A Time-Continuous Embedding Method for Scalar Hyperbolic Conservation Laws on Manifolds

Yinghua Wang¹ · Bao-Shan Wang² · Leevan Ling³ · Wai Sun Don²

Received: 25 November 2021 / Revised: 14 September 2022 / Accepted: 19 September 2022 / Published online: 9 November 2022 © The Author(s) 2022

Abstract

A time-continuous (tc-)embedding method is first proposed for solving nonlinear scalar hyperbolic conservation laws with discontinuous solutions (shocks and rarefaction waves) on codimension 1, connected, smooth, and closed manifolds (surface PDEs or SPDEs in $\mathbb{R}^2$ and $\mathbb{R}^3$). The new embedding method improves upon the classical closest point (cp-)embedding method, which requires re-establishments of the constant-along-normal (CAN-)property of the extension function at every time step, in terms of accuracy and efficiency, by incorporating the CAN-property analytically and explicitly in the embedding equation. The tc-embedding SPDEs are solved by the second-order nonlinear central finite volume scheme with a nonlinear minmod slope limiter in space, and the third-order total variation diminished Runge-Kutta scheme in time. An adaptive nonlinear essentially non-oscillatory polynomial interpolation is used to obtain the solution values at the ghost cells. Numerical results in solving the linear wave equation and the Burgers’ equation show that the proposed tc-embedding method has better accuracy, improved resolution, and reduced CPU times than the classical cp-embedding method. The Burgers’ equation, the traffic flow problem, and the Buckley-Leverett equation are solved to demonstrate the robust performance of the tc-embedding method in resolving fine-scale structures efficiently even in the presence of a shock and the essentially non-oscillatory capturing of shocks and rarefaction waves on simple and complex shaped one-dimensional manifolds. Burgers’ equation is also solved on the two-dimensional torus-shaped and spherical-shaped manifolds.
Keywords Closest point embedding · Surface PDEs · Central finite volume · ENO interpolation · Burgers’ · Traffic flow · Buckley-Leverett

1 Introduction

Partial differential equations on a surface (surface PDEs or SPDEs) have received growing interest over the last years. Problems of interest areas arise in a wide range of fields such as image processing (e.g., the restoration of a damaged pattern on a surface [4] and image of the human brain [24]), biological systems (e.g., study domain formation in vesicles [2] and wound healing process [27]), mathematical physics (e.g., flows and solidification on surfaces [25, 26]) and computer graphics (e.g., real-time fluid visualization on surfaces [1]).

Instead of studying linear and quasi-linear SPDEs with smooth solutions ([12, 14] for level set approach, [31] for a semi-Lagrangian method, [37] for meshfree finite difference method, [3, 7, 10, 22, 28–30] for closest point (cp-)embedding methods) on Cartesian grids and orthogonal gradient grids, as a proof of concept, this study explores the application of the embedding methods and develops a numerical scheme, based on our extensive knowledge of essentially non-oscillatory shock-capturing schemes, to solve the (nonlinear scalar) hyperbolic conservation laws (Burgers’ equation) with a discontinuous solution on one-dimensional manifolds. For the rest of the paper, the manifold is implicitly referred to as the one-dimensional, connected, smooth, closed, and complex (or simple) manifold unless specified otherwise.

After reviewing some necessary definitions and assumptions in Sec. 2, we propose a new time-continuous (tc-)embedding method based on the classical closest point (cp-)embedding in [30] in Sect. 3. The new tc-embedding method improves upon the classical cp-embedding method, which requires re-establishments of the constant-along-normal (CAN-)property of the extension function at every time step, in terms of accuracy and efficiency by incorporating the CAN-property analytically and explicitly in the embedding equation. The tc-embedding SPDEs are solved by the second-order nonlinear central finite volume scheme (CFV) with a nonlinear minmod slope limiter in space, and the third-order total variation diminished Runge–Kutta scheme (RK-TVD) in time. An adaptive nonlinear essentially non-oscillatory polynomial (ENO) interpolation is used to obtain the solution values at the ghost cells required by the CFV scheme. The main contributions of this study are in the following two areas. They are i) a novel time-continuous (tc-)embedding that holds for all time continuously in contrast to the classical cp-embedding that is only true at a fixed time (see Theorem 1 for the formal statement), and ii) the first attempt in solving hyperbolic SPDEs with discontinuous solutions on a manifold. In Sect. 4, we detail a numerical recipe (CFV, RK-TVD, and ENO interpolation) for solving the resulting embedded SPDEs. In Sect. 5.1, accuracy and efficiency tests of the tc-embedding method versus the cp-embedding method are given. The results indicate that the tc-embedding method has a lesser numerical dissipation, resolves the fine-scale structures more efficiently, and is roughly two and half times faster CPU times than the cp-embedding method. In Sect. 5.2, the Burgers’ equation with convex flux, the traffic flow problem with concave flux, and the Buckley-Leverett equation with non-convex flux demonstrate the essentially non-oscillatory shock-capturing property and the efficient resolution of fine-scale structures of the tc-embedding method on a complex manifold. In Sect. 5.3, numerical results are also given for extending the tc-embedding method for solving the Burgers’ equation on two-dimensional torus-shaped and spherical-shaped manifolds embedded in . Conclusion and future works are given in Sect. 6.
2 PDEs on Manifolds

Let \( \mathcal{M} \subseteq \mathbb{R}^2 \) be a (one-dimensional, connected, smooth, and closed) manifold with well-defined unit outward normal vector and tangent space, denoted by \( \hat{n} \in \mathbb{R}^2 \) and \( T_{\mathcal{M}} \), respectively, for every point \( p \in \mathcal{M} \).

Also, let \( Q_{\mathcal{M}} : \mathcal{M} \times \mathbb{R}^+ \rightarrow \mathbb{R} \) be some conserved variables associated with a smooth flux function \( F : \mathbb{R} \rightarrow \mathbb{R} \). In this paper, we consider SPDEs of scalar hyperbolic type for the unknown conserved function \( Q_{\mathcal{M}} \in H^1(H^2(\mathcal{M}); [0, T]) \) in the form of

\[
\begin{aligned}
\frac{\partial}{\partial t} Q_{\mathcal{M}} + v_{\mathcal{M}} \cdot \nabla_{\mathcal{M}} F(Q_{\mathcal{M}}) &= 0, \quad \text{on } \mathcal{M}, \quad t \in (0, T], \\
Q_{\mathcal{M}}(\cdot, 0) &= Q_{\mathcal{M}, 0}, \quad \text{on } \mathcal{M},
\end{aligned}
\]

for some known vector field \( v_{\mathcal{M}} : \mathcal{M} \rightarrow T_{\mathcal{M}} \).

\[2.1 \text{ Embedded PDEs}\]

For some \( \delta > 0 \), we begin by defining the narrow band domain around the manifold by

\[\Omega := \left\{ x \in \mathbb{R}^2 : \inf_{p \in \mathcal{M}} \| x - p \| < \delta \right\} \supseteq \mathcal{M}.\]

The goal of this section is to recast the well-posed PDEs on \( \mathcal{M} \) in (1) in \( \Omega \). Firstly, we need the relationship between surface and Euclidean gradient operators on time-independent functions on the manifold and their extensions.

**Proposition 1** ([9]) Let \( \mathcal{M} \) be \( C^3 \)-smooth and \( q_E \in (C^2 \cap H^{2+\frac{1}{2}})(\Omega) \) be an extension of some manifold function \( q_{\mathcal{M}} \in H^2(\mathcal{M}) \) with \( q_E|_{\mathcal{M}} = q_{\mathcal{M}} \), then we have

\[\nabla_{\mathcal{M}} q_{\mathcal{M}} = \nabla q_E - \hat{n} \frac{\partial}{\partial \hat{n}} q_E, \quad \text{on } \mathcal{M}.\]

**Proposition 1** gives a strong motivation to extend manifold functions \( Q_{\mathcal{M}} \) to \( Q_E \) in such a way that the normal-derivative \( \partial_{\hat{n}} Q_E \) remains zero everywhere on \( \mathcal{M} \). Note that this is not yet a unique extension operator.

Before proceeding, let us define the closest point mapping \( \text{cp} : \Omega \subset \mathbb{R}^2 \rightarrow \mathcal{M} \) by

\[\text{cp}(x) = \arg \inf_{p \in \mathcal{M}} \| p - x \|_{\ell^2(\mathbb{R}^2)},\]

on \( \Omega \) in (3) with some sufficiently small \( \delta \) [30]. The existence of such \( \Omega \), in which \( \text{cp} \) is well-defined, is guaranteed provided that \( \mathcal{M} \) has enough smoothness; see [8]. For any (time-independent) manifold function \( Q_{\mathcal{M}} \in \mathcal{M} \rightarrow \mathbb{R} \), the constant-along-normal (CAN) extension is given by

\[Q_{\text{cp}} := Q_{\mathcal{M}} \circ \text{cp} : \Omega \rightarrow \mathbb{R},\]

whose function value is constant along with normal directions of \( \mathcal{M} \).
So far, we follow the setting of the closest point embedding method [30]. For any fixed
$t_0 \geq 0$, the SPDEs in (1) can be embedded to $\Omega \supset M$ using CAN-extension $Q_{cp}$ to $Q_M(\cdot, t_0)$ and yield

\[
\begin{align*}
\frac{\partial}{\partial t} Q_{cp} + (v_M \circ cp) \cdot \nabla F(Q_{cp}) &= 0, \quad \text{in } \Omega, \quad t = t_0, \\
Q_{cp}(\cdot, t_0) &= Q_M(cp(\cdot), t_0), \quad \text{in } \Omega, \\
\frac{\partial}{\partial n} Q_{cp} &= 0, \quad \text{on } \partial \Omega, \quad t = t_0.
\end{align*}
\]

(7)

Since the CAN-property of $Q_{cp}$ carries over to $F(Q_{cp})$, Proposition 1 ensures that

\[
\nabla_M F(Q_M) = (\nabla F(Q_{cp}))|_M \quad \text{on } M \text{ and } t = t_0.
\]

(8)

The solution of the embedded PDEs (7), when restricted to the surface $M$, coincides with
the solution of the SPDEs (1) with $Q_M(\cdot, t_0) = Q_{cp}(\cdot, t_0)|_M$ at the fixed $t_0$.

