Lanczos Advection: Near-Orthogonal Semi-Lagrangian Reconstruction for Fluid Simulation

SEUNG-WOOK KIM, Hankuk University of Foreign Studies, South Korea
KIWON UM, LTCI, Telecom Paris, IP Paris, France

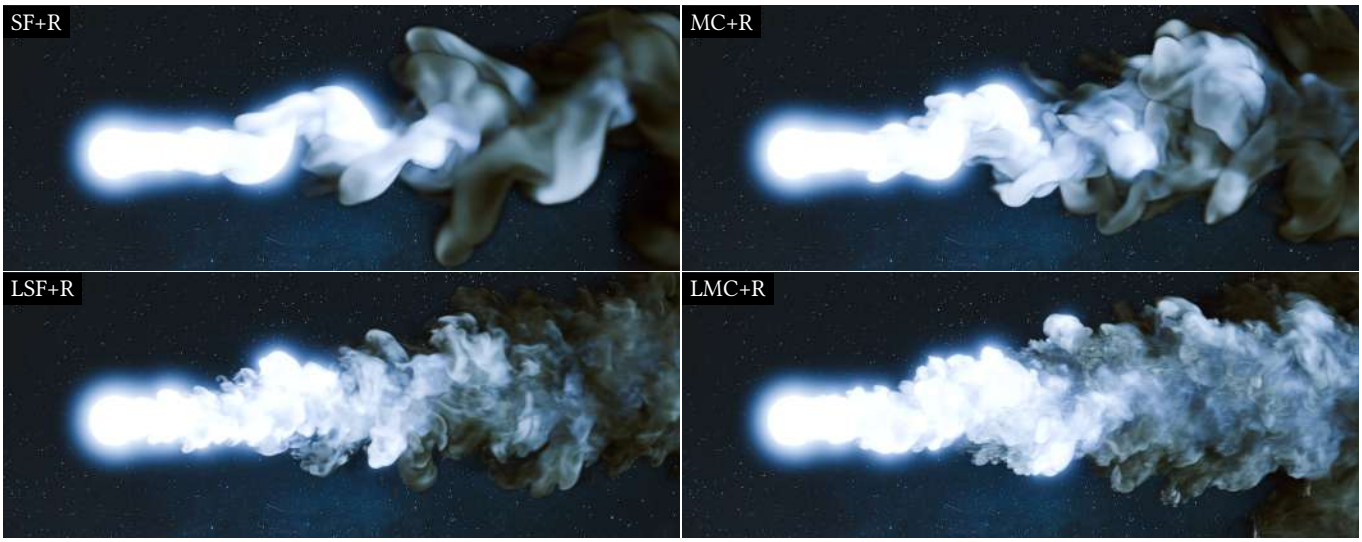
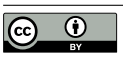


Fig. 1. Meteor. Each baseline solver of SF+R and MC+R is augmented to LSF+R and LMC+R with our extension. Our Lanczos-based advection operator preserves vorticity and generates noticeably richer, small-scale turbulence than their baseline counterparts even with a large CFL number (> 8).

Producing vivid fluid details, such as the sharp spatial structure of interfaces and vortices, is essential for high-quality fluid simulations in computer graphics. The advection solver is a vital component of fluid simulation, and the semi-Lagrangian method is widely adopted as a central building block for many advanced advection methods thanks to its unconditional stability. Despite these notable improvements, the numerical dissipation and energy loss inherent in conventional linear interpolation remain considerable drawbacks, compromising visual fidelity. This paper proposes a novel extension of the core semi-Lagrangian advection algorithm to significantly alleviate numerical dissipation and energy loss. Our extension uses a dimension-splitting weighted sum with a windowed sinc (also known as Lanczos) kernel, leveraging the properties of the sinc function to construct a bounded, near-orthogonal semi-Lagrangian reconstruction. Our experiments, comparing various advection methods and their discrete operators, demonstrate that our extension significantly improves the orthogonality of the operator, energy-retaining behavior and enables richer visual details.

CCS Concepts: • **Computing methodologies** → **Physical simulation**; *Animation*; Modeling and simulation.

Authors' Contact Information: Seung-wook Kim, seungwook.kim@hufs.ac.kr, Hankuk University of Foreign Studies, South Korea; Kiwon Um, kiwon.um@telecom-paris.fr, LTCI, Telecom Paris, IP Paris, France.



This work is licensed under a Creative Commons Attribution 4.0 International License.
© 2026 Copyright held by the owner/author(s).
ACM 1557-7368/2026/12-ART245
https://doi.org/10.1145/3842541

Additional Key Words and Phrases: Fluid simulation, Semi-Lagrangian advection, Lanczos kernel, Bounded near-orthogonal reconstruction, Energy retention, Vorticity, Numerical dissipation

ACM Reference Format:
Seung-wook Kim and Kiwon Um. 2026. Lanczos Advection: Near-Orthogonal Semi-Lagrangian Reconstruction for Fluid Simulation. *ACM Trans. Graph.* 45, 6, Article 245 (December 2026), 18 pages. https://doi.org/10.1145/3842541

1 Introduction

Visual details of fluid flow, such as vortices, are crucial to high-quality fluid simulations for computer graphics applications. To obtain the desired visual appearance of fluids, the Navier-Stokes equations (NSEs), which describe the motion of fluids as non-linear partial differential equations, have been thoroughly studied with numerical methods. The numerical integration of NSEs naturally adopts discretization and numerical schemes and thus ought to manage numerical dissipation. Such numerical dissipation often characterizes the visual quality of fluid simulations. The advances in many numerical methods for NSEs reflect the great effort to reduce numerical dissipation and eventually improve the visual quality of fluid simulations.

A challenge, among others, of numerical simulations for high-quality fluids lies in the non-linear advection part of the NSEs. For this vital step, the semi-Lagrangian method, well-known as *stable fluids* (SF) [Stam 1999], has been widely used. This method solves the advection part using backward tracing and linear interpolation, which significantly contributes to numerical dissipation. However,

despite the drawback, thanks to its unconditionally stable nature and straightforward algorithm to implement, the core routine of the semi-Lagrangian method has been an important building block of many advanced advection methods, e.g., back-and-forth error compensation and correction (BFECC) [Kim et al. 2005], MacCormack (MC) [Selle et al. 2008], and monotonic constrained interpolation profile (MCIP) [Song et al. 2005]. Consequently, however, those methods inevitably inherit the initial drawback of the semi-Lagrangian method although their entire advection routines yield more accurate advection with less numerical dissipation.

On the other hand, the computational fluid dynamics community has developed sophisticated high-order reconstruction advection schemes, most notably the essentially non-oscillatory (ENO) [Harten et al. 1987], weighted ENO (WENO) [Jiang and Shu 1996], and targeted ENO (TENO) [Fu et al. 2016] families. These schemes have achieved remarkable success in shock-capturing simulations by adaptively selecting or weighting candidate stencils to avoid interpolation across discontinuities. Despite their effectiveness, the energy behavior of their resulting advection operators has not been explored enough

Acknowledging the popularity of the semi-Lagrangian advection method, we focus on tackling a source of numerical dissipation in its core algorithm, i.e., interpolation after backward tracing. This paper introduces a new extension of the semi-Lagrangian advection algorithm aiming at an orthogonal operator form. To this end, the construction of our advection operator utilizes a Lanczos kernel function, which is one kind of windowed sinc function. Our method leverages the mathematical properties of the sinc function, particularly as its shifted functions form a set of orthonormal bases, which helps to construct a bounded, near-orthogonal semi-Lagrangian reconstruction. The resulting advection scheme exhibits energy-retention benefits and reduces numerical dissipation without a significant sacrifice of stability and computational efficiency.

