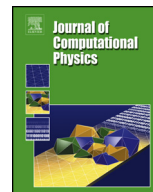




Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A Characteristic Mapping Method for the three-dimensional incompressible Euler equations

Xi-Yuan Yin^a, Kai Schneider^b, Jean-Christophe Nave^{a,*}

^a Department of Mathematics and Statistics, McGill University, Montréal, Québec H3A 0B9, Canada

^b Institut de Mathématiques de Marseille (I2M), Aix-Marseille Université, CNRS, 13453 Marseille Cedex 13, France



ARTICLE INFO

Article history:

Received 30 July 2021

Received in revised form 18 August 2022

Accepted 19 December 2022

Available online 13 January 2023

Keywords:

Fluid dynamics

Euler equations

Characteristic Mapping Method

Gradient-augmented level-set method

ABSTRACT

We propose an efficient semi-Lagrangian Characteristic Mapping (CM) method for solving the three-dimensional (3D) incompressible Euler equations. This method evolves advected quantities by discretizing the flow map associated with the velocity field. Using the properties of the Lie group of volume preserving diffeomorphisms $SDiff$, long-time deformations are computed from a composition of short-time submaps which can be accurately evolved on coarse grids. This method is a fundamental extension to the CM method for two-dimensional incompressible Euler equations [1]. We take a geometric approach in the 3D case where the vorticity is not a scalar advected quantity, but can be computed as a differential 2-form through the pullback of the initial condition by the characteristic map. This formulation is based on the Kelvin circulation theorem and gives point-wise a Lagrangian description of the vorticity field. We demonstrate through numerical experiments the validity of the method and show that energy is not dissipated through artificial viscosity and small scales of the solution are preserved. We provide error estimates and numerical convergence tests showing that the method is globally third-order accurate.

© 2023 Elsevier Inc. All rights reserved.

1. Introduction

Turbulence remains one of the oldest and most challenging research problems in both pure and applied science; the high Reynolds number limit of the Navier-Stokes equations is of particular interest and developments in scientific computing are useful in advancing the frontiers of our understanding of fluid dynamics in a highly nonlinear regime. Efficient and precise numerical schemes for simulating incompressible inviscid fluids in three-dimensional space are an essential component, however the multiscale nature of turbulent flows along with the computationally demanding high dimensionality requires the development of specialized algorithms. Different directions have been pursued so far.

Among the Eulerian ansatz using a fixed computational grid, Fourier pseudo-spectral methods are certainly very attractive discretizations [2], requiring nevertheless some viscous or hyperviscous regularization, see for instance the discussion in [3]. However one drawback of Eulerian schemes is the Courant-Friedrichs-Lewy (CFL) condition which constrains the size of time steps in relation to the spatial discretization scales. This implies that the complexity of the simulations with N grid points in each spatial direction is proportional to N^4 or even worse [4]. The progress of high-resolution numerical simulation using Fast Fourier Transforms is hence limited and directly linked to the development of supercomputers following Moore's

* Corresponding author.

E-mail address: jcnave@math.mcgill.ca (J.-C. Nave).

law. Furthermore, Eulerian methods are also prone to artificial dissipation and special care needs to be taken in the spatial resolution of the solution and when using spectral dealiasing; spatial adaptivity and parallel implementation can often play an important role in improving computational efficiency [5–7]. A comparative study of spatial discretization and spectral dealiasing methods for the Euler equations can be found in [8]. Generally, high-resolution computational grids are needed to reduce the effects of dissipation, currently pseudo-spectral simulations with up to 12288^3 grid points can be performed [9]. Recently, Reference Mapping methods have also gained in popularity in the field of solid and elastic mechanics [10–13]. These Eulerian methods approach the problem through a similar perspective as Characteristic Mapping methods by discretizing a motion map instead of the transported quantities.

The Lagrangian ansatz, and in particular the semi-Lagrangian approach which combines Lagrangian time integration with Eulerian grids, does not suffer from a time-step restriction due to the CFL condition, see e.g. Staniforth and Côté [14] (1991) and references therein. Hence they are well suited for advection-dominated problems. Purely Lagrangian approaches include the vortex blobs methods [15–20], and the vortex particle and filament methods [21,17,22–24]. These methods are characterized by a particle-based discretization of the vorticity field; the motion of the fluid is idealized as the transport of a collection of point vortices or compactly supported vortex blobs and the velocity of each particle can be computed from vortices using the Biot-Savart law. These methods are inherently spatially adaptive since the representation of the vorticity field is reduced to a collection of point vortices concentrated where the vorticity is important. Furthermore, they are more effective in avoiding artificial viscous dissipation compared to their Eulerian counterparts. Some drawbacks include the difficulty in the representation and controlled resolution of Eulerian quantities. Methods for transferring Lagrangian quantities to fixed Eulerian grids include the vortex-in-cell methods [25–28] and the Cauchy-Lagrangian frameworks [29].

Various Eulerian or semi-Lagrangian methods have been used to provide evidence of singularity for the 3D Euler equations [30–33] or nonsingular super-exponential growth in the maximum vorticity [34–36]. For instance a pseudospectral computation of an axisymmetric solution suggesting finite-time blow-up was performed by Kerr in 1993 [37]. Some recent computations including new test cases for potential singularities in 3D Euler have been proposed in [38,39].

The work in this paper can be seen as a proof of concept for a novel semi-Lagrangian numerical method, which allows a very detailed investigation of singularities in 3D Euler. To this end we propose a geometric method for the 3D incompressible Euler equations in its vorticity form. The method uses the numerical framework of the Gradient-Augmented Level-Set methods [40] and Reference Map methods [41] and consists of a semi-Lagrangian discretization of the backward flow map, called the characteristic map, generated by the velocity field. As the name suggests, the Characteristic Mapping (CM) methods are intended as generalizations of the method of characteristics from standard PDE theory for transport equations. The method of characteristics reformulates a transport equation as ODEs along characteristic curves, similarly, the CM methods reformulate a Lie-transport equation as differential equations along pullbacks by the characteristic map. The diffeomorphism property of the map not only allows for a point-to-point correspondence (relabeling symmetry) but also provides a transformation of the tangent space at each point, which is crucial for the Lie-advection of differential forms considered in this paper. The method presented in this work is based on a previous work on the 2D incompressible Euler equations using the CM method [1], which we extend and generalize here to the three-dimensional case. Compared to the 2D equations, the 3D Euler equations present several significant challenges. Firstly, the presence of an additional vortex stretching term requires a more geometric formulation of the CM method in order to be seamlessly incorporated in the framework: a direct treatment of the vortex stretching as a source term would not conform to the characteristic structure of the method, nullifying some of its numerical qualities. Secondly, due to the vortex stretching term, we no longer have conservation of any L^p norms, including the L^∞ norm. Indeed, in the 2D case, the scalar vorticity is an advected quantity with infinitely many Casimir invariants, which can be preserved numerically [42], and small scale features appear only from rapid growth in vorticity gradient. In the 3D case, the question of finite-time blow-up in the solutions of incompressible Euler equations with smooth initial data is a notoriously difficult open problem in the theory of PDEs and is related to the Clay institute Millennium problem on the Navier-Stokes equations [43]. From a numerical point of view, the rapid growth in both the magnitude of the vorticity and its gradient further increases the difficulty in providing sufficient spatial resolution of the solution. For the CM method, the dependency of vorticity on the spatial deformations or rate of strain tensor would then involve the Jacobian of the characteristic map in the computation of the vorticity, thus increasing the regularity requirements on the method. Lastly, the higher dimensionality further emphasizes on the computational efficiency of the method and on high order accuracy.

The rest of the paper is organized as follows: in section 2 we recall the CM framework presented in [1] and generalize it through a geometric reformulation in the context of the 3D incompressible Euler equations. In section 3, we provide some details on the numerical implementation of the method together with formal error estimates supported by convergence tests. Section 4 contains numerical tests and discussions; In particular we simulate the anti-parallel axisymmetric perturbed vortex tubes tests similar to those appearing in [37] and [36]. Finally, in section 5 we make some concluding remarks and propose future directions of work.

2. Mathematical formulation

We present here the Characteristic Mapping method for the incompressible Euler equations in three-dimensional space. This work is the natural continuation of the framework presented in [41,44,1] and extends the Characteristic Mapping method for the 2D Euler equations in [1] to the 3D case. The main challenge in the 3D case is the inclusion of the vortex stretching term in a way that is compatible with the CM method and preserves its arbitrary resolution and low-dissipation

properties. For this, we expand on the CM framework by including a more geometric formulation of the problem in terms of differential forms.

2.1. Characteristic Mapping Method

The Characteristic Mapping (CM) method consists of computing the diffeomorphic transformations of the domain generated by a given velocity field. For a given smooth and divergence-free velocity field \mathbf{u} on a three-dimensional domain $U \in \mathbb{R}^3$, we consider the family of characteristic curves $\boldsymbol{\gamma}(t)$ parametrized by their initial condition $\boldsymbol{\gamma}_0$:

$$\frac{d}{dt}\boldsymbol{\gamma}(t) = \mathbf{u}(\boldsymbol{\gamma}(t), t), \quad (1a)$$

$$\boldsymbol{\gamma}(0) = \boldsymbol{\gamma}_0. \quad (1b)$$

We define the characteristic map $\mathbf{X}_{[t_1, t_2]}$ associated with the velocity \mathbf{u} to be the solution operator for the characteristic curves, that is,

$$\mathbf{X}_{[t_1, t_2]}(\boldsymbol{\gamma}(t_1)) = \boldsymbol{\gamma}(t_2) \quad (2)$$

for all times t_1, t_2 and for all characteristic curves $\boldsymbol{\gamma}$. One can check that, keeping t_0 fixed, the characteristic map satisfies the following equations

$$\partial_t \mathbf{X}_{[t_0, t]} = \mathbf{u}(\mathbf{X}_{[t_0, t]}, t), \quad (3a)$$

$$(\partial_t + \mathbf{u} \cdot \nabla) \mathbf{X}_{[t, t_0]} = 0. \quad (3b)$$

The map $\mathbf{X}_{[t_1, t_2]}$ can be thought of as a transformation of the space from time t_1 to t_2 following the flow, i.e. vortex lines are transported. There is no requirement that $t_1 < t_2$, if $t_1 < t_2$, the map is forward in time and we will call it the forward map, if $t_1 > t_2$, we call it the backward map. It is straightforward to check the following properties of the characteristic maps:

$$\mathbf{X}_{[t_1, t_2]} \circ \mathbf{X}_{[t_0, t_1]} = \mathbf{X}_{[t_0, t_2]}, \quad (4a)$$

$$\mathbf{X}_{[t_0, t_1]}^{-1} = \mathbf{X}_{[t_1, t_0]}, \quad (4b)$$

$$\mathbf{X}_{[t_0, t_0]} = \mathbf{x}, \quad (4c)$$

for arbitrary t_0, t_1, t_2 . Indeed, for a given divergence-free velocity field \mathbf{u} , the characteristic maps \mathbf{X} are elements of $SDiff(U)$, the Lie group of volume-preserving diffeomorphisms of the domain U , with the space of divergence-free vector fields as its Lie algebra. The underlying theory for the characterization of the Euler equations as geodesic flow in the space of volume preserving diffeomorphisms can be found in the works of Arnol'd [45]. For simplicity of notation, we will denote the forward map $\mathbf{X}_{[0, t]}(\mathbf{x})$ as $\mathbf{X}_F(\mathbf{x}, t)$ and the backward map $\mathbf{X}_{[t, 0]}(\mathbf{x})$ as $\mathbf{X}_B(\mathbf{x}, t)$ when the time-interval of mapping is not emphasized. As a function of t , we can formally see \mathbf{X}_F as the integral curve of the time-dependent velocity field \mathbf{u} on $SDiff(U)$ starting from identity; \mathbf{X}_B is the corresponding inverse element of the group for each time t . The group property (4a) can also be used to express a long-time map as the composition of many short time submaps; this was studied in [1]. Although not completely necessary to the method, this decomposition brings significant numerical advantages since each submap can be computed and stored separately and higher numerical resolution can be obtained through their composition. We call this submap decomposition or remapping step and will discuss this in further details in section 3.5.

The characteristic maps act as solution operators to the transport equations. Consider the advection equation for a scalar ϕ under the velocity field \mathbf{u} :

$$(\partial_t + \mathbf{u} \cdot \nabla)\phi = 0, \quad (5)$$

$$\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}). \quad (6)$$

From the method of characteristics, we know that $\frac{d}{dt}\phi(\boldsymbol{\gamma}(t), t) = 0$ for any characteristic $\boldsymbol{\gamma}$ given by (1). It follows that

$$\phi(\mathbf{x}, t) = \phi_0(\mathbf{X}_B(\mathbf{x}, t)). \quad (7)$$

In geometric terms, since ϕ is Lie advected by \mathbf{u} , i.e. $(\partial_t + \mathcal{L}_{\mathbf{u}})\phi = 0$, we have that ϕ is given by the pullback $\mathbf{X}_B^*\phi_0$. This is also called the relabeling symmetry or back-to-label map. Indeed, \mathbf{X}_B allows us to switch between Lagrangian and Eulerian frames. One can think of \mathbf{X}_B as identifying the characteristic curve passing through \mathbf{x} at time t and returning the location of the corresponding particle in U at time 0 which, by convention, we use as the Lagrangian reference space. Here, ϕ_0 , the initial condition of the scalar, is a 0-form, however this is true for higher degree forms and will be our main tool for solving the Euler equations in vorticity form.

2.2. The Euler equations

We consider the incompressible Euler equations on a three-dimensional domain U , for simplicity, we take U to be the periodic cube \mathbb{T}^3 .

$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} = \frac{1}{\rho} \nabla p, \tag{8a}$$

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{8b}$$

where ρ is the scalar density and p is the pressure. For the incompressible equations, the density is assumed to be constant, in which case the continuity equation (8b) reduces to the divergence-free condition

$$\nabla \cdot \mathbf{u} = 0. \tag{9}$$

We let $\mathbf{w} = \nabla \times \mathbf{u}$ be the vorticity vector field, which is divergence-free by construction. The vorticity equation can be expressed as follows

$$\partial_t \mathbf{w} + (\mathbf{u} \cdot \nabla) \mathbf{w} = (\mathbf{w} \cdot \nabla) \mathbf{u} - \mathbf{w}(\nabla \cdot \mathbf{u}), \tag{10a}$$

$$\nabla \cdot \mathbf{u} = 0. \tag{10b}$$

Since we take a geometric approach, it is more convenient to express the vorticity equation in terms of differential forms. Define the vorticity 2-form $\omega = \star(\mathbf{w}^\flat)$, where \flat is the lowering of the tensor index and \star is the Hodge-star operator (see Lang [46] chapter 14 section 4). Formally, the 2-form ω performs linear measurements on infinitesimal 2D surfaces by dot product of \mathbf{w} with the surface normal; by definition of curl, this measurement yields the total circulation of \mathbf{u} along the surface boundary. We also refer to the book of Arnol'd [47], in particular chapter 7, for a thorough exposition on the relevant differential geometry concepts. In this paper, all vectors are denoted by bold letters, and differential forms, by italic Greek letters. The vorticity equation (10a) is equivalent to the Lie-transport equation of the vorticity 2-form ω :

$$\partial_t \omega + \mathcal{L}_{\mathbf{u}} \omega = 0, \tag{11a}$$

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0. \tag{11b}$$

The Cartan formula for the Lie derivative: $\mathcal{L}_{\mathbf{v}} \phi = d(\iota_{\mathbf{v}} \phi) + \iota_{\mathbf{v}} d\phi$, where $\iota_{\mathbf{v}}$ is the interior product, is used to recover the original vorticity vector equation¹:

$$\begin{aligned} (\star(\partial_t \omega + \mathcal{L}_{\mathbf{u}} \omega))^\sharp &= \partial_t \mathbf{w} + (\star d(\iota_{\mathbf{u}} \omega))^\sharp + (\star \iota_{\mathbf{u}} d\omega)^\sharp = \partial_t \mathbf{w} + \nabla \times (\mathbf{w} \times \mathbf{u}) \\ &= \partial_t \mathbf{w} + \mathbf{w}(\nabla \cdot \mathbf{u}) - (\mathbf{w} \cdot \nabla) \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{w} = 0, \end{aligned} \tag{12}$$

where $\mathbf{w} \times \mathbf{u}$ is the Lamb vector. This means that the vorticity 2-form is conserved in the sense of the Kelvin circulation theorem, i.e. it is Lie-advected or ‘‘frozen into’’ the flow. We note that this uses the general barotropic compressible version of the vorticity equation. In principle, in the incompressible case, the compression term $\mathbf{w}(\nabla \cdot \mathbf{u})$ vanishes, however, this would mean that the Lie-advected vorticity would be allowed to intensify due to numerical errors on the divergence-free condition.