Numerically, the spatial operator (in this case, the gradient of the flux) in the embedded
PDEs (7) can be discretized by any finite volume scheme. The resulting semi-discrete tem-
poral ODEs can be integrated to advance the solution of the SPDEs (1) near the surface of
the manifold for one time step (or one stage of a multi-stage Runge-Kutta scheme). Note that
there is no temporal treatment in this formulation. More precisely, once (7) is advanced in
time, the numerical solution $Q_{cp}(\cdot, t_0 + \Delta t)$ will violate the CAN-property for any $\Delta t > 0$.
An interpolation process of all grid points is needed to reconstruct the CAN-property of the
numerical solution, which is mathematically equivalent to constructing a new CAN-extension
based on $Q_M(\cdot, t_0 + \Delta t)$. This classical (spatial) cp-embedding method was commonly used
for solving time-dependent problems, see [21–23, 28, 30].

3 Time-Continuous Embedded PDEs

We desire to have the CAN-extension $Q_E$ of the exact solution $Q_M$ to (1) such that

\[
Q_E(x, t) := Q_M(cp(x), t) \quad \text{for } x \in \Omega \text{ and } t \in [0, T],
\]

(9)

and solve the tc-embedded PDEs in the form of

\[
\begin{align*}
\frac{\partial}{\partial t} Q_E + v_E \cdot \nabla F(Q_E) &= 0, \quad \text{in } \Omega, \quad t \in (0, T], \\
Q_E(\cdot, 0) &= Q_M(cp(\cdot), 0), \quad \text{in } \Omega, \\
\frac{\partial}{\partial n} Q_E &= 0, \quad \text{on } \partial \Omega, \quad t \in (0, T],
\end{align*}
\]

(10)

where $v_E = (u, v)$ is the vector field with $u$ and $v$ as its x- and y-direction components,
respectively, defined in the Theorem (1) below. For clarity of later discussion, we define the
continuous spatial operator $L$ as

\[
L(Q_E) := v_E \cdot \nabla F(Q_E) = u \frac{\partial F(Q_E)}{\partial x} + v \frac{\partial F(Q_E)}{\partial y},
\]

(11)

and (10) can be written compactly as

\[
\begin{align*}
\frac{\partial}{\partial t} Q_E + L(Q_E) &= 0, \quad \text{in } \Omega, \quad t \in (0, T], \\
Q_E(\cdot, 0) &= Q_M(cp(\cdot), 0), \quad \text{in } \Omega, \\
\frac{\partial}{\partial n} Q_E &= 0, \quad \text{on } \partial \Omega, \quad t \in (0, T].
\end{align*}
\]

(12)
Although the notation difference between (7) and (12) is small, the implication is huge in the computational aspect. Instead of embedding the original problem (1) in the form of (7) at every time step, we only need to tc-embed (1) once and solve (12) until we reach the final time \( T \). It is because, as shown later, the extended velocity field \( v_E \) depends exclusively on the geometric properties of the manifold, which is time-independent in this study.

One can easily see that the requirement (9) implies
\[
\frac{\partial}{\partial t} Q_E(x, t) = \frac{\partial}{\partial t} Q_M(cp(x), t), \quad x \in \Omega, \quad t \geq 0.
\]
The realization of the proposed tc-embedding hence lies on finding an appropriate extended velocity field \( v_E : \Omega \to \mathbb{R}^2 \) to the vector field \( v_M : M \to \mathbb{R}^2 \). To do so, we define the signed-distance function \( \text{dist} : \Omega \times M \to \mathbb{R} \) by
\[
\text{dist}(x, cp(x)) := \text{sign}(\hat{n} \cdot (x - cp(x)))\|x - cp(x)\|\ell_2(\mathbb{R}^2).
\]
Now, the following theorem provides the mean to adjust the extended velocity field \( v_E \) so that the CAN-property of \( Q_E \) is preserved over time.

**Theorem 1** Suppose that \( M \) is smooth enough and \( \Omega \) in (3) is chosen so that Proposition 1 holds. For any \( q : \Omega \to \mathbb{R} \) to be any (time-independent) \( C^1 \)-function satisfying the CAN-property with respect to the manifold \( M \). If an extended velocity field \( v_E : \Omega \to \mathbb{R}^2 \) to \( v_M \) in (1) defined by
\[
v_E(x) := \left( 1 + \text{dist} (x, cp(x))\tilde{\kappa}(cp(x)) \right)v_M(cp(x)), \quad x \in \Omega,
\]
where \( \kappa(cp(x)) \) and \( \tilde{\kappa}(cp(x)) \) denoting the absolute and signed curvature of \( M \) at \( cp(x) \), respectively, finite and exist, then
\[
v_M(cp(x)) \cdot \nabla_M q(cp(x)) = v_E(x) \cdot \nabla q(x).
\]

**Proof** Due to the CAN-property, it suffices to show that
\[
v_M(cp(x)) \cdot \nabla q(cp(x)) = v_E(x) \cdot \nabla q(x),
\]
for any fixed \( x \in \Omega \supset M \). Note that the right-hand side of (14) is a \( v_M \)-directional derivative of \( q \) at \( cp(x) \in M \). Consider a (local) arc-length parameterization \( \gamma(s) \) of the manifold \( M \) defined by its arc-length variable \( s \in N_\epsilon(0) \) in some \( \epsilon \)-neighborhood around the origin with \( \gamma(0) = cp(x) \) and orientation given by the unit tangent vector \( \frac{d}{ds} \gamma(0) = v_M/\|v_M\| =: \hat{T}(0) \). Now we have
\[
v_M(cp(x)) \cdot \nabla q(cp(x)) = \|v_M(cp(x))\|\frac{\partial}{\partial T} q(cp(x)).
\]
Obviously, \( v_E \) must point to the \( \hat{T} \) direction. We only need to study the scaling effect when the point of evaluation in the directional-derivative \( \frac{\partial}{\partial T} q \) is shifted from \( cp(x) \) to \( x \).

Let \( D := \text{dist}(x, cp(x)) \), that is a fixed real number because \( x \) is considered as fixed. Since the outward unit normal vector \( \hat{n}(cp(x)) = \hat{n}(\gamma(0)) \) of \( M \) depends on the orientation of the manifold (definition of counterclockwise) and the normal vector \( \hat{N}(0) \) obtained from \( \gamma(s) \) depends on the orientation of the curve \( \gamma(s) \), we have
\[
x = \gamma(0) + D\hat{N}(\gamma(0)) = \gamma(0) \pm D\hat{N}(0).
\]
We begin by considering the directional derivative as a derivative on \( \gamma(s) \), i.e., Gateaux derivative, and shifted the point of evaluation using the CAN-property as follows:

\[
\frac{\partial}{\partial \hat{T}} q(\gamma(x)) := \lim_{h \to 0} \frac{q(\gamma(h)) - q(\gamma(0))}{h} = \lim_{h \to 0} \frac{q(\gamma(h) \pm D\hat{N}(h)) - q(\gamma(0) \pm D\hat{N}(0))}{h} = \lim_{h \to 0} \frac{q(x + h(\hat{T}(0) \pm D\hat{N}(0)) + O(h^2)) - q(x)}{h} =: \left\| \hat{T}(0) \pm D\hat{N}(0) \right\| \frac{\partial}{\partial \hat{T}} q(x).
\]

The last equality uses another definition for directional derivative based on the rate of change in the direction \( \hat{T} \pm D\hat{N} \), i.e., Fréchet derivative. Using the Frenet-Serret formula and the fact that \( M \) is planar, we find that

\[
\left\| \hat{T}(s) \pm D\hat{N}'(s) \right\| = \left\| \hat{T}(s) \pm D\kappa(s)\hat{T}(s) \right\| = 1 + D\kappa(s), \quad \text{for } s \in \mathcal{N}_\epsilon(0). \quad (16)
\]

The scaling factor we desired is given by (16) at \( s = 0 \), and hence, the formula for \( v_E \) in (13) is deduced. \( \Box \)

In the proof above, we have used the definition of the absolute and signed curvatures, that is, \( \tilde{\kappa}(\gamma(x)) = \pm \kappa(\gamma(x)) \). Once again, if the \( v_M \circ \gamma(x) \) does not change in time, \( v_E \) in (13) is time-independent. It can be pre-computed once at each point \( x \) in the initialization stage of the algorithm and used subsequently in the time-stepping scheme.

Now that the embedding process is completed solving the SPDEs (1) can be transformed into solving the tc-embedded PDEs (12) in a narrow computational domain \( \Omega \) with any suitable spatial solver in the two-dimensional Euclidean space. In the coming sections, we will describe the tc-embedding method for solving (12) based on the governing equations and the CAN-property in its solution, i.e., instead of the zero Neumann boundary conditions. Using this alternative approach, we propose an easy recipe to propagate non-smooth solutions on one-dimensional manifolds.

### 4 Implementation of the Tc-Embedding Method

This section will describe the numerical method in solving tc-embedded PDEs (12) with the spatial operator \( L \) (11) in a computational tube of radius \( \delta \).