The key contributions of this work can be summarized as follows:

- A novel extension of the core semi-Lagrangian advection algorithm is introduced, realized as a bounded, near-orthogonal reconstruction whose discrete advection operator delivers energy-retention benefits. The extended advection method results in richer visual details of fluid flow, such as vortices and sharp spatial structures of interfaces.
- A deeper insight into advection is studied with a systematic analysis via a discrete advection operator viewpoint. Our experiments provide a better understanding of different advection schemes, revealing a quantitative comparison in the preservation of physical quantities, which pertains to the visual quality of flow details.
- The experiments of our extension applied to different advection methods are presented, showcasing significant improvements in visual details for all of the base methods, which indicates its great potential for further applications.

2 Related Work

Minimizing numerical dissipation remains a central challenge in physics-based fluid simulation. The numerical dissipation, referred to as artificial viscosity, easily smears out interface structures and suppresses turbulent features, which are important for high-quality visual simulation of fluids. Consequently, researchers have explored numerous strategies to mitigate or compensate for these artifacts, aiming to achieve a desired balance among numerical accuracy, computational overhead, and ease of implementation. In this section, we first discuss the "advection" solvers that try to tackle numerical dissipation, as our work lies in improving this particular component of fluid simulation. For a broader overview of the fight against numerical dissipation in fluid simulations, we refer to the thorough review by Wang et al. [2024].

The semi-Lagrangian method introduced by Stam [1999] remains popular, thanks to its unconditional stability and straightforward algorithm. However, it is known to incur significant smoothing because of an inaccurate approximation of backward tracing and accumulation of interpolation. Multiple methods have tried to tackle this drawback. Notably, MC [Selle et al. 2008] and BFECC [Kim et al. 2005] used the SF advection routine as their sub-routine and applied an error correction to reduce dissipation and capture finer details. MCIP [Song et al. 2005], unsplit CIP (USCIP) [Kim et al. 2008a], quadratic upstream interpolation for convective kinematics (QUICK) [Molemaker et al. 2008], and constrained moving least-squares interpolation profile (CMIP) [Kim and Hong 2013] methods elevated the order of accuracy utilizing derivatives or flux with non-negligible additional computational cost.

The accumulation of interpolation errors, recognized as a long-standing limitation of conventional advection, has been drastically addressed by *flow map* (also called *characteristic map*) methods [Tessendorf 2015; Tessendorf and Pelfrey 2011]. These methods explicitly tracked the forward or backward (or even both) *mapping functions* in time and reduced the interpolation error accumulation significantly. Sato et al. [2018; 2017] proposed a *long-term* semi-Lagrangian approach that evolves the backward characteristic mapping more accurately. The BiMocq [Qu et al. 2019] framework blended the forward and backward flow maps to enhance long-range consistency, curbing the usual smearing artifact. More recently, the covector fluids method [Nabizadeh et al. 2022] unified a flow map formulation with an "impulse-based" solver to preserve circulation and momentum, and Yin et al. [2023] extended the characteristic mapping concept to three-dimensional Euler equations. Despite the higher fidelity of flow map-based methods, they come with increased memory demands since storing and updating global mapping functions for each time step are not negligible. As a modern solution to this storage challenge, *neural networks* can be considered to more compactly represent velocity fields as in neural flow maps [Deng et al. 2023].

Given the discrete nature of numerical methods, the interpolation kernel naturally plays a critical role in both the conventional semi-Lagrangian or flow map frameworks. Instead of a simple linear interpolation, B-spline interpolation has been employed in these frameworks due to its favorable properties: B-splines enforce continuity of derivatives across the entire grid and exhibit better conservation properties with respect to the L^2 and C^1 norms compared to piecewise Lagrange polynomials of equivalent order [Bermejo 1995; Riishøjgaard et al. 1998]. Quadratic B-splines have also been explored for their natural C^1 continuity, enabling backward semi-Lagrangian methods with reduced numerical dissipation [Gagniere

et al. 2020]. In recent particle-based flow map methods, B-spline kernels have been utilized for efficient and smooth particle-to-grid transfers [Cao et al. 2022], combining the advantages of Lagrangian particle tracking with Eulerian grid-based projection. Investigation into the trade-off between dissipation and dispersion in different interpolation kernels, such as Hermite splines, Catmull-Rom splines, and Lagrange polynomials, continues to aim for improving advection accuracy [Kim et al. 2008a; Staniforth and Côté 1991].

In the domain of computational fluid dynamics, high-order shock-capturing schemes have been developed to address numerical dissipation while maintaining stability near discontinuities. The ENO schemes [Harten et al. 1987; Shu and Osher 1988] introduced an adaptive stencil selection procedure that automatically chooses the locally smoothest stencil, thereby avoiding interpolation across discontinuities. Building upon ENO, the WENO schemes [Jiang and Shu 1996; Liu et al. 1994] employ a convex combination of all candidate stencils rather than selecting only the smoothest one, achieving higher order accuracy while retaining excellent shock-capturing capability. The WENO schemes have become widely adopted for simulating compressible flows with shocks, contact discontinuities, and complex flow structures [Shu 1998]. More recently, the TENO schemes [Fu et al. 2016, 2017] have been proposed, which either apply a candidate stencil with optimal weight or eliminate it entirely when a genuine discontinuity is detected. The TENO schemes demonstrate significantly reduced numerical dissipation compared to the classical WENO while preserving robust shock-capturing capabilities, making them particularly suitable for turbulence-resolving simulations [Hamzehloo et al. 2021]. Although these high-order reconstruction schemes were originally developed for compressible flow simulations governed by the Euler equations, their underlying principles of adaptive stencil weighting and discontinuity detection offer valuable insights for designing less dissipative interpolation schemes in semi-Lagrangian advection.

As the numerical dissipation is tightly coupled with discretization, a straightforward approach to curtail the dissipation is to refine the computational grid, thus reducing truncation errors. However, this can be prohibitively expensive. Alternatively, adaptive mesh refinement (AMR) or octree-based methods [Losasso et al. 2004] attempted to concentrate resources in the regions of interest where high vorticity or boundaries exist. Although these strategies decreased global computational costs, they often complicated the pressure projection, which is generally the most critical part, potentially adding overhead when handling dynamic or irregular domains [Cohen et al. 2010; Feldman et al. 2005; Shah et al. 2004].

Another important line of work has focused on preserving or reintroducing small-scale details lost in coarse meshes. Vorticity confinement [Fedkiw et al. 2001] and vortex particle method [Selle et al. 2005; Yoon et al. 2009] augmented underresolved vortices with additional injection of energy although excessive tuning of the confinement term can produce noisy turbulence [He and Lau 2013; He et al. 2011]. Other solutions, such as wavelet turbulence [Kim et al. 2008b] and k–ε models [Narain et al. 2008], enriched low-cost flow simulations by injecting synthetic, fine-scale structures.

In the domain of signal processing, interpolation has been explored to restrain sampling-induced smoothing in grid operations. In particular, windowed sinc (i.e., Lanczos) filters appear across wave- and flow-based simulators as high-fidelity samplers: for off-grid pressure sampling in finite-difference time-domain acoustics where the strong accuracy was reported [Bilbao 2021]; at multiresolution interfaces in seismic finite-difference schemes where a Lanczos downsampling filter stabilizes transfers between coarse and fine grids [Kristek et al. 2010]; for volumetric resampling before isosurface extraction in large fluid volumes [Krüger et al. 2006]; and as scale-separating low-pass filters on gridded ocean velocities [Haza et al. 2016]. Foundational analyses of Lanczos filtering on grid-based fields from meteorology further explain its reduced Gibbs oscillations and favorable spectral characteristics for interpolation and resampling [Duchon 1979].