Remark 1. One can alternatively check that with \mathbf{w} evolving strictly under the incompressible equations, the 2-form $\rho \omega$ is Lie-advected, that is¹,

$$(\star(\partial_t \rho \omega + \mathcal{L}_{\mathbf{u}} \rho \omega))^\sharp = \rho(\partial_t \mathbf{w} + (\mathbf{u} \cdot \nabla) \mathbf{w} - (\mathbf{w} \cdot \nabla) \mathbf{u}) = 0,$$

where ρ is still assumed to satisfy the continuity equation (8b) (in case the discretized \mathbf{u} is not exactly divergence-free). Then the vorticity field can be obtained from the Lie-advected $\rho \omega$ by scaling by ρ^{-1} , this cancels any stretching of vortices due to artificial volume compression from numerical errors in $(\nabla \cdot \mathbf{u})$.

¹ Using the following identities:

$$\begin{aligned} (\iota_{\mathbf{v}} \phi)^\sharp &= \mathbf{v} \cdot \phi^\sharp \quad \text{when } \phi \text{ is a 1-form,} & (\iota_{\mathbf{v}} \phi)^\sharp &= (\star \phi)^\sharp \times \mathbf{v} \quad \text{when } \phi \text{ is a 2-form,} \\ (d\phi)^\sharp &= \nabla \phi \quad \text{when } \phi \text{ is a 0-form, and} & (d\phi)^\sharp &= \nabla \times \phi^\sharp \quad \text{when } \phi \text{ is a 1-form.} \end{aligned}$$

In the context of incompressible fluids, the above simply reduces to the statement that the vorticity 2-form is Lie-advected by the velocity field. This gives us an expression of the vorticity as the pullback of the initial condition by the characteristic map:

$$\omega(\cdot, t) = \mathbf{X}_{[t,0]}^* \omega_0, \tag{13}$$

where the superscript asterisk denotes pullback. For a mapping $F : U \rightarrow U$ the pullback F^* is the dual operator to the pushforward operator denoted by the subscript asterisk F_* . The pullback of a k -form η is defined by $(F^*\eta)(\mathbf{v}) = \eta(F_*\mathbf{v})$ where \mathbf{v} is an arbitrary k -vector representing an infinitesimal k -dimensional oriented parallelogram and the pushforward $F_*\mathbf{v}$ is its image under the mapping F . Hence, for the 2-form ω , by the generalized Stokes' theorem, equation (13) is equivalent to the conservation of circulation along all closed curves transported by the forward flow map.

Equation (13) provides many simplifications both numerically and in the analysis mainly due to the fact that pullback commutes with exterior derivatives. For instance, in the study of the Euler equations through Clebsch variables, one makes the simplifying assumption that the initial velocity 1-form is given by $f dg + d\psi$ for some scalar functions f, g and ψ . The initial vorticity is then given by $df \wedge dg$. Applying (13) to this initial condition and commuting pullback and d , we get that the vorticity 2-form at time t is given by

$$\omega(\cdot, t) = d(f \circ \mathbf{X}_{[t,0]}) \wedge d(g \circ \mathbf{X}_{[t,0]}), \tag{14}$$

that is, it is sufficient to solve the advection equations for f and g and reconstruct the vorticity by a cross product of their gradients. We note that the helicity scalar field is defined as $h = \mathbf{u} \cdot \mathbf{w}$ which corresponds to the volume form $\mathbf{u}^\flat \wedge \omega$. This implies that the Clebsch variable representation is limited to cases where h is exact, i.e. $h = d\phi$ for some 2-form ϕ . It follows that total helicity is 0 for flows admitting Clebsch variables, i.e. non-helical flows. A generalized version of the Clebsch approach has been studied in [48], these Generalized Clebsch variables can be used to represent any initial condition, including helical flows. In fact, the initial velocity expansion (18) used in this paper can be seen as a special case of these variables.

For the numerical method described here, we proceed in the following general setting. We assume that there exists closed 1-forms denoted (by abuse of notation) $d\theta_1, d\theta_2, \dots, d\theta_n$ and scalar functions u_1, u_2, \dots, u_n such that the initial velocity 1-form \mathbf{u}^\flat can be expressed as

$$\mathbf{u}^\flat = \sum_{k=1}^n u^k d\theta_k. \tag{15}$$

Then, the initial vorticity form is given by

$$\omega_0 = \sum_{k=1}^n du^k \wedge d\theta_k. \tag{16}$$

This gives us a closed expression for the vorticity depending only on $\mathbf{X}_{[t,0]}$ and the initial condition:

$$\omega(\cdot, t) = \sum_{k=1}^n d(u^k \circ \mathbf{X}_{[t,0]}) \wedge \mathbf{X}_{[t,0]}^* d\theta_k. \tag{17}$$

We do not require that the $d\theta_k$ 1-forms be exact, as long as the pullback is easy to compute. In fact, for the algorithm implemented in this work, on the 3D torus, the $d\theta_k$ forms, with $n = 3$, are simply the coordinate covectors $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$, u_k are the corresponding coordinate values of \mathbf{u}_0 and the pullback $\mathbf{X}_{[t,0]}^* d\theta_k$ is given by $\partial_k \mathbf{X}_{[t,0]}$.

Using the following expansion for the initial velocity

$$\mathbf{u}_0^\flat = u^1(1 \ 0 \ 0) + u^2(0 \ 1 \ 0) + u^3(0 \ 0 \ 1), \tag{18}$$

we get that the vorticity vector at time t is given by

$$\mathbf{w}(\cdot, t) = \sum_{k=1}^3 (\nabla u^k \cdot \nabla \mathbf{X}_{[t,0]}) \times \nabla \mathbf{X}_{[t,0]}, \tag{19}$$

which further simplifies to

$$w^i(\cdot, t) = \epsilon_{ijk} \epsilon_{abc} w_0^a \partial_j \mathbf{X}_{[t,0]}^b \partial_k \mathbf{X}_{[t,0]}^c, \tag{20}$$

in summation notation, where ϵ are the Levi-Civita symbols.

Upon further inspection, the above expression is Cramer's rule expansion of

$$\mathbf{w}(\cdot, t) = \det(\nabla \mathbf{X}_{[t,0]}) (\nabla \mathbf{X}_{[t,0]})^{-1} \mathbf{w}_0(\mathbf{X}_{[t,0]}) = (\nabla \mathbf{X}_{[t,0]})^{-1} \mathbf{w}_0(\mathbf{X}_{[t,0]}), \tag{21}$$

where the determinant factor can be omitted since the maps are volume preserving.

Remark 2. The basis 1-forms $d\theta_k$ are chosen here to express general initial conditions on the torus. In specific cases, for instance in the presence of Clebsch variables, the number of basis 1-forms can be reduced to improve computational performance. That is, the computation of the characteristic map allows for a flexible framework where the vorticity at time t can be constructed by a pullback formula (17), not limited to the formula in (21).

Remark 3. An equivalent formulation can be obtained from a Lagrangian perspective by considering the forward map. Following characteristic curves γ , we see that the vorticity field satisfies

$$\frac{d}{dt} \mathbf{w}(\gamma(t), t) = \nabla \mathbf{u} \cdot \mathbf{w}(\gamma(t), t). \tag{22}$$

Noticing that the gradient of the forward map evolves according to $\partial_t \nabla \mathbf{X}_F = \nabla \mathbf{u} \cdot \nabla \mathbf{X}_F$, one can show that the vorticity field at \mathbf{X}_F is given by

$$\mathbf{w}(\mathbf{X}_F, t) = \nabla \mathbf{X}_F \cdot \mathbf{w}_0(\mathbf{x}). \tag{23}$$

This is Cauchy’s Lagrangian formula used in many Lagrangian particle approaches [16,49]. Composing the above equation with \mathbf{X}_B to return to Eulerian frame and applying the inverse function theorem we get

$$\mathbf{w}(\cdot, t) = (\nabla \mathbf{X}_B)^{-1} \mathbf{w}_0(\mathbf{X}_B). \tag{24}$$

The factor $\det(\nabla \mathbf{X}_B)$ discrepancy with (21) does not show up in the incompressible case since the transformations are volume preserving. In fact, using that $\rho\omega$ is Lie-advected and $\rho(\mathbf{x}, t) = \rho_0(\mathbf{X}_B) \det(\nabla \mathbf{X}_B)$, isolating ω from $\rho\omega$ would remove the determinant factor. The CM method for a compressible flow has been studied in [50] in the context of diffusion-driven density transport.

Remark 4. The evolution of the vorticity 2-form through pullback by \mathbf{X}_B is the infinitesimal expression of the Kelvin circulation theorem which states that the total circulation along a closed curve passively carried by the fluid flow is constant. As a matter of fact, equation (19) can be obtained directly by applying the Kelvin circulation theorem to the definition of the curl operator. Formally, considering that $\mathbf{w} \cdot \mathbf{n} = \lim_{|S| \rightarrow 0} \frac{1}{|S|} \int_{\partial S} \mathbf{u} \cdot d\mathbf{s}$ for some infinitesimal surface S with unit normal \mathbf{n} , we apply the Kelvin circulation theorem to ∂S , moving it to its position and shape at time $t = 0$, to obtain $\mathbf{w} \cdot \mathbf{n} = \lim_{|S| \rightarrow 0} \frac{1}{|S|} \int_{\mathbf{X}_B(\partial S)} \mathbf{u}_0 \cdot d\mathbf{s} = \lim_{|S| \rightarrow 0} \frac{1}{|S|} \int_{\partial S} \mathbf{u}_0(\mathbf{X}_B) \cdot d\mathbf{X}_B(\mathbf{s})$ which yields equation (19) after taking the limit.

This also relates the CM method to the Kelvin-filtered turbulence models which can roughly be summarized by the vorticity equation

$$\partial_t \mathbf{w} + (\mathbf{v} \cdot \nabla) \mathbf{w} = (\mathbf{w} \cdot \nabla) \mathbf{v} \tag{25}$$

where \mathbf{v} is a filtered version of the velocity field \mathbf{u} , for instance $\mathbf{v} = (I - \alpha^2 \Delta)^{-1} \mathbf{u}$. We refer to [51] for a review on the theory of these nonlinearly dispersive equations. In the inviscid case, the vorticity is given through pullback by a modified flow map; for the Kelvin filtered equations, the modified flow map is obtained from the filtered velocity field \mathbf{v} , in the CM framework, the modification on the flow map is a result of a combination of filtering and numerical errors.

Remark 5. The pullback formulation allows us to quickly check the conservation of total helicity. Indeed, given that the vorticity is the curl of the velocity, i.e. $d\mathbf{u}^b = \omega$, we have that by the Helmholtz decomposition theorem, there exists a 1-form η and a scalar ψ such that $\mathbf{u}^b = \eta + d\psi$ and consequently, the vorticity can be written as $\omega = d\eta$. Furthermore, since $\omega = \mathbf{X}_B^* \omega_0$, we also have that there exists some η_0 such that $\eta = \mathbf{X}_B^* \eta_0$ and $d\eta_0 = \omega_0$. The local helicity is given by $\mathbf{u} \cdot \mathbf{w}$ which corresponds to the 3-form $\mathbf{u}^b \wedge \omega = \eta \wedge d\eta + d\psi \wedge d\eta$. Noting that $d\psi \wedge d\eta = d(\psi d\eta)$ is exact and so has vanishing total integral, we have

$$\int_U \mathbf{u}^b \wedge \omega = \int_U \eta \wedge d\eta = \int_U \mathbf{X}_B^*(\eta_0 \wedge d\eta_0) = \int_{\mathbf{X}_B(U)} \eta_0 \wedge d\eta_0 = \int_U \mathbf{u}_0^b \wedge \omega_0. \tag{26}$$

This property relies only on the fact that both η and ω evolve through pullback by the same \mathbf{X}_B .

The derivations in this section allow us to express the evolution of the vorticity field using the characteristic map. The evolution of the characteristic maps is in turn given by the velocity field, which we compute from the vorticity field using the Biot-Savart law $\mathbf{u} = -\Delta^{-1} \nabla \times \mathbf{w}$. The fully coupled vorticity-characteristic map equations are as follows

$$(\partial_t + \mathbf{u} \cdot \nabla) \mathbf{X}_B = 0 \tag{27a}$$

$$\mathbf{w}(\cdot, t) = (\nabla \mathbf{X}_B)^{-1} \mathbf{w}_0(\mathbf{X}_B) \tag{27b}$$

$$\mathbf{u} = -\Delta^{-1} \nabla \times \mathbf{w}. \tag{27c}$$

This system of equations is central to the Characteristic Mapping approach. The numerical method presented here can be summarized as a semi-Lagrangian evolution of the map according to (27a), a direct sampling of the vorticity in (27b) by map interpolation and a pseudospectral computation of the Biot-Savart kernel for the velocity field according to (27c).

3. Numerical implementation

The numerical approach of the CM method for the 3D incompressible Euler equations largely follows the framework of the Gradient-Augmented Level-Set (GALS) [40] and Jet-Scheme methods [52]. The numerical approach is based on the CM method for linear advection [44] and for the 2D Euler equations [1]. It is also related to other mapping based methods used in solid and elastic mechanics [10–13]. In this section, we will first present the general numerical framework for the CM method. We will then discuss the specific implementation details used for the numerical experiments in this work.

The computational method is mainly based on the three equations in (27). Numerically, this corresponds to evolving a characteristic map $\mathcal{X}_{[t_n,0]}$ in a finite dimensional discretization space \mathcal{V} approximating $Diff(U)$, the space of diffeomorphisms on U . Then at every time step, the discretized vorticity $\tilde{\mathbf{w}}^n$ and velocity $\tilde{\mathbf{u}}^n$ will be reconstructed by pullback using $\mathcal{X}_{[t_n,0]}$. For the rest of this paper, we will denote by the script letter \mathcal{X} the approximation of the characteristic map \mathbf{X} in \mathcal{V} , the superscript n on a variable will denote the evaluation of said variable at time t_n and the tilda indicates an approximation or modified equation. The numerical method comprises the three following parts:

1. A discretized velocity field $\tilde{\mathbf{u}}^n$ at time t_n , (assuming the characteristic map $\mathcal{X}_{[t_n,0]}$ is known). This is given by the Biot-Savart law

$$\tilde{\mathbf{u}}^n = -\Delta^{-1} \nabla \times ((\nabla \mathcal{X}_{[t_n,0]})^{-1} \mathbf{w}_0(\mathcal{X}_{[t_n,0]})) \tag{28}$$

computed using spectral Fast-Fourier transform methods.

2. A numerical approximation $\tilde{\mathbf{X}}_{[t_{n+1},t_n]}$ of the one-step map $\mathbf{X}_{[t_{n+1},t_n]}$. For instance, a first order approximation would give

$$\tilde{\mathbf{X}}_{[t_{n+1},t_n]}(\mathbf{x}) \approx \mathbf{x} - \Delta t \tilde{\mathbf{u}}^n(\mathbf{x}). \tag{29}$$

In the current implementation, we use a third-order Runge-Kutta backward in time integrator with a Hermite cubic time-interpolation of the velocity field. A Hermite in time interpolation is used instead of the Lagrange interpolation in [1] to improve accuracy.

3. A time update for the characteristic map based on the group property (4a). The map $\mathcal{X}_{[t_{n+1},0]}$ at time t_{n+1} is given by

$$\mathcal{X}_{[t_{n+1},0]} = \mathcal{H} \left[\mathcal{X}_{[t_n,0]} \circ \tilde{\mathbf{X}}_{[t_{n+1},t_n]} \right], \tag{30}$$

for some interpolation operator $\mathcal{H} : Diff(U) \rightarrow \mathcal{V}$.

In the following sections, we will examine each component of the algorithm in detail.

3.1. Spatial discretization

In the method presented here, the domain $U = \mathbb{T}^3$ is discretized using a Cartesian meshgrid denoted \mathbf{G} . In 3D, we label the grid points of \mathbf{G} by $\mathbf{x}_{i,j,k}$, with each $\mathbf{x}_{i,j,k}$ located at the lower corner of the cell $C_{i,j,k}$. The cells are uniform with cell widths $\Delta x^1, \Delta x^2$, and Δx^3 in the three coordinate directions. Given a grid \mathbf{G} , we define the Hermite cubic interpolation operator (for details see e.g. [5,40]) using piecewise smooth basis functions which are tricubic in each cell $C_{i,j,k}$. These basis functions are constructed by tensor product of the 1D Hermite cubic basis functions Q_i , with Q_0 interpolating function value, and Q_1 , the derivative:

$$q_0(s) = (1 + 2|s|)(1 - |s|)^2, \tag{31a}$$

$$q_1(s) = s(1 - |s|)^2, \tag{31b}$$

$$Q_i(s) = q_i(s) \mathbf{1}_{[-1,1]}(s), \tag{31c}$$

where $\mathbf{1}$ denotes the indicator function. The Q_0 and Q_1 functions form the shape functions corresponding to the function value and derivative interpolation on a 1D grid. They have the property that $\partial^a Q_b(s) = \delta_b^a \delta_s^0$ for $a, b \in \{0, 1\}$ and $s \in \{-1, 0, 1\}$. The 3D shape functions on the grid \mathbf{G} are then defined by

$$H_{\mathbf{a}}^{\mathbf{i}}(\mathbf{x}) = \prod_{m=1}^3 Q_{a_m} \left(\frac{x^m - x_{\mathbf{i}}^m}{\Delta x^m} \right) (\Delta x^m)^{a_m} \tag{32}$$

where $\mathbf{x} = (x^1, x^2, x^3)$, $\mathbf{i} = (i_1, i_2, i_3)$ and $\mathbf{a} = (a_1, a_2, a_3)$.

