



An conservative anti-diffusion technique for the level set method[☆]



Jean-Luc Guermond^a, Manuel Quezada de Luna^a, Travis Thompson^{b,*}

^a Department of Mathematics, Texas A&M University, College Station, TX 77843, United States

^b Department of Comp. and Appl. Mathematics, Rice University, Houston, TX 77005, United States

ARTICLE INFO

Article history:

Received 20 August 2016

Received in revised form 31 January 2017

Keywords:

Conservative

Level set

Two phase flow

Finite volume

Finite element

Entropy viscosity

ABSTRACT

A novel conservative level set method is introduced for the approximation of two-phase incompressible fluid flows. The method builds on recent conservative level set approaches and utilizes an entropy production to construct a balanced artificial diffusion and artificial anti-diffusion. The method is self-tuning, maximum principle preserving, suitable for unstructured meshes, and neither re-initialization of the level set function nor reconstruction of the interface is needed for long-time simulation. Computational results in one, two and three dimensions are presented for finite element and finite volume implementations of the method.

© 2017 Elsevier B.V. All rights reserved.

1. Introduction

The method of level sets was first introduced by Osher and Sethian [1] in the late 1980s as a technique for capturing evolving interfaces and tracking the propagation of fronts. The traditional level set method is useful for handling complex topological dynamics but suffers from a lack of mass conservation. In the present paper we present a novel conservative level set technique using a new compression strategy; the underlying idea is to utilize a corrective flux constructed from entropy principles. A first-order application of the approach yields a provable maximum principle. The first-order scheme can be extended to a high-order non-linear compression, and flux-corrected transport techniques can be applied to retrieve the maximum principle.

Conservative level set methods have been the focus of recent research. Techniques for enforcing mass conservation in level set methods differ depending on the context of the method. The use of an entropy-production to inform the assembly of a precise anti-diffusion performs similarly to hybrid particle-type methods (e.g., Enright et al. [2], Ianniello and Di Mascio [3]). The proposed construction is, however, less complex, and straight-forward to implement. In addition, the proposed method does not require solving a re-initialization or interface reconstruction subproblem (see e.g., Fedkiw et al. [4], Olsson and Kreiss [5]). Further, the non-compressive first-order variant of our method does not rely on heuristics such as limiters or explicit mass redistribution (see e.g., Olsson and Kreiss [5], Chiu and Lin [6]) to address the usual undershoots, and overshoots.

[☆] This material is based upon work supported in part by the National Science Foundation grants DMS-1217262, DMS-1620058, DMS-1619892, by the Air Force Office of Scientific Research, USAF, under grant/contract numbers FA99550-12-0358, FA9550-15-1-0257, and by the Army Research Office under grant/contract number W911NF-15-1-0517.

* Corresponding author.

E-mail addresses: guermond@math.tamu.edu (J.-L. Guermond), mquezada@math.tamu.edu (M.Q. de Luna), tthompson@rice.edu (T. Thompson).

The current work offers other improvements over many contemporary algorithms, and specific comparisons are discussed in the computational results, Sections 7–12. For instance, the proposed approach does not invoke a characteristic mesh size; this is important for general unstructured meshes as the notion of local mesh size can be problematic to define (see Section 2.4 in Guermond and Nazarov [7] for a discussion on this topic). Secondly, the current approach details how to construct a first-order, diffusion corrected, viscous tensor which is maximum principle preserving, as proven in Guermond and Popov [8], provided the transport velocity is incompressible. Third, a high-order extension of the diffusion corrected dissipation is explained; this extension ensures that the artificial dissipation vanishes outside of a neighborhood of the level set iso-surface. Finally, it is shown how this extension can be made maximum-principle preserving through the adaptation of the flux-corrected transport ideas of Boris–Book–Zalesak. A similar methodology has been proposed in Chiu and Lin [6, Eq. (18)] and, in some sense, our work can be viewed as an extension thereof. In a wider, more historical, context the proposed method can be seen as combining the level set method with the one-step re-initialization heuristics of Coupez [9], the theory of artificial compressors vis-a-vis Harten [10,11], Olsson and Kreiss [5] and the entropy production principles of Guermond et al. [12].

The paper is organized as follows. Section 2 gives a brief literature survey of historical topics related to the discussions in the sections that follow. Section 3 discusses the preliminaries of artificial compression as a viscous correction and introduces Eq. (10). This equation is the motivation of the method proposed in the paper and can be viewed as a one-stage version of the classical artificial compression method proposed by Harten [10,11]. Section 4 describes the full method and implementation in the context of finite elements. Section 5 describes the analogous implementation details in the finite volume framework. Section 6 details an extension of the methods to a maximum principle preserving technique; this is done by adapting the flux-corrected transport methodology of Boris–Book–Zalesak. Sections 7–12 present numerical illustrations of the performance of the method in one, two, and three dimensions for a variety of benchmark test problems. The computations and results discussed in Sections 7–12 demonstrate the efficacy of the technique for interface capturing. The computation of secondary terms, such as surface tension or stresses, is out of the scope the current work and is therefore not discussed.

2. Brief overview of the literature

This section gives a brief survey of the topics which constitute a historical foundation for the proposed method. Readers who are already familiar with numerical front tracking methods, the level set method, the role of re-initialization in the level set method, and the connection between re-initialization of Heaviside level set functions and anti-diffusion should advance to Section 3.

2.1. Numerical front tracking and the level set method

Numerical techniques for the evolution of interfaces and free surfaces are an active area of research. Popular Eulerian approaches for the transport of an interface include volume tracking methods and level set methods. More extensive sources for the historical development of these methods and their variants are found in, respectively, Rider and Kothe [13] and Osher and Fedkiw [14]. A drawback of the volume tracking methods is the difficulty in a-posteriori interface reconstruction while level set methods can suffer from loss of area enclosed by the interface. Hybrid methods have been proposed to address these issues such as coupling volume tracking and the level set method (CLSVOF) (Sussman and Puckett [15]), the addition of marker particles (Enright et al. [2]), and the extension to oriented marker particles with reconstruction (Ianniello and Di Mascio [3]).

The level set method typically refers to the transport of a smooth distance function whose zero isosurface is the interface to be tracked. The method has been used successfully in many fluid flow applications, see e.g., Sussman et al. [16], Osher and Sethian [1], Gibou et al. [17], Ville et al. [18], Bonito et al. [19], but it can also be used in other contexts like shape optimization as in Dapogny et al. [20], Yamada et al. [21].

If the advection velocity field is incompressible, the transport problem can be recast in conservative form which allows for the use of smoothed Heaviside level set functions, conservative discretizations, and anti-diffusion techniques for enhancing mass conservation.

Given a velocity field \mathbf{u} , the basic equation for the level set method is

$$\phi_t + \mathbf{u} \cdot \nabla \phi = 0. \quad (1)$$

Depending on the specifics of the problem, the above equation is often rewritten in alternative forms. For instance if \mathbf{u} is divergence free, (1) can be expressed in the conservation form $\phi_t + \nabla \cdot (\mathbf{u}\phi) = 0$ which can be discretized using conservative numerical schemes. In the context of curvature driven flows Eq. (1) can be rewritten as follows:

$$\phi_t + u_N \|\nabla \phi\|_{\ell^2} = 0, \quad (2)$$

which is of Hamilton–Jacobi type, allowing access to fast numerical schemes for solving problems in this category. Both Eqs. (1) and (2) are often referred to as ‘the level set equation’. The reader is referred to Fedkiw et al. [4], and the sources therein, for a large selection of references on the many applications of level set methods. We also refer to Dapogny et al. [20, Section 4] for a review of important mathematical properties of the signed distance function.

2.2. Reinitialization and level set methods

To minimize area loss, level set methods typically use re-initialization procedures although interface reconstruction has also been used in place of re-initialization in Ianniello and Di Mascio [3]. The type of reinitialization procedure depends on the context of the problem and the type of level set function used. The method described in this manuscript uses smoothed Heaviside level set functions and the level set equation in conservative form. The re-initialization procedure used is therefore a shock capturing anti-diffusion technique.

The classical level set method uses a smooth distance function for its level set function ϕ . Therefore the re-initialization problem is aimed at preserving the property $\|\nabla\phi\|_{L^2} = 1$ which is necessary and sufficient for ϕ to be a distance function. Historically, level set methods were implemented in two stages: (i) the first one consists of solving the level set equation (1), or one of its variations, for a few time steps; (ii) in the second stage the time stepping is suspended and a subproblem is solved to reinitialize the level set function. The usual re-initialization approach for distance level set functions consists of solving the unsteady Hamilton–Jacobi problem

$$\partial_t \phi = S(\phi)(1 - \|\nabla\phi\|_{L^2}), \quad \phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}), \tag{3}$$

to steady state, where $S(\cdot)$ is a smoothed approximation of the sign function, see e.g., Sussman et al. [22]. Fast methods for solving unsteady Hamilton–Jacobi problems is a well established area of ongoing research.

When \mathbf{u} is divergence free the level set Eq. (1) can be recast in conservative form. In this case a smooth Heaviside type function can be used as the level set function and the level set equation is often discretized using a conservative method limiting oscillations at the interface; these methods include the total variation diminishing (TVD) methods, and essentially non-oscillatory (ENO/WENO) type approaches discussed in Harten [23]. The re-initialization for smoothed Heaviside type level set functions is an anti-diffusive artificial compression technique, see e.g., Harten [10,11]. This is precisely the approach followed in Olsson and Kreiss [5] where a TVD scheme with Superbee limiter on a uniform mesh is coupled with the artificial compression subproblem

$$\partial_t \phi + \nabla \cdot \left(\phi(1 - \phi) \frac{\nabla\phi}{\|\nabla\phi\|_{L^2}} \right) = 0, \quad \phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}). \tag{4}$$

Recently, researchers have put forth methods combining these two steps into a single-step method, see e.g., Coupez [9, Eq. (16)], Ville et al. [18, Eq. (25)] and Chiu and Lin [24]. In the present work a modified equation is introduced which lifts the two step, artificial compression approach of Olsson and Kreiss [5] to a one step method. The procedure yields a modified level set equation similar to that of Chiu and Lin [24].

3. Principle of the artificial-viscosity-based compression method

This section introduces Eq. (10) which is the motivation of the method proposed in the paper. Artificial compression methods were proposed by Harten [10,11] for use in the numerical solution of conservation laws. Harten introduced the technique with the one dimensional, possibly stabilized, conservation equation

$$\partial_t \phi + \partial_x f(\phi) = \partial_x(\mu \partial_x \phi), \quad \phi(x, 0) = \begin{cases} \phi_L & \text{if } x < 0 \\ \phi_R & \text{if } x > 0. \end{cases} \tag{5}$$

with convex flux f , artificial viscosity $\mu \geq 0$, and Riemann initial data. The artificial compression method proceeds by augmenting (5) with an additional flux term $g(\phi)$ to produce

$$\partial_t \phi + \partial_x (f(\phi) + g(\phi)) = \partial_x(\mu \partial_x \phi). \tag{6}$$

The function $g(\phi)$ is called an “artificial compression function” and in practice it may be any function satisfying the compatibility condition:

$$g(\phi) = 0, \quad \forall \phi \notin (\phi_R, \phi_L), \quad \text{and} \quad g(\phi) \text{sign}(\phi_R - \phi_L) > 0, \quad \forall \phi \in (\phi_R, \phi_L). \tag{7}$$

The classical artificial compression approach to solving (6) is a two-stage method. First, (5) is solved followed by a sub-timestep problem of the form $\partial_t \hat{\phi} + \partial_x g(\hat{\phi}) = 0$ with initial data equal to the current numerical solution; see Harten [10,11] for more details. An anti-diffusive formulation which can be solved in one stage is enabled by the careful selection of the artificial compression function $g(\phi)$. Introducing the notation $z^+ = \max(z, 0)$, for reasons which will be clear shortly, a simple way to ensure that the compatibility condition is satisfied is to choose $g(\phi)$ proportional to $[(\phi - \phi_L)(\phi_R - \phi)]^+ \text{sign}(\phi_R - \phi_L)$. When $\mu \neq 0$, in (6), the profile of the wave connecting ϕ_L to ϕ_R is smooth and monotone, and we have $\text{sign}(\phi_R - \phi_L) = \partial_x \phi / |\partial_x \phi|$. It is then natural to set

$$g(\phi) = c\mu [(\phi - \phi_L)(\phi_R - \phi)]^+ \frac{\partial_x \phi}{|\partial_x \phi|}, \tag{8}$$

where $c \geq 0$ is a non-negative parameter. It can now be seen that the use of the positive part $[\cdot]^+$ in (8) ensures that a numerical approximation to $g(\phi)$ avoids overshoots and undershoots. Note that since μ scales like a wave speed times a length scale, and $g(\phi)$ must scale like a wave speed times ϕ , the parameter c must scale like $c \sim c_{\text{comp}}\mathcal{L}^{-1}$ where c_{comp} is a universal constant and \mathcal{L} is some length scale. In the context of an approximation method with mesh size h , we expect the compression to be active over a few mesh cells; it is then natural to set $\mathcal{L} = h$. Eq. (6) can then be rewritten

$$\partial_t \phi + \partial_x \left(\mathbf{f}(\phi) - \mu \left(1 - c_{\text{comp}} \frac{[(\phi - \phi_L)(\phi_R - \phi)]^+}{h|\partial_x \phi|} \right)^+ \partial_x \phi \right) = 0. \tag{9}$$

Hence, artificial compression can be achieved for any artificial-viscosity-based approximation method by redefining the viscosity appropriately. The generalization of (9) in higher dimension is

$$\partial_t \phi + \nabla \cdot \left(\mathbf{f}(\phi) - \mu \left(1 - c_{\text{comp}} \frac{[(\phi - \phi_L)(\phi_R - \phi)]^+}{h\|\nabla \phi\|_{\ell^2}} \right)^+ \nabla \phi \right) = 0, \tag{10}$$

where $\|\cdot\|_{\ell^2}$ is the Euclidean norm. The use of the positive part $[\cdot]^+$ in (10) guarantees that numerical approximations to the effective viscosity are non-negative. Furthermore, since Eq. (10) incorporates the anti-diffusion of $g(\phi)$ directly into its formulation it can be interpreted as an artificial compression method, in the spirit of Harten, amenable to direct discretization.