Recent advances in addressing numerical dissipation for fluid simulations have tackled diverse sources of error. Among others, our approach focuses on the semi-Lagrangian advection routine, particularly its unavoidable "interpolation" step. Unlike earlier higher-order interpolation or *a posteriori* injection schemes, our method proposes a carefully constructed weight function, leveraging the properties of the windowed sinc function, yielding a bounded, near-orthogonal semi-Lagrangian reconstruction whose advection operator retains more kinetic energy than baseline interpolants in our experiments. This strategy not only enables its seamless integration into an existing semi-Lagrangian-based framework but also opens avenues for combining our extension with other advanced techniques.

3 Advection Operator

When various partial differential equation (PDE) problems are solved, including NSEs for fluids, the advection term frequently appears to be challenging, and thus critical for the solution of the problem. For fluid flows, the advection of the momentum equation forms a non-linear PDE as follows:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = 0, \tag{1}$$

where $\mathbf{u}$ is the velocity field and t denotes the time.

In order to solve the advection equation of Eq. (1), when a numerical scheme is realized using n discrete nodes and a time step size of Δt, an advection operator $\mathbf{A}[\mathbf{u}^n] \in \mathbb{R}^{n \times n}$ can be defined as follows:

$$\mathbf{u}^{n+1} = \mathbf{A}[\mathbf{u}^n]\,\mathbf{u}^n. \tag{2}$$

Note that, for example, the typical semi-Lagrangian scheme with a single backward tracing leads $\mathbf{A}[\mathbf{u}^n]\mathbf{u}^n$ to naturally be a (bi- or tri-)linear weighted sum of $\mathbf{u}^n$ at each back-traced location. It is worth noting that many variants for improved advection eventually adapt different weighted-sum strategies.

The advection operator $\mathbf{A}[\mathbf{u}^n]$ of Eq. (2) (henceforth, $\mathbf{A}$ for brevity) is desired to uphold the following properties, which are important for both stable and reliable solutions:

- **Normalization condition**: Each row of $\mathbf{A}$ sums to one, i.e.,

$$\sum_j A_{ij} = 1 \quad \text{for all } i,$$

  where A_{ij} is the element of $\mathbf{A}$ at i-th row and j-th column. This condition, also referred to as the *partition of unity* property, ensures that the resulting value at each node is a proper weighted sum of neighboring values. This prevents undesired inflation and deflation of values in advection.

- **Momentum preservation**: Each column of $\mathbf{A}$ sums to one, i.e.,
$$\sum_i A_{ij} = 1 \quad \text{for all } j.$$
This property ensures that the total contribution of $\mathbf{u}(t)$ from each node to its neighboring nodes remains consistent; hence, the momentum is conserved during the advection process.
- **Energy preservation**: The operator $\mathbf{A}$ is orthogonal, i.e.,
$$\mathbf{A}^\top \mathbf{A} = \mathbf{I},$$
where $\mathbf{I}$ is the identity matrix. This property ensures that the velocity norm remains consistent, holding $\|\mathbf{u}(t+\Delta t)\|^2 = \|\mathbf{A}\mathbf{u}(t)\|^2 = \|\mathbf{u}(t)\|^2$; thus, the kinetic energy is preserved after advection.

Given the desired properties above, our goal is to build an advection operator that satisfies the properties as much as possible, particularly gauging the energy retention using the residual analysis described in Section 5.

4 Distance-Based Kernel Function

At the core of the semi-Lagrangian advection algorithm for a physical quantity field (such as velocity and density), the value at each node is updated with an evaluation of the quantity at each back-traced position using a set of known neighboring node values; this evaluation is often called "interpolation". To evaluate the quantity at an arbitrary position, a weighted sum is adopted in general; hence, a *distance-based kernel function*, $K(\mathbf{r})$, can be used for determining the weights for the neighboring values. To fulfill the aforementioned properties for the advection operator, the kernel function is desired to be defined with the following properties:

- **Normalization on gathering**: The kernel function satisfies the partition of unity property, i.e.,
$$\sum_i K(\mathbf{x} - \mathbf{x}_i) = 1, \quad \text{for any } \mathbf{x},$$
where $\mathbf{x}_i$ denotes the neighboring nodes contributing to the evaluation at the back-traced position $\mathbf{x}$.
- **Normalization on scattering**: The kernel function also reversely satisfies the partition of unity property, i.e.,
$$\sum_j K(\mathbf{x}_i - \mathbf{x}_j) = 1, \quad \text{for any } \mathbf{x}_i,$$
where $\mathbf{x}_j$ denotes the back-traced positions that receive the contributions of the node at $\mathbf{x}_i$ for evaluation. This normalization guarantees the entire transfer of the i-th node's quantity, thus the quantity conservation.
- **Autoconvolution approximating a delta**: The autoconvolution of the kernel function approximates a Kronecker delta function, δ, i.e.,
$$(K * K)(\mathbf{r}) \approx \delta(\mathbf{r}),$$
where $*$ denotes the convolution. This property implies that the kernel can potentially form the orthogonal basis of the advection operator $\mathbf{A}$, which is the structural reason for the bounded near-orthogonality of our reconstruction and the energy-retention benefits we observe in practice.

4.1 Cardinal Sine Kernel

To serve the desired properties, we first examine the sinc (also called *cardinal sine*) function in a continuous domain. We recognize the fundamental properties of the sinc function help construct a bounded, near-orthogonal semi-Lagrangian reconstruction whose discrete advection operator delivers energy-retention benefits.

Let us first consider a one-dimensional (1D) domain and define the (normalized) sinc function:
$$\mathrm{sinc}(x) = \frac{\sin(\pi x)}{\pi x}.$$
This function has several key properties:

- **Normalization**: The integral of sinc from $-\infty$ to ∞ is one, i.e.,
$$\int_{-\infty}^{\infty} \mathrm{sinc}(x)\,\mathrm{d}x = 1.$$
- **Shift-invariant normalization**: The integral is invariant to any non-zero shift τ, i.e.,
$$\int_{-\infty}^{\infty} \mathrm{sinc}(x-\tau)\,\mathrm{d}x = 1.$$
- **Orthogonality of shifted kernels**: Shifted sinc functions form an exact orthonormal family on integer shifts, i.e., for $j, k \in \mathbb{Z}$,
$$\int_{-\infty}^{\infty} \mathrm{sinc}(x-j)\ \mathrm{sinc}(x-k)\,\mathrm{d}x = \mathrm{sinc}(j-k) = \delta_{jk},$$
where δ_{jk} is the Kronecker delta, whereas for non-integer offsets τ the same integral equals $\mathrm{sinc}(\tau)$, which decays like $1/|\tau|$ and thus serves as the continuous extension of the Kronecker pattern.

For higher-dimensional domains whose dimension is d, the sinc-based weight function can be defined using a product form as follows:
$$\mathrm{sinc}_d(\mathbf{x}) = \prod_{k=1}^{d} \frac{\sin(\pi x_k)}{\pi x_k},$$
where x_k denotes each element of $\mathbf{x} \in \mathbb{R}^d$. Note that the 1D sinc kernel for each dimension independently contributes to the evaluation, which still holds the normalization property, i.e.,
$$\int_{\mathbb{R}^d} \mathrm{sinc}_d(\mathbf{x})\,\mathrm{d}\mathbf{x} = \prod_{k=1}^{d} \Big(\int_{-\infty}^{\infty} \mathrm{sinc}(x_k)\,\mathrm{d}x_k \Big) = 1.$$
The product structure transfers the integer-shift orthonormality to higher dimensions: for $\mathbf{j}, \mathbf{k} \in \mathbb{Z}^d$,
$$\int_{\mathbb{R}^d} \mathrm{sinc}_d(\mathbf{x}-\mathbf{j})\ \mathrm{sinc}_d(\mathbf{x}-\mathbf{k})\,\mathrm{d}\mathbf{x} = \prod_{\ell=1}^{d} \delta_{j_\ell k_\ell} = \delta_{\mathbf{jk}}.$$
Hence, on a unit-spaced grid, shifted copies of sinc_d form an orthonormal family, and any advection operator built from this family inherits exact column orthogonality. This is the precise statement that motivates our search for a compactly supported approximation of sinc_d.