On grid points \mathbf{x}_r of \mathbf{G} , these basis functions satisfy

$$\partial^{\mathbf{b}} H_{\mathbf{a}}^i(\mathbf{x}_r) = \delta_{\mathbf{a}}^{\mathbf{b}} \delta_r^i. \tag{33}$$

The $H_{\mathbf{a}}^i$ functions are supported on the 8 cells surrounding the grid point \mathbf{x}_i , are tricubic in each cell and are globally C^1 with continuous mixed derivatives of the form $\partial^{\mathbf{b}}$ with $b_m \leq 1$ for $m = 1, 2, 3$. We define the Hermite cubic interpolation operator for a smooth function f to be

$$\mathcal{H}_{\mathbf{G}}[f](\mathbf{x}) = \sum_{\mathbf{a} \in \{0,1\}^3} \sum_{\mathbf{s} \in \mathbf{G}} \partial^{\mathbf{a}} f(\mathbf{x}_i) H_{\mathbf{a}}^i(\mathbf{x}). \tag{34}$$

The minimum regularity requirement for this to be well-defined is that $\partial^{\mathbf{b}} f$ is continuous for all mixed partial derivatives involving at most one derivative in each Cartesian coordinate. We denote $\mathcal{K}^1(U) = \{f \in C^1(U) \mid \partial^{\mathbf{b}} f \in C^0(U) \text{ for } \mathbf{b} \in \{0, 1\}^3\}$. Then $\mathcal{H}_{\mathbf{G}} : C^3(U) \rightarrow \mathcal{K}^1(U)$ is a projection operator. Furthermore, for $f \in C^4(U)$, it is known that

$$\|\partial^{\mathbf{a}} f - \partial^{\mathbf{a}} \mathcal{H}_{\mathbf{G}}[f]\|_{\infty} \lesssim \Delta x^{4-|\mathbf{a}|} |f|_{C^4}, \tag{35}$$

where $|\mathbf{a}| = a_1 + a_2 + a_3$ and $|f|_{C^4} = \sum_{|\mathbf{b}|=4} \|\partial^{\mathbf{b}} f\|_{\infty}$ is the C^4 seminorm; more detailed error bounds can be found in [53].

For the characteristic maps, \mathcal{X} is given by three coordinate functions. We defined the discretization subspace $\mathcal{V} = (\mathcal{K}^1(U))^3$ and use coordinate-wise Hermite cubic interpolation as interpolation operator for the evolution of the characteristic maps. Here we note that \mathcal{V} is not necessarily contained in $SDiff(U)$ or even $Diff(U)$ as there is no *a priori* constraint on the determinant or the invertibility of the interpolated map. However, when interpolating a volume-preserving diffeomorphism, the error on the Jacobian determinant is $\mathcal{O}(\Delta x^3)$ and therefore, for sufficiently well-behaved maps, the interpolant will be a diffeomorphism. For longer time simulations, the characteristic map will develop strong small scale features which cannot be resolved using a fixed grid, in those cases, a remapping method will be employed to decompose the transformation; we will examine this in later sections.

3.2. Velocity interpolation

Using the Hermite cubic interpolation in the previous section, the numerical characteristic map $\mathcal{X}_{[t_n,0]}$ is defined as a diffeomorphism of the domain U . We define the numerical vorticity $\tilde{\mathbf{w}}^n$ through pullback by $\mathcal{X}_{[t_n,0]}$:

$$\tilde{\mathbf{w}}^n(\mathbf{x}) = (\nabla \mathcal{X}_{[t_n,0]})^{-1} \mathbf{w}_0(\mathcal{X}_{[t_n,0]}(\mathbf{x})), \tag{36}$$

where the gradient of $\mathcal{X}_{[t_n,0]}$ is directly evaluated from the interpolant. This defines $\tilde{\mathbf{w}}^n$ as a C^0 vector field on U . The numerical velocity $\tilde{\mathbf{u}}^n$ is in turn computed from the convolution of the Biot-Savart kernel with $\tilde{\mathbf{w}}^n$; we will do this using Fourier spectral methods. We will discretize the velocity field on a grid \mathbf{V} and denote by $\mathcal{F}_{\mathbf{V}}$ the discrete Fourier transform computed by FFT on the grid \mathbf{V} . Sampling $\tilde{\mathbf{w}}^n$ on \mathbf{V} using (36) and applying a forward Fourier transform yields a truncated Fourier series for the vorticity field. Computing the Biot-Savart kernel in frequency space then gives us a Fourier series representation for the velocity. Finally, we define the numerical velocity $\tilde{\mathbf{u}}^n$ as the Hermite interpolant of this truncated Fourier series; this allows us to evaluate the velocity at arbitrary locations in the domain without having to compute the inverse Fourier transform at non-uniform grid points. The definition of the numerical velocity at t_n can be summarized as follows:

$$\tilde{\mathbf{u}}^n(\mathbf{x}) = \mathcal{H}_{\mathbf{V}} \left[\mathcal{F}_{\mathbf{V}}^{-1} \left[-\Delta^{-1} \nabla \times \mathcal{F}_{\mathbf{V}}[\tilde{\mathbf{w}}^n] \right] \right] (\mathbf{x}). \tag{37}$$

In the above equation, it is understood that in order to define the Hermite cubic interpolant for the velocity, the required mixed partial spatial derivatives are computed directly from the Fourier series.

Similarly, from $\partial_t \mathbf{w} = (\mathbf{w} \cdot \nabla) \mathbf{u} - (\mathbf{u} \cdot \nabla) \mathbf{w}$, we can also discretize the time derivative of \mathbf{u} at t_n :

$$\partial_t \tilde{\mathbf{u}}^n(\mathbf{x}) = \mathcal{H}_{\mathbf{V}} \left[\mathcal{F}_{\mathbf{V}}^{-1} \left[-\Delta^{-1} \nabla \times \mathcal{F}_{\mathbf{V}}[(\tilde{\mathbf{w}}^n \cdot \nabla) \tilde{\mathbf{u}}^n - (\tilde{\mathbf{u}}^n \cdot \nabla) \tilde{\mathbf{w}}^n] \right] \right] (\mathbf{x}). \tag{38}$$

The data $\tilde{\mathbf{u}}^n$ and $\partial_t \tilde{\mathbf{u}}^n$ at time steps t_n allow us to locally approximate the velocity using a 4-dimensional time-space Hermite cubic interpolant. For the one-step map (29), we will define an approximate $\tilde{\mathbf{u}}$ in the interval $[t_n, t_{n+1}]$ by extending the interpolant obtained from the velocity data \mathbf{u}^{n-1} and \mathbf{u}^n in the interval $[t_{n-1}, t_n]$. This gives the following definition for the numerical velocity field:

$$\begin{aligned} \tilde{\mathbf{u}}(\mathbf{x}, t) = & \left(q_0(t - t_{n-1}) \tilde{\mathbf{u}}^{n-1}(\mathbf{x}) + q_0(t - t_n) \tilde{\mathbf{u}}^n(\mathbf{x}) \right) \\ & + \Delta t \left(q_1(t - t_{n-1}) \partial_t \tilde{\mathbf{u}}^{n-1}(\mathbf{x}) + q_1(t - t_n) \partial_t \tilde{\mathbf{u}}^n(\mathbf{x}) \right) \quad \text{for } t \in [t_n, t_{n+1}], \end{aligned} \tag{39}$$

using the Hermite basis functions given in (31). We note that since for each n , $\tilde{\mathbf{u}}^n$ and $\partial_t \tilde{\mathbf{u}}^n$ are Hermite interpolants of divergence-free vector fields, the modified velocity field $\tilde{\mathbf{u}}(\mathbf{x}, t)$ is a linear combination of divergence-free velocity fields and is also divergence-free at all time up to interpolation error. This error can be reduced by refining the velocity interpolation grid which can be achieved by a zero-padding in frequency space before taking the inverse Fourier transform.

The one-step map in the interval $[t_n, t_{n+1}]$ is then obtained from the backward in time flow of the approximate velocity field $\tilde{\mathbf{u}}$. We define the numerical one-step map $\tilde{\mathbf{X}}_{[t_{n+1}, t_n]}$ pointwise using a third order Runge-Kutta integration of $\tilde{\mathbf{u}}$:

$$\tilde{\mathbf{X}}_{[t_{n+1}, t_n]} = \int_{t_{n+1}}^{t_n} \tilde{\mathbf{u}}(\tilde{\mathbf{X}}_{[\tau, t_n]}, \tau) d\tau. \tag{40}$$

Together with the time update step given in (30), the time stepping scheme for the map can then be described as a semi-Lagrangian method with a Hermite cubic in time extrapolation of the transport velocity.

The one-step map is used in the time update of the characteristic map (30); it is therefore only evaluated at grid points. However, in order to compute the chain rules for the derivatives required to define the Hermite interpolant, we also need the mixed partial derivatives of $\tilde{\mathbf{X}}_{[t_{n+1}, t_n]}$ at grid points. This is done using a 4th order version of the ϵ -difference scheme described in [54]. The ϵ -difference schemes introduce an L^∞ error of order $\epsilon^4 + \delta \sum_{k=0}^3 \Delta x^k \epsilon^{-k}$ where δ is the machine precision. For all computations presented in this paper, we used $\epsilon = 2.5 \times 10^{-3}$ which corresponds to an error term of at most 10^{-8} and effectively less than 10^{-11} , if $\Delta x < 0.1$.

Remark 6. Evaluating the extrapolation formula (39) at $t = t_{n+1}$ would generally not give the same velocity as $\tilde{\mathbf{u}}^{n+1}$ which is obtained by vorticity pullback followed by the Biot-Savart law. Indeed, the extrapolation of the velocity in the interval $[t_n, t_{n+1}]$ is used only to evolve the characteristic map to time t_{n+1} . The velocity at t_{n+1} is then reconstructed using (36) and (37), similar to a predictor-corrector approach. This also implies that numerical errors in the extrapolation are not directly carried and amplified in the next time step.

3.3. Error estimates

We will examine in this section the numerical error on the characteristic map and its relation to the error on the vorticity field. We will try to characterize the nature of the numerical error and provide some estimates. We use as starting assumption that the numerical map \mathcal{X}_B is consistent with the exact map \mathbf{X}_B in the $C^{1,\alpha}$ norm for some $\alpha \in (0, 1)$, that is the error is $o(1)$. This will allow us to estimate the global $C^{1,\alpha}$ error to third-order in time and space by omitting higher order terms in the error. The consistency assumption is then implied for short-time since the initial numerical map $\mathcal{X}_{[0,0]} = \mathbf{x}$ is exact. In order to preserve the advective structure of the error, we define the following error map:

$$\mathbf{E}_{[0,t_n]} := \mathcal{X}_{[t_n,0]} \circ \mathbf{X}_{[t_n,0]}^{-1} = \mathcal{X}_{[t_n,0]} \circ \mathbf{X}_{[0,t_n]} = \mathcal{X}_B \circ \mathbf{X}_F, \tag{41}$$

which measures by how much the diffeomorphism $\mathcal{X}_{[t_n,0]}$ differs from the inverse of the exact forward flow map $\mathbf{X}_{[0,t_n]}$. Indeed, since the composition of the forward and backward maps $\mathbf{X}_{[t_n,0]} \circ \mathbf{X}_{[0,t_n]} = \mathbf{x}$ gives the identity map, we have that

$$\mathbf{E}_{[0,t_n]} = \mathbf{x} + (\mathcal{X}_{[t_n,0]} - \mathbf{X}_{[t_n,0]}) \circ \mathbf{X}_{[0,t_n]}, \tag{42}$$

that is, the deviation of $\mathbf{E}_{[0,t_n]}$ from the identity map is the numerical error of the map evaluated at the pushforward location. This is a Lagrangian representation of the error since $\mathbf{x} - \mathbf{E}_{[0,t_n]}(\mathbf{x})$ essentially gives the time t_n map error for a particle starting at \mathbf{x} at time 0. We also note that since \mathcal{X}_B is a C^1 diffeomorphism by construction, it follows that all maps considered here are also C^1 diffeomorphisms. Thus left and right inverses exist, are equal and are also diffeomorphisms.

We also define the auxiliary “modified map”

$$\tilde{\mathbf{X}}_{[t_n,0]} := \tilde{\mathbf{X}}_{[t_1,0]} \circ \tilde{\mathbf{X}}_{[t_2,t_1]} \circ \dots \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}, \tag{43}$$

where $\tilde{\mathbf{X}}_{[t_k,t_{k-1}]}$ is the one-step map given in (40) obtained from RK3 integration on the numerical interpolated velocity field.

The full error map is decomposed as follows:

$$\mathbf{E}_{[0,t_n]} = \mathcal{X}_{[t_n,0]} \circ \tilde{\mathbf{X}}_{[t_n,0]}^{-1} \circ \tilde{\mathbf{X}}_{[t_n,0]} \circ \mathbf{X}_{[t_n,0]}^{-1}. \tag{44}$$

We let $\Phi_{[0,t_n]} = \mathcal{X}_{[t_n,0]} \circ \tilde{\mathbf{X}}_{[t_n,0]}^{-1}$ and $\Psi_{[0,t_n]} = \tilde{\mathbf{X}}_{[t_n,0]} \circ \mathbf{X}_{[t_n,0]}^{-1}$ and compute their time-evolution as follows:

$$\begin{aligned} \Phi_{[0,t_n]} &= \mathcal{H}_M \left[\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]} \right] \circ \left(\tilde{\mathbf{X}}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]} \right)^{-1} \\ &= \mathcal{H}_M \left[\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]} \right] \circ \left(\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]} \right)^{-1} \circ \mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_{n-1},0]}^{-1} \\ &= \xi_n \circ \Phi_{[0,t_{n-1}]}, \end{aligned} \tag{45}$$

where we defined a one-step error $\xi_n := \mathcal{H}_M [\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}] \circ (\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]})^{-1}$. We note that ξ_n is the error due to Hermite interpolation since

$$\begin{aligned} \xi_n - \mathbf{x} &= \left(\mathcal{H}_M [\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}] - \mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]} \right) \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}^{-1} \circ \mathcal{X}_{[t_{n-1},0]}^{-1} \\ &= \left(\mathcal{H}_M [\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}] - \mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]} \right) \circ \tilde{\mathbf{X}}_{[t_n,0]}^{-1} \circ \Phi_{[0,t_{n-1}]}^{-1}. \end{aligned} \tag{46}$$

We define the following interpolation error

$$\varphi_n := \mathcal{H}_M [\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}] - \mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}, \tag{47}$$

we then obtain that

$$\Phi_{[0,t_n]} = (\mathbf{x} + (\xi_n - \mathbf{x})) \circ \Phi_{[0,t_{n-1}]} = \Phi_{[0,t_{n-1}]} + \varphi_n \circ \tilde{\mathbf{X}}_{[t_n,0]}^{-1} = \mathbf{x} + \sum_{k=1}^n \varphi_k \circ \tilde{\mathbf{X}}_{[t_k,0]}^{-1}. \tag{48}$$

A similar derivation gives us

$$\Psi_{[0,t_n]} = \eta_n \circ \Psi_{[0,t_{n-1}]}, \tag{49}$$

where $\eta_n := (\tilde{\mathbf{X}}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}^{-1}) \circ (\tilde{\mathbf{X}}_{[t_{n-1},0]} \circ \mathbf{X}_{[t_n,t_{n-1}]})^{-1}$. We note that η is the velocity approximation error which gives the discrepancy between the true flow and the flow obtained from the modified velocity $\tilde{\mathbf{u}}$.

$$\eta_n - \mathbf{x} = \left(\tilde{\mathbf{X}}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}^{-1} - \tilde{\mathbf{X}}_{[t_{n-1},0]} \circ \mathbf{X}_{[t_n,t_{n-1}]}^{-1} \right) \circ \mathbf{X}_{[t_n,0]}^{-1} \circ \Psi_{[0,t_{n-1}]}^{-1}. \tag{50}$$

We define the following modified flow error

$$\psi_n = \tilde{\mathbf{X}}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]} - \tilde{\mathbf{X}}_{[t_{n-1},0]} \circ \mathbf{X}_{[t_n,t_{n-1}]}, \tag{51}$$

i.e. the map evolution error due to errors in the approximated velocity field; we note that this term is approximately $\Delta t \nabla \tilde{\mathbf{X}}_{[t_{n-1},0]} \cdot (\tilde{\mathbf{u}} - \mathbf{u})$. This gives us

$$\Psi_{[0,t_n]} = (\mathbf{x} + (\eta_n - \mathbf{x})) \circ \Psi_{[0,t_{n-1}]} = \Psi_{[0,t_{n-1}]} + \psi_n \circ \mathbf{X}_{[t_n,0]}^{-1} = \mathbf{x} + \sum_{k=1}^n \psi_k \circ \mathbf{X}_{[t_k,0]}^{-1}. \tag{52}$$

This allows us to write the error map as

$$\mathbf{E}_{[0,t_n]} = \left(\mathbf{x} + \sum_{k=1}^n \varphi_k \circ \tilde{\mathbf{X}}_{[t_k,0]}^{-1} \right) \circ \left(\mathbf{x} + \sum_{k=1}^n \psi_k \circ \mathbf{X}_{[t_k,0]}^{-1} \right). \tag{53}$$

We use the fact that for two $C^{1,\alpha}$ diffeomorphisms, \mathbf{f} and \mathbf{g} , the composition $\mathbf{f} \circ \mathbf{g}$ is also $C^{1,\alpha}$ with $\|\mathbf{f} \circ \mathbf{g}\|_{C^{1,\alpha}} \leq C \|\mathbf{f}\|_{C^{1,\alpha}} \|\mathbf{g}\|_{C^{1,\alpha}}^{1+\alpha}$, to estimate the norm of the Lagrangian displacement error $\epsilon_{[0,t_n]} := \mathbf{x} - \mathbf{E}_{[0,t_n]}$:

$$\|\epsilon_{[0,t_n]}\|_{C^{1,\alpha}} \lesssim \sum_{k=1}^n \|\psi_k\|_{C^{1,\alpha}} + \left(1 + \sum_{k=1}^n \|\psi_k\|_{C^{1,\alpha}} \right)^{1+\alpha} \sum_{k=1}^n \|\varphi_k\|_{C^{1,\alpha}}, \tag{54}$$

where the terms dependent on $\mathbf{X}_{[t_n,0]}$ have been absorbed in the constants of the inequality; $\tilde{\mathbf{X}}_{[t_n,0]}$ is also approximated with $\mathbf{X}_{[t_n,0]}$ using the consistency assumption. Here we use the notation $A \lesssim B$ to denote that there exists some constant c such that $A < cB$. In this section, the constant will depend on the dimension, the domain, the constants involved in the norms, and the solution \mathbf{u} as well as \mathbf{X}_B .