A simple analysis for the case of $\mathbf{f}(\phi) = \mathbf{u}\phi$ with $\nabla \cdot \mathbf{u} = 0$ allows for a demonstration of the effect of artificial compression on (10). Suppose that $0 \leq \phi \leq 1$ and the transition interface is embedded as the iso-value $\{\phi = \frac{1}{2}\}$, i.e., the Riemann data are such that $\phi_L, \phi_R \in \{0, 1\}$. Let $\mathbf{x}_0(t)$ be a point on the moving interface, and assume that the velocity \mathbf{u} is locally constant in a neighborhood of \mathbf{x}_0 , and $\phi(\mathbf{x} - \mathbf{u}t)$ is time-independent, i.e., the interface is just transported and is at equilibrium. Assume also that μ is locally constant. Denoting $\varphi(\mathbf{x}) = \phi(\mathbf{x} - \mathbf{u}t)$ and letting s be the signed distance along the direction of the gradient of φ , we infer that $\mu \left(1 - \frac{c_{\text{comp}}}{h} \frac{\varphi(1-\varphi)}{\|\nabla \varphi\|_{\ell^2}} \right) \nabla \varphi = 0$, i.e., φ solves the following ODE, $\partial_s \varphi - \alpha \varphi(1 - \varphi) = 0$, $\varphi(0) = \frac{1}{2}$, where $\alpha = c_{\text{comp}}h^{-1}$. The solution to this ODE is

$$\varphi(s) = \frac{1}{2} \left(1 + \tanh \left(c_{\text{comp}} \frac{s}{2h} \right) \right). \tag{11}$$

We interpret this result by saying that at equilibrium the compression balances exactly the artificial viscosity, and the level set adopts the classical hyperbolic tangent profile of width $2h/c_{\text{comp}}$. The constant c_{comp} can be used to further control the width of the interface.

This simple example demonstrates the advantage of incorporating the viscosity into the artificial compressor. Compared to Harten [10], Olsson and Kreiss [5] the formulation (10) allows for the combination of the transport step and the compression step. Another important aspect of the formulation is that it is fully conservative, i.e., assuming that $\nabla \cdot \mathbf{u} = 0$ and there is no flux at the boundary of the domain, the quantity $\int_{\Omega} \phi \, dx$ is constant over time. Finally, arriving at Eq. (10) does not require any heuristic reasoning, a particular flux $\mathbf{f}(\phi)$, or the a-priori knowledge of a particular solution such as the “phase field kernel”-motivated derivation of Chiu and Lin [6]. The theory of artificial compression methods of Harten [10,11] provides the provable conclusion that, since the compatibility condition (7) is satisfied, the viscous profile for the original problem, (5) with $\mu = 0$, and the model problem (10) coincide, and that as $\mu \downarrow 0$ these viscous profiles converge in the L^1 sense to the physically relevant solution.

4. Implementation of the method with finite elements

The artificial viscosity technique that we propose consists of three steps: (i) first we introduce a first-order maximum preserving artificial viscosity method, see Section 4.1; (ii) some of this viscosity is removed by considering an entropy residual, see Section 4.2; (iii) the compression technique described in (10) is applied, see Section 4.3. In computational practice the parameters presented in the formulation of the finite method, e.g. c_E and c_{comp} in Eqs. (18) and (20), are taken to be $c_E = c_{\text{comp}} = 1$.

4.1. Finite element-based first-order viscosity

We first describe the first-order viscosity method in the context of finite elements as introduced in Guermond and Popov [8]. Consider the following scalar conservation equation as model problem

$$\partial_t \phi(\mathbf{x}, t) + \nabla \cdot \mathbf{f}(\phi) = 0, \quad \mathbf{x} \in D, \quad t > 0, \tag{12}$$

where $D \subseteq \mathbb{R}^d$ is an open and connected domain and the initial data is given by $\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x})$. To simplify the presentation, we assume that the initial data or/and the flux \mathbf{f} are such that the flux at the boundary of D is zero. The Cauchy problem has

a unique entropy solution, i.e., a weak solution satisfying all the entropy inequalities $\partial_t E(\phi) + \nabla \cdot \mathbf{F}(\phi) \leq 0$ where $E(\phi)$ is any convex entropy function and $\mathbf{F}(\phi) = \int_0^\phi E'(v) \mathbf{f}'(v) dv$ is the associated entropy flux. For any unit vector \mathbf{n} and any pair of real numbers v_L, v_R we denote by $\lambda_{\max}(\mathbf{f}, \mathbf{n}, v_L, v_R)$ the fastest wave speed in the one-dimensional Riemann problem

$$\partial_t v + \partial_x(\mathbf{n} \cdot \mathbf{f}(v)) = 0 \quad v(x, 0) = \begin{cases} v_L & \text{if } x < 0 \\ v_R & \text{if } x > 0. \end{cases} \tag{13}$$

For instance $\lambda_{\max}(\mathbf{f}, \mathbf{n}, v_L, v_R) = |\mathbf{u} \cdot \mathbf{n}|$ if $\mathbf{f}(v) = \mathbf{u}v$.

Let $(\mathcal{T}_h)_{h>0}$ be a sequence of shape-regular matching meshes. Let $(\widehat{K}, \widehat{P}, \widehat{\Sigma})$ be a reference element. The shape functions on the reference element are denoted $\{\theta_i\}_{i \in \{1:n_{\text{sh}}\}}$. We assume that the basis $\{\widehat{\theta}_i\}_{i \in \{1:n_{\text{sh}}\}}$ has the following key properties: $\widehat{\theta}_i(\mathbf{x}) \geq 0$, $\sum_{i \in \{1:n_{\text{sh}}\}} \widehat{\theta}_i(\widehat{\mathbf{x}}) = 1$, for all $\widehat{\mathbf{x}} \in \widehat{K}$. These properties hold for linear Lagrange elements and for Bernstein–Bezier finite elements, see e.g., Schumaker [25, Chap. 2]. For any mesh cell $K \in \mathcal{T}_h$ we denote by $T_K : \widehat{K} \rightarrow K$ geometric map between the reference element \widehat{K} and the cell K . We define the scalar-valued space

$$P(\mathcal{T}_h) = \{v \in C^0(D; \mathbb{R}) \mid v|_K \circ T_K \in \widehat{P}, \forall K \in \mathcal{T}_h\}, \tag{14}$$

where \widehat{P} is the reference polynomial space. The global shape functions in $P(\mathcal{T}_h)$ are denoted by $\{\psi_i\}_{i \in \{1:I\}}$. Recall that these functions form a basis of $P(\mathcal{T}_h)$. We denote by S_i the support of ψ_i and by $\mathcal{I}(S_i)$ the collection of the indices of the shape functions whose support has an intersection with S_i of nonzero measure. We define $m_{ij} = \int_D \psi_i \psi_j dx$ and $m_i = \int_D \psi_i dx$. Let M^C denote the consistent mass matrix, with entries m_{ij} , and M^L denote the lumped mass matrix with entries m_i .

We now describe the explicit first-order maximum principle preserving method from [8]. The method is described with explicit Euler time stepping, but higher-order accuracy in time can be achieved by using any strong stability preserving (SSP) method (see Gottlieb et al. [26] for a review on SSP methods). Let $\phi_h^n = \sum_{i \in \{1:I\}} \Phi_i^n \psi_i^n$ be the approximation of ϕ at time t^n . For any pair $i, j \in \{1 : I\}$ we define the vector $\mathbf{c}_{ij} := \int_D \psi_i \nabla \psi_j dx$ and the unit vector $\mathbf{n}_{ij} := \mathbf{c}_{ij} / \|\mathbf{c}_{ij}\|_{\ell^2}$ where $\|\cdot\|_{\ell^2}$ denotes the Euclidean norm. We now introduce the artificial viscosity coefficients

$$d_{ij}^n = \max(\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \Phi_i^n, \Phi_j^n) \|\mathbf{c}_{ij}\|_{\ell^2}, \lambda_{\max}(\mathbf{f}, \mathbf{n}_{ji}, \Phi_j^n, \Phi_i^n) \|\mathbf{c}_{ji}\|_{\ell^2}), \quad i \neq j. \tag{15}$$

The first-order approximation of ϕ at time t^{n+1} is then defined by

$$m_i \frac{\Phi_i^{n+1} - \Phi_i^n}{\tau} + \sum_{j \in \mathcal{I}(S_i)} (\mathbf{c}_{ij} \cdot (\mathbf{f}(\Phi_j^n) - \mathbf{f}(\Phi_i^n)) - d_{ij}^n (\Phi_j^n - \Phi_i^n)) = 0. \tag{16}$$

Theorem 1 (Guermond and Popov [8]). *The scheme (15)–(16) is maximum principle preserving under CFL condition for any Lipschitz flux \mathbf{f} .*

4.2. Entropy viscosity

We describe the entropy-viscosity method in this section. Let $\widetilde{\phi}_h^{n+1} = \sum_{i \in \{1:I\}} \widetilde{\Phi}_i^n \psi_i$ be the Galerkin approximation of ϕ at t^{n+1} , i.e., $\sum_{j \in \mathcal{I}(S_i)} m_{ij} \widetilde{\Phi}_j^{n+1} = \sum_{j \in \mathcal{I}(S_i)} m_{ij} \Phi_j^n - \frac{\tau}{m_i} \sum_{j \in \mathcal{I}(S_i)} \mathbf{c}_{ij} \cdot (\mathbf{f}(\Phi_j^n) - \mathbf{f}(\Phi_i^n))$. Let E be an entropy. For any pair of indices $i, j \in \{1 : I\}$ such that $\text{int}(S_i \cap S_j) \neq \emptyset$, we define the normalized entropy residual

$$R_i^n = \frac{1}{E_i^{\max} - E_i^{\min}} \int_D \left(\frac{\widetilde{\phi}_h^{n+1} - \phi_h^n}{\tau} + \mathbf{f}'(\phi_h^n) \cdot \nabla \phi_h^n \right) E'(\phi_h^n) \psi_i dx, \quad \forall i \in \{1 : I\}, \tag{17}$$

where $E_i^{\max} = \max_{j \in \mathcal{I}(S_i)} |E(\Phi_j^n)|$ and $E_i^{\min} = \min_{j \in \mathcal{I}(S_i)} |E(\Phi_j^n)|$. (In all the applications shown in Sections 7–12 we use the entropy $E(\Phi) = -\log(|\Phi(1 - \Phi)| + \epsilon)$ with $\epsilon = 1 \times 10^{-14}$ and $0 \leq \Phi \leq 1$.) Note that the entropy residual is zero when $E(\phi) = \phi$; hence this regularization is useful only for nonlinear entropies. Then the entropy viscosity coefficients are defined by

$$d_{ij}^{E,n} = \min(d_{ij}^n, c_E \max(|R_i^n|, |R_j^n|)), \quad i \neq j, \tag{18}$$

where c_E is a user-defined non-dimensional parameter. Extensive numerical experiments have shown that $c_E = 1$ may be taken. The entropy-viscosity solution is obtained by replacing d_{ij}^n by $d_{ij}^{E,n}$ and by replacing the lumped mass matrix by the consistent mass matrix in (16):

$$\sum_{j \in \mathcal{I}(S_i)} m_{ij} \frac{\Phi_j^{n+1} - \Phi_j^n}{\tau} + \sum_{j \in \mathcal{I}(S_i)} (\mathbf{c}_{ij} \cdot (\mathbf{f}(\Phi_j^n) - \mathbf{f}(\Phi_i^n)) - d_{ij}^{E,n} (\Phi_j^n - \Phi_i^n)) = 0. \tag{19}$$

Remark 4.1. The consistent mass matrix is used in (19) to approximate the time derivative. This detail is important since it eliminates the second-order dispersion errors; we refer to Christon et al. [27], Guermond and Pasquetti [28], Thompson [29] for details.

Section 6 discusses the extension of (19) to a maximum preserving scheme. More on the entropy-viscosity technique, its attributes, and its applications are discussed in [7,30,12,31].

4.3. Artificial compression

Let us now assume that the level set function ϕ is in the range $[0, 1]$, i.e., $\Phi_L, \Phi_R \in \{0, 1\}$ in (10). Then, following the principles of the artificial compression technique broached in Section 3, the entropy viscosity is refined by setting $\Phi_{ij}^n := \frac{1}{2}(\Phi_i^n + \Phi_j^n)$ and

$$d_{ij}^{C,n} = d_{ij}^{E,n} \left[1 - c_{\text{comp}} \frac{[\Phi_{ij}^n(1 - \Phi_{ij}^n)]^+}{|\Phi_i^n - \Phi_j^n|} \right]^+ \tag{20}$$

where c_{comp} is a user-defined parameter. It is demonstrated numerically that this parameter can be taken to be 1 without any noticeable drawbacks. In all the finite element applications shown in Sections 7–12 we use $c_{\text{comp}} = 1$. Using the positive part of the correction, in (20) avoids the introduction of any negative viscosity. The approximate level set solution is obtained by replacing $d_{ij}^{E,n}$ by $d_{ij}^{C,n}$ in (19):

$$\sum_{j \in \mathcal{I}(S_i)} m_{ij} \frac{\Phi_j^{n+1} - \Phi_j^n}{\tau} + \sum_{j \in \mathcal{I}(S_i)} (\mathbf{c}_{ij} \cdot (\mathbf{f}(\Phi_j^n) - \mathbf{f}(\Phi_i^n)) - d_{ij}^{C,n}(\Phi_j^n - \Phi_i^n)) = 0. \tag{21}$$

Finally, in the process of conducting numerical computations, the use of the anti-diffusion (20) to reduce the artificial viscosity produced no notable difference in the standard CFL restriction.