Let us assume a measure-preserving diffeomorphism for the backward tracing operator $\boldsymbol{\Phi}$ of the semi-Lagrangian advection algorithm, i.e., $\mathbf{x}_b = \boldsymbol{\Phi}(\mathbf{x})$, where $\boldsymbol{\Phi}(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}^d$. Since the fluid

velocity is divergence-free, this mapping is volume-preserving; thus, $\det\left(\frac{\partial \Phi(\mathbf{x})}{\partial \mathbf{x}}\right) = 1$. Consequently,

$$d\mathbf{x} = \left|\det\left(\frac{\partial \Phi(\mathbf{x})}{\partial \mathbf{x}}\right)\right| d\mathbf{x}_b,$$

hence

$$\int_{\mathbb{R}^d} \mathrm{sinc}_d\big(\Phi(\mathbf{x})\big)\, d\mathbf{x} = \int_{\mathbb{R}^d} \mathrm{sinc}_d(\mathbf{x}_b)\, d\mathbf{x}_b = 1. \tag{3}$$

Note that Eq. (3) implies that all the weights for back-traced positions (i.e., $\mathbf{x}_b$) are integrated into one, indicating the normalization on scattering property.

These properties make sinc an ideal kernel for our objective: if each column of the discrete advection operator behaves as a shifted sinc basis under a smooth, volume-preserving backtrace, then *column interactions* (i.e., correlations among columns) are minimized, thus the operator becomes close to orthogonal, improving kinetic-energy retention. This perspective also clarifies why simple low-order kernels (e.g., the triangular kernel behind linear interpolation) remain noticeably more dissipative on unit-spaced grids: their self-overlap is substantially larger, leading to stronger correlations among columns.

4.2 Lanczos Kernel

The preceding discussion motivates the sinc kernel as the ideal cardinal interpolation kernel: shifted sinc functions provide exact reconstruction for band-limited data and form the classical basis of the Shannon–Whittaker sampling formula [Shannon 1949; Unser 2000]. However, the ideal sinc kernel is not directly suitable for a grid-based semi-Lagrangian stencil because it has infinite support and decays slowly. In practice, a finite approximation is required.

We therefore use the Lanczos kernel, a compactly supported windowed-sinc kernel:

$$L_a(x) = \begin{cases} \mathrm{sinc}(x)\ \mathrm{sinc}\left(\frac{x}{a}\right), & |x| < a, \\ 0, & \text{otherwise}, \end{cases} \tag{4}$$

where a is the Lanczos window, also known as a sinc window or Lanczos sigma factor. Lanczos sigma factors are commonly used to reduce Gibbs-type oscillations caused by abrupt spectral or spatial truncation [Duchon 1979; Lanczos 1988].

For our advection operator, the relevant theoretical connection to sinc is minimal. First, L_a remains cardinal on the integer grid:

$$L_a(m) = \begin{cases} 1, & m = 0, \\ 0, & m \in \mathbb{Z},\ m \neq 0, \end{cases} \tag{5}$$

so grid-aligned samples are reproduced exactly. Second, for any fixed x,

$$\lim_{a\to\infty} L_a(x) = \mathrm{sinc}(x), \tag{6}$$

thus, increasing the support radius moves the compact kernel toward the ideal sinc kernel discussed in Sec. 4.1. Accordingly, we do not claim that a finite-radius Lanczos operator is exactly orthogonal. Rather, Lanczos provides a practical compact approximation to the ideal sinc construction while reducing the ringing associated with naive truncation. This practical property motivates us to use the Lanczos kernel for building a discrete advection operator to improve its orthogonality

Fig. 2 compares the sinc and Lanczos kernels as well as their numerical approximations with the 2^{14} samples in the frequency domain of $[-50, 50]$. The left column shows both kernels in the spatial domain. The right column presents them in the frequency domain after the Fourier transform. The analytic function of the sinc kernel in the frequency domain is a rectangular function (shown in red at the top). Its numerical approximation (shown in blue at the top), however, exhibits oscillations as a result of a typical truncation of the infinite support. These oscillations can adversely affect the stability and accuracy of simulations, causing undesired over- and under-shootings. The Lanczos kernel, on the other hand, presents a significantly better numerical approximation (shown in green at the bottom) for its analytic function (shown in red at the bottom) in the frequency domain, demonstrating its superiority in avoiding such oscillations. Moreover, we underline that the finite support of the Lanczos kernel is computationally more attractive for numerical implementation. Therefore, we realize our advection operator using the Lanczos kernel.

4.3 Advection Operator Construction

We adopt the aforementioned Lanczos kernel of Eq. (4) as our weight function for the advection operator and use a dimension-splitting evaluation. Let $\mathbf{x}_i$ denote the node location for a field value to be advected in a d-dimensional space and $\mathbf{x}_{b\leftarrow i}$ denote the back-traced location from $\mathbf{x}_i$. The weight for a node at $\mathbf{x}_j$ contributing to $\mathbf{x}_{b\leftarrow i}$ can be evaluated as follows:

$$W(\mathbf{x}_{b\leftarrow i}, \mathbf{x}_j) = \prod_{k=1}^{d} L_a(x_{b\leftarrow i,k} - x_{j,k}), \tag{7}$$

where $x_{b\leftarrow i,k}$ and $x_{j,k}$ denote the k-th coordinate of $\mathbf{x}_{b\leftarrow i}$ and $\mathbf{x}_j$, respectively. We note that our weight function does not hold the partition of unity property itself; thus, a typical normalization applies to each row of our operator, i.e.,

$$A_{ij} = \frac{W(\mathbf{x}_{b\leftarrow i}, \mathbf{x}_j)}{\sum_j W(\mathbf{x}_{b\leftarrow i}, \mathbf{x}_j)}. \tag{8}$$

5 Orthogonality of Advection Operators

We investigate the orthogonality of our advection operator and other conventional ones. The orthogonality is measured using the deviation of $\mathbf{A}^\top\mathbf{A}$ from the identity matrix $\mathbf{I}$ as follows:

$$\sigma(\mathbf{A}) = \frac{1}{n}\left\|\mathbf{A}^\top\mathbf{A} - \mathbf{I}\right\|_{\mathrm{F}}, \tag{9}$$

where n is the total degrees of freedom (i.e., the number of discrete values for the advected field) and $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm. The factor $1/n$ is chosen so that $\sigma(\mathbf{A})$ remains comparable across grid resolutions, since for a row-stochastic operator with bounded local support the Frobenius norm of $\mathbf{A}^\top\mathbf{A} - \mathbf{I}$ grows at most linearly with n. A smaller $\sigma(\mathbf{A})$ indicates that the operator $\mathbf{A}$ is closer to orthogonal, thus potentially exhibiting better kinetic-energy retention behavior; this is the regime in which our experiments exhibit better kinetic-energy retention rather than a guaranteed preservation property.