We shall denote the computational domain in the two-dimensional Cartesian domain as \( \Omega \) (bounded within two pink lines) where the one-dimensional manifold \( M \) (black line) is embedded inside as illustrated in Fig. 1 with a circular manifold. With the Cartesian domain \( \Omega \), we discretize the domain with uniformly spaced computational cells with cell sizes \( \Delta x \) and \( \Delta y \) in the horizontal \( x \)-direction and the vertical \( y \)-direction, respectively. In Fig. 1, the cell centers \( x = (x_i, y_j) \) are shown. Following the suggestion in [28], we restrict the computational tube radius \( \delta \) to be

\[
\delta \geq \max \left( p\sqrt{\Delta x^2 + \Delta y^2} + p/2 \max(\Delta x, \Delta y), \delta^* \right), \quad (17)
\]

where the critical radius \( \delta^* \) satisfies the condition \( \delta^* \max(\kappa(x) : x \in M) < 1 \), where \( \kappa \) is the curvature of \( M \). Since we will be using a second-order finite volume scheme (CFV) and
Fig. 1 Simple illustration of the distribution of data points
$X = X_I \cup X_G \subset \Omega$ in the computational tube with the sets of interior $X_I$ (blue) and ghost $X_G$ (red) points. The black line represents the manifold $\mathcal{M}$ (Color figure online)

A second-order ENO interpolation, to solve (12) at a given $x = (x_i, y_j)$ (excluding the ghost cells), the computational stencil is $p = 2$ cells in all directions. This lower bound of $\delta$ ensures that there is always a sufficient number of computational cells available in the neighborhood of a given cell center $x$ to perform the necessary spatial operations given the order of the numerical scheme $p$.

Together, as illustrated in Fig. 1, we define a set of uniform spaced cell center (grid) points $(x_i, y_j)$ with cell size $\Delta x$ and $\Delta y$ in $\Omega$ by

$$X := \Omega \cap ((\Delta x \mathbb{Z}) \otimes (\Delta y \mathbb{Z})),$$

(18)

contained inside the computational tube of radius $\delta$. The set $X$ is further sub-divided into two subsets:

- The set of interior cells (blue solid symbol)
  $$X_I := \{ x \in X : \text{for computing the flux gradient in (11)} \},$$

- The set of ghost cells (red hollow symbol)
  $$X_G := X \setminus X_I.$$

Ghost cells $X_G$ are near-boundary cells that cannot be updated due to missing stencil’s information, but are needed to compute the flux gradients in (11) for those cells at the edges of $X_I$. The size of $X_G$ generally grows linearly as the spatial order of the scheme. For the second-order scheme used here, one cell is needed inside the domain boundary.

At any given time $t \geq 0$, we can use the available numerical solution $Q_E(X, t)$ at $X$ to compute the numerical flux gradient at $X_I$ by

$$\nabla F(Q_E)\bigg|_{X_I} = \left( \frac{\partial F(Q_E)}{\partial x}, \frac{\partial F(Q_E)}{\partial y} \right)^T \bigg|_{X_I},$$

which will be discussed in greater detail below. Together with the extended velocity field $v_E = (u, v)$ defined in (13), the semi-discrete approximation $\mathcal{L}_N$ of the spatial operator $\mathcal{L}$ defined in (11) becomes

$$\mathcal{L}_N(Q_E)\bigg|_{X_I} = v_E \cdot \nabla F(Q_E)\bigg|_{X_I} = \left( u \frac{\partial F(Q_E)}{\partial x} + v \frac{\partial F(Q_E)}{\partial y} \right) \bigg|_{X_I}. \quad (19)$$
Finally, we obtain the semi-discrete system of ODEs

\[
\begin{align*}
\frac{d}{dt} Q_E(X_I, t) &= -\mathcal{L}_N(Q_E(X, t)) \\
Q_E(X, 0) &= Q_M(cp(X), 0),
\end{align*}
\]

(20)

where the initial condition is specified by applying the cp-mapping to the initial conditions in (1). We note that while the system of ODEs is evaluated only at the interior points \(X_I\), the initial condition must be made available in the whole computational tube \(X\) that includes the ghost cells \(X_G\).

The numerical solution \(Q_E(X_I, t)\) of (20) can be solved by advancing in time via the explicit third-order total variation diminishing (TVD) low-storage Runge-Kutta (RK-TVD) scheme [13] from time \(t = t_n\) with the discrete solution \(Q^n_I = Q^n_E(X_I) = Q_E(X_I, t_n)\), to time \(t = t_{n+1} = t_n + \Delta t\) with a time step \(\Delta t\), \(Q^{n+1}_I = Q^{n+1}_E(X_I) = Q_E(X_I, t_{n+1})\), \(n = 0, 1, 2, \ldots\) as

\[
\begin{align*}
Q^{(1)}_I &= Q^n_I - \Delta t \mathcal{L}_N (Q^n_I), \\
Q^{(2)}_I &= \frac{3}{4} Q^n_I + \frac{1}{4} Q^{(1)}_I - \frac{1}{4} \Delta t \mathcal{L}_N (Q^{(1)}_I), \\
Q^{n+1}_I &= \frac{1}{3} Q^n_I + \frac{2}{3} Q^{(2)}_I - \frac{2}{3} \Delta t \mathcal{L}_N (Q^{(2)}_I),
\end{align*}
\]

(21)

with the given initial condition \(Q^n_0 = Q_E(X_I, 0)\) (20). \(\{Q^{(k)}_I = Q_E(X_I, t_k), k = \{1, 2, \ldots\}\}\) are the intermediate solutions and times \((t_1 = t_n + \Delta t, t_2 = t_n + \Delta t/2)\) at the kth RK-TVD stage. The time step \(\Delta t\) at each RK-TVD step is determined by the CFL stability condition

\[
\Delta t \left( \alpha \left\| \frac{\partial F}{\partial Q} \right\|_p \right) = \text{CFL} < 1,
\]

for some \(p\)-norm \((p = 1, 2, \infty)\) and \(\alpha = \max_{x \in X} |F/\partial Q|\) is the maximum characteristic speed of the hyperbolic conservation laws on \(X\). The CFL stability condition for an explicit numerical SPDE solver depends not only on the time step \(\Delta t\), grid spacing \(\Delta r\), and spectral radius of the Jacobian of the flux, but also on the geometric properties, namely, curvature \(\kappa\) expressed via \(v_E\) (13), of the given manifold. For a manifold with a large curvature \(\kappa\), the time step \(\Delta t\) will be reduced accordingly. In the following discussion, for simplicity, we shall take \(\Delta x = \Delta y\) and CFL = 0.45 unless stated otherwise. Readers are reminded that \(t_k, k = \{1, 2\}\) are the intermediate times between the times \(t_n\) (at the time step \(n\)) and \(t_{n+1}\) (at the time step \(n+1\)).

Since the tc-embedded PDEs (9) with the extended velocity field \(v_E\) satisfies the CAN-property intrinsically, the Neumann boundary condition in (12) is discarded in exchange for finding the solution \(Q_E(x, t)\) at the ghost cells \(x \in X_G\).

Given the numerical solutions \(\{Q^{(k)}_X = Q_E(X, t_k), k = \{n, 1, 2, n+1\}\}\) the two missing components of the numerical scheme in the discussion are:

1. At the beginning of each RK-TVD stage with \(k = \{n, 1, 2\}\) and for all \(x \in X_I\), computing the flux gradient \(\nabla F(Q_E(X_I, t_k))\) using the shock-capturing central finite volume (CFV) scheme with the nonlinear minmod slope limiter (see Sect. 4.1), and
2. At the end of each RK-TVD stage with \(k = \{1, 2, n+1\}\) and for all \(x \in X_G\), updating the numerical solution for the ghost cells \(Q_E(X_G, t_k)\) with the essentially non-oscillatory (ENO) interpolation at \(\xi = cp(x_g) \in M\), which satisfies the CAN-property, based on a few specially selected \(Q_E(X_I, t_k)\) in the closest neighboring cells around \(\xi\) (see Sect. 4.2).
4.1 Nonlinear Construction of the Flux Gradient $\nabla F(Q_E(X_I, t_k))$

At any given time $t = t_k$, $k = \{n, 1, 2\}$, the numerical flux gradient

$$\nabla F(Q_E) \bigg|_{X_I} = \left( \frac{\partial F(Q_E)}{\partial x}, \frac{\partial F(Q_E)}{\partial y} \right)^T \bigg|_{X_I},$$

at a given cell center $x = (x_i, y_j) \in X_I$ is approximated via a second-order shock-capturing nonlinear central finite volume (CFV) scheme with the nonlinear minmod slope limiter [19] in the two-dimensional Euclidean space, which is a class of nonlinear numerical schemes designed for solving nonlinear hyperbolic partial differential equations. Since the computational stencils $X$ are embedded in the Cartesian rectangular domain with uniform cell-sizes $\Delta x$ and $\Delta y$ in both $x$- and $y$-directions, respectively, the numerical approximation of the flux gradients can be carried out simply in a dimension-by-dimension order. Without loss of generality, we will only briefly describe a generic numerical framework in computing the $x$-component $\frac{\partial F}{\partial x}$ via a one-dimensional CFV scheme by assuming the solution $Q_E$ and the flux $F$ are functions of $x$ alone and independent of $y$. The same numerical framework for computing the $y$-component $\frac{\partial F}{\partial y}$ can be carried out similarly by assuming the solution $Q_E$ and the flux $F$ are functions of $y$ alone and independent of $x$. Therefore, we will describe the one-dimensional CFV scheme’s generic numerical framework without the burden of irrelevant superscript and subscript notations needed in a multi-dimensional case. Furthermore, since we are only interested in finding the spatial gradient of the flux at a given time $t$, the time-dependency of the solution $Q_E$ since we are only interested in finding the spatial gradient of the flux at a given time $t$, the time-dependency of the solution $Q_E$ and the flux $F$ and their corresponding subscript and superscript notations are also ignored in the discussion below.