5. Implementation of the method with finite volumes

We now describe how the method can be implemented with finite volumes method on Cartesian grids. For brevity we restrict ourselves to two space dimensions and to uniform grids. The extension to nonuniform grids and to three space dimensions is evident. For the finite volume method, the parameters presented in the formulation described in this section, see e.g. Eqs. (25) and (29), are $c_E = \frac{1}{10}$, and $c_{\text{comp}} = \frac{2}{5}$.

5.1. Finite volumes and first-order viscosity

The dimensions of the cell (i, j) are denoted h_x and h_y and the mass of the cell is $m_{ij} = h_x h_y$. The dependent variable is cell-centered, $\phi_{i,j}$, and the fluxes are face-centered. Let $\Gamma_{i \pm \frac{1}{2}}^n, \Gamma_{j \pm \frac{1}{2}}^n$ be the four faces of the cell (i, j) with unit outward normals $\mathbf{n}_{i \pm \frac{1}{2}}^n$ and $\mathbf{n}_{i, j \pm \frac{1}{2}}^n$ and measure h_y and h_x , respectively. The flux on these faces is denoted $\mathbf{n}_{i \pm \frac{1}{2}}^n \cdot \frac{1}{2}(\mathbf{f}(\phi_{i \pm 1, j}) + \mathbf{f}(\phi_{i, j}))$ and $\mathbf{n}_{i, j \pm \frac{1}{2}}^n \cdot \frac{1}{2}(\mathbf{f}(\phi_{i, j \pm 1}) + \mathbf{f}(\phi_{i, j}))$. To make the similarities with the finite element algorithm more apparent, we define a global set of indices by setting $\alpha := (i, j)$ and $\Phi_\alpha := \phi_{i, j}$. The mass of the cell α is $m_\alpha = h_x h_y$ and we define the lumped mass matrix M^L to be the diagonal matrix with entries m_α . We introduce the set of neighboring indices, $\mathcal{I}(S_\alpha) := \{(i, j), (i \pm 1, j), (i, j \pm 1)\}$. We then set

$$\mathbf{c}_{\alpha\beta} := \begin{cases} \frac{h_y}{2} \mathbf{n}_{i \pm \frac{1}{2}, j} & \text{if } \beta = (i \pm 1, j) \\ \frac{h_x}{2} \mathbf{n}_{i, j \pm \frac{1}{2}} & \text{if } \beta = (i, j \pm 1). \end{cases}$$

Notice that $\sum_{\alpha \neq \beta \in \mathcal{I}(S_\alpha)} \mathbf{c}_{\alpha\beta} = 0$ for any α . Taking inspiration from (15), we set

$$d_{\alpha\beta}^n = \max(\lambda_{\max}(\mathbf{f}, \mathbf{n}_{\alpha\beta}, \Phi_\alpha^n, \Phi_\beta^n) \|\mathbf{c}_{\alpha\beta}\|_{\ell^2}, \lambda_{\max}(\mathbf{f}, \mathbf{n}_{\beta,\alpha}, \Phi_\beta^n, \Phi_\alpha^n) \|\mathbf{c}_{\beta\alpha}\|_{\ell^2}), \quad \alpha \neq \beta. \tag{22}$$

The first-order finite volume approximation of ϕ at time t^{n+1} is defined by

$$m_\alpha \frac{\Phi_\alpha^{n+1} - \Phi_\alpha^n}{\tau} + \sum_{\beta \in \mathcal{I}(S_\alpha)} (\mathbf{c}_{\alpha\beta} \cdot (\mathbf{f}(\Phi_\beta^n) - \mathbf{f}(\Phi_\alpha^n)) - d_{\alpha\beta}^n(\Phi_\beta^n - \Phi_\alpha^n)) = 0. \tag{23}$$

Note that the arguments from Guermond and Popov [8] are still valid in this case; hence, the scheme (22)–(23) is maximum principle preserving, under CFL condition, for any Lipschitz flux.

5.2. Entropy viscosity

Since it is known that the use of the consistent mass matrix corrects the second-order dispersive error, see Christon [32], Guermond and Pasquetti [28], Thompson [29], we introduce the consistent matrix M^C with coefficients

$$m_{\alpha\beta} = m_\alpha \begin{cases} \frac{1}{6} & \text{if } \alpha \neq \beta \in \mathcal{I}(S_\alpha) \\ 1 - \frac{2d}{6} & \text{if } \alpha = \beta, \end{cases} \tag{24}$$

where d is the space dimension; here $d = 2$. To compute the entropy residual as in the finite element algorithm. First, we estimate the solution at t^{n+1} without viscosity, i.e., $\sum_{\beta \in \mathcal{I}(S_\alpha)} m_{\alpha\beta} \tilde{\Phi}_\beta^{n+1} = \sum_{\beta \in \mathcal{I}(S_\alpha)} m_{\alpha\beta} \Phi_\beta^n - \frac{\tau}{m_\alpha} \sum_{\beta \in \mathcal{I}(S_\alpha)} \mathbf{c}_{\alpha\beta} \cdot (\mathbf{f}(\Phi_\beta^n) - \mathbf{f}(\Phi_\alpha^n))$. Let E be an entropy. (In all the applications shown in Sections 7–12 we use the entropy $E(\Phi) = -\log(|\Phi(1-\Phi)| + \epsilon)$ with $\epsilon = 1 \times 10^{-14}$ and $0 \leq \Phi \leq 1$.) We define the entropy residual at α by $R_\alpha^n := \sum_{\beta \in \mathcal{I}(S_\alpha)} \frac{m_{\alpha\beta}}{\tau} (E(\tilde{\Phi}_\beta^{n+1}) - E(\Phi_\beta^n)) + \sum_{\beta \in \mathcal{I}(S_\alpha)} \mathbf{c}_{\alpha\beta} \cdot \mathbf{f}'(\Phi_\alpha^n) (E(\Phi_\beta^n) - E(\Phi_\alpha^n))$. Then we define the entropy viscosity by

$$d_{\alpha\beta}^{E,n} = \min(d_{\alpha\beta}^n, c_E \max(|R_\alpha^n|, |R_\beta^n|)), \quad \alpha \neq \beta. \tag{25}$$

The entropy viscosity solution is obtained by replacing $d_{\alpha\beta}^n$ by $d_{\alpha\beta}^{E,n}$ and by using the consistent mass matrix:

$$\sum_{\beta \in \mathcal{I}(S_\alpha)} m_{\alpha\beta} \frac{\Phi_\beta^{n+1} - \Phi_\beta^n}{\tau} + \sum_{\beta \in \mathcal{I}(S_\alpha)} (\mathbf{c}_{\alpha\beta} \cdot (\mathbf{f}(\Phi_\beta^n) - \mathbf{f}(\Phi_\alpha^n)) - d_{\alpha\beta}^{E,n} (\Phi_\beta^n - \Phi_\alpha^n)) = 0. \tag{26}$$

In the massively parallel code that we developed and which is documented in Guermond and Minev [33], the mass matrix is inverted approximately by replacing $(M^C)^{-1}$ by $(1 - (M^L)^{-1}(M^C - M^L))(M^L)^{-1}$.

5.3. Consistent mass matrix

To illustrate our point regarding the use of the consistent mass matrix in the finite volumes context, consider the semi-discrete version of the one-dimensional transport equation $\partial_t \phi + u \partial_x \phi = 0$ on a uniformly spaced mesh with constant velocity u :

$$\partial_t \phi_i + \frac{u}{2h} (\phi_{i+1} - \phi_{i-1}) = 0. \tag{27}$$

The Taylor analysis shows that the exact solution satisfies $\partial_t \phi(x_i) + \frac{u}{2h} (\phi(x_{i+1}) - \phi(x_{i-1})) + u \frac{h^2}{6} \partial_{xxx} \phi(x_i) + O(h^4) = 0$. The leading truncation error term is therefore dispersive. Using the consistent mass matrix approach couples the time derivative at neighboring cell centers and yields the discretization:

$$\partial_t \phi_i + \frac{1}{6} (\partial_t \phi_{i-1} - 2\partial_t \phi_i + \partial_t \phi_{i+1}) + \frac{\beta}{2h} (\phi_{i+1} - \phi_{i-1}) = 0. \tag{28}$$

The consistency analysis based on Taylor expansion shows that the exact solution satisfies $\partial_t \phi(x_i) + \frac{1}{6} (\partial_t \phi(x_{i-1}) - 2\partial_t \phi(x_i) + \partial_t \phi(x_{i+1})) \approx \partial_t \phi(x_i) - \beta \frac{h^2}{6} \partial_{xxx} \phi(x_i) + O(h^4)$ so that the dispersive error term of (27) is offset in (28); hence (28) is a better approximation than (27), see Guermond and Pasquetti [28, Prop 2.2]. The effect of the consistent mass matrix (24) is to correct the dispersion error along the x and y directions.

5.4. Artificial compression

Finally the compression is realized exactly as in the finite element case. We set $\Phi_{\alpha\beta}^n := \frac{1}{2}(\Phi_\alpha^n + \Phi_\beta^n)$ and the corrected artificial viscosity is defined by

$$d_{\alpha\beta}^{C,n} = d_{\alpha\beta}^{E,n} \left[1 - c_{\text{comp}} \frac{[\Phi_{\alpha\beta}^n (1 - \Phi_{\alpha\beta}^n)]^+}{|\Phi_\alpha^n - \Phi_\beta^n|} \right]^+, \tag{29}$$

where c_{comp} is a user-defined non-dimensional parameter. The approximate level set solution is obtained by replacing $d_{\alpha\beta}^{E,n}$ by $d_{\alpha\beta}^{C,n}$ in (26):

$$\sum_{\beta \in \mathcal{I}(S_\alpha)} m_{\alpha\beta} \frac{\Phi_\beta^{n+1} - \Phi_\beta^n}{\tau} + \sum_{\beta \in \mathcal{I}(S_\alpha)} (\mathbf{c}_{\alpha\beta} \cdot (\mathbf{f}(\Phi_\beta^n) - \mathbf{f}(\Phi_\alpha^n)) - d_{\alpha\beta}^{C,n} (\Phi_\beta^n - \Phi_\alpha^n)) = 0. \tag{30}$$

In order to illustrate that the method is not too much sensitive to the choice of the parameters c_E and c_{comp} , all the finite volume computations reported in Sections 7–12 have been done with $c_E = \frac{1}{10}$, and $c_{\text{comp}} = \frac{2}{5}$ (see (25) and (29)).

6. An extension of the method to preserve the maximum principle

The low-order method described in Section 4.1 preserves the discrete maximum principle locally; that is to say, given the solution Φ_h^n at time t^n , the low-order solution at time t^{n+1} satisfies

$$\min_{j \in \mathcal{I}(S_i)} \Phi_j^n =: \Phi_i^{\min} \leq \Phi_i^{n+1} \leq \Phi_i^{\max} := \max_{j \in \mathcal{I}(S_i)} \Phi_j^n,$$

but there is no guarantee that the solution given by either (21) or (30) has this property. We describe in this section an adaption of the Flux Corrected Transport (FCT) methodology from Boris and Book [34] and Zalesak [35] to the compression method described in Section 4.3 and Section 5.4. We refer to Kuzmin et al. [36] for more details and to Guermond et al. [31] for an example on the FCT method using continuous Galerkin finite elements.

For convenience we rewrite the low- and the high-order methods (16) and (21) (or (23) and (26) in the finite volume context) as follows:

$$M^L \left(\frac{\Phi^L - \Phi^n}{\tau} \right) + K(\Phi^n) + D^L \Phi^n = 0, \tag{31a}$$

$$M^C \left(\frac{\Phi^H - \Phi^n}{\tau} \right) + K(\Phi^n) + D^C \Phi^n = 0, \tag{31b}$$

where M^L and M^C are the lumped and consistent mass matrices, respectively, K is the column vector with entries $\sum_{j \in \mathcal{I}(S_i)} c_{ij} \cdot (\mathbf{f}(\Phi_j^n) - \mathbf{f}(\Phi_i^n))$, D^L is the first-order viscosity matrix and D^C is the high-order compressive viscosity matrix. The entries of D^L are d_{ij}^L as defined in (15) (or (22) in the finite volume context), with the convention that $d_{ii}^L = -\sum_{j \neq i} d_{ij}^L$. The entries of D^C are $d_{ij}^{C,n}$ as defined in (20) (or (29) in the finite volume context) with the convention that $d_{ii}^{C,n} = -\sum_{j \neq i} d_{ij}^{C,n}$. Note that both D^L and D^C are symmetric matrices. The high-order method can be rewritten as follows:

$$M^L(\Phi^H - \Phi^L) = (M^L - M^C)(\Phi^H - \Phi^n) + \tau(D^L - D^C)\Phi^n. \tag{32}$$

Note that for any $i = 1, \dots, N$, we have $\sum_j (M^L - M)_{ij} = 0$ by definition of the lumped mass matrix, and $(D^L - D^C)_{ii} = -\sum_{j \neq i} (D^L - D^C)_{ij}$, by definition of the diagonal terms in D^L and D^C . Therefore,

$$\begin{aligned} [(D^L - D^C)\Phi^n]_i &= \sum_j (D^L - D^C)_{ij} \Phi_j^n = \sum_{j \neq i} (D^L - D^C)_{ij} \Phi_j^n + (D^L - D^C)_{ii} \Phi_i^n \\ &= \sum_{j \neq i} (D^L - D^C)_{ij} (\Phi_j^n - \Phi_i^n) = \sum_j (D^L - D^C)_{ij} (\Phi_j^n - \Phi_i^n), \end{aligned}$$

and since $D^L - D^C$ is symmetric, the matrix with entries $(D^L - D^C)_{ij} (\Phi_j^n - \Phi_i^n)$ is skew-symmetric. Similarly, $\sum_j (M^L - M)_{ij} (\Phi_j^n - \Phi_i^n) = \sum_j (M^L - M)_{ij} (\delta \Phi_j - \delta \Phi_i)$, where $\delta \Phi := \Phi^H - \Phi^n$, and again the matrix with entries $(M^L - M)_{ij} (\delta \Phi_j - \delta \Phi_i)$ is skew-symmetric. Let us then introduce the so-called flux correction matrix F with entries

$$f_{ij} := (M^L - M)_{ij} (\delta \Phi_j - \delta \Phi_i) + \tau (D^L - D^C)_{ij} (\Phi_j^n - \Phi_i^n).$$