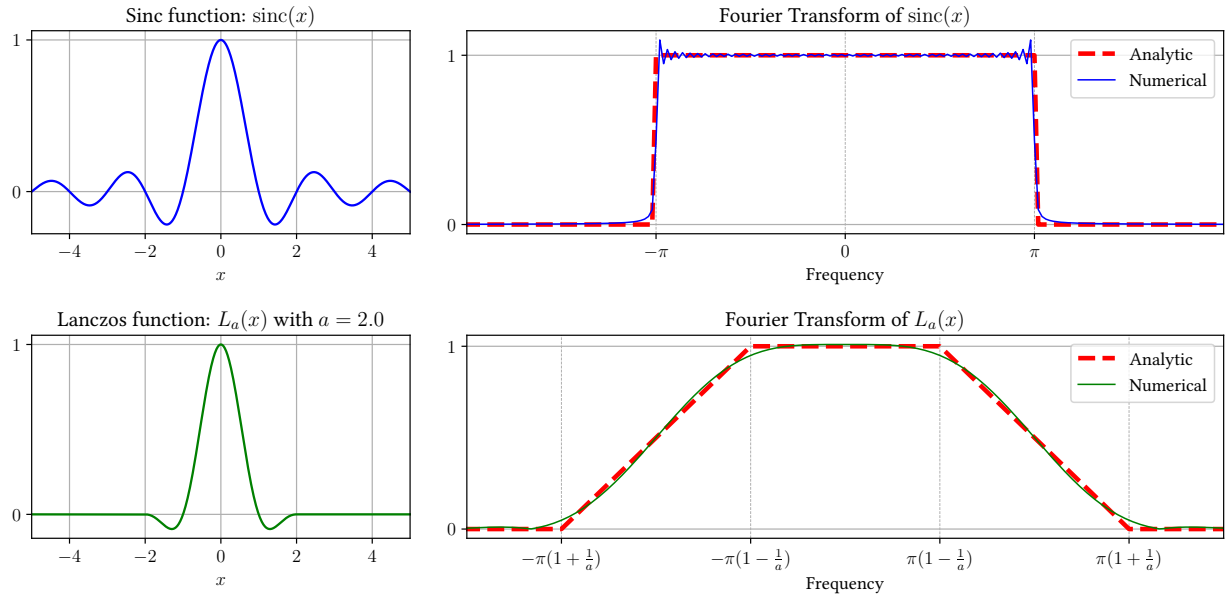


Fig. 2. Comparison of the sinc and Lanczos kernels. **Left**: The sinc (top) and Lanczos (bottom) kernels are shown in the spatial domain. **Right**: They are shown in the frequency domain after the Fourier transform. The numerical Fourier transform of the sinc kernel exhibits oscillations due to a typical truncation of its infinite support, while the Lanczos kernel shows a better approximation with reduced oscillations, highlighting its effectiveness in minimizing numerical artifacts. The comparison here isolates the numerical effect of windowing both functions, rather than presenting the sinc's inherent property.

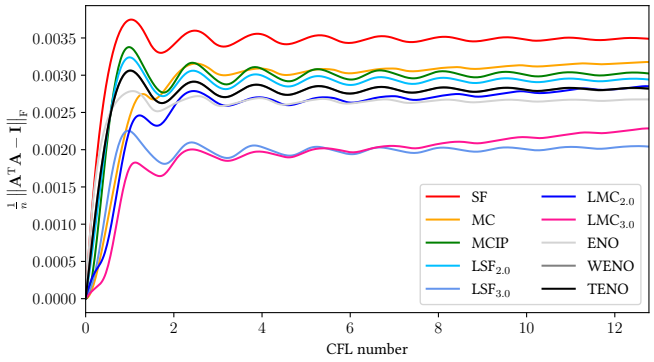


Fig. 3. Orthogonality deviation of different advection operators.

As an example for the investigation, we consider a rotational flow whose domain is $[-0.5, 0.5]^2$ initialized as follows:

$$(u, v) = (-y, x)\,. \tag{10}$$

The velocity field is stored on a staggered marker-and-cell (MAC) grid with the regularly spaced cells of 256^2. This velocity field is updated for a single time step using an advection operator $\mathbf{A}$ as in Eq. (2). For comparison, we examine different advection operators obtained by using exact solutions of direct backward trace with varying time step size, Δt.

Fig. 3 compares the different advection operators: semi-Lagrangian (SF) [Stam 1999], MacCormack (MC) [Selle et al. 2008], monotonic CIP (MCIP) [Song et al. 2005], ENO [Harten et al. 1987], WENO [Jiang and Shu 1996] and TENO [Fu et al. 2016] and our extension applied to two base methods (i.e., SF and MC). In our notation for LSF and LMC, "L" indicates our extension with the Lanczos kernel, and the subscript value denotes the support radius, a, of Eq. (4). For the different advection operators, the graph plots deviation from the orthogonality $\sigma(\mathbf{A})$ of Eq. (9) with respect to the time step size Δt.

As shown in Fig. 3, our extension for both SF and MC significantly improves the orthogonality of the operator. $\text{LSF}_{2.0}$ presents better orthogonality than the higher-order advection scheme, i.e., MCIP. We note that $\text{LSF}_{3.0}$ and $\text{LMC}_{3.0}$ present better orthogonality than $\text{LSF}_{2.0}$ and $\text{LMC}_{2.0}$. However, the Lanczos kernel with a large support radius converges to sinc, potentially leading to numerical instability as shown in Fig. 2 and requiring more computations. Thus, we use the support radius of 2.0 for both numerical stability and efficiency. We will, henceforth, omit the subscript for the support radius, unless we distinguish our experiments with different values.

6 Implementation

We apply our extension to the core algorithm of the semi-Lagrangian scheme, which consists of two main steps: (1) backward tracing and (2) evaluation of a weighted sum. Instead of the low-order (e.g., bi- or tri-linear) interpolation used in the conventional implementation, our method employs the Lanczos-based weight function and hence improves the evaluation of the weighted sum. The extended advection scheme achieves a substantial improvement in energy preservation without an enormous modification of the conventional solver.

Our extension can improve the advection algorithm mainly in two operations. First, in the spatial discretization of a MAC grid [Harlow and Welch 1965], the backward tracing velocity at each node (where each physical quantity is stored, e.g., the cell center for a marker density or the face center for a velocity component) can be evaluated using our method. In the conventional algorithm, the backtracing velocity at each node is computed using a typical linear approximation. One may replace this linear approximation with our Lanczos kernel. However, we note that it does not result in significant differences as shown in Fig. 15, hence we use the linear approximation in our implementation. Second, after backtracing, our method can be applied to the evaluation of the physical quantity. At each back-traced location $\mathbf{x}_{b\leftarrow i}$, the advected quantity is evaluated using our Lanczos kernel. We find that the latter primarily contributes to the improved quality.

As the Lanczos kernel gives zero for every integer input except zero, we can assume a unit grid spacing (i.e., $\Delta x = \Delta y = \Delta z = 1$ in 3D) for the weight evaluation of each node. The support radius, a, of the kernel then easily decides the number of nodes that contribute to the weighted sum. This radius is typically set to two or three in various applications [Turkowski 1990]. Our implementation uses the radius of 2.0 as discussed in Sec. 5. Consequently, for a weighted sum in the d dimension, the 1D Lanczos kernel is evaluated $4d$ times in our dimension-splitting calculation when $a = 2.0$. We note that, when the contributing nodes refer to the boundary, we simply use a zero physical quantity instead of adjusting their weights.

Interpolation limiter: Despite the advantage of the Lanczos kernel for constructing an orthogonal advection operator, its advantage comes with a trade-off between the reduction in aliasing artifacts and the preservation of sharp features. Because the Lanczos kernel can assign negative weights, the range of interpolated values can be wider than the range of known sample values. In practice, this may cause an overshoot or an undershoot, leading to visually or physically undesirable results (e.g., ringing artifacts and negative densities) and affecting the simulation stability.