The φ_n error is an error pertaining the numerical resolution of \mathcal{X}_B , it can be controlled as long as we can control the higher derivatives of $\mathcal{X}_{[t_{n-1},0]} \circ \tilde{\mathbf{X}}_{[t_n,t_{n-1}]}$. The stability analysis of a similar methods has been studied in [55].

The ψ_n error is a feedback between the map error and the velocity error (and also numerical integration), with $\psi_n \approx \nabla \tilde{\mathbf{X}}_{[t_{n-1},0]} \Delta t (\tilde{\mathbf{u}} - \mathbf{u})$ and $\|\psi_n\|_{C^{1,\alpha}} \lesssim \Delta t \|\tilde{\mathbf{u}}^{n-1} - \mathbf{u}^{n-1}\|_{C^{1,\alpha}}^{1+\alpha}$. We first bound the error on the vorticity which will require the Eulerian version of the error map. Define

$$\mathbf{E}_{[t_n,0]} := \mathbf{X}_{[t_n,0]}^{-1} \circ \mathcal{X}_{[t_n,0]} = \mathbf{X}_{[0,t_n]} \circ \mathbf{E}_{[0,t_n]} \circ \mathbf{X}_{[t_n,0]}, \tag{55}$$

here the conjugation exactly serves the purpose of changing the error from a Lagrangian frame to an Eulerian frame.

We note that

$$\mathbf{w}(\mathbf{x}, t_n) = (\nabla \mathcal{X}_B)^{-1} \mathbf{w}_0(\mathcal{X}_B) \quad \text{and} \quad \tilde{\mathbf{w}}^n(\mathbf{x}) = (\nabla \mathcal{X}_B)^{-1} \mathbf{w}_0(\mathcal{X}_B), \tag{56}$$

So we have that, using $\mathcal{X}_B = \mathbf{X}_B \circ \mathbf{E}_B$ and $(\nabla \mathcal{X}_B)^{-1} = (\nabla \mathbf{E}_B)^{-1} (\nabla \mathbf{X}_B)^{-1}|_{\mathbf{E}_B^{-1}}$

$$\tilde{\mathbf{w}}^n = (\nabla \mathbf{E}_B)^{-1} \mathbf{w}^n(\mathbf{E}_B) \quad \text{i.e.} \quad \tilde{\omega}^n = \mathbf{E}_B^* \omega^n. \tag{57}$$

The numerical vorticity is the pullback of the exact vorticity by the error map. This was expected since the exact map was decomposed into the composition of the numerical map and the error map.

A first look at the velocity error, letting $\epsilon_B = \mathbf{x} - \mathbf{E}_B$ assuming ϵ_B small, we have

$$\begin{aligned} \mathbf{w}^n - \tilde{\mathbf{w}}^n &\approx (\mathbf{I} - (\nabla \mathbf{E}_B)^{-1}) \mathbf{w}^n(\mathbf{E}_B) + \nabla \mathbf{w}^n \cdot (\mathbf{x} - \mathbf{E}_B) \\ &\approx \nabla \epsilon_B \mathbf{w}^n + \nabla \mathbf{w}^n \epsilon_B, \end{aligned} \tag{58}$$

up to second order terms of $\mathcal{O}(\|\epsilon_B\|^2)$.

This gives us $\|\mathbf{w}^n - \tilde{\mathbf{w}}^n\|_{\mathcal{C}^{0,\alpha}} \lesssim \|\epsilon_B\|_{\mathcal{C}^{1,\alpha}}$ for some $\alpha \in (0, 1)$, which after Biot-Savart, yields

$$\|\mathbf{u}^n - \tilde{\mathbf{u}}^n\|_{\mathcal{C}^{1,\alpha}} \lesssim \|\epsilon_B\|_{\mathcal{C}^{1,\alpha}} \tag{59}$$

We use this to control the ψ_n error which is due to the difference between the numerical and the exact velocities as well as the interpolation and integration schemes both of which have 4th order local truncation error.

$$\|\psi_n\|_{\mathcal{C}^{1,\alpha}} \lesssim \Delta t \|\epsilon_{[t_{n-1},0]}\|_{\mathcal{C}^{1,\alpha}}^{1+\alpha} + \mathcal{O}(\Delta t^4). \tag{60}$$

Since the Lagrangian and Eulerian error maps are related by conjugation by the map $\mathbf{X}_{[t_n,0]}$, we have that $\|\epsilon_{[t_n,0]}\|_{\mathcal{C}^{1,\alpha}} \lesssim \|\epsilon_{[0,t_n]}\|_{\mathcal{C}^{1,\alpha}}^{1+\alpha}$ and vice-versa. Therefore, we can write the following estimate for the Eulerian error, up to leading order terms, using (54):

$$\|\epsilon_{[t_n,0]}\|_{\mathcal{C}^{1,\alpha}}^{1/(1+\alpha)} \lesssim \sum_{k=1}^n \Delta t \|\epsilon_{[t_{k-1},0]}\|_{\mathcal{C}^{1,\alpha}}^{1+\alpha} + \sum_{k=1}^n \|\varphi_k\|_{\mathcal{C}^{1,\alpha}}. \tag{61}$$

This can be majorized by the ODE by approximating the discrete sum of order Δt terms with the integral from 0 to $t = n\Delta t$ and taking a time derivative.

$$\dot{\epsilon}_B = (1 + \alpha) \epsilon_B^{\alpha/(1+\alpha)} (\epsilon_B^{1+\alpha} + A), \tag{62}$$

where $A = \mathcal{O}(\Delta t^3) + \frac{1}{\Delta t} \|\varphi_k\|_{\mathcal{C}^{1,\alpha}}$, which is the CM advection \mathcal{C}^1 error and is $\mathcal{O}(\Delta t^3 + \Delta x^2)$.

We note that the Hölder- α norm was introduced artificially to gain the full 2 degrees of regularity from the Poisson equation in the Biot-Savart law. We can therefore pick $\alpha > 0$ arbitrarily small, in which case, the map error estimate ϵ_B in (62) solves a regularly perturbed 1st order linear ODE with a source term of $\mathcal{O}(\Delta t^3 + \Delta x^2)$. We recall that this is built on the assumption that the modified flow map $\tilde{\mathbf{X}}_{[t_n,0]}$ is consistent with $\mathbf{X}_{[t_n,0]}$ so that higher order error terms can be omitted; this is true since the initial $\tilde{\mathbf{X}}_{[t_0,0]}$ is exact and the time-evolution of the error is third-order in Δx and Δt according to the above derivations. It is also assumed that $\mathcal{X}_{[t_n,0]} \circ \tilde{\mathbf{X}}_{[t_{n+1},t_n]}$ is well represented by Hermite interpolation, i.e. that the spatial resolution is high enough and that the grid data of $\mathcal{X}_{[t_i,0]}$ do not oscillate unboundedly. This can be in part controlled by having high enough resolution and also by a remapping method discussed in the following section. The stability of the CM method was also discussed in [40,44,1] and convergence of similar methods using Hermite interpolation was proven in [55]. Overall, the CM method has $\mathcal{O}(\Delta x^2 + \Delta t^3)$ error in \mathcal{C}^1 norm which would translate to a 3rd order global convergence in L^∞ norm.

3.4. Numerical error tests

We provide here two numerical tests for the error estimates derived above. As a sanity check, we test the method on the stationary Arnold-Beltrami-Childress (ABC) flow so that the numerical solution can be compared against a known exact solution. We will also perform a second test using a standard Taylor-Green vortex initial condition, numerical results will be compared against a high-resolution reference test.

Both tests are performed on a periodic domain $[-2\pi, 2\pi]^3$. The ABC flow initial condition is given by

$$\mathbf{w}_0(x, y, z) = \frac{1}{2} \begin{pmatrix} \cos(y) + \sin(z) \\ \cos(z) + \sin(x) \\ \cos(x) + \sin(y) \end{pmatrix}, \tag{63}$$

and the Taylor-Green initial condition is given by

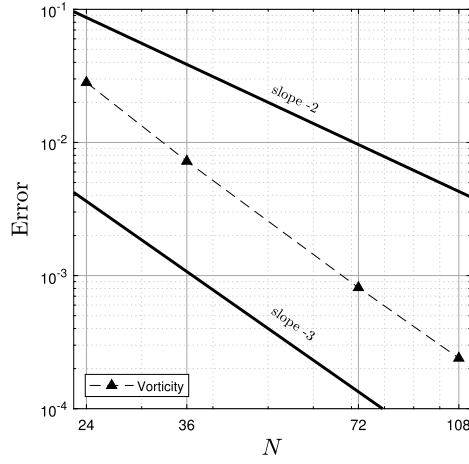


Fig. 1. L^∞ vorticity error for the ABC test at $T_f = 2$. Numerical solution is directly compared to the exact solution $\mathbf{w}(\mathbf{x}, t) = \mathbf{w}_0(\mathbf{x})$.

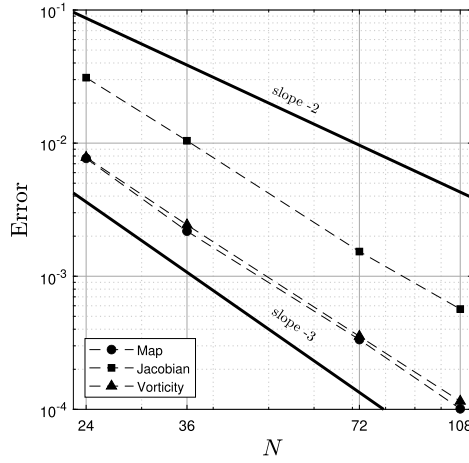


Fig. 2. L^∞ map, Jacobian and vorticity errors for the Taylor-Green Vortex test at $T_f = 2$. Error is calculated by comparing results against a $N = 216$ test.

$$\mathbf{w}_0(x, y, z) = \begin{pmatrix} \cos\left(\frac{x}{2}\right) \sin\left(\frac{y}{2}\right) \sin(z) \\ \sin\left(\frac{x}{2}\right) \cos\left(\frac{y}{2}\right) \sin(z) \\ -\sin\left(\frac{x}{2}\right) \sin\left(\frac{y}{2}\right) \cos(z) \end{pmatrix}. \tag{64}$$

We run each test on increasingly finer grids of N cells per dimension for both \mathbf{M} and \mathbf{V} and using $N/12$ time steps to reach the final time. In both cases, the final time is $T_f = 2$, so that $\Delta x = \frac{4\pi}{N}$ and $\Delta t = \frac{24}{N}$. For the ABC test, we measure the vorticity error using the exact solution $\mathbf{w}^n = \mathbf{w}_0$ and for the Taylor-Green vortex test, we measure the \mathbf{w} , \mathcal{X}_B and $\nabla \mathcal{X}_B$ errors at grid points by comparing against a reference higher resolution test with $N = 216$. We note that for both tests the maximum velocity throughout the simulation is greater than 0.9 at all times so that Δt exceeds the CFL condition. The maximum vorticity for the ABC test is constant in time, for the Taylor-Green test, a 20% growth is observed over the $[0, 2]$ time interval. Figs. 1 and 2 show the L^∞ errors for both tests at $T_f = 2$; the errors are computed directly from grid values and confirm the expected 3rd order error.

3.5. Submap decomposition

The error estimates in section 3.3 show that the error is advective in nature since the numerical solution can be written as the pullback of the exact solution by an error map; therefore, viscous and hyper-viscous type dissipation in the solution are avoided. Indeed, by computing the characteristic map, the vorticity field is provided functionally and can be evaluated anywhere on the domain by interpolating \mathcal{X}_B . Instead of directly evolving the vorticity on some grid, in which case $\tilde{\mathbf{w}}^n$ depends on the grid values of $\tilde{\mathbf{w}}^{n-1}$, the vorticity in the CM method is in principle “reconstructed” at every step using $\tilde{\omega}^n = \mathcal{X}_{[t_n, 0]}^* \omega_0$. For traditional grid-based methods, $\tilde{\mathbf{w}}^n$ is obtained from grid values of $\tilde{\mathbf{w}}^{n-1}$ and typically carries a viscous or hyper-viscous error of the form $\epsilon \Delta^p \tilde{\mathbf{w}}^{n-1}$, in the CM method, the error is instead advective with $\tilde{\omega}^n = \mathbf{E}_B^* \omega^n$, thus

preventing the loss of subgrid scales due to artificial viscosity. One implication is that the CM method can better avoid artificial merging of vortices at the subgrid scale in particular in the study of vortex tube reconnection problems. The error in this case is of elastic type. Indeed, since \mathcal{X}_B is evolved using the GALs method, the leading order spatial errors on the map are of the form $\epsilon \Delta^p \mathcal{X}_B$. This can be seen as an elasticity term which dampens extensive deformation of the domain under the \mathcal{X}_B mapping. The evolution of the elasticity error is governed by the φ term in the error map.

The φ map representation error depends on the \mathbf{M} grid and the regularity of the characteristic map \mathbf{X}_B . For Hermite cubic interpolation, this error roughly scales with the 4th spatial derivative of \mathbf{X}_B . At time $t = 0$, this error is 0 since $\mathbf{X}_{[0,0]}$ is the identity map, then the error increases with time as the characteristic map develops more complicated spatial features. We limit the growth of this error by periodically reinitializing the characteristic map using the group property of the flow maps. Indeed, a time t characteristic map can be decomposed as follows using (4a):

$$\mathbf{X}_{[t,0]} = \mathbf{X}_{[T_1,0]} \circ \mathbf{X}_{[T_2,T_1]} \circ \dots \circ \mathbf{X}_{[T_{m-1},T_{m-2}]} \circ \mathbf{X}_{[t,T_{m-1}]}, \tag{65}$$

for $0 = T_0 < T_1 < \dots < T_{m-1} < T_m = t$. The T_i are remapping times, in each interval $[T_i, T_{i+1}]$, the evolution of the characteristic map is given by the $T_{i+1} - T_i$ time flow of equations (27) with $\mathbf{w}(\mathbf{x}, T_i)$ as initial condition.

Remark 7. Note that the T_i refer to the remapping times and t_n are the time steps of the scheme with $t_n - t_{n-1} = \Delta t$ small, approximating the limit to 0. On the other hand $T_i - T_{i-1}$ is $\mathcal{O}(1)$, its purpose is to subdivide the time interval $[0, T_{final}]$ into shorter subintervals where the characteristic maps are better behaved.