The above arguments show that $f_{ij} = -f_{ji}$, i.e., F is skew-symmetric. Then the update for the high-order solution (32) can be rewritten as

$$\Phi_i^H = \Phi_i^L + m_i^{-1} \sum_{j \in \mathcal{S}_i} f_{ij}. \tag{33}$$

This equality shows that the flux correction matrix improves the accuracy of the low-order method, but it is also responsible for the high-order solution to possibly jump out of bounds; i.e., the high-order solution may violate the maximum principle. The idea behind FCT is to limit the flux correction when it induces a violation of the maximum principle. Following Zalesak [35] (see Eq. (10)–(13) [35]) we introduce a symmetric flux limiter matrix α , with entries α_{ij} , and we compute the high-order update as follows:

$$\Phi_i^{n+1} = \Phi_i^L + m_i^{-1} \sum_{j \in \mathcal{S}_i} \alpha_{ij} f_{ij}, \tag{34}$$

where the entries of the flux limiter matrix are computed as follows:

$$\alpha_{ij} := \begin{cases} \min(R_i^+, R_j^-) & \text{if } f_{ij} \geq 0 \\ \min(R_i^-, R_j^+) & \text{otherwise,} \end{cases} \tag{35a}$$

where

$$R_i^+ := \begin{cases} \min\left(1, \frac{Q_i^+}{P_i^+}\right) & \text{if } P_i^+ \neq 0 \\ 1 & \text{otherwise,} \end{cases} \quad R_i^- := \begin{cases} \min\left(1, \frac{Q_i^-}{P_i^-}\right) & \text{if } P_i^- \neq 0 \\ 1 & \text{otherwise,} \end{cases} \tag{35b}$$

$$P_i^+ := \sum_j \max(0, f_{ij}), \quad P_i^- := \sum_j \min(0, f_{ij}), \tag{35c}$$

$$Q_i^+ := m_i(\Phi_i^{\max} - \Phi_i^L), \quad Q_i^- := m_i(\Phi_i^{\min} - \Phi_i^L). \tag{35d}$$

Remark 6.1 (*Maximum Principle*). Assume that Φ^L satisfies the local discrete maximum principle; i.e., $\Phi_i^{\min} \leq \Phi_i^L \leq \Phi_i^{\max}$ for all $i = 1, \dots, N$. Then the solution of (34) satisfies the local discrete maximum principle; i.e., $\Phi_i^{\min} \leq \Phi_i^{n+1} \leq \Phi_i^{\max}$ for all $i = 1, \dots, N$. We prove this statement by following Guermond et al. [31] (see also Kuzmin et al. [36, p. 182]). Assume that $P_i^+ \neq 0$, then using (35) we obtain

$$\begin{aligned} m_i(\Phi_i^{n+1} - \Phi_i^L) &= \sum_j \alpha_{ij} f_{ij} \leq \sum_{j:f_{ij} \geq 0} \alpha_{ij} f_{ij} = \sum_{j:f_{ij} \geq 0} \min(R_i^+, R_j^-) f_{ij} \leq \sum_{j:f_{ij} \geq 0} R_i^+ f_{ij} \\ &\leq \frac{Q_i^+}{P_i^+} \sum_{j:f_{ij} \geq 0} f_{ij} = \frac{Q_i^+}{P_i^+} \sum_{j \in \delta_i} \max(0, f_{ij}) = Q_i^+ = m_i(\Phi_i^{\max} - \Phi_i^L); \end{aligned}$$

therefore, $\Phi_i^{n+1} \leq \Phi_i^{\max}$. If $P_i^+ = 0$, then

$$m_i(\Phi_i^{n+1} - \Phi_i^L) \leq \sum_{j:f_{ij} \geq 0} R_i^+ f_{ij} = \sum_{j:f_{ij} \geq 0} f_{ij} = P_i^+ = 0,$$

and, provided $\Phi_i^L \leq \Phi_i^{\max}$, we infer that $P_i^+ = 0 \leq m_i(\Phi_i^{\max} - \Phi_i^L)$, which implies $\Phi_i^{n+1} \leq \Phi_i^{\max}$. The lower bound $\Phi_i^{\min} \leq \Phi_i^{n+1}$ is proven similarly. \square

Remark 6.2 (*Mass Conservation*). The method (34) is mass conservative in the following sense:

$$\int_{\Omega} \phi_h^n(\mathbf{x}) \, d\mathbf{x} = \int_{\Omega} \phi_h^0(\mathbf{x}) \, d\mathbf{x}, \quad \forall n \geq 0.$$

To establish this property, consider the row sum of (34) and use the properties $\alpha_{ij} = \alpha_{ji}$ and $f_{ij} = -f_{ji}$ to see that:

$$\sum_i m_i(\Phi_i^{n+1} - \Phi_i^L) = \sum_i \sum_{j \in \delta_i} \alpha_{ij} f_{ij} = \sum_{i,j} \alpha_{ij} f_{ij} + \alpha_{ji} f_{ji} = \sum_{i,j} \alpha_{ij} (f_{ij} - f_{ij}) = 0. \quad \square$$

7. A one-dimensional validation

To directly illustrate the effectiveness of the entropy viscosity stabilization the one-dimensional linear transport equation, $\partial_t \phi + \partial_x \phi = 0$, over the periodic domain $(0, 1)$ is solved. Piecewise linear \mathbb{P}_1 finite elements and the strong stability preserving SSPRK(3,3) version of the algorithm (21) (together with (15), (18) and (20)) are utilized. The initial data is $\phi_0(x) = 1$ if $0.4 \leq x \leq 0.7$ and $\phi_0(x) = 0$ otherwise. We choose the entropy to be $E(\phi) = -\log(|\phi(1 - \phi)| + \epsilon)$ with $\epsilon = 10^{-14}$. The mesh is composed of 100 cells, i.e., 100 grid points. Three solutions are computed at $T = 1, T = 10$, and $T = 100$ in each panel.

We show in the left panel of Fig. 1 the solutions obtained by using the first-order viscosity only, i.e., the scheme (15)–(16). The method is monotonic but very diffusive; actually it is $\mathcal{O}(h^{\frac{1}{2}})$ accurate in the L^1 -norm with this particular initial data. The solution computed with the entropy viscosity is shown in the central panel, i.e., the scheme (18)–(19) with $c_E = 1$. The superiority of the entropy viscosity method over the first-order viscosity solution is clear. Note that, although the entropy viscosity is mainly localized in the two regions where the graph of the solution goes from 0 to 1, the dissipation accumulates in time and the graph of the approximate solution is eventually flattened. We show in the right panel of the figure the solutions obtained by using the compression technique, i.e., the scheme (20)–(21) with $c_{\text{comp}} = 1$. The effects of the entropy viscosity and the compression are evident.

8. A rotating disk in two dimensions

We now test the method on the linear transport equation $\partial_t \phi + \mathbf{u} \cdot \nabla \phi = 0$ using three numerical methods: the first one uses second-order finite volumes on Cartesian meshes, the second one uses unstructured \mathbb{P}_1 finite elements, the third one uses \mathbb{Q}_1 finite elements. The computational domain is the disk $\Omega = \{\mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x}\|_{\ell^2} \leq 1\}$ and the velocity is the solid

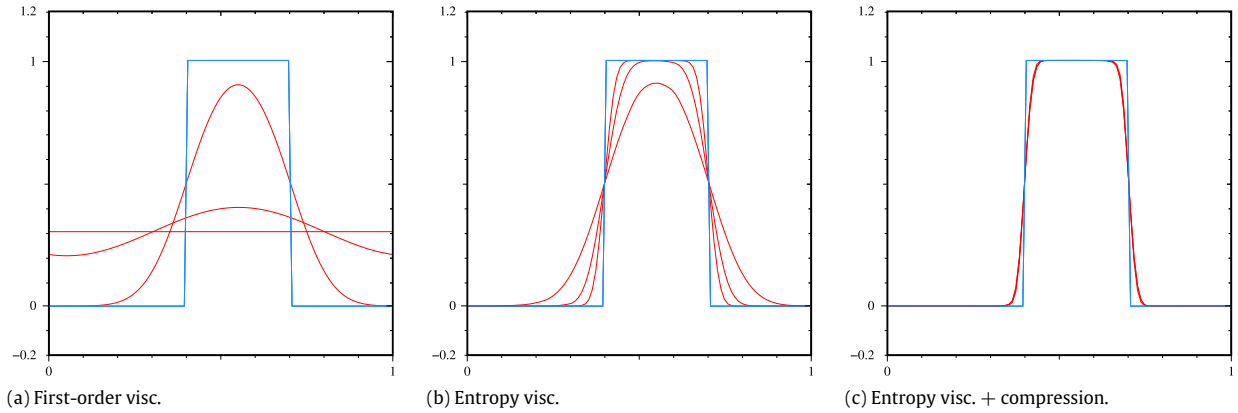


Fig. 1. Linear transport: exact and approximate solutions at $T = 1, 10, 100$ using 100 grid points.

rotation about the origin of angular velocity 2π , i.e., $\mathbf{u} = 2\pi(-y, x)$. The initial data is $\phi_0(\mathbf{x}) = 1$ if $\|\mathbf{x} - \mathbf{x}_0\|_{\ell^2} \leq 0.25$ with $\mathbf{x}_0 = (\frac{1}{2}, 0)$ and $\phi_0(x) = 0$ otherwise. We choose the entropy to be $E(\phi) = -\log(|\phi(1 - \phi)| + \epsilon)$, $\epsilon = 10^{-14}$.

8.1. Finite volumes

We start with second-order finite volumes on a Cartesian grid with the SSPRK(3,3) version of the algorithm (30) (together with (22), (25), (29)). An important capability of a level set algorithm for conservative two-phase flow, as mentioned in Olsson and Kreiss [5], is the demonstration of sustained duration conservation of area enclosed by the interface level set for a normative test case. In order to demonstrate the long-term conservative properties of the compression method, we use two error indicators introduced in Olsson and Kreiss [5], Enright et al. [2] to measure interface differences between the exact solution ϕ and the approximate solution ϕ_h . These are the interface error defined in (36) and the area conservation error defined in (37):

$$I_{\text{Err}}(\phi_h) = \frac{1}{L} \int_{\Omega} |H(\phi - \frac{1}{2}) - H(\phi_h - \frac{1}{2})| dx \tag{36}$$

$$A_{\text{Err}}(\phi_h) = \frac{1}{\int_{\Omega} H(\phi - \frac{1}{2}) dx} \left| \int_{\Omega} H(\phi - \frac{1}{2}) - H(\phi_h - \frac{1}{2}) dx \right|. \tag{37}$$

The quantity L is the measure of the interface level set $\{\phi = \frac{1}{2}\}$ (a length in 2D and a surface area in 3D) and $H(\cdot)$ is the Heaviside function. We test the finite volume method by letting the initial data revolve through thirty two full revolutions. The test was conducted for various discretization levels; the results are reported in Table 1. The average area conservation error is plotted for each discretization level. The computations are done with fixed time step $\Delta t = 3.25 \times 10^{-4}$ and the conservation of area error (37) was recorded every three hundred time steps. The average order of area convergence for the finite volume method is seen to be 1.82; see Table 1. An analysis of the left panel in Table 1 shows that the drift in area conservation error is substantial for $h \approx 2 \times 10^{-2}$ and nears 15% over thirty revolutions. Conversely, for $h \approx 4.95 \times 10^{-3}$ the area difference varies by only about 1% over thirty revolutions and $h \approx 2.488 \times 10^{-3}$ sees a variation in the area enclosed by the interface of less than 0.3% over thirty turns.

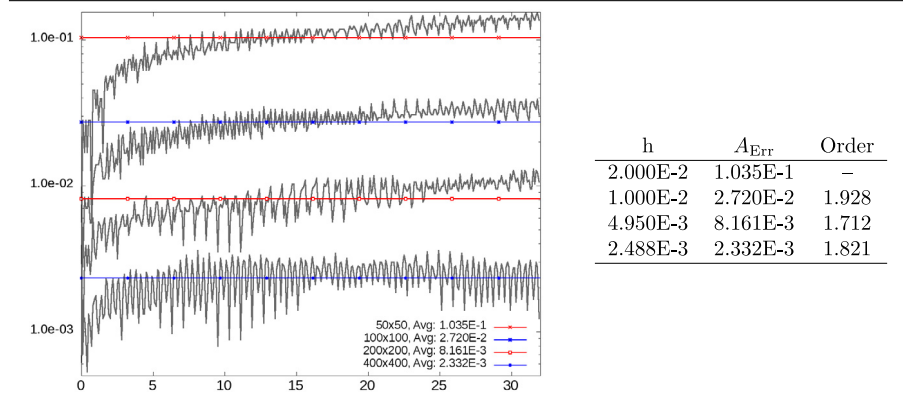
A comparison of Tables 1, 3, and 4 exemplifies the strong similarities between the three computational approaches examined; i.e., \mathbb{P}_1 finite elements, \mathbb{Q}_1 finite elements, and finite volumes.