To mitigate these issues, we propose a *soft-bound* limiter that validates the Lanczos sample against a trusted trilinear source evaluated at the same back-traced location, presenting a good compromise between simulation stability and visual quality. Let $q_{\text{lan}} = \sum_j A_{ij}\, q_j$ denote the normalized Lanczos sum from Eq. (8), and let q_{lin} denote the standard trilinear interpolation evaluated from the eight stencils $\{q_{c_0}, \ldots, q_{c_7}\}$. From those same stencils, we extract the local source bounds

$$q_{\min} = \min_m q_{c_m}, \qquad q_{\max} = \max_m q_{c_m}, \tag{11}$$

optionally limited to any physical bounds (e.g., non-negativity for densities). The limiter then proceeds in two stages. If the Lanczos candidate falls outside the source bounds, i.e., $q_{\text{lan}} \notin [q_{\min}, q_{\max}]$, we take the trilinear source:

$$q_i \leftarrow q_{\text{lin}}. \tag{12}$$

Otherwise, we accept the Lanczos correction $\Delta q = q_{\text{lan}} - q_{\text{lin}}$ but cap its magnitude to a small fraction λ of the local span:

$$q_i \leftarrow q_{\text{lin}} + \alpha\, \Delta q, \qquad \alpha = \min\left(1,\ \frac{\lambda\,(q_{\max} - q_{\min})}{|\Delta q|}\right), \tag{13}$$

Algorithm 1 Extended Advection

```
procedure ADVECT(q)
    for each node value q_i at x_i on grid do
        x_{b←i} ← backtrace from x_i
        q_lin ← trilinear interpolation at x_{b←i}
        [q_min, q_max] ← source bounds from the 8 surrounding corners
        q_lan ← Σ_j q_j W(x_{b←i}, x_j) / Σ_j W(x_{b←i}, x_j)   ▷ Eq. (7), Eq. (8)
        if q_lan ∉ [q_min, q_max] then
            q_i ← q_lin                                          ▷ hard revert
        else
            Δq ← q_lan − q_lin
            α ← min(1, λ(q_max − q_min)/|Δq|)
            q_i ← q_lin + α Δq                                   ▷ soft bound, Eq. (13)
        end if
    end for
end procedure
```

where we set $\lambda = 0.02$ for velocity samples, and let $\alpha = 1$ (i.e., direct acceptance of q_{lan}) for scalar quantities as the source envelope bounds the values sufficiently. This soft-bound design preserves the orthogonality benefit of the Lanczos kernel inside the source envelope while preventing the rare overshoots that would otherwise destabilize the simulation. **Algorithm 1** summarizes our extended advection algorithm.

Boundedness of the limiter: Although nonlinear, the soft-bound limiter admits a clean local maximum-principle interpretation. Since the trilinear source is a convex combination of the eight surrounding values, $q_{\text{lin}} \in [q_{\min}, q_{\max}]$. If the Lanczos candidate falls outside this interval, the limiter reverts to q_{lin}, which lies in $[q_{\min}, q_{\max}]$ by construction. Otherwise the output $(1-\alpha)\, q_{\text{lin}} + \alpha\, q_{\text{lan}}$ with $\alpha \in [0, 1]$ is again a convex combination of two values inside $[q_{\min}, q_{\max}]$. Therefore, every reconstructed value satisfies $q_i^{n+1} \in [q_{\min}, q_{\max}]$, the interpolation step is L^∞-nonexpansive with respect to the input grid, and the accepted Lanczos correction obeys $|q_i^{n+1} - q_{\text{lin}}| \le \lambda(q_{\max} - q_{\min})$, so λ acts as a trust-region radius measured against the local source variation. Setting $\lambda = 1$ recovers $\alpha = 1$ for the scalar case, since $q_{\text{lin}}, q_{\text{lan}} \in [q_{\min}, q_{\max}]$ implies $|\Delta q| \le q_{\max} - q_{\min}$, while $\lambda = 0.02$ caps the velocity correction at 2% of the local source span. To ensure numerical robustness, our implementation sets $q_i^{n+1} \leftarrow q_{\text{lin}}$ whenever $|\Delta q| \le \varepsilon^2$, which is consistent with the same boundedness property. We emphasize that this is an L^∞ guarantee for the reconstruction step alone and not a global stability theorem for the coupled advection-projection solver.

7 Experimental Results

The experiments demonstrate that our extended advection method produces vivid fluid flows, exhibiting energy-retention benefits attributable to the bounded near-orthogonal structure of the reconstruction, which helps develop richer vortices and complex flow details. Fig. 1 showcases the evolution of great visual details of complex turbulent flows produced by our method compared to its baseline methods.

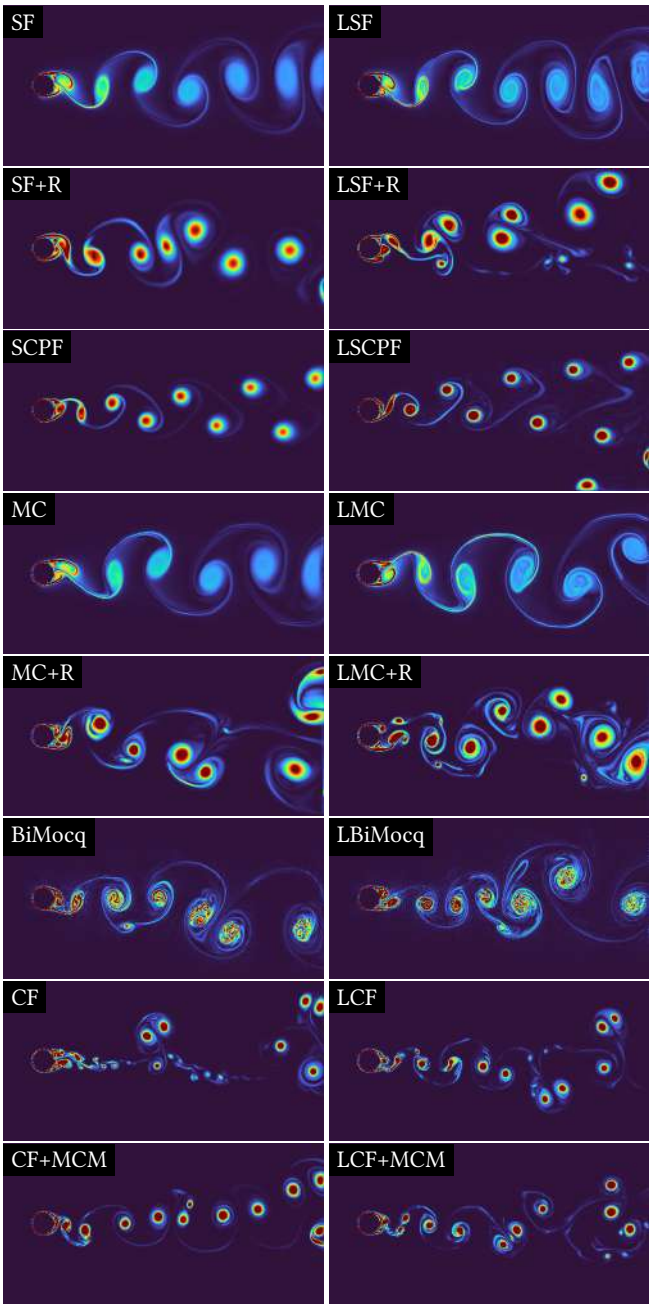


Fig. 4. 2D Kármán vortex street. Baseline methods (left column) are compared with our Lanczos-enhanced versions (right column). Ours retain sharper shear layers, leading to a visibly distinct and more coherent vortex street downstream of the circle.

We examine our method through various examples in both two and three dimensions. The enhancement made by our method is highlighted in comparison with widely used methods such as semi-Lagrangian (SF) [Stam 1999], circulation-preserving (SCPF) [Elcott et al. 2007], MacCormack (MC) [Selle et al. 2008], monotonic CIP (MCIP) [Song et al. 2005], ENO [Harten et al. 1987], WENO [Jiang and Shu 1996] and TENO [Fu et al. 2016]. Moreover, our extension is experimented with advanced methods, such as the reflection solver (R) [Zehnder et al. 2018], bidirectional mapping (BiMocq) [Qu et al. 2019], covector fluids (CF) [Nabizadeh et al. 2022], and the method of characteristic map (MCM) [Sato et al. 2018], which are all known to be effective for producing great fluid details.