In a one-dimensional generic numerical framework of the CFV scheme, at a given cell $I_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ with the cell center $x_i$, cell boundaries $x_{i\pm\frac{1}{2}}$, and cell-size $\Delta x$, the computational stencil $\{I_{i-1}, I_i, I_{i+1}\}$ and their corresponding numerical solution $\{Q_{i-1}, Q_i, Q_{i+1}\}$ are employed to compute the gradient of the flux written in the conservative form at $x_i$ exactly as

$$\frac{\partial F}{\partial x} \bigg|_{x_i} = \frac{\mathcal{F}_{i+\frac{1}{2}} - \mathcal{F}_{i-\frac{1}{2}}}{\Delta x}, \quad (22)$$

to guarantee the numerical solution of the CFV scheme is a weak solution of the hyperbolic conservation laws. The numerical flux $\mathcal{F}_{i+\frac{1}{2}}$ at the cell boundary $x_{i+\frac{1}{2}}$ can be approximated by the global Lax-Friedrichs (LF) flux as

$$\mathcal{F}_{i+\frac{1}{2}} = \frac{1}{2} \left[ F(Q_{i+\frac{1}{2}}) + F(Q_{i+\frac{1}{2}}^-) \right] - \frac{1}{2} \alpha \left( Q_{i+1}^+ - Q_{i+\frac{1}{2}}^- \right), \quad (23)$$

where $\alpha = \max_Q | \frac{\partial F}{\partial Q} |$ is the global maximum speed of propagation.

To obtain a formal second-order accuracy when the solution $Q(x)$ is sufficiently smooth, the numerical solution $Q_{i+\frac{1}{2}}^\pm$ at the (+) right and (-) left sides of the cell boundary $x_{i+\frac{1}{2}}$, respectively, are reconstructed via a piece-wise linear polynomial, as

$$Q_{i+\frac{1}{2}}^+ = Q_{i+1} - \frac{\Delta x}{2} (Q_x)_{i+1}, \quad \text{and} \quad Q_{i+\frac{1}{2}}^- = Q_{i} + \frac{\Delta x}{2} (Q_x)_{i}, \quad (24)$$

where the slopes are approximated as

$$(Q_x)_{i+1} = \frac{Q_{i+1} - Q_i}{\Delta x}, \quad \text{and} \quad (Q_x)_{i} = \frac{Q_{i} - Q_{i-1}}{\Delta x}.$$
Due to the non-linearity of the hyperbolic PDEs, a finite-time singularity (discontinuity) might formed. To mitigate the Gibbs oscillations from polluting the solution and destabilizing the numerical scheme, the slope \((Q_x)_i\) should be limited. Therefore, we apply the generalized nonlinear minmod slope limiter [19]

\[
(Q_x)_i = \minmod \left( \frac{Q_{i+1} - Q_i}{\Delta x}, \frac{Q_{i+1} - Q_{i-1}}{2\Delta x}, \frac{Q_i - Q_{i-1}}{\Delta x} \right), \quad \omega \in [0, 2],
\]  

(25)

where the general multivariate minmod function is

\[
\minmod (x_1, x_2, \ldots) = \begin{cases} 
\min_i \{x_i\}, & \text{if } x_i > 0 \ \forall \ i, \\
\max_i \{x_i\}, & \text{if } x_i < 0 \ \forall \ i, \\
0, & \text{otherwise}.
\end{cases}
\]

(26)

In this study, unless otherwise specified, we take \(\omega = 1.25\) for all the numerical examples.

4.2 Essentially Non-Oscillatory (ENO) Interpolation for Ghost Cells \(X_G\)

At any given time \(t = t_k, k = \{1, 2, n + 1\}\), the numerical solution at all ghost cells \(Q_E(X_G, t_k)\) are needed to be specified to allow an update of the numerical flux gradient in the interior cells \(\nabla F(X_I, \cdot)\) in the next RK-TVD stage/step. Using the CAN-property of the embedding method and the closest point mapping \(\text{cp}(x)\), the value \(Q_E(x_g, t_k)\), \(x_g \in X_G\) are mapped to the closest point \(\xi_M \in \mathcal{M}\) via the cp-mapping (5), that is, \(\xi_M = \text{cp}(x_g)\) and \(Q_E(x_g, t_k) = Q_M(\xi_M, t_k)\). By denoting \(\xi_M = x_0 = (x_0, y_0)\), we shall employ a two-dimensional Lagrange interpolation polynomial \(P(x_0) = Q_E(x_g, t_k) = f(x_0)\) with the known data \(Q_E(X_I, t_k) = f(X_I)\) at the end of the RK-TVD stage/step. Without loss of generality, we shall ignore the time-dependency of the variables and their corresponding superscript notations in the discussion for simplicity.

One can argue that a linear (e.g. central polynomial) interpolation method should work well if the underlining function is sufficiently smooth. However, one cannot assume any smoothness of the nonlinear hyperbolic PDEs’ solution in general. The solution can become discontinuous at any spatial location and time. Hence, one should not use the linear high order interpolation method near a discontinuity lest the interpolation across the discontinuity will result in numerical oscillations. To get around this predicament, a nonlinear interpolation method that biases the interpolation stencils adaptively according to the local smoothness of the data should be employed instead. Hence, to maintain the formal high order of the numerical scheme, a more computational expensive ENO interpolation process (see [15–18, 32, 33, 35, 36] for details) is employed to avoid the interpolation across any potential discontinuity. Furthermore, since the data \(f(X_I)\) are distributed uniformly on a two-dimensional Cartesian grid, a one-dimensional ENO interpolation process can be applied to the two-dimensional data in a tensorial, aka, dimension-by-dimension, manner.

Hence, the Lagrange interpolation polynomial can be written as

\[
P(x, y) = \sum_{k=n_j}^{n_j+M} \sum_{m=n_j}^{n_j+K} f_{km} l_k(x) l_m(y),
\]

(27)

where \(f_{km} = f(x_k, y_m)\), and \(l_k(x)\) and \(l_m(y)\) are the one-dimensional Lagrange interpolation polynomial of degrees \(K\) and \(M\), respectively.

The index \(k \) (\(m\)) will be biased according to the local smoothness of the solution that would minimize two \((K + 1)\)th-order \(((M + 1)\)th-order) divided differences of \(f_{km}\) if a data
point is added either to the right (top) or to the left (bottom) of the $K$th-order ($M$th-order) divided difference stencil, and $n_i$ ($n_j$) represents the left-most (bottom-most) index. The process of moving the indices $k$ and $m$ above will give the adaptive ENO interpolation. This is essentially a one-dimensional process, and hence we will describe it in a generic one-dimensional framework by assuming the data is a function of $x$ only and independent of $y$ for simplicity.

We will begin by giving the construction of the $K$th-order divided difference;

- The 0th-order divided difference of $f_i$ is defined by:
  \[ f[x_i] = f(x_i). \]  
  \[ (28) \]

- The $k$th-order divided difference is then defined recursively by:
  \[ f[x_i, \ldots, x_{i+k}] = \frac{f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}, \quad k = 1, \ldots, K. \]  
  \[ (29) \]

The interpolation polynomial, in general, can be written as
\[ P(x) = \sum_{k=n_i}^{n_i+K} f_k l_k(x). \]  
\[ (30) \]

The $n_i$ is the left-most index determined by the local smoothness of $f(x)$ at $x_i$ as measured by the $K$th-order divided differences.

- For the starting point $n_i$ with the given $x_0$:
  
  We first locate the nearest neighboring point $x_i$ such that $|x_i - x_0| \leq \Delta x/2$ and set $n_i = i$.

- For the 0th-order divided difference $f[x_n_i]$:
  $f[x_n_i]$ is formed across the one-point ENO stencil $S_1 = \{x_{n_i}\}$.

- For the 1st-order divided difference $f[x_{n_i}, x_{n_i+1}]$:
  - If $x_{n_i} \geq x_0$, then $n_i$ is updated by $n_i \leftarrow (n_i - 1)$.
  - $f[x_{n_i}, x_{n_i+1}]$ is formed across the two-points ENO stencil $S_2 = \{x_{n_i}, x_{n_i+1}\}$.

- For the $k$th-order divided difference $f[x_{n_i}, \ldots, x_{n_i+k}]$ ($k = 2, \ldots, K$):
  
  Given the $k$-points ENO stencil $S_k = \{x_{n_i}, \ldots, x_{n_i+k-1}\}$,
  
  - if $|f[I_k \cup x_{n_i-1}]| < |f[I_k \cup x_{n_i+k}]|$, then $n_i$ is updated by $n_i \leftarrow (n_i - 1)$.
  - $f[x_{n_i}, \ldots, x_{n_i+k}]$ is formed across the $(k+1)$-points ENO stencil $S_{k+1} = \{x_{n_i}, \ldots, x_{n_i+k}\}$.

5 Numerical Results

To demonstrate the numerical performance of the cp- and tc-embedding methods (refer collectively as embedding methods), we present several classical benchmark problems in one-dimensional scalar hyperbolic conservation laws with a linear, and both convex and non-convex nonlinear fluxes $F(Q, \mathcal{M})$ on a manifold.

The tested problems are given on three representative manifolds with an increasing complexity
\[ \mathcal{M}_k : p = (R_x \cos(\theta), R_y \sin(\theta)), \quad k = 1, 2, 3, \]

where $R_x$ and $R_y$ are the radii in the $x$- and $y$-directions as given in Table 1.
The shapes of the three manifolds of $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$.

**Table 1** The radius $(R_x, R_y)$ with $\alpha = \frac{\pi}{4} \theta + \frac{\pi}{8}$, $\beta = e^{\cos(2\theta)}$, the arc-length $C_k$, the computational Cartesian domain $x \times y$, the numbers of grid points $N_x \times N_y$, and the mesh resolution $\Delta x = \Delta y$ of the three manifolds $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$.

|        | $\mathcal{M}_1$ | $\mathcal{M}_2$ | $\mathcal{M}_3$ |
|--------|-----------------|-----------------|-----------------|
| $R_x$  | 1               | 1.25            | (1 + (\beta - 1) \cos^2(\theta))(1 + \sin^2(\alpha)) |
| $R_y$  | 1               | 0.75            | (1 + (\beta - 1) \cos^2(\theta))(1 + \sin^2(\alpha)) |
| $C_k$  | $2\pi$         | $\approx 6.38174971$ | $\approx 20.56468763$ |
| $x \times y$  | $[-2, 2] \times [-2, 2]$ | $[-2, 2] \times [-2, 2]$ | $[-5.8, 4.2] \times [-2.8, 2.2]$ |
| $N_x \times N_y$  | 160 $\times$ 160 | 160 $\times$ 160 | 400 $\times$ 200 |
| $\Delta x = \Delta y$  | $\frac{1}{10}$ | $\frac{1}{10}$ | $\frac{1}{10}$ |

The manifolds $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$ are a unit circle, an ellipse, and a complex closed curve, respectively, as illustrated in Fig. 2.

