, M., & Yebari, N. (2020). A Conservative Semi-Lagrangian Finite Volume Method for Convection–Diffusion Problems on Unstructured Grids. *Journal of Scientific Computing*, 85(1), Article 11. <https://doi.org/10.1007/s10915-020-01316-8>

For final citation and metadata, visit Durham Research Online URL:
<https://durham-repository.worktribe.com/output/1233015>

Copyright statement: This content can be used for non-commercial, personal study.