For the numerical NSEs solver, we employ a staggered MAC grid and adopt the standard operator splitting scheme [Chorin 1967] with a projection solver. In the 3D examples, we use a fast multigrid projection method [Sun et al. 2025], while a multigrid solver and a direct solver with Cholesky decomposition are used in the 2D examples. We note that all simulations are implemented in Taichi [Hu et al. 2019] and the open source provided with the work of CF [Nabizadeh et al. 2022]. Performance is measured on a workstation with an AMD Ryzen 5 7600 6-Core CPU of 3.80 GHz and an NVIDIA GeForce RTX 4080 SUPER GPU. For all comparisons, the marker density field is advected by the same scheme as the velocity field of the corresponding solver, so that any difference in visual detail can be attributed to the advection operator itself. No vorticity confinement is applied to any baseline or to our variants, and viscosity is set to zero unless otherwise stated. The projection solver and boundary treatment are kept identical across all methods within each scene.

Fig. 4 presents the results of 2D Kármán vortex street simulations in which the vorticity is visualized. We impose a constant inflow velocity at the left boundary, and the flow strikes a circular obstacle, generating a steady pattern of vortex shedding downstream. Compared to each baseline method shown on the left, our extension shown on the right improves all the baseline methods, presenting much sharper and richer vortical structures, thanks to the energy-retention benefits of our bounded near-orthogonal reconstruction and the reduced numerical dissipation that follows from it. As our extension works on top of a baseline method, it is worth noting that our extension does not improve the noisy flow presented in the baseline method, as shown in the results of BiMocq and LBiMocq.

Fig. 6 shows a vortex-dominant leapfrogging flow example. Each baseline method is notably improved with our extension, retaining sharp eddies and vivid vortices over time. Compared to the baseline methods, our extension significantly reduces smearing of fine-scale structures and demonstrates its superiority in preserving details and vorticity throughout the run.

In order to further investigate the energy-retention behaviour of different methods, we set up a 2D decaying turbulence scenario, starting from a random initial velocity field. In this example, we compare our variants with SF, MC, MCIP, ENO, WENO, and TENO. Fig. 8 shows that LMC presents stronger vortices and visually sharper structures of flows, showing the advantages of our bounded, near-orthogonal semi-Lagrangian reconstruction and the energy-retention benefits that follow from it. As shown in the graph of Fig. 9, all methods gradually lose the initial kinetic energy over time as a consequence of numerical dissipation. However, LMC better retains the initial kinetic energy over time.

In addition to the velocity flow experiments, we set up a density diffusion example to examine our approach in a visual context. This example sets up the initial marker density field of the SIGGRAPH

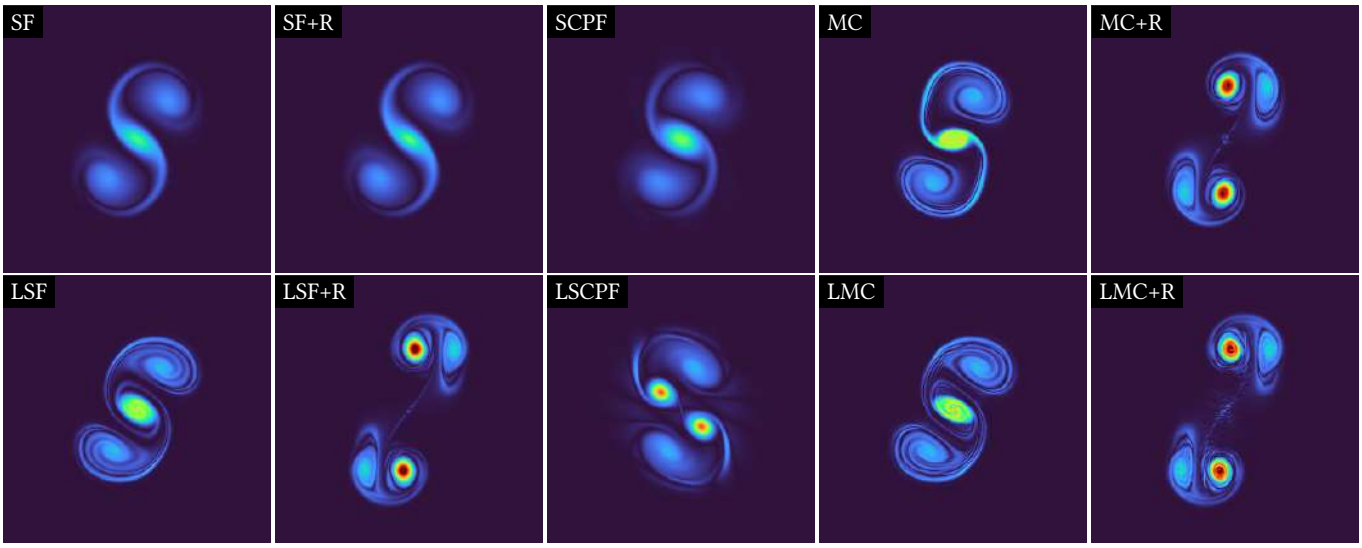


Fig. 5. 2D Taylor vortex. The frames at $t = 150\Delta t$ are visualized. Our Lanczos extension exhibits crisper eddies and stronger vorticity than the baseline solvers.

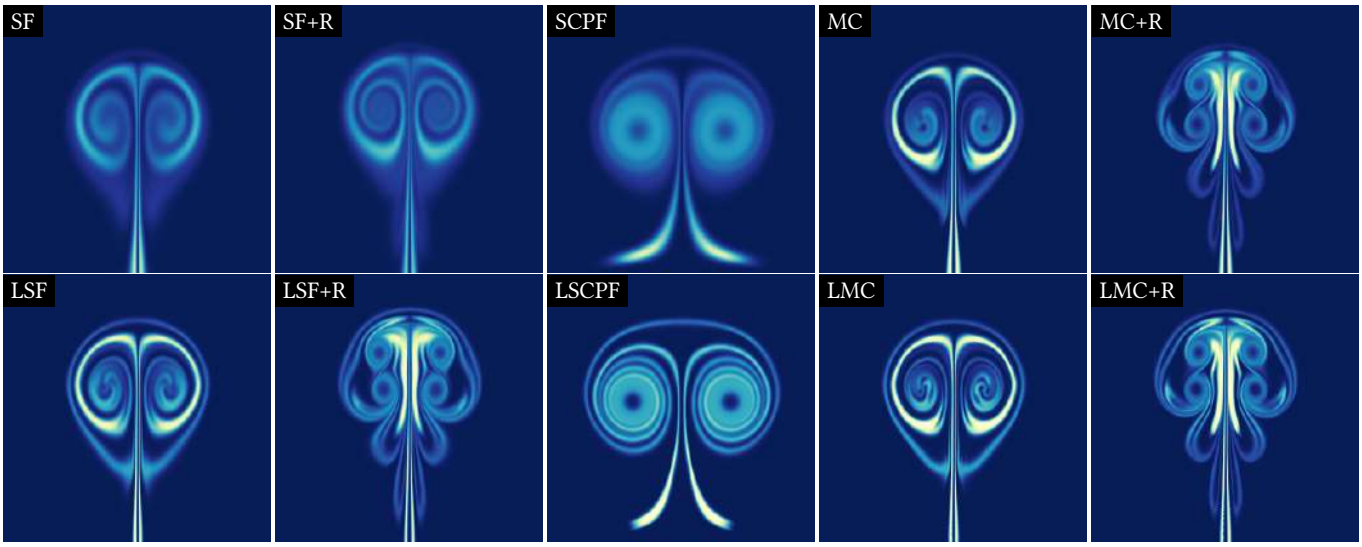


Fig. 6. 2D Leap-frogging vortex pair. While the baseline methods diffuse the vortices after a few crossings, our Lanczos versions improve all the methods, keeping them longer and sharper throughout the interaction.

logo shape as shown in the top left of Fig. 7. The density field is, then, advected by an incompressible flow driven by the downward buoyancy force exerted by the marker density field itself. As shown in Fig. 7, our extension helps all baseline methods produce far more intricate details and crisper interfaces, yielding significantly sharper and visually more faithful results.