The counterclockwise direction is defined as a positive direction. Therefore, the tangential vector at $(x, y)$ is $v_{\mathcal{M}}(c_p(x)) = \frac{p'(\theta)}{||p'(\theta)||_2} \in T_{\mathcal{M}}$, and the arc-length$^1$ of the manifold $\mathcal{M}_k$ is

$$C_k = \int_0^{2\pi} ||p'(\theta)||_2 d\theta. \quad (31)$$

Figure 3 shows the global and local extended velocity field $v_E \in \Omega$ around the manifold $\mathcal{M}_3$ as depicted by (13). The velocity $v_E$ increases (decreases) in speed gradually from the $v_{\mathcal{M}}$ on the manifold away radially outward (inward) according to the locally signed curvature $\kappa(c_p(x))$ and follows the same direction as the direction of the tangent vector at a given point $x$ on $\mathcal{M}_3$.

The computational Cartesian domain $x \times y$, the number of grid points $N_x \times N_y$, and the mesh resolution $\Delta x = \Delta y$ and other relevant parameters employed in this study are summarized in Table 1 unless specified otherwise.

Under this spatial discretization setting, the corresponding collocating points on the manifolds will not be uniformly distributed and tend to cluster around a large curvature. For presentations of the results in the following numerical experiments, by the second-order

$^1$ The Matlab `integral` command, which uses global adaptive quadrature and certain (default) error tolerances, is used to compute the arc-length.
ENO interpolation from the collocation points on the manifold, we will obtain and plot 500 data points of the numerical results and the reference solutions distributed evenly along the angle $\theta$ of the manifolds.

By considering a (local) parameterization $\gamma(s)$ with $s$ being the arc-length of the manifold $M_k$, one has

$$ v_M \cdot \nabla_M F = \frac{\partial F}{\partial y} \frac{\partial y}{\partial s} = \frac{\partial F}{\partial s}. $$

(32)

The SPDEs of scalar hyperbolic type (1) can be written in the equivalent form of

$$ \begin{cases} 
\frac{\partial Q_M}{\partial t} + \frac{\partial F_M}{\partial s} = 0, & \text{on } M, \ t \in (0, T], \ s \in [0, C_k], \\
Q_M(s, 0) = Q_0(s), & \text{on } M.
\end{cases} 
$$

(33)

We will begin by examining the accuracy and rate of convergence of the cp- and tc-embedding methods in solving a linear wave advection with a smooth solution on the manifolds. We will then examine the essentially non-oscillatory shock-capturing performance by solving the inviscid nonlinear Burgers’ equation with a composite-waves initial condition.
on the manifold $M_1$. The numerical dissipation property and the resolution efficiency in resolving fine-scale structures in the complex solution are also studied.

Finally, several sets of benchmark one-dimensional shock-tube problems are simulated on $M_3$ with the tc-embedding method are presented. In particular, the nonlinear hyperbolic conservation laws with a convex flux (e.g., inviscid Burgers’ equation) and a concave flux (e.g., traffic flow problem) that forms isolated shock or isolated rarefaction wave, and a non-convex flux (e.g., Buckley-Leverett equation) that forms compound wave are used to validate the performance of the tc-embedding method on a complex manifold.

5.1 The Accuracy, Resolution, and Efficiency of the Embedding Methods

As for any newly developed numerical scheme, it is essential to examine the accuracy and rate of convergence of the scheme in solving a linear wave equation with a smooth solution. The $L^1$, $L^2$, and $L^\infty$ errors of the embedding methods in solving the linear wave equation show quantitatively the accuracy and the rate of convergence on the three manifolds $M_1$, $M_2$, and $M_3$. For a non-smooth solution, the nonlinear inviscid Burgers’ equation with the composite multi-waves initial condition is solved using the embedding methods to assess their relative performance in essentially non-oscillatory shock-capturing and resolution of smooth fine-scale structures. Their results are compared qualitatively with the reference solution on the manifold $M_1$. The CPU times of the embedding methods are also obtained to demonstrate their relative computational efficiency.

Example 1: Linear Wave Equation

The linear wave equation with a smooth solution is used to obtain the errors in examining the accuracy and the convergence rate of the embedding methods.

The linear flux $F$ is

$$F(Q_M) = Q_M,$$  \hspace{1cm} (34)

and the initial condition is

$$Q_0(s) = \sin(2\pi s/C_k).$$  \hspace{1cm} (35)

The exact solution is $Q_M(s, t) = Q_0(2\pi(s - t)/C_k)$. The final time is $T = C_k$.

Figure 4 shows the errors without the nonlinear minmod slope limiter ($\omega = 0$ in (25)) in the CFV scheme since the solution is smooth. In the figure, a thick black line with a slope of two is included for easy reference when comparing the convergence rate of different methods. The errors of the cp- and tc-embedding methods are shown with the dashed and solid lines, respectively. It is easy to see that the errors of the tc-embedding method are about ten times smaller than the errors of the cp-embedding method for all three manifolds. The convergence rates of both embedding methods, as expected, are second-order.

Since we are attempting to solve the hyperbolic conservation laws with non-smooth solutions, the nonlinear minmod slope limiter needs to be activated to suppress any Gibbs oscillations that might appear in the solution. Figure 5 shows the errors in the CFV scheme with the nonlinear minmod slope limiter ($\omega = 1.25$ in (25)) activated. Here, we include another thick black line with a slope of one as the limiter might reduce the convergence rate by a half order. It is easy to see that the errors of the tc-embedding method are still smaller than the errors of the cp-embedding method for all three manifolds. The convergence rate is reduced slightly to around 1.5 in the $L^2$ norm and 1.2 in the $L^\infty$ norm. These results are
consistent with the results in [19]. In summary, the results here show that the tc-embedding method is more accurate than the cp-embedding method, and both methods converge at the same rate as the mesh is refined.

Example 2: Inviscid Burgers’ Equation with Composite Waves IC

The inviscid Burgers’ equation with the composite waves initial condition is employed to demonstrate the essentially non-oscillatory shock-capturing capability near the discontinuities and the high-resolution at the fine-scale smooth structures of the scheme.

The nonlinear convex flux $F$ is

$$F(Q_M) = \frac{1}{2} Q_M^2,$$

and the initial condition consisting of composite waves is

$$Q_0(s) = \begin{cases} 
\sin(\pi \eta), & -4 \leq \eta < -1, \\
3, & -1 \leq \eta < -0.5, \\
1, & -0.5 \leq \eta < 0, \\
3, & 0 \leq \eta < 0.5, \\
2, & 0.5 \leq \eta < 1, \\
\sin(4\pi \eta), & 1 \leq \eta \leq 4, 
\end{cases}$$

(37)
Fig. 6 Example 2 (The Burgers’ equation with composite waves on the unit circle $\mathcal{M}_1$): (Left) The solution $Q_{\mathcal{M}}(s, T)$ at the final time $T = \frac{1}{20} C_1$ and (Right) the zoomed-in figure computing by the tc-embedding and the cp-embedding methods with the mesh resolutions $\Delta x = \Delta y = \frac{1}{30}$ and $\Delta x = \Delta y = \frac{1}{40}$.

Table 2 The CPU times (in seconds) of the wave equation with smooth initial condition on $\mathcal{M}_1$ (Example 1) at final time $T = C_1$ and speedup factors (SF) of the cp- and tc-embedding methods

| $\mathcal{M}_1$ with cells | CPU\textsubscript{cfv} | CPU\textsubscript{eno} | CPU\textsubscript{total} |
|-----------------------------|-------------------------|-------------------------|---------------------------|
| $\Delta x$ | $n(X_I)$ | $n(X_G)$ | cp | tc | SF | cp | tc | SF | cp | tc | SF |
| 1/10 | 360 | 128 | 7.2 | 9.0 | 0.8 | 37.6 | 13.8 | 2.7 | 44.9 | 22.9 | 2.0 |
| 1/20 | 720 | 248 | 23.0 | 27.0 | 0.9 | 138.0 | 42.4 | 3.3 | 161.3 | 69.5 | 2.3 |
| 1/40 | 1432 | 492 | 88.9 | 93.5 | 1.0 | 526.2 | 147.7 | 3.6 | 616.6 | 243.0 | 2.5 |
| 1/80 | 2848 | 1004 | 333.1 | 355.7 | 0.9 | 2025.8 | 582.7 | 3.5 | 2368.5 | 947.5 | 2.5 |
| 1/160 | 5704 | 2000 | 1300.1 | 1360.8 | 1.0 | 8171.7 | 2217.3 | 3.7 | 9579.9 | 3693.7 | 2.6 |

with $\eta = 4 \left( \frac{2s}{C_k} - 1 \right)$. The final time is $T = \frac{1}{20} C_k$. Since there is no exact solution to the Burgers’ equation with this complex initial condition, the solution of the cp-embedding method with the mesh resolution $\Delta x = \Delta y = \frac{1}{250}$ is used as the reference solution.

In Fig. 6, the solution $Q_{\mathcal{M}}(s, T)$ at the final time $T = \frac{1}{20} C_1$ on the manifold $\mathcal{M}_1$ computed by the embedding methods with the mesh resolution $\Delta x = \Delta y = \frac{1}{30}$ and $\Delta x = \Delta y = \frac{1}{40}$ are shown. The solution on the manifold exhibits multiple shocks and fine-scale structures. Both embedding methods can capture multiple discontinuities in an essentially non-oscillatory manner as the solution evolves in space and time, forming discontinuities from an otherwise smooth solution. If not well-captured by the locally adaptive numerical dissipation at a strong gradient, the Gibbs oscillations will pollute the smooth fine-scale structures. Furthermore, as shown in the right sub-figure of Fig. 6, the tc-embedding method has less numerical dissipation than the cp-embedding method allowing an efficient resolution of the fine-scale structures, which is a very important feature for a long time simulation.