At any t , the vorticity is given by the following pullback using submap decomposition:

$$\mathbf{w}(\mathbf{x}, t) = (\nabla \mathbf{X}_{[t,T_{m-1}]})^{-1} \dots (\nabla \mathbf{X}_{[T_2,T_1]})^{-1} (\nabla \mathbf{X}_{[T_1,0]})^{-1} (\mathbf{w}_0 \circ \mathbf{X}_{[T_1,0]} \circ \dots \circ \mathbf{X}_{[t,T_{m-1}]}) \tag{66}$$

Numerically, this means that we can compute each submap individually and use

$$\tilde{\mathbf{w}}(\cdot, t) = \left(\prod_{i=0}^{m-1} (\nabla \mathcal{X}_{[T_{m-i},T_{m-i-1}]})^{-1} \right) \mathbf{w}_0 (\mathcal{X}_{[T_1,0]} \circ \dots \circ \mathcal{X}_{[t,T_{m-1}]}) \tag{67}$$

to compute the pullback. Each submap will only perform the mapping in the time subinterval $[T_i, T_{i+1}]$ and remapping times can be either fixed or chosen dynamically so that each submap can be well represented using the grid \mathbf{M} . Indeed, after each remapping, we essentially solve a separate Euler equation in the time interval $[T_i, T_{i+1}]$, the characteristic map is reinitialized with the identity map. This means that the spatial representation of the current submap is again exact and will accumulate error over $[T_i, T_{i+1}]$, until an error threshold is exceeded and remapping is triggered. In terms of the error estimates in section 3.3, the remapping resets the φ error of the i^{th} submap to 0, which prevents further accumulation of spatial interpolation errors due to the fixed resolution of the \mathbf{M} grid. One can therefore control the accumulation of the elasticity type error by changing the frequency of the remapping. More frequent remapping reduces the effect that the \mathbf{M} grid has on the spatial features of the map; in the extreme case where remapping is done at every time step, the φ error can no longer accumulate and numerical error is reduced to ψ only. For the numerical experiments presented in this paper, we use the volume-preservation error of the \mathcal{X}_B map as remapping criterion, that is, the remapping time T_i is chosen to be the first time t such that the error $|\det \nabla \mathcal{X}_{[t,T_{i-1}]} - 1|$ is greater than some chosen tolerance. On one hand, this serves as an *a posteriori* estimate of the C^1 error of the map and on the other hand, this allows us to guarantee that the composition of all submaps yields a diffeomorphism and provides some control on the overall volume-preserving property of the characteristic map.

3.6. Implementation summary

The previous subsections contain the numerical tools for implementing the CM method for the 3D incompressible Euler equations, we give here a short summary of the method in pseudocode format. We note that the method uses two discretization grids, a grid \mathbf{M} for representing the numerical map $\mathcal{X}_{[t_n,0]}$ and a grid \mathbf{V} for sampling the numerical vorticity \mathbf{w}^n and computing the Biot-Savart law using Fourier spectral methods. These two grids do not need to have the same resolution, in fact, using the submap decomposition method described in the previous subsection, a short time characteristic map can be represented on a coarse grid \mathbf{M} . The \mathbf{V} grid used to represent the vorticity needs to be fine enough to avoid sampling errors. Indeed, as the flow evolves, the vorticity can develop small scale features and high gradients. If \mathbf{V} is not fine enough to resolve \mathbf{w}^n , this can cause aliasing errors in the Fourier transform. One way to reduce the effect of under-sampling is to mollify \mathbf{w}^n in Fourier space which was studied in [1], however this also reduces the accuracy of the scheme. Other possibilities for future investigation include the use adaptive mesh for the vorticity sampling and wavelet methods for the computation of the Biot-Savart law [56,57]. The CM method for 3D incompressible Euler equations can be summarized with the following pseudocode algorithm.

Algorithm 1 CM method 3D incompressible Euler equations.

Input: Initial vorticity \mathbf{w}_0 , grids \mathbf{M} and \mathbf{V} , time step Δt , final time T_f

- 1: Initialize $n \leftarrow 0, t_n \leftarrow 0, m \leftarrow 0, T_m \leftarrow 0$.
- 2: **while** $T_m < T_f$ **do**
- 3: $\mathcal{X}_{[t_n, T_m]} \leftarrow \mathbf{x}$ (identity map)
- 4: **while** $\|\det \nabla \mathcal{X}_{[t_n, T_m]} - 1\|_\infty < TOL$ **do**
- 5: Sample \mathbf{w}^n on grid points of \mathbf{V} . ▷ using (67)
- 6: $\mathcal{F}_V[\tilde{\mathbf{u}}^n] = -\Delta^{-1} \nabla \times \mathcal{F}_V[\tilde{\mathbf{w}}^n]$.
- 7: Compute $\partial^b \tilde{\mathbf{u}}^n$ for $\mathbf{b} \in \{0, 1\}^3$ in Fourier space. Define $\tilde{\mathbf{u}}^n(\mathbf{x})$. ▷ using (37)
- 8: Compute $(\tilde{\mathbf{w}}^n \cdot \nabla) \tilde{\mathbf{u}}^n - (\tilde{\mathbf{u}}^n \cdot \nabla) \tilde{\mathbf{w}}^n$ on \mathbf{V} and define $\partial_t \tilde{\mathbf{u}}^n(\mathbf{x})$. ▷ using (38)
- 9: Define $\tilde{\mathbf{u}}(\mathbf{x}, t)$, by linear combination of spatial interpolants. ▷ using (39)
- 10: Compute $\partial^b \tilde{\mathbf{X}}_{[t_n + \Delta t, t_n]}$ on \mathbf{M} using RK3 integration of $\tilde{\mathbf{u}}$. ▷ using (40)
- 11: Update characteristic map $\mathcal{X}_{[t_{n+1}, T_m]} \leftarrow \mathcal{H}_M \left[\mathcal{X}_{[t_n, T_m]} \circ \tilde{\mathbf{X}}_{[t_n + \Delta t, t_n]} \right]$.
- 12: $t_{n+1} \leftarrow t_n + \Delta t, n \leftarrow n + 1$.
- 13: **end while**
- 14: $T_{m+1} \leftarrow t_n, m \leftarrow m + 1$.
- 15: **end while**

Remark 8. Since the discrete vorticity evaluation at line 5 in Algorithm 1 will eventually produce aliasing errors from undersampling, we will usually introduce a low-pass filter or a Fourier truncation at lines 6 and 8. This is the effective scale cut-off of the velocity field governing the discrete flow. As a rule of thumb, we pick this low-pass filter to be the coarsest scale in the discretization and the grid \mathbf{V} to be the finest scale with \mathbf{M} at an intermediate scale. This is to ensure that the map grid has enough resolution to represent a short-time deformation generated by the filtered velocity field, and also that the \mathbf{V} grid is fine enough so that the Hermite interpolation of the filtered velocity is accurate and preserves the divergence-free property.

4. Numerical tests

In this section, we present several numerical tests computed using the CM method for the 3D incompressible Euler equations. The algorithm is implemented in C using OpenMP parallelization and Discrete Fourier transforms are performed using the FFTW library [58]. The tests in this section are performed on a personal computer with an AMD Ryzen 7 4800H CPU with 8 cores (16 threads) and 16GB of RAM. All interpolation routines are implemented with parallel for-loops iterating over all query points using all 16 threads and all FFT routines also use the multi-threaded implementation included in the FFTW package; for these tests, a wallclock computation time is recorded. The larger FFT computations for the spectrum plots are performed on a cluster computer.

4.1. Perturbed antiparallel vortex tubes

The question of finite-time blow-up in the solution of the 3D incompressible Euler equations is an important open problem in mathematics. One extensively studied initial condition for potentially generating a finite-time blow-up are the perturbed antiparallel vortex tubes studied by Kerr in 1993 [37]. In the viscous case, for the Navier-Stokes equations, this initial condition evolves into a vortex reconnection in the process of which a topological change of the vortex cores occurs.

The initial condition can be constructed as the pullback of two antiparallel vortex tubes by a shear-deformation of the $[-2\pi, 2\pi]^3$ periodic domain. The initial vorticity field is antisymmetric across the $z = 0$ plane with each half-space containing a vortex tube of opposite orientation. We construct the initial condition \mathbf{w}_0 as follows. Consider the unperturbed vortex tube in the $z > 0$ half-space given by

$$\boldsymbol{\varphi}_+(x, y, z) = \exp\left(\frac{-r^2}{1-r^2} + r^4(1+r^2+r^4)\right) \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{if } r < 1, \tag{68}$$

and is the zero vector if $r \geq 1$. Here, r is the scaled distance from the vortex core given by

$$r(x, y, z) = R^{-1} \sqrt{(x-x_0)^2 + (z-z_0)^2}. \tag{69}$$

This forms a vortex tube oriented in the y -direction centered at $x = x_0, z = z_0$ and supported in a tube of radius R . One can check that this initial condition is divergence-free and therefore a valid vorticity field on the flat 3-torus. The pair of antiparallel vortex tubes is given by

$$\boldsymbol{\varphi}(x, y, z) = \boldsymbol{\varphi}_+(x, y, z) - \boldsymbol{\varphi}_+(x, y, -z). \tag{70}$$

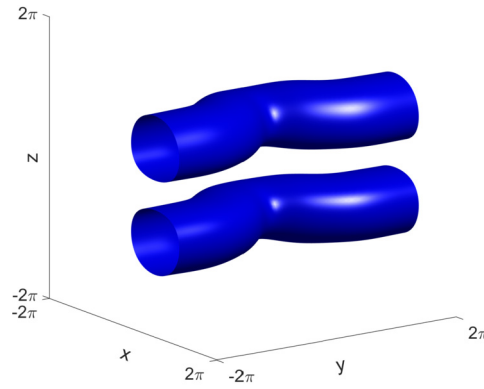


Fig. 3. Initial vortex tubes for the Kerr initial condition [37]. The figure shows the level-set surface of $|\mathbf{w}_0|$ at 0.4015 which is 60% of the maximum value.

The vortex tubes are perturbed by the following domain deformation:

$$\mathbf{T} : \begin{pmatrix} x \\ y \\ z \end{pmatrix} \mapsto \begin{pmatrix} x + \delta_x \cos\left(\frac{\pi}{L_x} s(y)\right) \\ y \\ z + \delta_z \cos\left(\frac{\pi}{L_z} s(y)\right) \end{pmatrix}, \tag{71}$$

where

$$s(y) = y + L_y \delta_{y2} \sin(\pi y/L_y) + L_y \delta_{y1} \sin(y + L_y \delta_{y2} \sin(\pi y/L_y)). \tag{72}$$

The perturbed vortex tubes are then defined as the pullback $(\mathbf{T}^{-1})^* \varphi$. A closed form expression can be obtained using $(\mathbf{T}^{-1})^* \varphi = \nabla \mathbf{T}|_{\mathbf{T}^{-1}} \varphi(\mathbf{T}^{-1})$ and the fact that \mathbf{T} is a shear-deformation, and for a fixed y , \mathbf{T} is simply a translation on the x - z plane.

The initial vorticity \mathbf{w}_0 used in [37,36,39] is defined as a filtered and rescaled version of the above perturbed vortex tubes given by

$$\mathbf{w}_0 = 8K * (\mathbf{T}^{-1})^* \varphi. \tag{73}$$

The exact expression for the filter K might have been slightly different in references [37] and [36], here we use the filter K defined in Fourier space by $\hat{K}(\xi) = \exp(-0.05(\xi_1^4 + \xi_2^4 + \xi_3^4))$ where ξ_i are the integer wave numbers. The specific parameters for the initial condition, taken from [36], are $R = 0.75$, $\delta_{y1} = 0.5$, $\delta_{y2} = 0.4$, $\delta_x = -1.6$, $\delta_z = 0$, $x_0 = 0$, $z_0 = 1.57$, $L_x = L_y = 4\pi$, $L_z = 2\pi$. Fig. 3 shows a level-set surface of the initial condition.

We first perform a low-resolution simulation using a personal computer. The simulation is carried out on a \mathbf{M} grid of $64 \times 48 \times 32$ points and vorticity sampling is computed on a \mathbf{V} grid of $96 \times 72 \times 48$ points; here the z -direction is most finely sampled as important higher frequencies are expected to be produced in that direction, similarly, the y -direction has the coarsest representation. The resulting Fourier series is truncated at a radius of 32, the time steps are fixed at $1/50$ and the Jacobian determinant error tolerance $|\det \nabla \mathcal{X}_B - 1|$ for the remapping is set at 10^{-3} . The initial condition is defined on a 128^3 grid, we then compute its mixed-partial derivatives in Fourier space in order to define a Hermite cubic interpolant. The simulation is run until time $t = 17$ requiring a total of 79 submaps, taking under an hour of wallclock computation time. Fig. 4 shows the evolution of the vortex cores throughout the simulation and Table 1 contains the energy and helicity conservation errors as well as total enstrophy and maximum vorticity and velocity evaluated at regular time intervals. The energy is defined as the squared L^2 norm of the velocity $\|\mathbf{u}\|_{L^2}^2$ in the $[-2\pi, 2\pi]^3$ domain, the total enstrophy is $\|\mathbf{w}\|_{L^2}^2$ and the helicity is defined as the L^2 inner-product of the velocity and the vorticity, $H := (\mathbf{u}, \mathbf{w})_{L^2}$.

Remark 9. We note here that the Fourier truncation or filtering used in the computation of the velocity field from the sampled vorticity is not related to the 2/3 rule typically used in Fourier pseudo-spectral methods, whose purpose is to dealias spurious modes generated by the frequency convolutions when computing the nonlinear term in physical space in each time step. In the CM method, there is no direct time-stepping of the velocity and vorticity fields, at each step, the vorticity is reconstructed by direct sampling of the functional expression in equation (67). Since all map computations are carried out on a coarse grid \mathbf{M} , the purpose of the filtering is to ensure that the velocity field defined from the sampled vorticity is sufficiently band-limited so that the backward flow map it generates is regular enough to be accurately represented on \mathbf{M} . In the extreme cases where the vorticity field exhibits important subgrid scales, essentially discontinuous from a numerical point of view, this filtering can help prevent the Gibbs phenomenon from generating spurious oscillations in the entire domain. Ultimately, the size of the truncation would scale with the resolution of \mathbf{M} to maintain consistency.

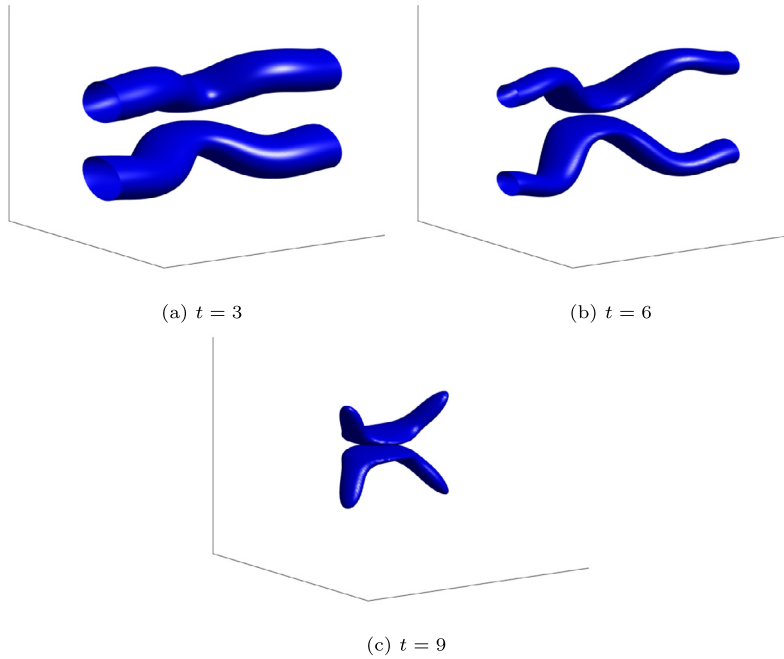


Fig. 4. Evolution of the vortex cores for the Kerr initial condition. We show level-set surfaces of $|w|$ at 0.4105, 0.5435 at 0.7519 for times 3, 6 and 9 respectively, which is 60% of the maximum value. Figures are generated using a 128^3 grid.

Table 1

Evolution of total enstrophy, energy conservation relative error (divided by initial energy), helicity conservation error (initial helicity is 0), maximum vorticity and velocity, number of remaps and wallclock computation time of the Kerr initial condition using CM method for 3D Euler. Grid resolutions: $64 \times 48 \times 32$ for M , $96 \times 72 \times 48$ for V , $\Delta t = 1/50$, Fourier truncation at radius 32, remapping Jacobian determinant tolerance at 10^{-3} . All data in this table are evaluated using a grid of resolution 256^3 .

t	$\ w\ _{L^2}^2$	$\ u\ _{L^2}^2 / \ u_0\ _{L^2}^2 - 1$	$H - H_0$	$\ w\ _{L^\infty}$	$\ u\ _{L^\infty}$	n_{maps}	time (s)
0	67.2181	0.000	0.000	0.6691	0.7393	1	0
1	67.0990	1.341×10^{-6}	-5.619×10^{-14}	0.6666	0.7272	1	102
2	67.3909	1.744×10^{-6}	-1.717×10^{-13}	0.6715	0.7131	1	206
3	68.0925	1.989×10^{-7}	-2.849×10^{-13}	0.6841	0.7019	1	310
4	69.1954	-1.406×10^{-6}	3.333×10^{-13}	0.7387	0.7171	2	416
5	70.6844	-1.663×10^{-6}	5.680×10^{-13}	0.8181	0.7350	2	524
6	72.5405	-2.369×10^{-6}	8.913×10^{-13}	0.9059	0.7501	3	633
7	74.7460	-3.141×10^{-6}	1.129×10^{-12}	1.0046	0.7622	4	746
8	77.2912	-4.061×10^{-6}	1.218×10^{-12}	1.1179	0.7715	5	861
9	80.1845	-5.020×10^{-6}	-2.656×10^{-12}	1.2532	0.7785	6	981
10	83.4644	-6.245×10^{-6}	-7.347×10^{-12}	1.4181	0.7833	8	1108
11	87.2154	-7.793×10^{-6}	-4.307×10^{-12}	1.6296	0.7864	10	1240
12	91.5883	-1.012×10^{-5}	-3.304×10^{-12}	1.9121	0.7885	13	1382
13	96.8361	-1.246×10^{-5}	8.511×10^{-12}	2.2935	0.7920	18	1538
14	103.3978	-8.680×10^{-6}	3.923×10^{-11}	2.8495	0.8013	25	1719
15	112.1115	1.893×10^{-5}	4.295×10^{-11}	3.8549	0.8223	37	1935
16	124.5780	9.829×10^{-5}	-2.643×10^{-11}	5.6541	0.8545	56	2208
17	143.8254	2.633×10^{-4}	1.910×10^{-10}	9.1819	0.9052	79	2558

This filtering is not always necessary, without filtering, the effective truncation of the Fourier series for the velocity will be the grid size of M as higher frequencies in the flow cannot be represented on M . However, our numerical experiments suggest that a small amount of smoothing generates better results with more accurate energy and helicity conservation.