8.2. \mathbb{P}_1 finite elements

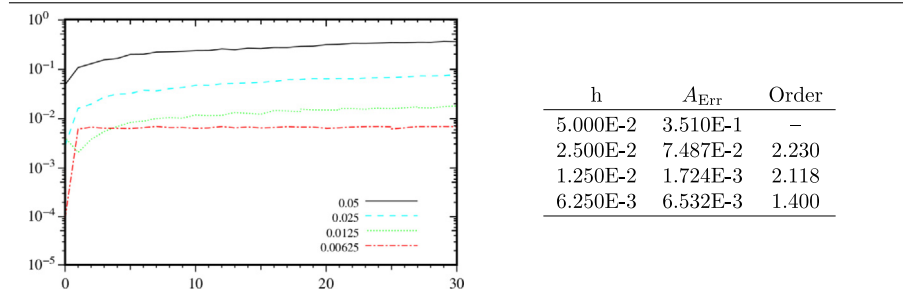
We now test the method with continuous piecewise linear finite elements on triangular Delaunay meshes and the SSPRK(3,3) version of the algorithm (21) (together with (15), (18), (20)). We systematically set $c_E = 1$ for the entropy viscosity coefficient. We show in Table 2 convergence tests on various nonuniform meshes of characteristic size $h = 0.1, 0.05, 0.025, 0.0125, 0.01$. We give the relative L^1 -norm of the error at $T = 1$, i.e., after one revolution. The table shows results without compression, $c_{\text{comp}} = 0$, and with compression, $c_{\text{comp}} = 1$. Note that for this problem the maximum convergence order achievable is 1 since the solution has only bounded variation regularity. We observe that the convergence order of the method with compression is close to 0.88 whereas it is close to 0.78 without compression. The difference is small, but this test shows that the method with compression performs better over long time simulations. We show in Fig. 2 contour lines of the solution at $T = 1$ on a mesh composed of 76 996 \mathbb{P}_1 finite elements; the typical meshsize is $h = 0.01$. The contour lines shown are $\phi \in \{0.1, 0.2, \dots, 0.9\}$. The left panel shows the first-order solution without compression; the center panel shows the entropy-viscosity solution; the right panel shows the entropy-viscosity solution with compression.

Table 1

Area conservation error vs. time (left) and average over 32 revolutions (right); Finite volumes.

**Table 2**Convergence tests using \mathbb{P}_1 elements. L^1 -norm of error at $T = 1$.

h	$c_{\text{comp}} = 1$	Order	$c_{\text{comp}} = 0$	Order
1.00E-1	6.768E-1		7.612E-1	
5.00E-2	3.704E-1	0.87	4.350E-1	0.81
2.50E-2	2.034E-1	0.87	2.529E-1	0.78
1.25E-2	1.142E-1	0.83	1.469E-1	0.78
1.00E-2	9.380E-2	0.88	1.230E-1	0.80

Table 3Area conservation error vs. time (left) and average over 30 revolutions (right); \mathbb{P}_1 finite elements.

The effect of the compression is again clear when comparing the center and right panels. Finally we do the area conservation test over 30 revolutions with $\text{CFL} = 0.25$. The area error was recorded at the end of every revolution. The results are shown in Table 3. The convergence rate of the finite element method on A_{Err} is between 1.5 and 2.2. The long time behavior of the method is similar to that of the finite volume method.

8.3. \mathbb{Q}_1 finite elements

We test the FCT version of the method with continuous \mathbb{Q}_1 finite elements on quadrangles using again the SSPRK(3,3) version of the algorithm (21) (together with (15), (18), (20)). The code is based on deal.ii, and is written in C++, see Bangerth et al. [37] for details.

We show in Table 4 the convergence of the L^1 -norm of the error with respect to the mesh size after one revolution. We also show in this table the behavior of interface error (36) and the area error (37) with respect to the mesh size.

We show in Fig. 3 contour lines of the solution at $T = 1$ on a mesh of mesh size $h = 1 \times 10^{-2}$. The contour lines are $\phi \in \{0.1, 0.2, \dots, 0.9\}$. The left panel shows the contour lines without compression, i.e., with $c_{\text{comp}} = 0$, and the right panel shows the contour lines with compression, i.e., with $c_{\text{comp}} = 1$. The effects of the compression are evident. In the right panel of the figure we show the graph of the solution along the cross section $x = 0.5$. In this figure we compare the solution obtained by using the FCT method, i.e., (34), and that without FCT, i.e., (33). This figure effectively illustrates the FCT method's ability to produce a solution which is free of under-shoots and over-shoots.

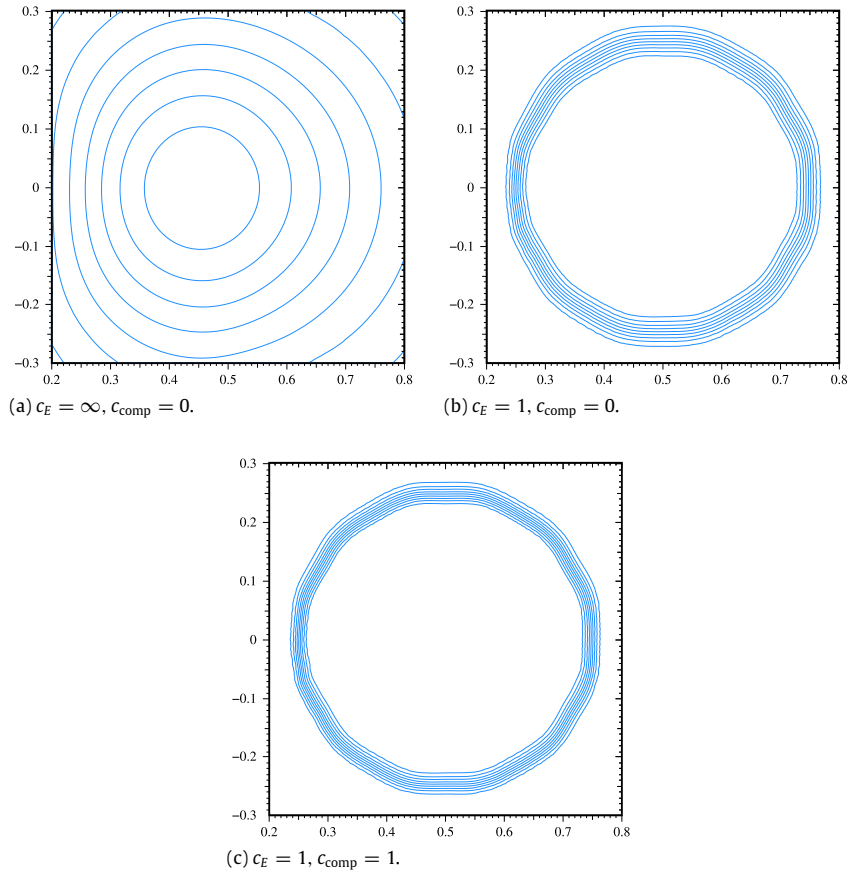


Fig. 2. 2D transport, solid rotation, \mathbb{P}_1 finite elements, effect of entropy viscosity and compression.

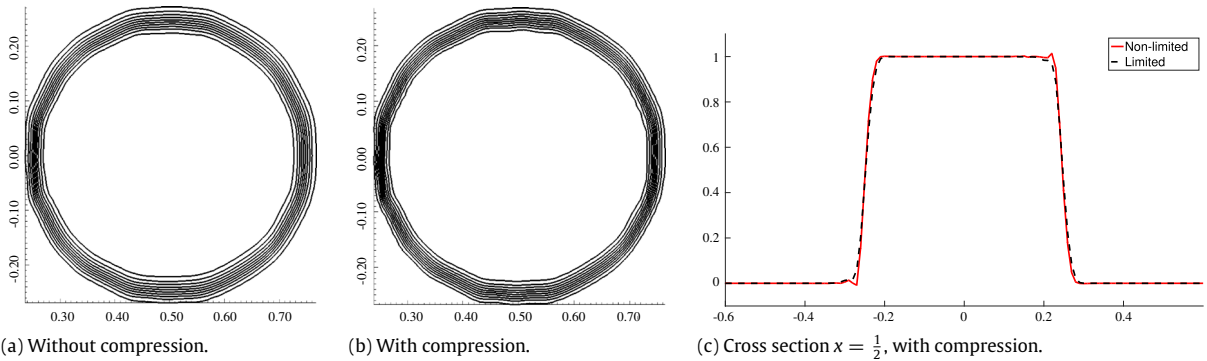


Fig. 3. (a) and (b) contour lines after one revolution using \mathbb{Q}_1 elements with the FCT method and $h = 1 \times 10^{-2}$. (c) Cross section at $x = 0.5$ of the graph of the solution using FCT (in dotted black) and without FCT (in solid red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

9. A Zalesak disk problem in two dimensions

We now consider a test first proposed in Zalesak [35]. The initial data is the characteristic function of a disc of radius $r = 0.15$ centered at the point $\mathbf{x}_0 := (0.5, 0.75)$ with a thin rectangular volume removed; for this test case the removed volume is $\{\mathbf{x} = (x, y) \in \mathbb{R}^2 \mid |x - 0.5| < 0.025, y - 0.75 < 0.1125\}$. The initial data undergoes rigid circular motion in the vector field $\mathbf{u} = (-2\pi(y - 0.5), 2\pi(x - 0.5))$ so that the exact solution coincides with the initial data in unit time. The interface arclength for this configuration is $L = 1.46724$ and the exact area is $A = 5.756 \times 10^{-2}$. For this problem the reconstruction of the thin slit in the interface is of primary importance.

Table 4

Convergence tests using \mathbb{Q}_1 elements with the FCT method; $T = 1$. L^1 -norm of the error (left); interface error (36) (center); area error (37) (right).

h	L^1 -norm	Order	I_{Err}	Order	A_{Err}	Order
1.00E-1	8.174E-02		4.695E-2		1.487E-1	
5.00E-2	4.676E-02	0.80	1.889E-2	1.31	5.493E-2	1.43
2.50E-2	2.697E-02	0.79	7.311E-3	1.36	2.245E-2	1.29
1.25E-2	1.543E-02	0.80	2.810E-3	1.37	7.761E-3	1.53

Table 5

Zalesak's disc with finite volumes: level set $\{\phi = \frac{1}{2}\}$ after one revolution (left); level set $\{\phi = \frac{1}{2}\}$ after two revolutions (center); interface error (36) after one and two revolutions (right).

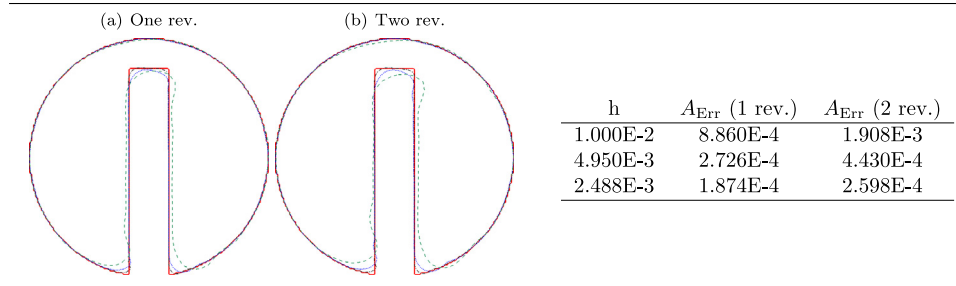


Table 6

Zalesak's disk with FCT and \mathbb{Q}_1 uniform meshes. (a) L^1 -norm of error; (b) Interface error after one and two revolutions; (c) Area error after one and two revolutions.

h	(a) L^1 -norm of error		(b) I_{Err}		(c) A_{Err}	
	1 rev.	2 rev.	1 rev.	2 rev.	1 rev.	2 rev.
2.50E-2	1.439E-2	1.607E-2	1.160E-2	1.256E-2	9.510E-2	9.239E-2
1.25E-2	8.401E-3	9.482E-3	4.259E-3	4.446E-3	2.021E-2	7.412E-3
6.25E-3	4.768E-3	5.448E-3	1.750E-3	2.010E-3	1.527E-3	4.412E-3

9.1. Finite volumes

We start the tests with the finite volume method. The interface error, defined by (36), is reported in Table 5 for various discretizations after one and two revolutions. The level sets $\{\phi = \frac{1}{2}\}$ for three different mesh resolutions after one and two revolutions are shown in the left and center panel of Table 5. The computations were carried out with various mesh discretizations: $h = 1 \times 10^{-2}$ (green dashed), $h = 4.950 \times 10^{-3}$ (blue dotted), and $h = 2.488 \times 10^{-3}$ (black dashed-dotted).

9.2. \mathbb{P}_1 finite elements

We report in this section results for the Zalesak test using continuous \mathbb{P}_1 finite elements. Fig. 4(a) and (b) depict the level sets $\{\phi = \frac{1}{2}\}$ on a Delaunay mesh with characteristic mesh size $h \approx 0.01$ after one and ten revolutions. This test shows that the method performs well over long time integration on moderately refined meshes. Fig. 4(c) shows the level sets $\{\phi = \frac{1}{2}\}$ after one revolution for four meshes $h \approx 0.025$, $h \approx 0.0125$, $h \approx 0.01$, and $h \approx 0.00625$.

9.3. \mathbb{Q}_1 finite elements

We continue with the Zalesak problem using \mathbb{Q}_1 finite elements limited with the FCT method. Table 6 gives the L^1 -norm of the error, the interface error, and the area error for the three discretizations, $h = 2.5 \times 10^{-2}$, $h = 1.25 \times 10^{-2}$ and $h = 6.25 \times 10^{-3}$, after one and two revolutions. The level sets $\{\phi = \frac{1}{2}\}$ for the three meshes after one revolution and two revolutions are shown in Fig. 5(a) and (b).