Finally, we present the CPU times of the embedding methods to demonstrate their relative computational efficiency. Both embedding methods are written in Matlab version R2019a and ran on a desktop computer equipped with a four cores Intel(R) Core(TM) i7-6700 CPU with a clock speed of 3.40 GHz. Tables 2 and 3 show the CPU times of the cp-embedding method (CPU\textsubscript{cp}) and the tc-embedding method (CPU\textsubscript{tc}), and the speedup factors (SF =
### Table 3
The CPU times (in seconds) of the wave equation with smooth initial condition on $\mathcal{M}_3$ (Example 1) after 20 Runge-Kutta time steps and speedup factors (SF) of the cp- and tc-embedding methods.

| $\Delta x$ | $n(X_I)$ | $n(X_G)$ | CPU$_{cfv}$ cp | CPU$_{cfv}$ tc | SF | CPU$_{eno}$ cp | CPU$_{eno}$ tc | SF | CPU$_{total}$ cp | CPU$_{total}$ tc | SF |
|-----------|----------|----------|----------------|----------------|----|----------------|----------------|----|----------------|----------------|----|
| 1/40      | 4641     | 1623     | 6.2            | 5.7            | 1.1| 31.1           | 8.0            | 3.9| 37.3           | 13.8           | 2.7|
| 1/60      | 7006     | 2459     | 9.1            | 8.5            | 1.1| 46.0           | 11.9           | 3.9| 55.3           | 20.6           | 2.7|
| 1/80      | 9301     | 3304     | 11.5           | 11.4           | 1.0| 61.6           | 16.1           | 3.8| 73.4           | 27.9           | 2.6|
| 1/100     | 11597    | 4112     | 14.0           | 13.7           | 1.0| 77.6           | 20.6           | 3.8| 92.3           | 35.1           | 2.6|
| 1/120     | 13973    | 4899     | 17.2           | 17.2           | 1.0| 94.6           | 24.6           | 3.9| 113.3          | 42.6           | 2.7|

CPU$_{cp}$/CPU$_{tc}$ for an increasing resolution. The CPU times are obtained by running the Example 1 on the unit circle $\mathcal{M}_1$ to the final time $T = C_1$ and the complex manifold $\mathcal{M}_3$ with 20 Runge-Kutta time steps. We shall also denote $n(X_I)$ and $n(X_G)$ as the number of cells in the sets $X_I$ and $X_G$, respectively. The total CPU times (CPU$_{total}$) of the embedding methods are mainly composed of the CPU times used in the CFV method (CPU$_{cfv}$) and the ENO interpolation (CPU$_{eno}$), that is, CPU$_{total} \approx$ CPU$_{cfv}$ + CPU$_{eno}$. In terms of CPU$_{cfv}$, the embedding methods are basically the same (SF $\approx$ 1). In terms of CPU$_{eno}$, the tc-embedding method is about 3.5 times faster than the cp-embedding method (SF $\approx$ 3.5). Although $n(X_I)/n(X_G) \approx 3$, the CPU$_{eno}$ dominates over the CPU$_{cfv}$. In terms of CPU$_{total}$, the tc-embedding method is about 2.5 times faster than the cp-embedding method (SF $\approx$ 2.5). The reason for such large speedup factors is that the former eliminates the costly computations in reconstructing the CAN-property for all $x \in X_I$ at each RK-TVD stage required by the latter.

In summary, the tc-embedding method has better accuracy, improved resolution, and reduced CPU time than the cp-embedding method.

#### 5.2 The One-Dimensional Benchmark Shocked Flow Problems on $\mathcal{M}_3$

This section presents the numerical results of solving the hyperbolic conservation laws on $\mathcal{M}_3$ with several benchmark one-dimensional scalar hyperbolic conservation laws with discontinuous solutions. Both the linear ($F'' = 0$), convex ($F'' > 0$), concave ($F'' < 0$), and non-convex ($F''$ changes sign) fluxes of the hyperbolic conservation laws are considered. The problems are the linear advection equation with a discontinuous initial condition (linear flux), the inviscid Burgers’ equation with a stationary shock and with an interaction of the moving shock and composite waves (convex flux), the traffic flow problem (concave flux), and the Buckley-Leverett equation (non-convex flux). These problems demonstrate the robust performance of the tc-embedding method in resolving fine-scale structures efficiently even in the presence of a shock and the essentially non-oscillatory capturing of complex shocks and rarefraction waves existed in the nonlinear PDEs. As the results computed on the other two manifolds $\mathcal{M}_1$ and $\mathcal{M}_2$ are essentially the same as the one computed on the $\mathcal{M}_3$ when the final time $T$ is scaled proportionally with their respective arc-length $C_k$; they are omitted here for saving space.

In the figure of each following example, three sub-figures are included. The solution at the final time $Q_{\mathcal{M}}(s, T)$, the space-time $(s-t)$ evolution of $Q_{\mathcal{M}}(s, t)$ in terms of arc-length $s$ and the space-time $(x-t)$ evolution of $Q_{\mathcal{M}}(x, t)$ in terms of $x \in \Omega$ in the left, middle, and right sub-figures, respectively. In the right sub-figure, the vertical axis $z$ represents the increasing...
Fig. 7 Example 3 (The linear advection of the multi-wave on $\mathcal{M}_3$): (Left) The solution $Q_M(s, T)$ at the final time $T = C_3$. The space-time evolution of the solution (Middle) $Q_M(s, t)$ in terms of arc-length $s$ and (Right) $Q_M(x, t)$ in terms of physical location $x$

time $t$, and the manifold $\mathcal{M}_3 \in \Omega$ resides on the $x − y$ plane. In the left sub-figure, the symbols are placed with a fixed spacing distance of 3% between symbols to avoid clustering of the symbols for clarity. For all the test problems, the solutions of the tc-embedding method with the mesh resolution $\Delta x = \Delta y = \frac{1}{1250}$ on $\mathcal{M}_3$ are used as the reference solutions. The solution’s space-time evolution is illustrated with color-flooded and black solid lines contour in the middle and right sub-figures.

**Example 3: Linear Advection Equation with Multi-Waves IC**

In this example, the simple linear advection problem is solved with the linear flux $F(Q_M)$ and a piece-wise continuous multi-waves initial condition $Q_0(s)$ in (33) as given below.

The linear flux $F$ is

$$F(Q_M) = Q_M,$$  \hspace{1cm} (38)

and the multi-wave initial condition consists of a smooth Gaussian function, a discontinuous square function, a piece-wise continuous triangle function, and a smooth elliptic function,

$$Q_0(s) = \begin{cases} 
\frac{1}{6}(G(\eta, \beta, z - \delta) + G(\eta, \beta, z + \delta) + 4G(\eta, \beta, z)), & -0.8 \leq \eta \leq -0.6, \\
1, & -0.4 \leq \eta \leq -0.2, \\
1 - |10(\eta - 0.1)|, & 0 \leq \eta \leq 0.2, \\
\frac{1}{6}(H(\eta, \alpha, a - \delta) + H(\eta, \alpha, a + \delta) + 4H(\eta, \alpha, a)), & 0.4 \leq \eta \leq 0.6, \\
0, & \text{otherwise}.
\end{cases}$$ \hspace{1cm} (39)

where $\eta = 2s/C_3 - 1$, $G(x, \beta, z) = e^{-\beta(x-z)^2}$, $H(x, \alpha, a) = \sqrt{\max(0, 1 - \alpha^2(x - a)^2)}$ with $a = 0.5$, $z = -0.7$, $\delta = 0.005$, $\alpha = 10$ and $\beta = \frac{\log 2}{360^2}$. The exact solution is $Q_M(s, t) = Q_0(2(s - t)/C_3 - 1)$. The final time is $T = C_3$.

As shown in Fig. 7, the tc-embedding method is capable of advecting the discontinuous solution in an essentially non-oscillatory manner while resolving the piece-wise smooth solutions well spatially and temporally. The result is very encouraging.

Next, we shall solve the SPDEs with a nonlinear convex and non-convex flux, giving rise to complex finite-time singularity even with a smooth initial condition.
Example 4: Inviscid Burgers’ Equation with Stationary Shock

Inviscid Burgers’ equation is the simplest prototype for the scalar hyperbolic conservation laws with a nonlinear convex flux that the solution can develop either an isolated shock wave (discontinuity in the solution) or an isolated rarefaction wave (discontinuity in the first derivative of the solution). They are the weak solutions of the inviscid Burgers’ equation and satisfy an entropy inequality with entropy function $\eta = Q^2$ [20]. The Burgers’ equation is a fundamental partial differential equation in applied mathematics, such as fluid mechanics and gas dynamics. Here we solve (33) on $\mathcal{M}_3$ with the nonlinear convex flux $F$ and a smooth initial condition $Q_0(s)$ given below with the tc-embedding method.

The nonlinear convex flux $F$ is

$$F(Q_\mathcal{M}) = \frac{1}{2} Q^2_\mathcal{M},$$

and the smooth initial condition is

$$Q_0(s) = \sin\left(\frac{2\pi s}{C_3}\right).$$

The final time is $T = \frac{1}{4} C_3$.

Figure 8 shows the numerical results of the simulations. As shown in the space-time evolution of the solution $Q_\mathcal{M}(s, t)$ in the middle sub-figure, the initial smooth function develops into a stationary discontinuity (shock) at arc-length $s = C_3/2$ and the wave breaking time $t = C_3/(2\pi)$, as predicted by the theory when the left- and right-going characteristics have collided together. The weak solution also satisfies the Rankine-Hugoniot condition for the shock speed. The simple piece-wise continuous numerical solution has no discernible Gibbs oscillations. The example demonstrates that the tc-embedding method can capture the shock wave adaptively in an essentially non-oscillatory manner while resolving the piece-wise smooth solutions well spatially and temporally.