Remark 10. We note that the vorticity maxima $\|w\|_{L^\infty}$ shown in Table 1 are lower bounds since they are evaluated using a 256^3 grid. The actual maximum vorticity of the numerical solution is higher. Using the arbitrary resolution property of the method, we can improve our lower bound by recursively refining the vorticity sampling around the estimated numerical maxima. For instance, on a grid of N^3 points, we can locate the vorticity maximum on the grid and resample the vorticity in a region of size $3\Delta x$, again using a grid of N^3 points. With $N = 256$, 3 iterations of the above procedure allows us to

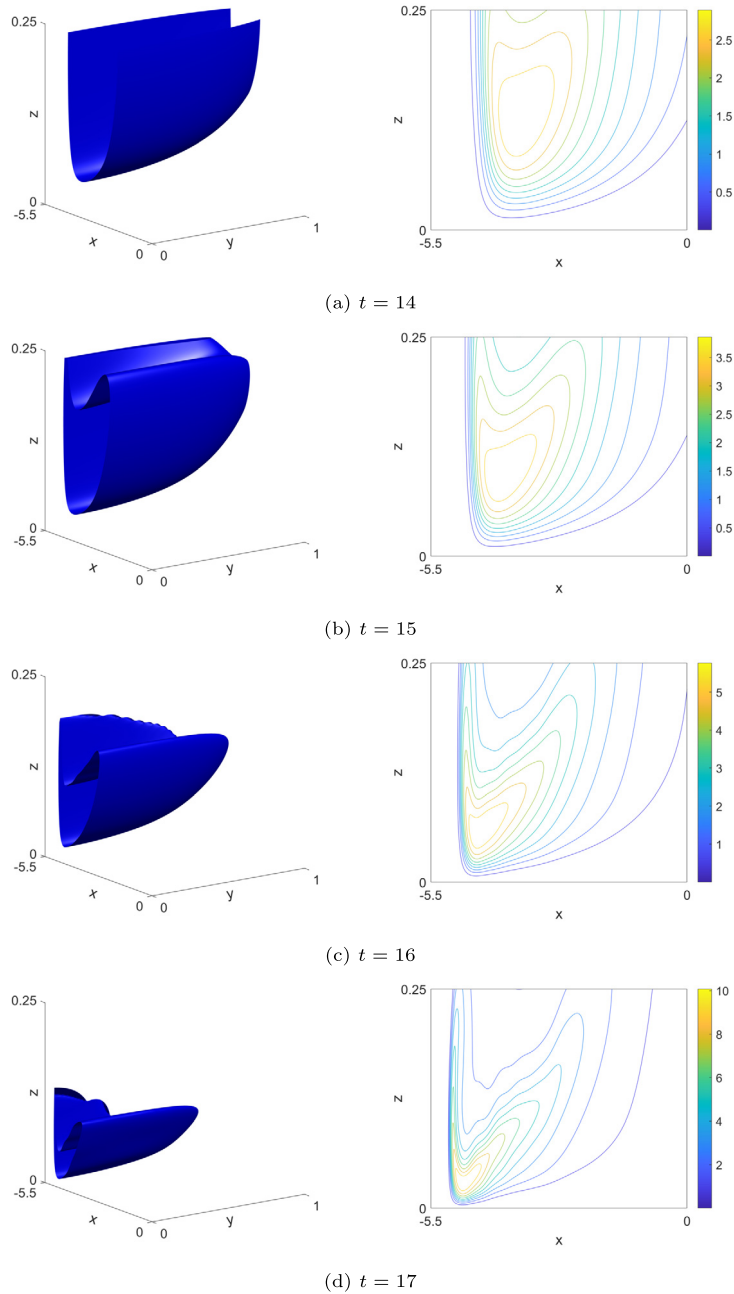


Fig. 5. Zoomed view of the vortex core from $t = 14$ to 17 and contour plot of the vorticity strength $|\mathbf{w}|$ across the symmetry plane $y = 0$; 10 isoline values evenly distributed from 0 to 2.8929, 3.8618, 5.7677 and 10.0798 are shown for times 14, 15, 16, and 17, respectively. Viewed domain: $[-5.5, 0] \times [0, 1] \times [0, 0.25]$. Contour plots are generated using 512^2 2D grids. We note that this is a much finer local sampling of the vorticity field compared to the data in Table 1 and thus able to resolve a higher vorticity maximum (see Remark 10). (Color online)

bound the vorticity maximum from below by 3.8618, 5.7674 and 9.5185 for times 15, 16 and 17 respectively. This shows reasonable agreement with the high resolution reference computations performed in [36].

The functional definition of the vorticity field through the pullback by \mathcal{X}_B provides arbitrary resolution of the solution independently of the discretization grids. This allows us to zoom in on the solution, in particular for larger times t where the vorticity starts developing significant small scale features. Fig. 5 shows zoomed views of the vortex tubes and contour plots of the vorticity intensity across the symmetry plane $y = 0$. We note that the vertical length of the viewed domain is 0.25, a bit more than the width of a single cell of the map grid \mathbf{M} . The domain deformation at time $t = 17$ cannot be represented properly using a single map on the \mathbf{M} grid, however, through the dynamic remapping method, the full deformation at time

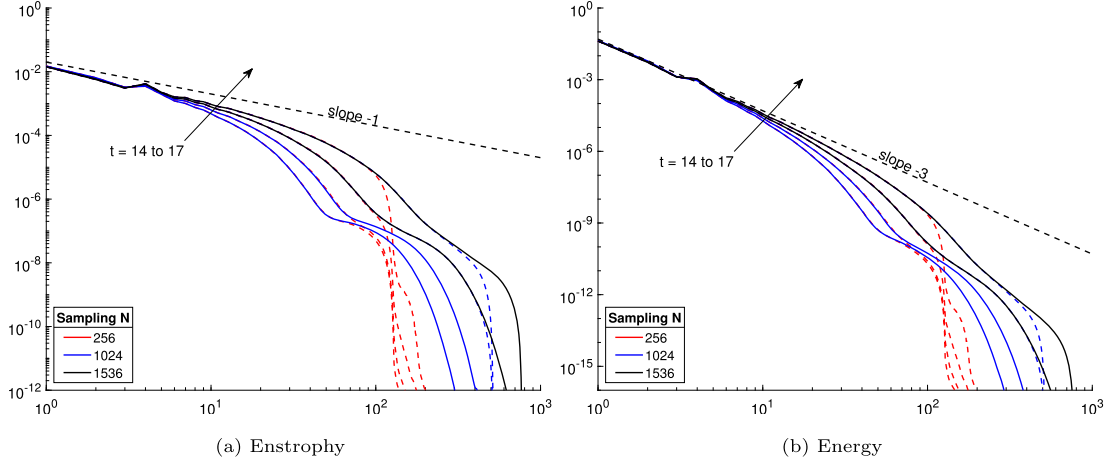


Fig. 6. Entropy and energy spectra at times $t = 14, 15, 16$ and 17 . Dotted black lines are the k^{-1} and k^{-3} curves, respectively. Successive finer sampling are shown, red curves are obtained from 256^3 sampling, blue curves from 1024^3 and black curves from 1536^3 sampling. The highest resolution sampling for each time t is drawn in full-line, coarser samplings for the same t are shown as dotted-lines. (Color online)

$t = 17$ can be represented using the composition of 79 short-time submaps defined on a coarse $64 \times 48 \times 32$ grid. The vorticity field $\tilde{\mathbf{w}}$ defined by pullback through the 79 submaps is therefore able to exhibit small scale features and high gradients as shown in Fig. 5.

As seen from Table 1, the remapping routine is triggered more frequently as the simulation approaches $t = 17$, meaning that the Jacobian determinant error accumulates at an increasing rate in part due to a lack of spatial resolution. One often used measurement of the smoothness of the solution is the isotropic spectrum of the Fourier series given by

$$Z(k) = \frac{1}{2} \sum_{|\xi| \in [k-\frac{1}{2}, k+\frac{1}{2})} |\mathcal{F}[\tilde{\mathbf{w}}](\xi)|^2 \quad (74a)$$

$$E(k) = \frac{1}{2} \sum_{|\xi| \in [k-\frac{1}{2}, k+\frac{1}{2})} |\mathcal{F}[\tilde{\mathbf{u}}](\xi)|^2 \quad (74b)$$

We plot the entropy and energy spectra of the solution at times $t = 14$ to 17 in Fig. 6.

The spectrum plots in Fig. 6 demonstrate a key property of the CM method: that the resolution scale of the map is not the dissipation scale of the vorticity solution, the vorticity field itself is not being dissipated in a viscous manner, and therefore can be resolved to arbitrary resolution. Indeed, although all computations were carried out on coarse grids and the Fourier support of the velocity field evolving the map is only a ball of radius 32, we can reconstruct the vorticity field by computing a fine grid pullback from evaluating the maps and applying equation (67). Fig. 6 shows that the vorticity and velocity fields obtained this way have the expected decay in their Fourier transforms. These can be compared with Figs. 17 and 18 in [36] and general agreement is found. The vorticity field $\mathbf{w}^n = (\nabla \mathcal{X}_{[t_n, 0]})^{-1} \mathbf{w}_0(\mathcal{X}_{[t_n, 0]}(\mathbf{x}))$, defined functionally (i.e. \mathcal{X}_B evaluated by interpolation and \mathbf{w}_0 by direct function evaluation), contains arbitrary fine scales until round-off errors dominate. The reason for this arbitrary resolution is twofold: Firstly, since the discretized quantity is the characteristic map, the pointwise definition of the vorticity field can in fact be viewed as Lagrangian. Indeed, consider a particle starting at position \mathbf{y} at time $t = 0$, its position at time t is given by the forward map $\mathbf{X}_F(\mathbf{y}, t)$ and the vorticity field at the particle location is given by $\mathbf{w}(\mathbf{X}_F(\mathbf{y}, t), t) = \nabla \mathbf{X}_F(\mathbf{y}, t) \cdot \mathbf{w}_0(\mathbf{y})$. In order to obtain the vorticity at an Eulerian point \mathbf{x} at time t , we plug in $\mathbf{y} = \mathbf{X}_B(\mathbf{x}, t)$ and use the inverse property (4b) to get $\mathbf{w}(\mathbf{x}, t) = (\nabla \mathbf{X}_B)^{-1} \mathbf{w}_0(\mathbf{X}_B)$. This means that by interpolating the discrete map \mathcal{X}_B at an Eulerian point \mathbf{x} , we are approximating a particle path and the associated local material deformation of a Lagrangian particle whose time t position is \mathbf{x} ; the time t vorticity is then directly constructed from the initial condition by applying the material deformation. This can also be expressed using the error map studied in 3.3, where the numerical vorticity field can be written as the pullback of the exact vorticity by an error map \mathbf{E}_B : $\tilde{\omega}^n = \mathbf{E}_B^* \omega^n$. Therefore, since \mathbf{E}_B is a C^1 diffeomorphism, as long as the error is controlled, the pullback $\mathbf{E}_B^* \omega^n$ will not destroy small scales. The functional definition of $\tilde{\mathbf{w}}^n$ by pullback therefore allows us to oversample the vorticity on a 1536^3 grid even though all computations were carried out on much coarser grids. For traditional Eulerian methods, in order to preserve these small scales and prevent large artificial dissipation, the vorticity field will have to be discretized and evolved on a 1536^3 grid throughout the entire computation. With the CM method, these scales are not lost to dissipation and can be obtained by a fine grid sampling. Secondly, the map error can be controlled using the submap decomposition method made possible by the group structure of the characteristic maps. Indeed, the map error \mathbf{E}_B arises in part from the error in approximating $SDiff(U)$ by a finite-dimensional interpolation space \mathcal{V} . As the flow develops, the small scale features in

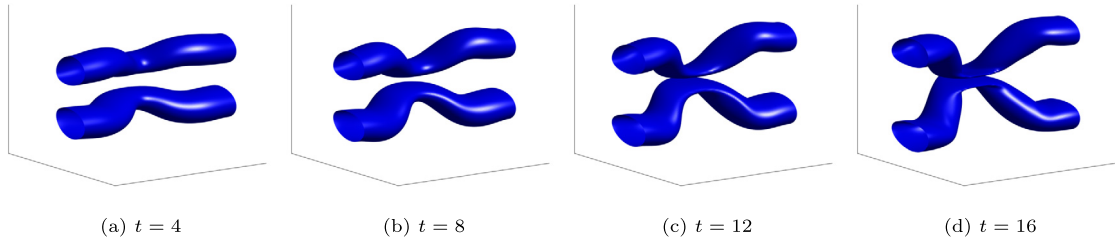


Fig. 7. Transport of the initial vortex core. We show level-set surfaces of ϕ at 0.4015, which is 60% of the maximum value. Figures are generated using a 128^3 grid.

\mathbf{X}_B not resolved in \mathcal{V} increase, adding to the E_B error. Through the remapping method, this representation error is reset to 0 for each submap since the initial condition for each new submap is the identity map, which is represented exactly in \mathcal{V} . Appropriate remapping therefore guarantees that each submap $\mathbf{X}_{[T_i, T_{i+1}]}$ can be well represented in \mathcal{V} and that its numerical error remains in the asymptotic regime, i.e. the omitted small scales are not significant enough to pollute the large scale, lower frequency features which carry most of the energy. The resulting global-time map \mathcal{X}_B is obtained as the composition of n_{maps} submaps; \mathcal{X}_B can be seen as an element of $\mathcal{V}^{n_{maps}}$ and therefore is able to represent the small scale features generated by the long-time flow through the composition of coarse grid maps.

Remark 11. The arbitrary resolution property of the method mainly refers to its ability of resolving spatial scales significantly smaller than the computational scales, as seen in the zoomed views in Fig. 5 and in the high frequency modes in the spectra in Fig. 6. Moreover, this resolution property also allows us to use different discretization scales for the map and for the velocity computation as was studied in [1], where highly different spatial resolutions were used. Here, due to more expensive computational requirements for 3D simulations, we have limited the vorticity sampling grid in this section to 50% finer than the map grid. A larger numerical error is therefore expected, especially towards larger times as the solution develops more and more finer scales that are not resolved on the vorticity sampling grid.

Another feature of the CM method is that we have access to the solution operator \mathbf{X}_B of the advection under the velocity field \mathbf{u} , this means that we can evolve passively advected quantities at no additional computational cost. This has several applications such as tracking passively transported fluid quantities or solute densities or visualization of the fluid flow. As example we solve the following scalar advection problem using the initial vorticity strength $|\mathbf{w}_0|$ as initial condition:

$$(\partial_t + \mathbf{u} \cdot \nabla)\phi = 0 \tag{75a}$$

$$\phi(\mathbf{x}, 0) = |\mathbf{w}_0(\mathbf{x})|. \tag{75b}$$

From 2.1, the solution to this advection equation is given by $\phi(\mathbf{x}, t) = |\mathbf{w}_0 \circ \mathbf{X}_{[t, 0]}|$. This gives us the evolution of the initial vortex strength as a passively advected quantity. In Fig. 7, we show a level-set surface of ϕ at 60% the maximum value; this allows us to track the motion of the initial vortex core transported under the fluid flow. We note that this does not correspond to the evolution of the actual vortex core as the vortex stretching can play an important role in moving the location of the vortex core.