10. A periodic swirling flow problem in two dimensions

In this section a transport problem is computed with the divergence free velocity \mathbf{u} given by:

$$\mathbf{u} = (-2 \sin(\pi y) \cos(\pi y) \sin^2(\pi x) \cos(\pi t/T), 2 \sin(\pi x) \cos(\pi x) \sin^2(\pi y) \cos(\pi t/T)). \tag{38}$$

The initial data for the problem is the characteristic function of a disk of radius $r = 0.15$ with initial position $(0.5, 0.75)$: $\phi_0(\mathbf{x}) = 1$ if $\|\mathbf{x} - (0.5, 0.75)\|_{\ell^2} \leq 0.15$ and $\phi_0(\mathbf{x}) = 0$ otherwise. The solution is periodic and modulated in time by

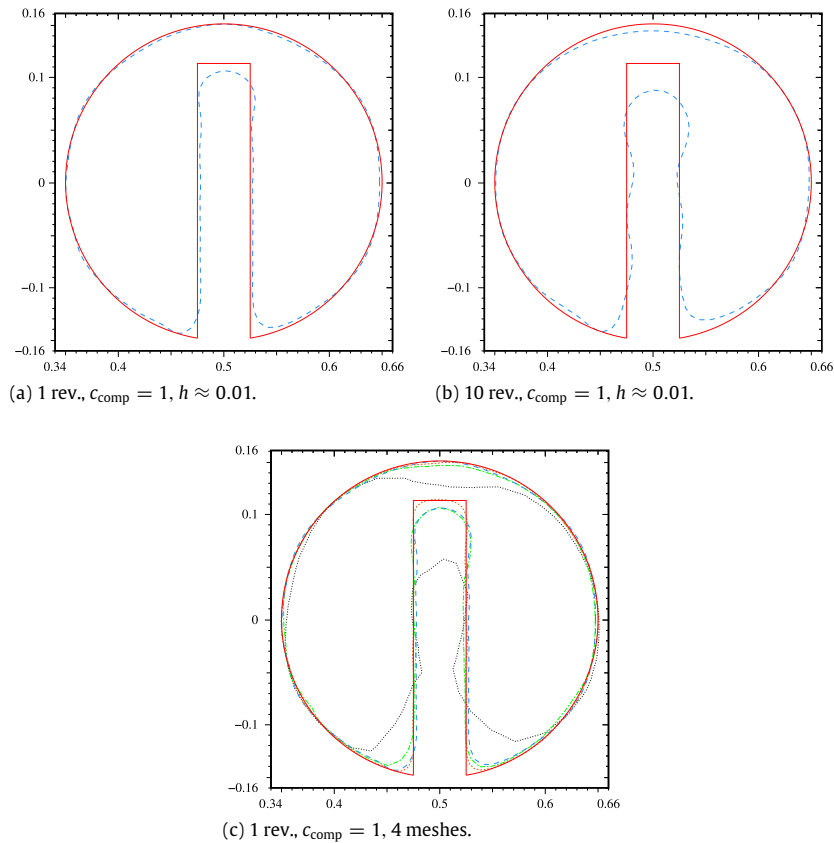


Fig. 4. Zalesak disk; \mathbb{P}_1 finite elements on Delaunay grids; Level set $\{\phi = \frac{1}{2}\}$. Panel 4(a)–(b): 1 and 10 revolutions, dark blue dashes, $h \approx 0.01$, 38 813 nodes. Panel 4(c): 1 revolution, Black dots, $h \approx 0.025$, 6532 nodes; green dots-dashes, $h \approx 0.0125$, 24 916 nodes; dark blue dashes, $h \approx 0.01$, 38 813 nodes; brown long dots, $h \approx 0.00625$, 98 648 nodes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

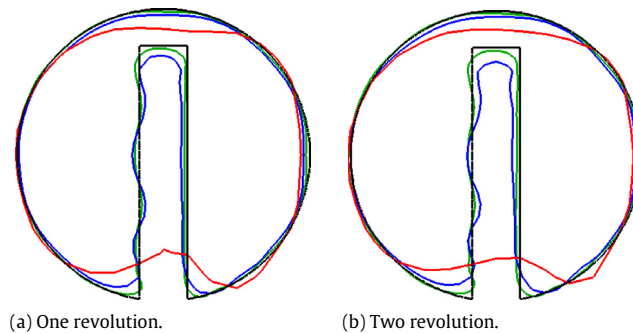


Fig. 5. Zalesak disk; FCT and \mathbb{Q}_1 finite elements. Fig. 5(a)–(b) One and two revolutions. Are shown the exact solution (black) and the numerical solution with $h = 2.5 \times 10^{-2}$ (red), $h = 1.25 \times 10^{-3}$ (blue), and $h = 6.25 \times 10^{-3}$ (green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the parameter T . The circular initial data is twisted into a thin vortex and returns to its initial configuration in time T ; as the period T is increased the ability of the method to capture thin filaments is tested. This test uses $T = 8$ to allow for comparison with Enright et al. [2]. As a consequence of the choice $T = 8$ the winding of the initial data forms very thin and elongated structures in the tail which are difficult to resolve.

10.1. Finite volumes

Fig. 6 shows half of a period to the periodic swirl problem, from $t = 0$ to $t = 4$, with the mesh resolution $h = 4.95 \times 10^{-3}$; the process then proceeds in reverse and the exact solution at the final time $t = 8$ coincides with the initial data.

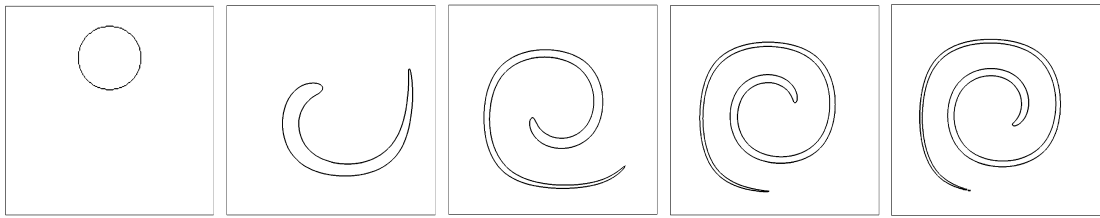


Fig. 6. Periodic vortex; Finite volumes; Level set $\{\phi = \frac{1}{2}\}$; $t = 0, 1, 2, 3, 4$; $h = 4.95 \times 10^{-3}$.

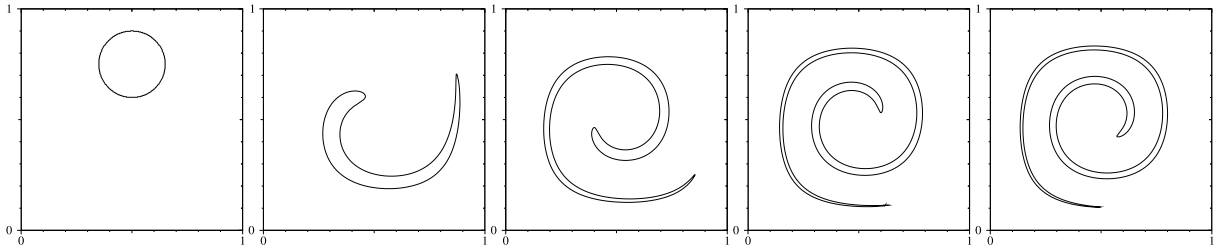


Fig. 7. Periodic vortex at times (from left to right) $t = 0, 1, 2, 3,$ and 4 using \mathbb{P}_1 elements with $h \approx 3.125 \times 10^{-3}$.

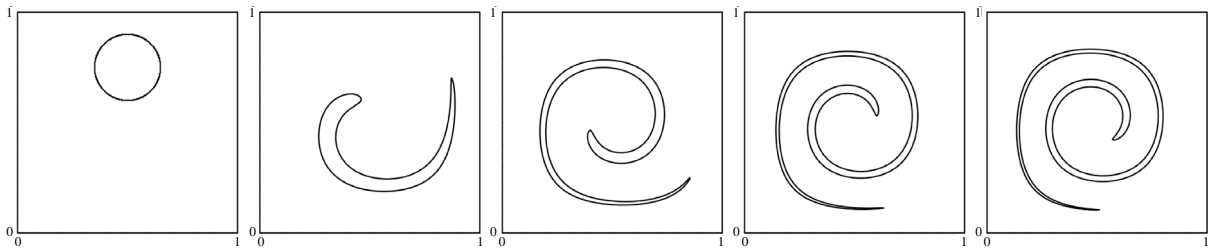


Fig. 8. Periodic vortex at times (from left to right) $t = 0, 1, 2, 3, 4$ using \mathbb{Q}_1 with the FCT method and $h = 2.5 \times 10^{-3}$.

At low discrete resolutions the tail disassociates and forms isolated structures; when the flow is reversed this leads to area loss. Measuring the area loss provides a metric for the ability of the method to capture thin structures. Table 7 gives both the area conservation error and the interface error as defined in (36) and (37), respectively, for four different meshes. The left panel shows the interface level set for three meshes: $h = 1 \times 10^{-2}$ (red), $h = 4.950 \times 10^{-3}$ (blue), and $h = 2.488 \times 10^{-3}$ (green) while the exact solution is shown in black. The results are qualitatively similar to those of particle level set methods as reported in Enright et al. [2].

10.2. \mathbb{P}_1 finite elements

We consider again the vortex in this section, but this time we use continuous \mathbb{P}_1 finite elements with $c_{\text{comp}} = c_E = 1$. We show in Fig. 7 the level set $\{\phi = \frac{1}{2}\}$ at $t = 0, 1, 2, 3, 4$ on a Delaunay mesh of characteristic size $h \approx 0.003125$ composed of 118 851 \mathbb{P}_2 degrees of freedoms. The left panel in Table 8 shows the level set $\{\phi = \frac{1}{2}\}$ at $T = 8$ for three different meshes of mesh size $h \approx 0.01$, $h \approx 0.00625$ and $h \approx 0.003125$. The L^1 -norm of the error at $T = 8$ and the quantities $I_{\text{Err}}, A_{\text{Err}}$ for these three meshes are reported in the table.

10.3. \mathbb{Q}_1 finite elements

In this section we solve the periodic swirl problem with \mathbb{Q}_1 finite elements, on uniform grids, and also utilize the flux-corrected transport modification. The maximum principle is guaranteed by the FCT technique. In Fig. 8, we consider $h = 2.5 \times 10^{-3}$ and show the solution at times $t = 0, 1, 2, 3,$ and 4 . In Table 9 we show the L^1 -norm of the error at $T = 8$ and the quantities $I_{\text{Err}}, A_{\text{Err}}$ for these three meshes after one cycle. The left panel in Table 9 shows the exact and the numerical solutions at $t = 8$ for different refinements.

11. Multiphase simulations

In this section we use the maximum principle preserving method with artificial compression presented in Section 6 along with an incompressible Navier–Stokes solver to simulate two-phase flow; the method is documented in Guermond and Salgado [38].

Table 7
Periodic vortex with finite volumes, one period.

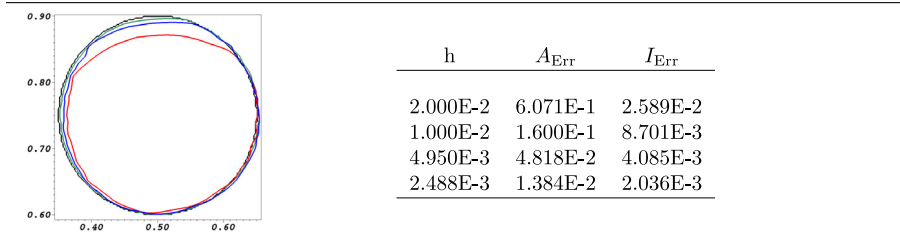


Table 8
Periodic vortex in 2D with \mathbb{P}_1 finite elements. Level set $\{\phi = \frac{1}{2}\}$ in the left panel at $T = 8$: $h \approx 0.01$ red line; $h \approx 0.00625$, dark blue line; $h \approx 0.003125$, green line. The table on the right shows the L^1 -norm of the error and the quantities $I_{\text{Err}}, A_{\text{Err}}$ after one cycle.

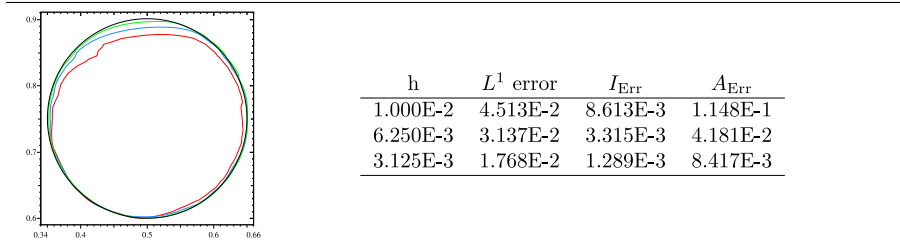
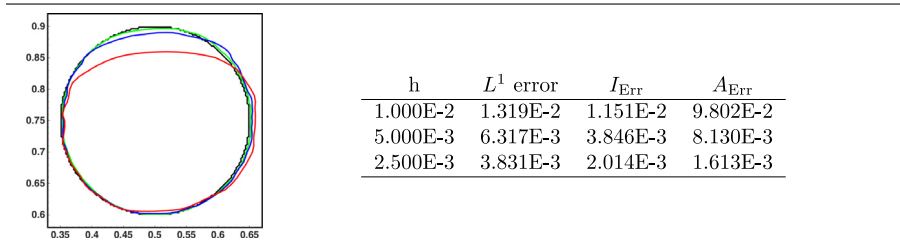


Table 9
Periodic vortex in 2D with \mathbb{Q}_1 finite elements with FCT. Level set $\{\phi = \frac{1}{2}\}$ in the left panel at $T = 8$: $h = 0.01$ red line; $h \approx 0.005$, dark blue line; $h \approx 0.0025$, green line. The table on the right shows the L^1 -norm of the error and the quantities $I_{\text{Err}}, A_{\text{Err}}$ after one cycle.