This problem is relatively easy to handle as the discontinuity is stationary in space and time. Next, we shall tackle the same problem but with moving shocks and with a complex initial condition consists of composite waves. This problem will challenge the space-time adaptivity characteristic of the tc-embedding method.
Example 5: Inviscid Burgers’ Equation with Composite Waves IC

This example is given to show the performance of the tc-embedding method on $\mathcal{M}_3$. The initial condition is complex, including four discontinuous with different jump ratios and two smooth sine functions with different wave frequencies.

The nonlinear convex flux $F$ is

$$F(Q_M) = \frac{1}{2} Q_M^2,$$

and the initial condition consisting of composite waves is given in (37). The final time is $T = \frac{1}{20} C_3$.

Figure 9 shows the Burgers’ equation with the composite waves initial condition. As shown in the space-time evolution of the solution $Q_M(s, t)$ in the middle sub-figure, the left-most sine wave develops into stationary shock at $s = C_3/8$, while the right-most high-frequency sine wave to the right of the strong shock develops into a train of shocklets. The two right-moving shocks are formed in the middle and moves to the right. They are captured essentially non-oscillatory as they evolve around the manifold $\mathcal{M}_3$ in space and time, as shown in the right sub-figure. This challenging example demonstrates the space-time adaptivity characteristic of the tc-embedding method on a non-trivial shape of the manifold.

Example 6: Traffic Flow Problem

In mathematics and transportation engineering, traffic flow is a fundamental theory to analyze and study the interactions between travellers (mainly cars) and infrastructure, with the aim of understanding the relationship between traffic flow, velocity and density, reducing the occurrence of accidents, and improving the efficiency movement of traffic.

The nonlinear concave flux $F$ is

$$F(Q_M) = v_{\text{max}} Q_M (1 - Q_M/Q_{\text{max}}),$$

where $Q_M$ is the cars’ density. We take $v_{\text{max}} = 1$ and $Q_{\text{max}} = 1$ as the limited speed and density, respectively. The initial condition is

$$Q_0(s) = \begin{cases} 1, & -0.2 \leq \eta \leq 0.6, \\ \frac{1}{2}, & \text{otherwise}, \end{cases}$$

Fig. 10 Example 6 (The traffic flow problem on $\mathcal{M}_3$): (Left) The solution $Q_M(s, T)$ at the final time $T = \frac{2}{5} C_3$. The space-time evolution of the solution (Middle) $Q_M(s, t)$ in terms of arc-length $s$ and (Right) $Q_M(x, t)$ in terms of physical location $x$.

where $\eta = \frac{2s}{C_3} - 1$. The final time is $T = \frac{2}{5} C_3$.

As shown in Fig. 10, there are two strong discontinuities (front and back of the traffic jam) in the initial conditions. The left discontinuity (shock) propagates to the left as the cars in the low-density region slow down and become congested as cars enter the high-density region. The right discontinuity evolves into a rarefaction wave as cars’ density reduces gradually, as the cars speeding up, from the high-density region toward the low-density region. The left-going shock and the left-going rarefaction waves are well-captured in an essentially non-oscillatory manner by the tc-embedding method on $\mathcal{M}_3$ spatially and temporally, as shown in the middle and right sub-figures.

The examples above show that the tc-embedding method performs equally well for both the convex and concave fluxes. We will conclude the study showing that the tc-embedding method also works well with a non-convex flux, where the second derivative of the flux changes sign, on $\mathcal{M}_3$.

Example 7: Buckley-Leverett Problem

In fluid dynamics, the Buckley-Leverett equation is a model for a two-phase oil–water immiscible displacement process in porous media in a one-dimensional or quasi-one-dimensional reservoir.

The nonlinear non-convex flux $F$ is

$$F(Q_M) = \frac{Q_M^2}{Q_M^2 + \mu(1 - Q_M)^2}, \quad (45)$$

with the viscosity $\mu = 1/4$. The initial condition is

$$Q_0(s) = \begin{cases} 1, & -0.5 \leq \eta \leq 0, \\ 0, & \text{otherwise,} \end{cases} \quad (46)$$

where $\eta = \frac{2s}{C_3} - 1$. The final time is $T = \frac{1}{5} C_3$.

The flux is S-shaped with an inflection point and, therefore, non-convex. It yields a rarefaction-shock compound wave, also known as the Buckley-Leverett (BL) profile, which consists of a shock wave immediately followed by a rarefaction wave (see [20] for details). Figure 11 shows that the BL profile is captured essentially non-oscillatory and sharply by the tc-embedding method. The space-time evolution of the solution shows the formation
Fig. 11 Example 7 (The Buckley-Leverett problem on $\mathcal{M}_3$): (Left) The solution $Q_\mathcal{M}(s, T)$ at the final time $T = \frac{1}{5} C_3$. The space-time evolution of the solution (Middle) $Q_\mathcal{M}(s, t)$ in terms of arc-length $s$ and (Right) $Q_\mathcal{M}(x, t)$ in terms of physical location $x$.

process of the BL profile as the shock waves move towards the right stably together with the expanding rarefaction fans attached.

5.3 The Burgers’ Equation on the Two-Dimensional Manifolds

This section presents preliminary numerical results for applying the tc-embedding method for solving the moving Burgers’ equation on two-dimensional manifolds embedded in $\mathbb{R}^3$. More specifically, they are the torus-shaped and spherical-shaped manifolds. We conjuncture that the generalization of Theorem 2 to $\mathbb{R}^3$ can be done by using extended velocity field as in (13) on codimension-2 manifolds constructed by slicing $\mathcal{M}$ along $v_\mathcal{M}$ direction. We leave the proof to our future work.

We first illustrate the performance of the tc-embedding method in solving SPDEs on a two-dimensional parametrized torus surface:

$$
\begin{align*}
    x(\theta, \phi) &= (R + r \cos \phi) \cos \theta, & y(\theta, \phi) &= (R + r \cos \phi) \sin \theta, & z(\theta, \phi) &= r \sin \theta.
\end{align*}
$$

(47)

where the parametrization parameters are $\theta \in [-\pi, \pi)$ and $\phi \in [-\pi, \pi)$. $R = 1$ is the distance from the center of the tube to the center of the torus, and $r = 0.5$ is the radius of the tube. The SPDEs Burgers’ equation simulates the finite-time formation of shock waves and their movement on a surface. The initial condition is

$$
Q_0(\theta + \phi) = \frac{1}{2} + \sin(\theta + \phi).
$$

(48)

The final time is $T = \pi$. In Fig. 12, the solutions $Q_\mathcal{M}(x, t)$ of the Burgers’ equation at times $(t = 0, 1, 2)$ on the torus surface computed by the tc-embedding methods with the mesh resolution $\Delta x = \Delta y = \Delta z = \frac{1}{32}$ are shown. The results show that the tc-embedding method can capture the formation of the strong gradient and shock sharply and essentially non-oscillatory and its later evolution on the torus-shaped manifold.

We conclude this study by performing a similar but slightly more complex evolution of four Gaussian-shaped spots distributed on a spherical-shaped surface:

$$
\begin{align*}
    x(\theta, \phi) &= R \sin \phi \cos \theta, & y(\theta, \phi) &= R \sin \phi \sin \theta, & z(\theta, \phi) &= R \cos \phi.
\end{align*}
$$

(49)
Fig. 12 The solution of the Burgers’ equation computed by the tc-embedding methods on the torus surface at times $t = 0, 1, 2$

Fig. 13 The solution of the Burgers’ equation computed by the tc-embedding methods on the spherical surface at times $t = 0, 1, 2$

where the parametrization parameters are $\theta \in [0, 2\pi)$ and $\phi \in [0, \pi)$. $R = 1$ is the radius. The initial condition is

$$Q_0(\theta, \phi) = C + \frac{1}{\sqrt{2\pi} \sigma} \sum_{i=1}^{4} e^{-\frac{(\theta-a_1 k_i)^2 + (\phi-a_2 h_i)^2}{2\sigma^2}} ,$$

(50)

where $C = 10^{-5}$, $a_1 = \frac{\pi}{4}$, $k_i = 2i - 1$, $a_2 = \frac{\pi}{5}$, $h_i = i$, and $\sigma = \frac{\pi}{8}$. The final time is $T = \pi$. In Fig. 13, the solutions $Q_M(x, t)$ of the Burgers’ equation at times $(t = 0, 1, 2)$ on the spherical surface computed by the tc-embedding methods with the mesh resolution $\Delta x = \Delta y = \Delta z = \frac{1}{12}$ are shown. The initial condition ($t = 0$) shows the four peaks on the spherical surface. Similar to the single shock formation in the previous example, the proposed algorithm can capture the formation of the strong gradient and shock sharply and essentially non-oscillatory. The movements and mutual interactions among the Gaussian functions are well-resolved, and discontinuities are well-captured, as depicted in the middle and right subfigures.

6 Conclusion

A new closest point time-continuous embedding (tc-embedding) method is developed for solving nonlinear scalar hyperbolic conservation law (PDEs) on one-dimensional, connected, smooth, and closed manifolds (SPDEs). It improves upon the classical closest point embedding (cp-embedding) method that requires a costly re-establishment of the exten-
tion function’s constant-along-normal (CAN) property at every time step; the tc-embedding method incorporates the CAN-property analytically and explicitly in the SPDEs directly. The main idea in the tc-embedding method for solving SPDEs on a one-dimensional manifold is that under certain minor restrictions, the time-independent extended velocity field only needs to be determined once at the initialization stage and then used subsequently for all times in the method; thus, a substantial saving in the computational times can be realized.