4.2. Perturbed perpendicular vortex tubes

Another test we performed is the merging of two perturbed perpendicular vortex tubes inspired by the tests in [18]. The initial condition is constructed in a similar fashion as in section 4.1. The unperturbed vortex tube is given by (68) with $R = 0.5, x_0 = 0, z_0 = -1$. We apply the perturbation transformations $\mathbf{T} : [-2\pi, 2\pi]^3 \rightarrow [-2\pi, 2\pi]^3$ given by a sinusoidal shear deformation $\mathbf{T} : x \mapsto x - 0.5 \sin(0.5y)$ and a reflection and translation map $\mathbf{R} : (x, y, z) \mapsto (y, x, z + 2)$. The two vortex tubes are then defined as

$$\boldsymbol{\varphi} = (\mathbf{T}^{-1})^* \boldsymbol{\varphi}_+ + (\mathbf{R}^{-1})^* (\mathbf{T}^{-1})^* \boldsymbol{\varphi}_+, \tag{76}$$

which corresponds to a sinusoidal vortex tube in the y -direction through $(x, z) = (0, -1)$ combined with a reflected tube in the x -direction through $(y, z) = (0, 1)$. The initial vorticity field is given as a scaled and filtered version of $\boldsymbol{\varphi}$ where the filter is the same as the one used in section 4.1,

$$\mathbf{w}_0 = 24K * \boldsymbol{\varphi}. \tag{77}$$

Fig. 8 shows a level-set surface of this initial condition.

These two vortex tubes will start rotating around each other, creating high vorticity gradients and significant small scale features. In the viscous case, the vortices are expected to collide at the intersection. Fig. 9 shows the evolution of the vortex

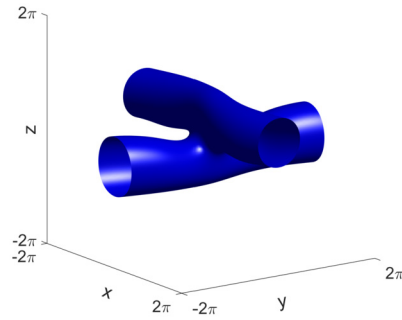


Fig. 8. Initial condition for the perpendicular vortex tubes test. The figure shows the level-set surface of $|\mathbf{w}_0|$ at 0.5402, which is 60% of the maximum value.

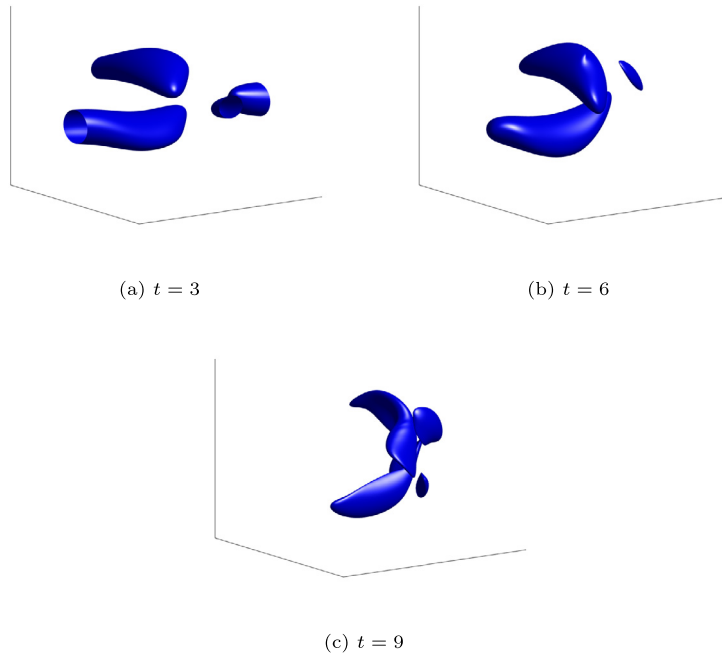


Fig. 9. Evolution of the vortex cores for the perpendicular vortex tubes test. We show the level-set surfaces of $|\mathbf{w}|$ at 0.6817, 0.8192 and 1.2450 for times 3, 6 and 9 respectively which is 60% of the maximum value. Figures are generated using a 324^3 grid.

tubes computed from a simulation using a 48^3 grid for both \mathbf{M} and \mathbf{V} , the Hermite cubic interpolation of the velocity field uses a 96^3 grid to ensure sufficient smoothness. The time step Δt is fixed at $1/50$, Fourier truncation radius at 32 and the Jacobian determinant error tolerance at 10^{-3} . Due to the significant localized small scale features, the vorticity computation uses the adaptive sampling described in the Appendix A. The mollifier μ_h is given by a tensor product of a 1D function $\cos^2\left(\frac{\pi s}{2h}\right)$ in each cell. The cell integral (A.1) is computed by numerical quadrature using equidistributed sample points in each cell. The number of sample points per cell is at minimum 2 points per dimension and for each cell, this number is allowed to increase adaptively depending on the range and total variation of \mathbf{w} in each cell. The total number of sample points is capped at 192^3 at which point all cells have their sample number rescaled down proportionally. The results of this test until time $t = 12$ are presented in Table 2. Fig. 9 shows the evolution of the vortex cores until time $t = 9$. To better visualize the flow, we also include in Fig. 10 the scalar advection of the initial vorticity strength $\phi_0 = |\mathbf{w}_0|$ as given by equation (75).

The small scale features that develop in the flow after $t = 9$ become very fine and make 3D visualization difficult. We present instead three 2D contour plots on the xy -, xz - and yz - planes, respectively. Figs. 11 and 12 show zoomed views on the lateral cuts of the vorticity magnitude and the advected initial vorticity strength. The highest zoom shows a domain of width $\pi/16$ corresponding to $1/64$ of the computational domain, i.e. smaller than a single cell of the advection grid. In Fig. 11, we can see a presence of a high vorticity gradient at time $t = 12$ and the formation of a vortex sheet. Fig. 12 shows more clearly the material deformations which lead to this high gradient. Indeed, the advected initial vorticity shows that two separate level-sets of the vorticity were pushed close together by the flow. The yellow region squeezed between the blue curves in 12b was formed from the flattening of the initial vortex tubes. However, the absence of a highly

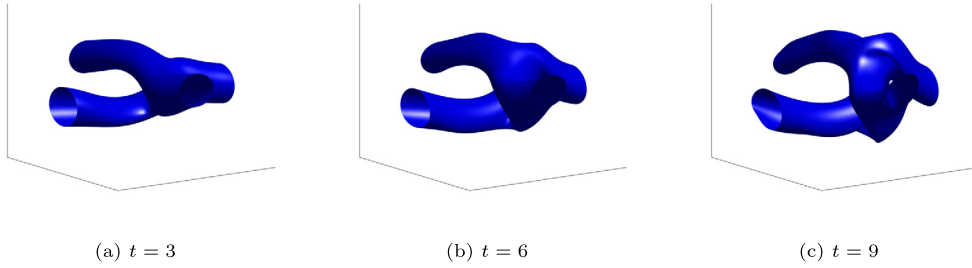


Fig. 10. Transport of the initial vortex core. We show level-set surfaces of ϕ at 0.5402, which is 60% of the maximum value. Figures are generated using a 324^3 grid.

Table 2

Evolution of total enstrophy, energy conservation relative error (divided by initial energy), helicity conservation relative error (divided by initial helicity), maximum vorticity and velocity, number of remaps and wallclock computation time of the perturbed perpendicular vortex tubes initial condition using CM method for 3D Euler. Grid resolutions: 48^3 for \mathbf{M} , 48^3 for \mathbf{V} , $\Delta t = 1/50$, adaptive sampling with mollifier convolution is used for the vorticity, Fourier truncation at radius 32, remapping Jacobian determinant tolerance at 10^{-3} . All data in this table are evaluated using a grid of resolution 256^3 .

t	$\ \mathbf{w}\ _{L^2}^2$	$\ \mathbf{u}\ _{L^2}^2 / \ \mathbf{u}_0\ _{L^2}^2 - 1$	$H/H_0 - 1$	$\ \mathbf{w}\ _{L^\infty}$	$\ \mathbf{u}\ _{L^\infty}$	n_{maps}	time (s)
0	125.7910	0.000	0.000	0.9004	0.9684	1	0
1	126.9290	-7.816×10^{-4}	2.296×10^{-3}	0.9183	0.9644	1	187
2	130.3522	-3.043×10^{-3}	8.889×10^{-3}	1.0317	0.9478	1	381
3	136.0808	-6.604×10^{-3}	1.901×10^{-2}	1.1361	0.9331	2	586
4	144.1831	-1.117×10^{-2}	3.160×10^{-2}	1.2187	0.9431	3	801
5	154.8535	-1.636×10^{-2}	4.557×10^{-2}	1.2885	0.9794	4	1027
6	168.4914	-2.178×10^{-2}	5.988×10^{-2}	1.3653	1.0208	8	1568
7	185.7193	-2.707×10^{-2}	7.354×10^{-2}	1.4806	1.0495	14	2386
8	207.4032	-3.201×10^{-2}	8.561×10^{-2}	1.6472	1.0636	22	3380
9	234.9474	-3.648×10^{-2}	9.518×10^{-2}	2.0750	1.0697	30	4691
10	271.2813	-4.048×10^{-2}	1.015×10^{-1}	3.3802	1.0602	40	6442
11	322.7244	-4.403×10^{-2}	1.038×10^{-1}	6.7721	1.0433	51	8489
12	400.7262	-4.733×10^{-2}	1.024×10^{-1}	14.1254	1.0929	64	10391

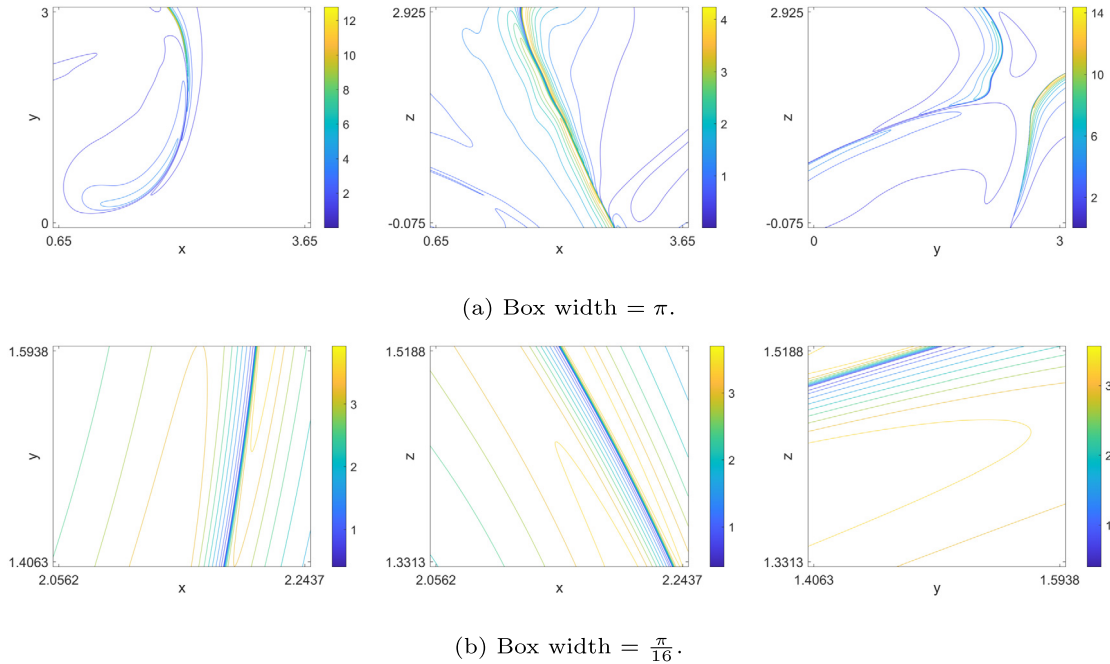


Fig. 11. Contour plot of vorticity strength $|\mathbf{w}|$ at time $t = 12$, centered at $(x, y, z) = (2.15, 1.5, 1.425)$; 10 isoline values evenly distributed in the range of each frame are shown. From left to right, contours on: xy - plane, xz -plane and yz -plane. The isoline ranges are $[0.0047, 12.7748]$, $[0.0075, 4.2412]$, $[0.0351, 14.3722]$, $[0.4072, 3.9227]$, $[0.4049, 3.6989]$ and $[0.4049, 3.5640]$. Figures are produced using a 1024^2 2D grid. We note that this is a much finer local sampling of the vorticity field compared to the data in Table 2 and thus able to resolve a higher vorticity maximum. (Color online)

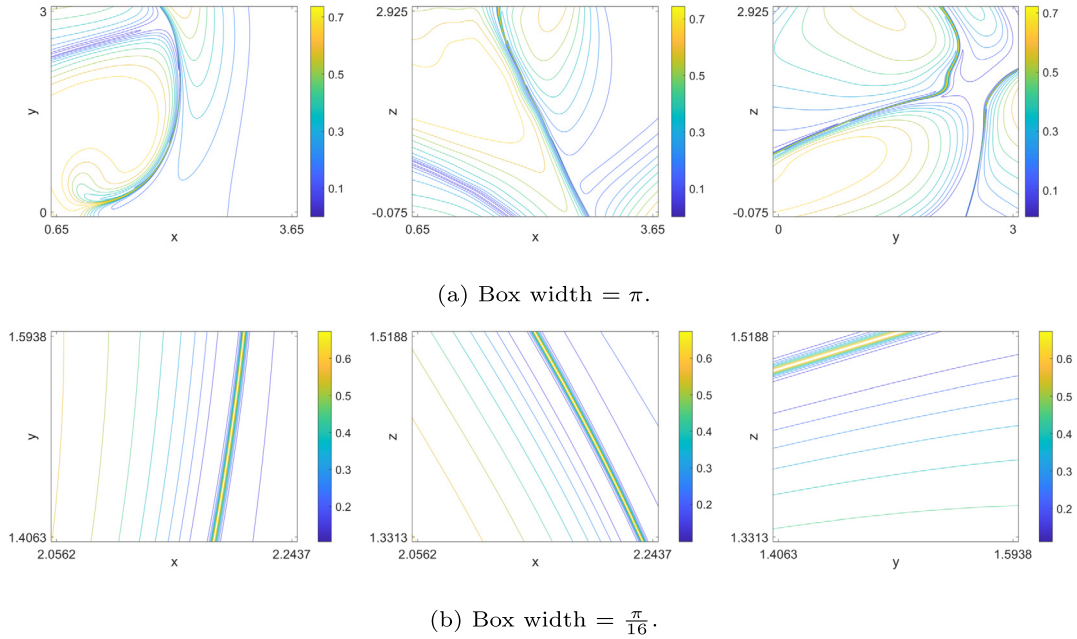


Fig. 12. Contour plot of the transported scalar ϕ (see equation (75)) at time $t = 12$, centered at $(x, y, z) = (2.15, 1.5, 1.42)$, 10 isoline values evenly distributed in the range of each frame are shown. From left to right, contours on: xy -plane, xz -plane and yz -plane. The isoline ranges are $[0.0007, 0.7435]$, $[0.0013, 0.7435]$, $[0.0102, 0.7220]$, $[0.1049, 0.6729]$, $[0.0977, 0.6729]$ and $[0.1117, 0.6729]$. Figures are produced using a 1024^2 2D grid. (Color online)

concentrated vorticity peak in 11 in contrast to 12 suggests that the vorticity direction was not fully aligned with the strain tensor eigenvector of the largest eigenvalue indicating that the vortex stretching term attenuated the vorticity gradient in this region. This nicely illustrates that vortex stretching can be quantified locally.

5. Conclusion

In this paper, we presented a novel numerical method for solving the 3D incompressible Euler equations. This method is based on the Gradient-Augmented Level-Set and Jet-Scheme frameworks and extends the previous work on the CM method for the 2D Euler equations, studied in [1]. Taking a more geometric approach, we proposed a reformulation of the CM framework which allows for the vortex stretching term in 3D to be seamlessly incorporated. The approach in this paper is summarized in equation (27) and can be described as follows: We evolve numerically the backward characteristic map, i.e. the backward-in-time flow map generated by the fluid velocity. This map is in fact the transition map between the Eulerian and Lagrangian coordinate charts. From the Kelvin circulation theorem, the time t vorticity field, expressed as a differential 2-form, can be computed from the pullback of the initial vorticity by the characteristic map, thereby allowing for vortex line deformation and stretching. The Biot-Savart law for the vorticity is then computed using Fourier spectral methods to provide the velocity field needed to further evolve the map.

This reformulation was motivated by the fact that direct integration of the vortex stretching term as a generic source term does not preserve the characteristic structure of the method and nullifies some of its numerical advantages. The geometric approach we proposed here allows us to retain all numerical properties of the CM method previously shown in [1]. Firstly, the functional representation of the vorticity field through pullback by the characteristic map preserves the fine scales generated by the inviscid flow. This is demonstrated by the tests in section 4, notably, the energy and enstrophy spectra plots in section 4.1 show that by oversampling the vorticity on a fine grid, high frequency features of the solution can be accurately reconstructed. Secondly, the rapid growth in the solution gradient can be efficiently resolved using the group structure of the characteristic map; the submap decomposition allows us to achieve multiplicative growth in the spatial resolution using only fast coarse grid map computations. As seen in section 4.1, the flow of the vortex reconnection test can be evolved up to time $t = 17$ using 79 coarse grid maps of size $64 \times 48 \times 32$, whereas purely Eulerian pseudospectral methods would require computations on grid sizes of order 1024^3 to achieve comparable results. Lastly, by defining the vorticity 2-form as the pullback of the initial condition by the map, we preserve the non-dissipative property of the CM method, this was shown theoretically in the error estimates in section 3.3 where we found that the error is in fact advective and that the numerical vorticity is related to the exact vorticity by pullback by an error map. Similar to the Lagrangian-Averaged Euler- α equations and the Kelvin-filtered turbulence models [51], the solution is obtained from a nonlinear dispersive perturbation of the equation. This was also supported by the numerical tests which showed that subgrid structures are preserved and that the spectrum plots of the solutions do not exhibit the exponential decay associated to artificial viscous dissipation; as a matter of fact, higher frequency modes can be reconstructed by finer samplings of the solution.