11.1. Overview of the methodology

The velocity and the pressure are computed using the method from [38], and the density is computed by solving the transport equation for the level set. The density and the viscosity fields are reconstructed from the level set function by employing the linear reconstruction:

$$\rho = \rho_{\text{water}}H_\epsilon(\phi) + \rho_{\text{air}}(1 - H_\epsilon(\phi)), \tag{39a}$$

$$\mu = \mu_{\text{water}}H_\epsilon(\phi) + \mu_{\text{air}}(1 - H_\epsilon(\phi)), \tag{39b}$$

where $\phi \in [0, 1]$ is the level set function and H_ϵ is a regularized Heaviside function. In particular, the set $\{\phi = 1\}$ represents water and the set $\{\phi = 0\}$ represents air. Ideally, the transition from 0 to 1 should be maintained as sharply as possible since values of ϕ in the range $(0, 1)$ give non-physical values for the reconstructed fields. This observation leads naturally to the definition of the regularized Heaviside function:

$$H_\epsilon(\phi) = \begin{cases} 1 & \text{if } 2\phi - 1 > \epsilon \\ 0 & \text{if } 2\phi - 1 < -\epsilon, \\ \frac{2\phi - 1 + \epsilon}{2\epsilon} & \text{otherwise.} \end{cases} \tag{40}$$

All the experiments reported in this section are done with quadrangular meshes and $\epsilon = h$ where $h = \Delta x = \Delta y$ is the cell size. The velocity is approximated using \mathbb{Q}_2 elements and the pressure is approximated using \mathbb{Q}_1 elements. Once the density and the viscosity are reconstructed from the level set function, the Navier–Stokes equations are solved to obtain the velocity field. The velocity field is then used to transport the level set. This process is repeated until the final time is reached.

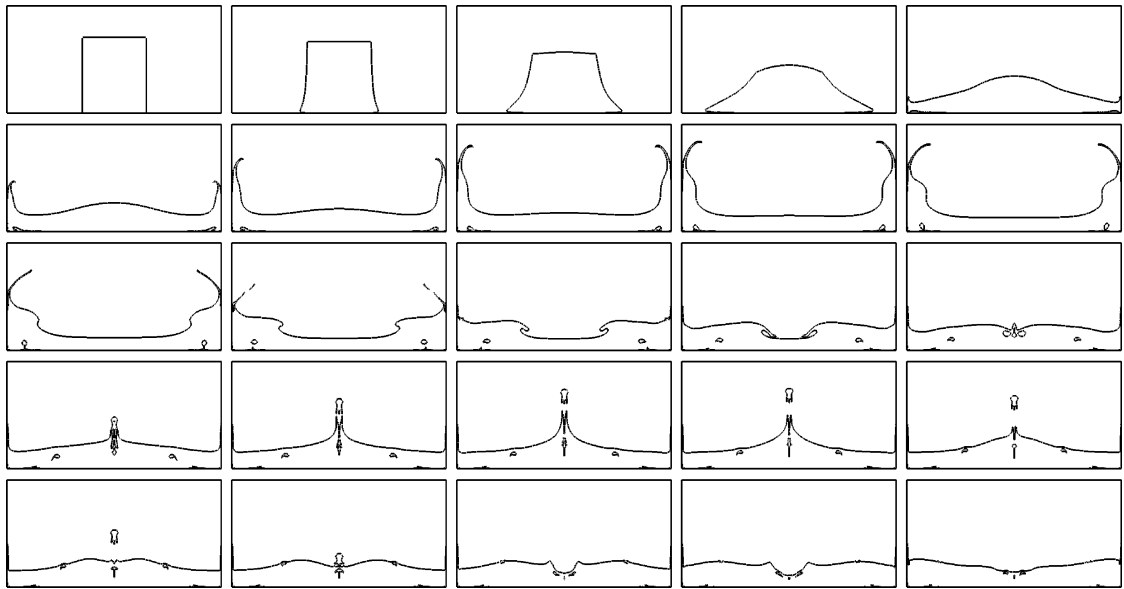


Fig. 9. Dam breaking problem at different times; $c_{\text{comp}} = 1$, $c_E = 1$. The times shown are (from left to right and top to bottom) $t = 0, 0.2, 0.4, 0.6, 0.8, 1, 1.2, 1.4, 1.6, 1.8, 2, 2.2, 2.4, 2.6, 2.8, 3, 3.2, 3.4, 3.6, 3.8, 4, 4.2, 4.4, 4.6$ and 4.8 .

11.2. Two-dimensional dam breaking

We consider the two-dimensional dam breaking problem on the domain $D = [0, 1] \times [0, 0.5]$. The initial data consists of water occupying the domain $W = \{(x, y) \mid |x - 0.5| < 0.15, y < 0.35\}$ and air occupying the domain $A = D \setminus W$. Both water and air are at rest at the initial time. At $t = 0$ we let the system evolve under the action of gravity; i.e., the column of water collapses under its own weight and spreads over the tank. The material parameters are given by

$$\rho_{\text{water}} = 1000, \quad \rho_{\text{air}} = 1, \quad \mu_{\text{water}} = 1, \quad \mu_{\text{air}} = 1.8 \times 10^{-2}, \quad (41)$$

and the gravity coefficient is $g = -1$. We impose $\mathbf{u} = 0$ at all the boundaries; i.e., we consider the no-slip boundary condition. The mesh size of the spatial discretization used is $h = 1.953 \times 10^{-3}$. The interface level set $\{\phi = 0.5\}$ is shown in Fig. 9 for various times.

11.3. Two-dimensional tank filling

The tank-filling test problem simulates water entering a tank filled with air. The domain is $D = [0, 0.4] \times [0, 0.4]$. The initial data consists of water occupying the domain $W = \{(x, y) \mid x < 0.01, |y - 0.325| < 0.025\}$ and air occupying the domain $A = D \setminus W$. Both water and air are at rest at the initial time. We use the same material parameters and the same gravity coefficient as in the previous section. The boundary conditions on the velocity are:

$$\mathbf{u} = \mathbf{u}_{\text{left}} := \begin{cases} (0.25, 0), & \forall x = 0, y \in [0.3, 0.35], \\ (0, 0), & \forall x = 0, y \notin [0.3, 0.35], \end{cases}$$

$$\mathbf{u} = \mathbf{u}_{\text{top}} := \begin{cases} (0, 0.25), & \forall x \in [0.3, 0.35], y = 0.4, \\ (0, 0), & \forall x \notin [0.3, 0.35], y = 0.4, \end{cases}$$

$\mathbf{u} =: \mathbf{u}_{\text{right}} = 0$ if $x = 0.4$ and $\mathbf{u} =: \mathbf{u}_{\text{bottom}} = 0$ if $y = 0$. The boundary condition for the level set is: $\phi = 1, \forall x = 0, y \in [0.3, 0.35]$; i.e., water is introduced. The mesh size of the spatial discretization used is $h = 1.562 \times 10^{-3}$. The interface level set $\{\phi = 0.5\}$ is shown, for various times, in Fig. 10.

12. Three-dimensional tests

This section illustrates the compression technique in three space dimensions. The computations are done in parallel using the massively parallel second-order finite volume code described in Guermond and Mineev [33]. The Brazos computational cluster at Texas A&M University and the Beacon computational cluster at the University of Tennessee were used for these computations.

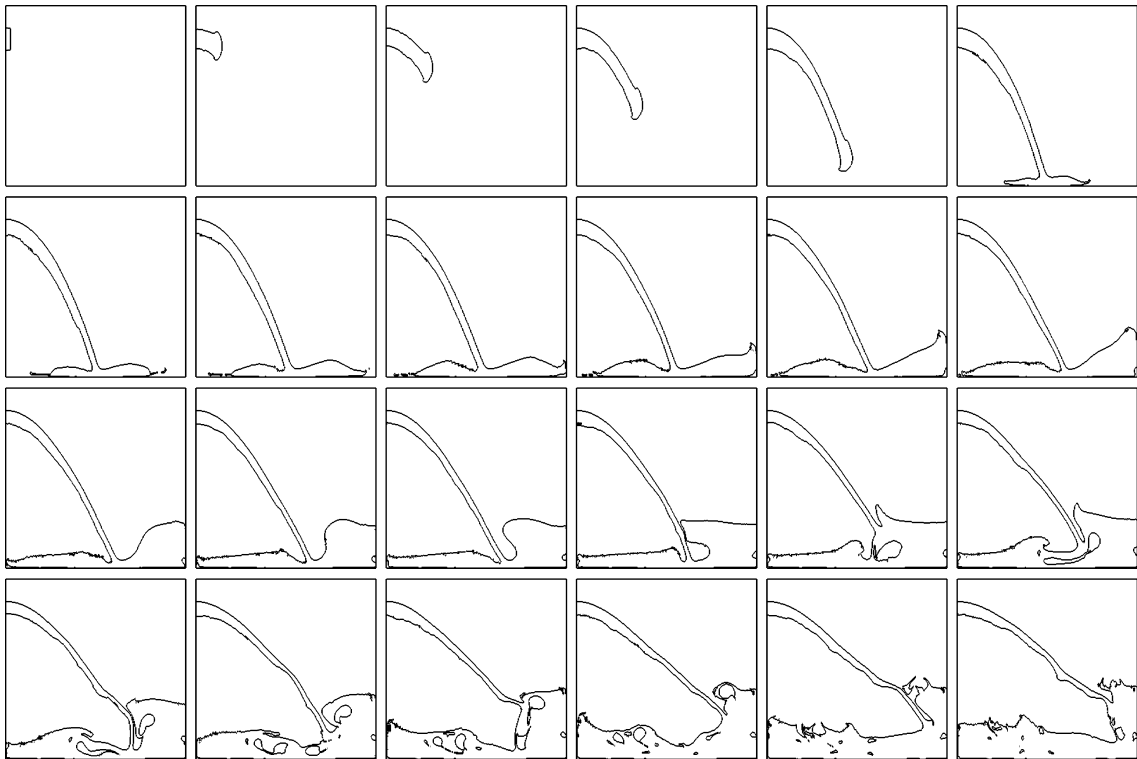


Fig. 10. Tank filling problem at different times; $c_{\text{comp}} = 1$, $c_E = 1$. The times shown are (from left to right and top to bottom) $t = 0, 0.2, 0.4, 0.6, 0.8, 1, 1.2, 1.4, 1.6, 1.8, 2, 2.2, 2.4, 2.6, 2.8, 3, 3.2, 3.4, 3.6, 3.8, 4, 4.2, 4.4$ and 4.6 .

Table 10
Zalesak's sphere; one revolution.

h	A_{Err}	I_{Err}
1.00E-2	4.60E-2	3.22E-3
5.00E-3	1.32E-2	1.17E-3
3.33E-3	5.06E-3	5.47E-4
2.50E-3	3.29E-3	4.09E-4

12.1. Zalesak's problem in three dimensions

This section revisits the Zalesak problem. The Zalesak sphere initial data is rotated by the vector field $\mathbf{u} = (-2\pi(y - 0.5), 2\pi(x - 0.5), 0)$. The initial data is the characteristic function of a sphere with radius $r = 0.15$ centered at the point $(0.5, 0.75, 0.75)$ with a thin rectangular volume removed; for this test case the removed volume is $R = \{\mathbf{x} = (x, y, z) \mid |x - 0.5| < 0.025, y - 0.75 < 0.1125, |z| < 1\}$. The surface area of the level set $\{\phi = \frac{1}{2}\}$ is $L = 2.3058848 \times 10^{-1}$. The exact solution coincides with the initial data in unit time.

The results of the parallel simulations are compared at $T = 1$ on various meshes, and results with, and without, compression are presented for comparison. Note again that these results have been produced by correcting the mass lumping effect to limit dispersion effects as explained in Section 5. The results are shown in Fig. 11 on three meshes: 50^3 , 100^3 and 200^3 . The effects of the compression are visible when one compares the panels in the top row with the panels in the bottom row in Fig. 11. The area and interface error measures are reported in Table 10.

12.2. LeVeque test in three dimensions

We conclude the three dimensional tests by a problem proposed in LeVeque [39]. The ambient, periodic vector field $\mathbf{u} = (u, v, w)$ is given by

$$\begin{aligned}
 u(x, y, z) &= 2 \sin(\pi x)^2 \sin(2\pi y) \sin(2\pi z) \cos(\pi t/T) \\
 v(x, y, z) &= -\sin(2\pi x) \sin(\pi y)^2 \sin(2\pi z) \cos(\pi t/T) \\
 w(x, y, z) &= -\sin(2\pi x) \sin(2\pi y) \sin(\pi z)^2 \cos(\pi t/T).
 \end{aligned}
 \tag{42}$$

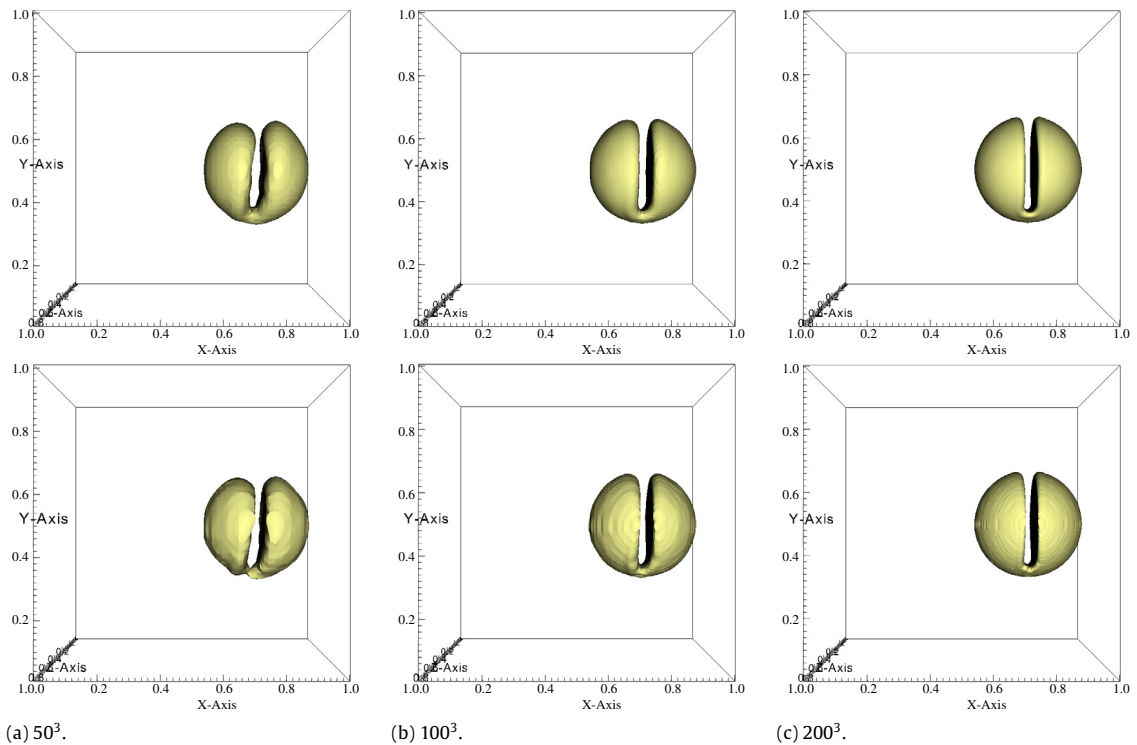


Fig. 11. Zalesak's problem, $T = 1$; (Top) with compression; (bottom) without compression.