Due to the nonlinear nature of the PDEs, finite-time singularities, such as shocks and rarefaction waves, can appear spontaneously in the solution. A second-order central finite volume scheme with a nonlinear minmod slope limiter is used to approximate the flux gradients discretely in a computational tube placed inside a Cartesian rectangular uniformly spaced mesh to mitigate Gibbs oscillations. It requires values at the ghost cells to compute the flux gradients at the boundary cell at the edges of the computational tube. By using the cp-mapping and CAN-property to map the location of a ghost cell to a point on the manifold, the ENO interpolation process updates the values of the ghost cells using data around the point on the manifold. The adaptive ENO interpolation is an essential technique in obtaining an essentially non-oscillatory value by avoiding interpolating across any potential discontinuity or high gradient in the solution. The third-order total variation diminishing Runge-Kutta (RK-TVD) scheme integrates the numerical solution in time.

We compared both methods’ accuracy on the linear advection equation with a smooth solution on simple and complex manifolds. While both methods are formally second order, the tc-embedding method is more accurate than the cp-embedding method overall. We have also compared both embedding methods’ resolution power and CPU times in solving the Burgers’ equation with a piece-wise continuous solution on a circular manifold. The results indicate that the tc-embedding method has a lesser numerical dissipation, resolves the fine-scale structures more efficiently, and is roughly two and half times faster CPU times than the cp-embedding method. Finally, several one-dimensional benchmark shocked flow problems, such as the Burgers’ equation with convex flux, the traffic flow problem with concave flux, and the Buckley-Leverett equation with non-convex flux, are used to demonstrate the essentially non-oscillatory shock-capturing property and the efficient resolution of fine-scale structures of the tc-embedding method on a complex manifold. The numerical results of the Burgers’ equation are also given for the two-dimensional torus-shaped and spherical-shaped manifolds.

These preliminary results show the potential synergy between the embedding method for solving SPDEs with smooth solutions and the nonlinear shock-capturing scheme for hyperbolic conservation laws with non-smooth (discontinuous) solutions. This study serves as the starting point, and there are more challenges in the theoretical and numerical aspects of combining both for solving more sophisticated hyperbolic SPDEs. As immediate follow-up work, extending the theoretical and numerical frameworks of the tc-embedding method for solving the scalar hyperbolic conservation laws on a two-dimensional complex connected manifold with a well-defined velocity field is challenging and will be investigated. As for future works, we will study if the tc-embedding method can help solve the one-dimensional or multi-dimensional system of hyperbolic conservation laws, for example, shallow water equations, Euler equations, multi-phases flow, and multi-components flow, on a stationary or moving manifold. Moreover, high-order shock-capturing schemes, with improved resolution, for example, the weighted ENO (WENO) finite volume/finite difference schemes [5, 6, 34] on a regular Cartesian mesh and the discontinuous Galerkin [11] method (DG) on a structured and unstructured triangular mesh, is also currently under consideration.

Acknowledgements The author (Yinghua Wang) would like to thank Nanjing Tech University for providing the startup funding (39804138) to support this work. The authors would like to acknowledge the funding
support of this research by the National Natural Science Foundation of China (11871443) and a Hong Kong Research Grant Council GRF Grant. The author (Don) also likes to thank the Ocean University of China for providing the startup funding (201712011) to support this work. The authors also thank Mr. Kang-Bo Tian for sharing the program on the tc-embedding methods on the two-dimensional manifold in $\mathbb{R}^3$.

**Author Contributions** The authors have contributed equally in this scholarly work.

**Funding** The authors have not disclosed any funding.

**Data Availability** No data sets were generated or analyzed during the current study.

**Declaration**

**Conflict of interest** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit [http://creativecommons.org/licenses/by/4.0/](http://creativecommons.org/licenses/by/4.0/).

**References**

1. Auer, S., Macdonald, C.B., Treib, M., Schneider, J., Westermann, R.: Real-time fluid effects on surfaces using the closest point method. Comput. Graph. Forum 31(6), 1909–1923 (2012)
2. Ayton, G.S., Mcwhirter, J.L., Mcmurtry, P., Voth, G.A.: Coupling field theory with continuum mechanics: a simulation of domain formation in giant unilamellar vesicles. Biophys. J. 88(6), 3855–3869 (2005)
3. Beale, J.T.: Solving partial differential equations on closed surfaces with planar cartesian grids. SIAM J. Sci. Comput. 42(2), A1052–A1070 (2020)
4. Bertalmio, M., Bertozzi, A.L., Sapiro, G.: Navier-Stokes, fluid dynamics, and image and video inpainting. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001. IEEE (2001)
5. Borges, R., Carmona, M., Costa, B., Don, W.S.: An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. J. Comput. Phys. 227(6), 3191–3211 (2008)
6. Castro, M., Costa, B., Don, W.S.: High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws. J. Comput. Phys. 230(5), 1766–1792 (2011)
7. Chen, M., Ling, L.: Kernel-based collocation methods for heat transport on evolving surfaces. J. Comput. Phys. 405, 109166 (2020)
8. Chen, Y.J., Macdonald, C.B.: The closest point method and multigrid solvers for elliptic equations on surfaces. SIAM J. Sci. Comput. 37(1), A134–A155 (2015)
9. Cheung, K.C., Ling, L.: A kernel-based embedding method and convergence analysis for surfaces PDEs. SIAM J. Sci. Comput. 40(1), A266–A287 (2018)
10. Chu, J., Tsai, R.: Volumetric variational principles for a class of partial differential equations defined on surfaces and curves. Res. Math. Sci. 5(2), 19 (2018)
11. Cockburn, B., Lin, S.-Y., Shu, C.-W.: TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems. J. Comput. Phys. 84(1), 90–113 (1989)
12. Flyer, N., Wright, G.B.: A radial basis function method for the shallow water equations on a sphere. Proc. R. Soc. A Math. Phys. 465(2016), 1949–1976 (2009)
13. Gottlieb, S., Shu, C.-W.: Total variation diminishing Runge-Kutta schemes. Math. Comput. 67(221), 73–85 (1998)
14. Greer, J.B.: An improvement of a recent Eulerian method for solving PDEs on general geometries. J. Sci. Comput. 29, 321–352 (2006)
15. Harten, A.: ENO schemes with subcell resolution. J. Comput. Phys. 83(1), 148–184 (1989)
16. Harten, A.: Multi-resolution analysis for ENO schemes. In: Hussaini, M.Y., Kumar, A., Salas, M.D. (eds.) Algorithmic Trends in Computational Fluid Dynamics. Springer, New York (1993)
17. Harten, A.: Adaptive multisolution schemes for shock computations. J. Comput. Phys. 115(2), 319–338 (1994)
18. Harten, A., Engquist, B., Osher, S., Chakravarthy, S.R.: Uniformly high order accurate essentially non-oscillatory schemes III. J. Comput. Phys. 131(1), 3–47 (1997)
19. Kurganov, A., Tadmor, E.: New high-resolution central schemes for nonlinear conservation laws and convection diffusion equations. J. Comput. Phys. 160(1), 241–282 (2000)
20. LeVeque, R.J.: Numerical Methods for Conservation Laws. Birkhäuser, Basel (1992)
21. Macdonald, C.B., Brandman, J., Ruuth, S.J.: Solving eigenvalue problems on curved surfaces using the closest point method. J. Comput. Phys. 230, 7944–7956 (2011)
22. Macdonald, C.B., Ruuth, S.J.: Level set equations on surfaces via the closest point method. J. Sci. Comput. 35, 219–240 (2008)
23. Macdonald, C.B., Ruuth, S.J.: The implicit closest point method for the numerical solution of partial differential equations on surfaces. SIAM J. Sci. Comput. 31(6), 4330–4350 (2010)
24. Mésmoli, F., Sapiro, G., Thompson, P.: Implicit brain imaging. NeuroImage 23, S179–S188 (2004)
25. Myers, T.G., Charpin, J.P.F.: A mathematical model for atmospheric ice accretion and water flow on a cold surface. Int. J. Heat Mass Transf. 47(25), 5483–5500 (2004)
26. Myers, T.G., Charpin, J.P.F., Chapman, S.J.: The flow and solidification of thin fluid film on an arbitrary three-dimensional surface. Phys. Fluids 14(8), 2788–2803 (2002)
27. Olsen, L., Maini, P.K., Sherratt, J.A.: Spatially varying equilibria of mechanical models: application to dermal wound contraction. Math. Biosci. 147(1), 113–29 (1998)
28. Petras, A., Ling, L., Ruuth, S.J.: An RBF-FD closest point method for solving PDEs on surfaces. J. Comput. Phys. 370, 43–57 (2018)
29. Piret, C.: The orthogonal gradients method: A radial basis functions method for solving partial differential equations on arbitrary surfaces. J. Comput. Phys. 231(14), 4662–4675 (2012)
30. Ruuth, S.J., Merriman, B.: A simple embedding method for solving partial differential equations on surfaces. J. Comput. Phys. 227(3), 1943–1961 (2008)
31. Shankar, V., Wright, G.B.: Mesh-free semi-Lagrangian methods for transport on a sphere using radial basis functions. J. Comput. Phys. 366, 170–190 (2018)
32. Shu, C.-W.: Numerical experiments on the accuracy of ENO and modified ENO schemes. J. Sci. Comput. 5, 127–149 (1990)
33. Shu, C.-W.: Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In: Quarteroni, A. (ed.) Advanced Numerical Approximation of Nonlinear Hyperbolic Equations. Springer, Berlin (1998)
34. Shu, C.-W.: High order weighted essentially nonoscillatory schemes for convection dominated problems. SIAM Rev. 51(1), 82–126 (2009)
35. Shu, C.-W.: Essentially non-oscillatory and weighted essentially non-oscillatory schemes. Acta Numer. 29, 701–762 (2020)
36. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. II. In: Yousuff Hussaini, M., Leer, B., Rosendale, J. (eds.) Upwind and High-Resolution Schemes. Springer, Berlin (1989)
37. Suchde, P., Kuhnert, J.: A meshfree generalized finite difference method for surface PDEs. Comput. Math. Appl. 78(8), 2789–2805 (2019)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.