This work constitutes a first investigation of the CM method for the 3D incompressible Euler equations, we demonstrated here the key properties of the method and studied its numerical accuracy. This opens numerous directions for future research. For instance, the inclusion of forcing terms which conform to the characteristic structure of the method could be of interest. Extensions to the framework to take into account different geometries and domain boundaries is also an important subject of further investigation. The application of spatial adaptivity was not studied in this work, this is an important direction for future research. Indeed, the general formulation of the method allows for the use of adaptive grids and spatial adaptivity for the GALS method has been studied in [5] in the context of a multiresolution 2D advection solver. Furthermore, the numerical tests in section 4.1 suggest that, with additional extensions in terms of spatial and temporal adaptivity or improvements in the discretization spaces and basis functions, more efficient higher resolution simulations using the CM method could produce new insights into the blow-up question for the Euler equations. We believe that the CM method provides a novel and unique approach for the simulation of inviscid flows, and offers a suitable framework for the numerical study of fluid dynamics.

CRedit authorship contribution statement

Xi-Yuan Yin: Conceptualization, Bibliography, Methodology, Software, Validation, Formal Analysis, Writing-Original Draft, Writing-Review & Editing, Visualization.

Kai Schneider: Methodology, Conceptualization, Formal Analysis, Editing Original Draft, Writing-Review.

Jean-Christophe Nave: Methodology, Conceptualization, Formal Analysis, Editing Original Draft, Writing-Review.

Declaration of competing interest

The authors do not have any conflict of interest with the work presented in this manuscript.

Data availability

Data will be made available on request.

Acknowledgements

The authors acknowledge partial funding from the Agence Nationale de la Recherche (ANR), grant ANR-20-CE46-0010-01. Additionally, JCN acknowledges partial support from the NSERC Discovery Grant program.

Appendix A. Vorticity sampling

The map error studied in section 3.3 can roughly be split into two main contributing parts: the map representation error arising from the interpolation of the flow generated by $\tilde{\mathbf{u}}$ and the velocity representation error $\tilde{\mathbf{u}} - \mathbf{u}$, these correspond to the φ and ψ errors respectively.

The ψ velocity representation error depend in part on the resolution of the \mathbf{V} grid used for the vorticity sampling. This grid needs to be fine enough to avoid Fourier aliasing from lack of resolution. One possible solution would be to use a dynamic sampling of the vorticity field such as through the use of oct-tree structured meshgrids. Here we will consider another adaptive sampling method: we directly evaluate a mollified version of the vorticity, that is defining $\tilde{\mathbf{w}} = \mu_h * \mathbf{w}^n$ where μ_h is a mollifier supported in a neighborhood of size h , this approach was studied to some extent in [1]. We pick h to be smaller than the cellwidth of \mathbf{V} , the evaluation of the mollified vorticity at grid points \mathbf{x}_i can then be expressed as the sum of the convolution integrals in all 8 cells C_{i+r} adjacent to \mathbf{x}_i :

$$(\mu_h * \mathbf{w}^n)(\mathbf{x}_i) = \sum_{\mathbf{r} \in \{-1,0\}^3} \int_{C_{i+r}} \mu_h(\mathbf{x}_i - \mathbf{y}) \mathbf{w}^n(\mathbf{y}) d\mathbf{y}. \quad (\text{A.1})$$

For instance, in the tests in section 4.2, the mollifier μ_h is chosen to be a 3D tensor of $\cos^2(x/h)$ supported in the cells adjacent to \mathbf{x}_i . The integral in each cell is computed using numerical quadrature and the number of quadrature points is then adapted to the local oscillations of \mathbf{w}^n to ensure the accuracy of the mollification. The resulting algorithm effectively computes a mollified vorticity field where subgrid oscillations are filtered out, by choosing an appropriate mollification scale, the \mathbf{V} grid can resolve the mollified vorticity without aliasing errors and the pointwise evaluation of $\mu_h * \mathbf{w}^n$ on \mathbf{V} is accurate as long as the sampling in each cell is sufficiently dense. This approach is also related to the Kelvin-filtered Euler equations. From the Kelvin circulation theorem, total circulation along a closed curve evolving under the flow is conserved. Both in the Kelvin-filtered equations and the CM method, circulation is conserved for closed curves evolving under a modified flow; in the Kelvin-filtered case, the modified flow arises from a filtered velocity field and in the CM case, the numerical flow map is modified by the φ and ψ errors, of which ψ contains the velocity filtering. In any case, the mollification of the transport velocity still deteriorates the accuracy of the method and, although the vorticity evolves

by pullback, dispersion is introduced in the form of a less energetic transport flow. The proper sampling of the vorticity is still subject of our current work, and future directions may include the use of adaptive meshes and wavelet transforms to reduce sampling errors.

References

- [1] X.-Y. Yin, O. Mercier, B. Yadav, K. Schneider, J.-C. Nave, A Characteristic Mapping method for the two-dimensional incompressible Euler equations, *J. Comput. Phys.* 424 (2021) 109781.
- [2] T. Ishihara, T. Gotoh, Y. Kaneda, Study of high-Reynolds number isotropic turbulence by direct numerical simulation, *Annu. Rev. Fluid Mech.* 41 (2009) 165–180.
- [3] M. Farge, N. Okamoto, K. Schneider, K. Yoshimatsu, Wavelet-based regularization of the Galerkin truncated three-dimensional incompressible Euler flows, *Phys. Rev. E* 96 (2017) 063119.
- [4] K. Schneider, D. Kolomenskiy, E. Deriaz, Is the CFL condition sufficient? Some remarks, in: C.A. de Moura, C.S. Kubrusly (Eds.), *The Courant–Friedrichs–Lewy (CFL) Condition*, Birkhäuser, 2013, pp. 139–146.
- [5] D. Kolomenskiy, J.-C. Nave, K. Schneider, Adaptive gradient-augmented level set method with multiresolution error estimation, *J. Sci. Comput.* 66 (1) (2016) 116–140.
- [6] A. Guittet, M. Theillard, F. Gibou, A stable projection method for the incompressible Navier–Stokes equations on arbitrary geometries and adaptive Quad/Octrees, *J. Comput. Phys.* 292 (2015) 215–238.
- [7] M. Mirzadeh, A. Guittet, C. Burstedde, F. Gibou, Parallel level-set methods on adaptive tree-based grids, *J. Comput. Phys.* 322 (2016) 345–364.
- [8] T. Grafke, H. Homann, J. Dreher, R. Grauer, Numerical simulations of possible finite time singularities in the incompressible Euler equations: comparison of numerical methods, *Physica D* 237 (14–17) (2008) 1932–1936.
- [9] T. Ishihara, Y. Kaneda, K. Morishita, M. Yokokawa, A. Uno, Second-order velocity structure functions in direct numerical simulations of turbulence with Re_λ up to 2250, *Phys. Rev. Fluids* 5 (10) (2020) 104608.
- [10] K. Kamrin, C.H. Rycroft, J.-C. Nave, Reference map technique for finite-strain elasticity and fluid–solid interaction, *J. Mech. Phys. Solids* 60 (11) (2012) 1952–1969.
- [11] B. Valkov, C.H. Rycroft, K. Kamrin, Eulerian method for multiphase interactions of soft solid bodies in fluids, *J. Appl. Mech.* 82 (4) (2015) 041011.
- [12] T. Bellotti, M. Theillard, A coupled level-set and reference map method for interface representation with applications to two-phase flows simulation, *J. Comput. Phys.* 392 (2019) 266–290.
- [13] M. Theillard, A volume-preserving reference map method for the level set representation, *J. Comput. Phys.* 442 (2021) 110478.
- [14] A. Staniforth, J. Côté, Semi-Lagrangian integration schemes for atmospheric models—a review, *Mon. Weather Rev.* 119 (9) (1991) 2206–2223.
- [15] A.J. Chorin, Numerical study of slightly viscous flow, *J. Fluid Mech.* 57 (4) (1973) 785–796.
- [16] J.T. Beale, A. Majda, Vortex methods. I. Convergence in three dimensions, *Math. Comput.* 39 (159) (1982) 1–27.
- [17] T.Y. Hou, J. Lowengrub, Convergence of the point vortex method for the 3-D Euler equations, *Commun. Pure Appl. Math.* 43 (8) (1990) 965–981.
- [18] R. Pelz, Locally self-similar, finite-time collapse in a high-symmetry vortex filament model, *Phys. Rev. E* 55 (2) (1997) 1617.
- [19] G.-H. Cottet, P. Koumoutsakos, M.L.O. Salihi, Vortex methods with spatially varying cores, *J. Comput. Phys.* 162 (1) (2000) 164–185.
- [20] M. Oliver, S. Shkoller, The Vortex Blob method as a second-grade non-Newtonian fluid, *Commun. Partial Differ. Equ.* 26 (2001).
- [21] L. Rosenhead, The point vortex approximation of a vortex sheet, *Proc. R. Soc. Lond. Ser. A* 134 (1932) 170–192.
- [22] G.-H. Cottet, J. Goodman, T.Y. Hou, Convergence of the grid-free point vortex method for the three-dimensional Euler equations, *SIAM J. Numer. Anal.* 28 (2) (1991) 291–307.
- [23] G. Winckelmans, A. Leonard, Contributions to vortex particle methods for the computation of three-dimensional incompressible unsteady flows, *J. Comput. Phys.* 109 (2) (1993) 247–273.
- [24] D. Rossinelli, M. Bergdorf, G.-H. Cottet, P. Koumoutsakos, GPU accelerated simulations of bluff body flows using vortex particle methods, *J. Comput. Phys.* 229 (9) (2010) 3316–3333.
- [25] I. Christiansen, Numerical simulation of hydrodynamics by the method of point vortices, *J. Comput. Phys.* 13 (3) (1973) 363–379.
- [26] B. Couet, O. Buneman, A. Leonard, Simulation of three-dimensional incompressible flows with a vortex-in-cell method, *J. Comput. Phys.* 39 (2) (1981) 305–328.
- [27] G.-H. Cottet, P. Poncet, Advances in direct numerical simulations of 3D wall-bounded flows by vortex-in-cell methods, *J. Comput. Phys.* 193 (1) (2004) 136–158.
- [28] I.F. Sbalzarini, J.H. Walther, M. Bergdorf, S.E. Hieber, E.M. Kotsalis, P. Koumoutsakos, PPM—a highly efficient parallel particle–mesh library for the simulation of continuum systems, *J. Comput. Phys.* 215 (2) (2006) 566–588.
- [29] O. Podvigina, V. Zheligovsky, U. Frisch, The Cauchy–Lagrangian method for numerical analysis of Euler flow, *J. Comput. Phys.* 306 (2016) 320–342.
- [30] W.T. Ashurst, D.I. Meiron, Numerical study of vortex reconnection, *Phys. Rev. Lett.* 58 (16) (1987) 1632.
- [31] R. Grauer, C. Mariani, K. Geraschewski, Adaptive mesh refinement for singular solutions of the incompressible Euler equations, *Phys. Rev. Lett.* 80 (19) (1998) 4177.
- [32] M.D. Bustamante, R.M. Kerr, 3D Euler about a 2D symmetry plane, *Physica D* 237 (14–17) (2008) 1912–1920.
- [33] T.Y. Hou, T. Jin, P. Liu, Potential singularity for a family of models of the axisymmetric incompressible flow, *J. Nonlinear Sci.* 28 (6) (2018) 2217–2247.
- [34] A. Pumir, E. Siggia, Collapsing solutions to the 3-D Euler equations, *Phys. Fluids A, Fluid Dyn.* 2 (2) (1990) 220–241.
- [35] M. Brachet, M. Meneguzzi, A. Vincent, H. Politano, P. Sulem, Numerical evidence of smooth self-similar dynamics and possibility of subsequent collapse for three-dimensional ideal flows, *Phys. Fluids A, Fluid Dyn.* 4 (12) (1992) 2845–2854.
- [36] T.Y. Hou, R. Li, Dynamic depletion of vortex stretching and non-blowup of the 3-D incompressible Euler equations, *J. Nonlinear Sci.* 16 (6) (2006) 639–664.
- [37] R.M. Kerr, Evidence for a singularity of the three-dimensional, incompressible Euler equations, *Phys. Fluids A, Fluid Dyn.* 5 (7) (1993) 1725–1746.
- [38] H. Moffatt, Y. Kimura, Towards a finite-time singularity of the Navier–Stokes equations. Part 2. Vortex reconnection and singularity evasion, *J. Fluid Mech.* 870 (2019).
- [39] J. Yao, F. Hussain, A physical model of turbulence cascade via vortex reconnection sequence and avalanche, *J. Fluid Mech.* 883 (2020).
- [40] J.-C. Nave, R.R. Rosales, B. Seibold, A gradient-augmented level set method with an optimally local, coherent advection scheme, *J. Comput. Phys.* 229 (10) (2010) 3802–3827.
- [41] H. Kohno, J.-C. Nave, A new method for the level set equation using a hierarchical-gradient truncation and remapping technique, *Comput. Phys. Commun.* 184 (6) (2013) 1547–1554.
- [42] J.C. Bowman, M.A. Yassaie, A. Basu, A fully Lagrangian advection scheme, *J. Sci. Comput.* 64 (1) (2015) 151–177.
- [43] C.L. Fefferman, Existence and smoothness of the Navier–Stokes equations, in: *The Millennium Prize Problems*, 2006, pp. 57–67.
- [44] O. Mercier, X.-Y. Yin, J.-C. Nave, The Characteristic Mapping method for the linear advection of arbitrary sets, *SIAM J. Sci. Comput.* 42 (3) (2020) A1663–A1685.

- [45] V.I. Arnol'd, Sur la géométrie différentielle des groupes de Lie de dimension infinie et ses applications à l'hydrodynamique des fluides parfaits, *Ann. Inst. Fourier* 16 (1966) 319–361.
- [46] S. Lang, *Fundamentals of Differential Geometry*, vol. 191, Springer Science & Business Media, 2012.
- [47] V.I. Arnol'd, *Mathematical Methods of Classical Mechanics*, vol. 60, Springer Science & Business Media, 2013.
- [48] J. Deng, T.Y. Hou, X. Yu, A level set formulation for the 3D incompressible Euler equations, *Methods Appl. Anal.* 12 (4) (2005) 427–440.
- [49] P. Constantin, An Eulerian–Lagrangian approach to the Navier–Stokes equations, *Commun. Math. Phys.* 216 (3) (2001) 663–686.
- [50] X.-Y. Yin, L. Chen, J.-C. Nave, A diffusion-driven characteristic mapping method for particle management, *SIAM J. Sci. Comput.* 43 (5) (2021) A3155–A3183.
- [51] C. Foias, D.D. Holm, E.S. Titi, The Navier–Stokes-alpha model of fluid turbulence, *Physica D* 152 (2001) 505–519.
- [52] B. Seibold, J.-C. Nave, R.R. Rosales, Jet schemes for advection problems, *Discrete Contin. Dyn. Syst., Ser. B* 17 (4) (2012) 1229–1259.
- [53] G. Birkhoff, M.H. Schultz, R.S. Varga, Piecewise Hermite interpolation in one and two variables with applications to partial differential equations, *Numer. Math.* 11 (3) (1968) 232–256.
- [54] P. Chidyagwai, J.-C. Nave, R.R. Rosales, B. Seibold, A comparative study of the efficiency of jet schemes, *Int. J. Numer. Anal. Model. Ser. B* 3 (3) (2012) 297–306.
- [55] J. Goodrich, T. Hagstrom, J. Lorenz, Hermite methods for hyperbolic initial-boundary value problems, *Math. Comput.* 75 (254) (2006) 595–630.
- [56] K. Schneider, N.-R. Kevlahan, M. Farge, Comparison of an adaptive wavelet method and nonlinearly filtered pseudospectral methods for two-dimensional turbulence, *Theor. Comput. Fluid Dyn.* 9 (3) (1997) 191–206.
- [57] K. Schneider, O.V. Vasilyev, Wavelet methods in computational fluid dynamics, *Annu. Rev. Fluid Mech.* 42 (2010) 473–503.
- [58] M. Frigo, S.G. Johnson, The design and implementation of FFTW3, in: Special Issue on “Program Generation, Optimization, and Platform Adaptation”, *Proc. IEEE* 93 (2) (2005) 216–231.