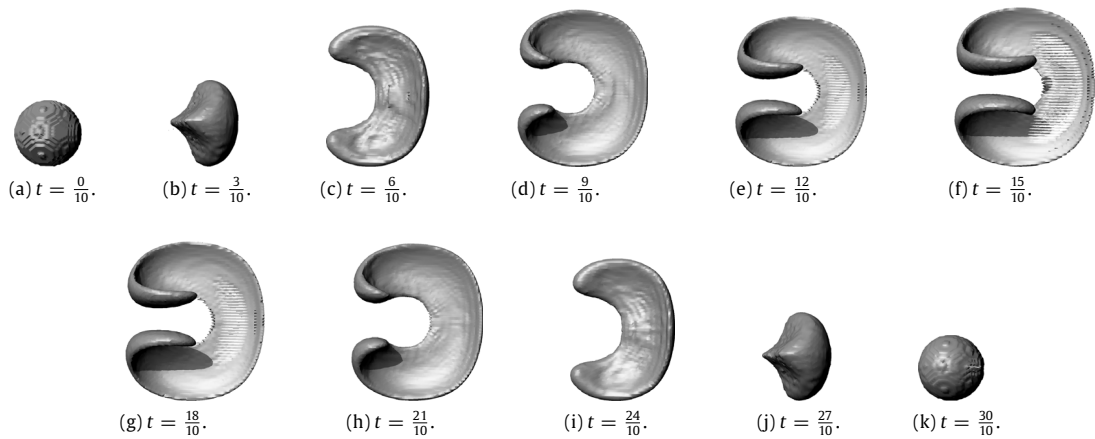


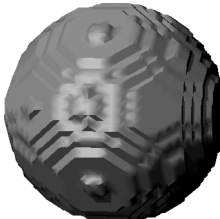
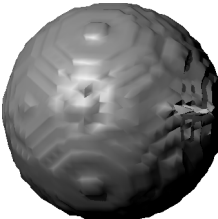
Fig. 12. 3D LeVeque test, $h = 1 \times 10^{-2}$, $t = 0$ to $t = 3$.

This divergence-free vector field combines deformations in the $x - y$ and $x - z$ planes. The initial data is the characteristic function of a sphere with radius $r = 0.15$ centered at the point $(0.35, 0.35, 0.35)$, and the time modulation was chosen as $T = 3$. The solution was computed using several discretizations of the unit cube, including the classical benchmark discretization of $100 \times 100 \times 100$. The surface area, used to measure interface error, is $L \approx 2.827 \times 10^{-3}$.

The interface level set entrained in the velocity field (41), develops very thin regions to the point of under-resolution, before returning to its initial position. A basis for comparison can be found in the works of Enright et al. [2], Ianniello and Di Mascio [3] in addition to Luo et al. [40], Wang et al. [41,42], which were suggested in the review process, although it does not appear in Chiu and Lin [6].

A time progression of the $\{\phi = \frac{1}{2}\}$ iso-surface interface for $h = 1 \times 10^{-2}$ is shown in Fig. 12(a)–(k). Area conservation error and interface error results are reported in Table 11(a). A comparison of the initial versus final time appears in panel Table 11(b) and (c). All these computations have been done for a CFL number of 0.75 without the use of the flux-corrected transport (FCT) described in Section 6. The absence of flux correction leads to slight overshoots interface iso-surface for

Table 11
3D LeVeque test, one period.

(a) Convergence test			(b) $t = 0, h = 1 \times 10^{-2}$	(c) $t = 3, h = 1 \times 10^{-2}$
h	A_{Err}	I_{Err}		
2.000E-2	1.794E-1	2.732E-1		
1.000E-2	4.179E-2	6.943E-2		
4.950E-3	4.806E-3	6.329E-3		
2.488E-3	9.392E-4	1.252E-3		

$t = 3$. The overall results are similar to those achieved by other methods; see for instance Enright et al. [2], Ianniello and Di Mascio [3], Luo et al. [40], Wang et al. [41,42].

13. Concluding remarks

In conclusion we have introduced a novel anti-diffusion method, based on entropy production, for use with the level set method. We have detailed the implementation of the method with linear continuous finite elements and finite volumes, and we have given explicit values for the constants in the methods which were used for all computations. It has been shown that the method is conservative and suitable for use on unstructured meshes; it can also be made (formally) second-order accurate in space and maximum principle preserving. Computational tests of the approach using well-known benchmark problems have been conducted in the context of both finite volumes and finite elements in one, two and three dimensions. The method proposed in the paper has recently been used to solve challenging magneto-hydrodynamics problems like the metal pad roll instability and the Tayler instability in liquid metal batteries, Cappanera et al. [43], Herreman et al. [44].

The authors would like to acknowledge the generous support of computational time on the Brazos cluster at Texas A& M and on the University of Tennessee's Beacon cluster located at the Joint Institute for Computational Sciences.

References

- [1] S. Osher, J. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations, *J. Comput. Phys.* 79 (1988) 12–49.
- [2] D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, *J. Comput. Phys.* 183 (1) (2002) 83–116.
- [3] S. Ianniello, A. Di Mascio, A self-adaptive oriented particles Level-Set method for tracking interfaces, *J. Comput. Phys.* 229 (4) (2010) 1353–1380.
- [4] R. Fedkiw, G. Sapiro, C.-W. Shu, Shock capturing, level sets, and PDE based methods in computer vision and image processing: a review of osher's contributions, *J. Comput. Phys.* 185 (2003) 309–341.
- [5] E. Olsson, G. Kreiss, A conservative level set method for two phase flow, *J. Comput. Phys.* 210 (2005) 225–246.
- [6] P.-H. Chiu, Y.-T. Lin, A conservative phase field method for solving incompressible two-phase flows, *J. Comput. Phys.* 230 (1) (2011) 185–204.
- [7] J.-L. Guermond, M. Nazarov, A maximum-principle preserving C^0 finite element method for scalar conservation equations, *Comput. Methods Appl. Mech. Engrg.* 272 (2014) 198–213.
- [8] J.-L. Guermond, B. Popov, Invariant domains and first-order continuous finite element approximation for hyperbolic systems, *SIAM J. Numer. Anal.* (ISSN: 0036-1429) 54 (4) (2016) 2466–2489. URL <http://dx.doi.org/10.1137/16M1074291>.
- [9] T. Coupez, Convection of local level set function for moving surfaces and interfaces in forming flow, in: *AIP Conference Proceedings*, Vol. 908, 2007.
- [10] A. Harten, The artificial compression method for computation of shocks and contact discontinuities. I. Single conservation laws, *Comm. Pure Appl. Math.* 30 (5) (1977) 611–638.
- [11] A. Harten, The artificial compression method for computation of shocks and contact discontinuities. III. Self-adjusting hybrid schemes, *Math. Comp.* 32 (1978) 363–389.
- [12] J.-L. Guermond, R. Pasquetti, B. Popov, Entropy viscosity method for nonlinear conservation laws, *J. Comput. Phys.* 230 (11) (2011) 4248–4267.
- [13] W. Rider, D. Kothe, Reconstructing volume tracking, *J. Comput. Phys.* 141 (1998) 112–152.
- [14] S. Osher, R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer-Verlag, 2003.
- [15] M. Sussman, E. Puckett, A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows, *J. Comput. Phys.* 162 (2000) 301–337.
- [16] M. Sussman, A. Almgren, J. Bell, P. Colella, L. Howell, M. Welcome, An adaptive level set approach for incompressible two-phase flows, *J. Comput. Phys.* 148 (1999) 81–124.
- [17] F. Gibou, R. Fedkiw, R. Caflisch, S. Osher, A level set approach for the numerical simulation of dendritic growth, *J. Sci. Comput.* 19 (2003) 324–344.
- [18] L. Ville, L. Silva, T. Coupez, Convected level set method for the numerical simulation of fluid buckling, *Internat. J. Numer. Methods Fluids* 66 (3) (2011) 324–344.
- [19] A. Bonito, J.-L. Guermond, S. Lee, Numerical simulations of bouncing jets, *Internat. J. Numer. Methods Fluids* 80 (1) (2016) 53–75.
- [20] C. Dapogny, A. Faure, G. Michailidis, G. Allaire, A. Couvelas, R. Estevez, Geometric constraints for shape and topology optimization in architectural design. working paper or preprint, Aug. 2016.
- [21] T. Yamada, K. Izui, S. Nishiwaki, A. Takezawa, A topology optimization method based on the level set method incorporating a fictitious interface energy, *Comput. Methods Appl. Mech. Engrg.* 199 (2010) 2876–2891.
- [22] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, *J. Comput. Phys.* 114 (1994) 146–159.
- [23] A. Harten, From artificial viscosity to eno schemes, in: J. Buchler (Ed.), *The Numerical Modelling of Nonlinear Stellar Pulsations*, in: *NATO ASI Series*, vol. 302, Springer, Netherlands, 1990, pp. 239–262.
- [24] P.-H. Chiu, Y.-T. Lin, A conservative phase field method for solving incompressible two-phase flows, *J. Comput. Phys.* 230 (2011) 185–204.
- [25] M.-j. Lai, L.L. Schumaker, Spline Functions on Triangulations, in: *Encyclopedia of Mathematics and its Applications*, vol. 110, Cambridge University Press, Cambridge, 2007.

- [26] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, *SIAM Rev.* 43 (1) (2001) 89–112. (electronic).
- [27] M.A. Christon, M.J. Martinez, T.E. Voth, Generalized Fourier analyses of the advection-diffusion equation-part i: one-dimensional domains, *Internat. J. Numer. Methods Fluids* 45 (8) (2004) 839–887.
- [28] J.-L. Guermond, R. Pasquetti, A correction technique for the dispersive effects of mass lumping for transport problems, *Comput. Methods Appl. Mech. Engrg.* 253 (2013) 186–198.
- [29] T. Thompson, A discrete commutator theory for the consistency and phase error analysis of semi-discrete C^0 finite element approximations to the linear transport equation, *J. Comput. Appl. Math.* 303 (2016) 229–248.
- [30] J.-L. Guermond, R. Pasquetti, Entropy-based nonlinear viscosity for fourier approximations of conservation laws, *C. R. Math. Acad. Sci. Paris* 346 (2008) 801–806.
- [31] J.-L. Guermond, M. Nazarov, B. Popov, Y. Yang, A second-order maximum principle preserving lagrange finite element technique for nonlinear scalar conservation equations, *SIAM J. Numer. Anal.* 52 (4) (2014) 2163–2182.
- [32] M. Christon, The influence of the mass matrix on the dispersive nature of the semi-discrete, second-order wave equation, *Comput. Methods Appl. Mech. Engrg.* 173 (1–2) (1999) 147–166.
- [33] J.L. Guermond, P.D. Mineev, A new class of massively parallel direction splitting for the incompressible Navier-Stokes equations, *Comput. Methods Appl. Mech. Engrg.* 200 (23–24) (2011) 2083–2093.
- [34] J. Boris, D. Book, Flux-corrected transport. I. Shasta, a fluid transport algorithm that works, *J. Comput. Phys.* 11 (1973) 38–69.
- [35] S.T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, *J. Comput. Phys.* 31 (3) (1979) 335–362.
- [36] D. Kuzmin, R. Löhner, S. Turek, Flux-Corrected Transport: Principles, Algorithms, and Applications. Scientific Computation, Springer, 2005.
- [37] W. Bangerth, R. Hartmann, G. Kanschat, deal.II – a general purpose object oriented finite element library, *ACM Trans. Math. Software* 33 (4) (2007) 24/1–24/27.
- [38] J.-L. Guermond, A. Salgado, A splitting method for incompressible flows with variable density based on a pressure Poisson equation, *J. Comput. Phys.* 288 (8) (2006) 2834–2846.
- [39] R.J. Leveque, High-resolution conservative algorithms for advection in incompressible flow, *SIAM J. Numer. Anal.* 33 (2) (1996) 627–665.
- [40] K. Luo, C. Shao, y. Yang, J. Fan, A mass conserving level set method for detailed numerical simulation of liquid atomization, *J. Comput. Phys.* 298 (2015) 495–519.
- [41] Z. Wang, J. Yang, F. Stern, A simple and conservative operator-splitting semi-lagrangian volume-of-fluid advection scheme, *J. Comput. Phys.* 231 (2012) 4981–4992.
- [42] Z. Wang, J. Yang, F. Stern, A new volume-of-fluid method with a constructed distance function on general structured grids, *J. Comput. Phys.* 231 (2012) 3703–3722.
- [43] L. Capanera, J.-L. Guermond, W. Herreman, C. Nore, Momentum-based approximation of incompressible multiphase fluid flows, *I. J. Numer. Methods Fluids* (2017) (submitted for publication).
- [44] W. Herreman, C. Nore, L. Capanera, J.-L. Guermond, Tayler instability in liquid metal columns and liquid metal batteries, *J. Fluid Mech.* 771 (2015) 79–114.