As a coupled flow of density and velocity fields, we set up a 2D smoke plume example. In this example, a source of marker density is placed at the bottom of the domain. The velocity field is, then, driven by the buoyancy force that is exerted by the marker density field. Fig. 10 compares the results among SF, MC, LSF, LMC, MCIP, ENO, WENO, and TENO. LSF and LMC produce richer and finer vortices as the smoke plume evolves and retain crisper structures with a longer lifetime of vortices.

To further investigate challenging time stepping, we compare different methods in a Taylor vortex separation case using a "larger CFL number," i.e., 2. Fig. 11 shows the results. Under this setting, most of the methods (i.e., SF, ENO, WENO, and TENO) suffer from substantial numerical diffusion, which smears the vortical structures and diminishes the apparent separation. In contrast, our LSF and MCIP maintain a clean, well-separated vortex pair, consistent with the expected physical behavior, indicating that these two schemes

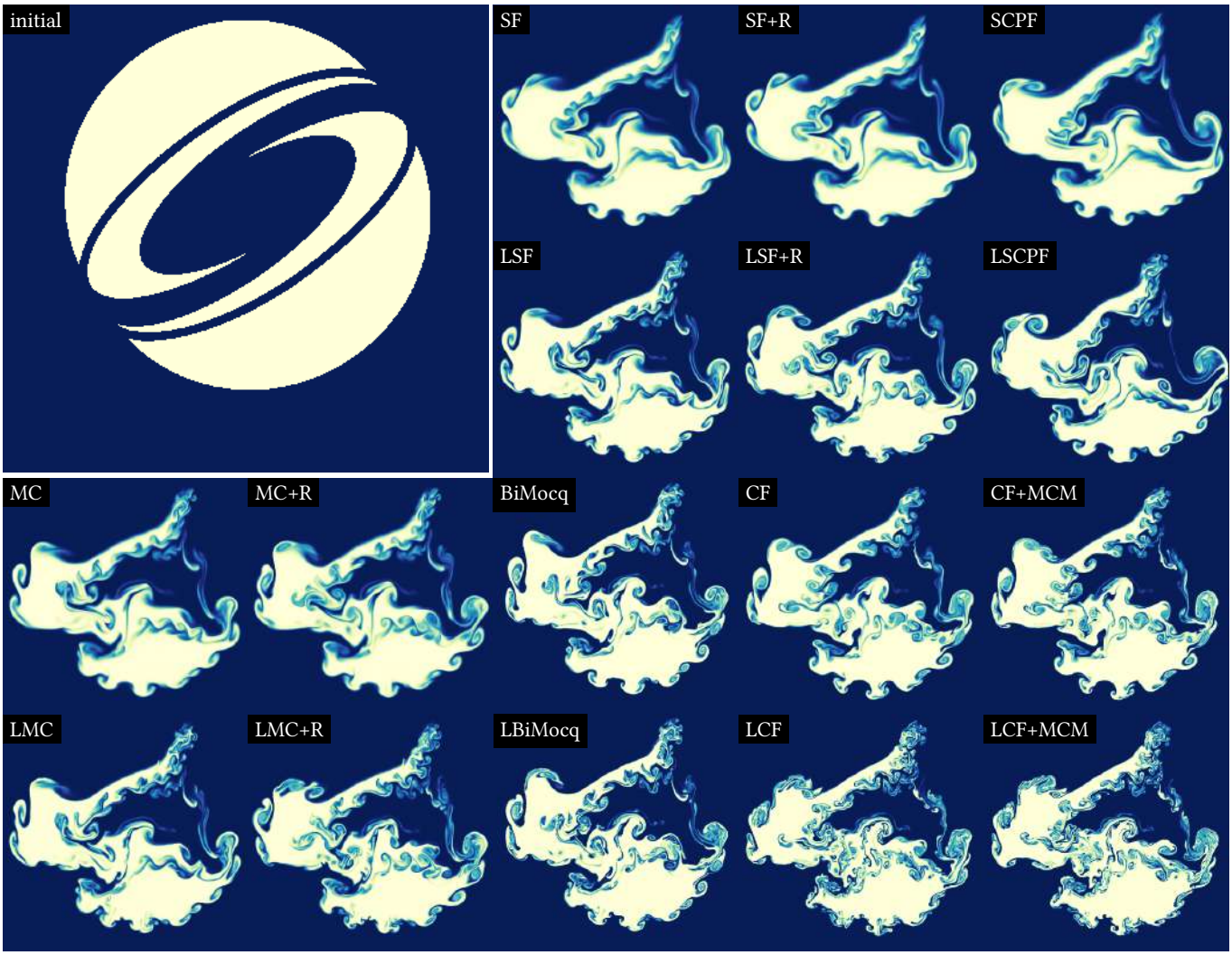


Fig. 7. SIGGRAPH Logo. A marker-density field initialized with the shape of the logo is transported by a density-driven flow. Our Lanczos operator maintains much sharper edges and vortical details.

better preserve coherent structures and resist excessive dissipation even when larger time step sizes are used.

In Fig. 12, we compare our method with the mass–momentum conserving (MMC) method [Lentine et al. 2011] in a simple density evolution example. While SF exhibits diminished fine-scale structure and MMC improves it to some extent, our LSF extension retains significantly improved small-scale vortical features. MMC enforces mass and momentum conservation as a global correction, which does not directly mitigate the dissipation inside the interpolation stencil itself, and the smearing of small vortices therefore remains largely unresolved. In contrast, our reconstruction yields sharper interior structures using a local interpolation step rather than an explicit global correction.

To see the effect of different support radii for the Lanczos kernel, we examine our extension with a single-vortex flow example. In this example, we compare LMC with different support radii ranging from 2.0 to 16.0 without our interpolation limiter. The results in Fig. 13 show that, as the support radius increases, the Lanczos filter approaches the sinc filter, which is conjecturally best for orthogonality. As a result, a larger radius filter develops finer details, and the kinetic energy decreases less, as shown in the graph. However, we notice that larger radii introduce stronger numerical dispersions.

Fig. 14 shows a 2D vortex lattice test, where we initialize a regular arrangement of coherent vortices and visualize the evolution of the vortices. This example is particularly sensitive to numerical dissipation: even mild smearing of the vortex cores can artificially promote interaction and lead to premature merging, breaking the intended separation and lattice-like organization. As shown in Fig. 14, the original LFM [Sun et al. 2025] and NFM [Deng et al. 2023] exhibit noticeable loss of separation as nearby vortices begin to merge,

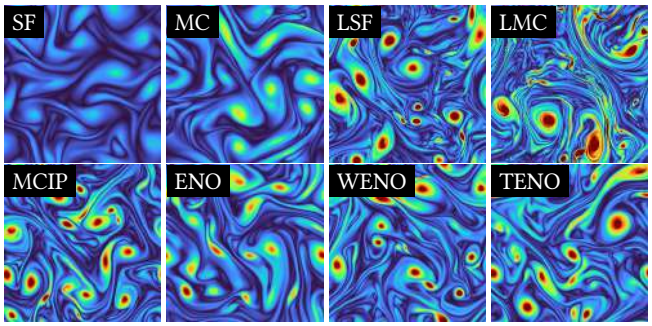


Fig. 8. 2D Decaying turbulence. The frames at $t = 1500\Delta t$ are visualized. The vortices stay longer with our Lanczos variants (LSF and LMC), while quickly decaying in the other methods, such as SF, MC, MCIP, ENO, WENO and TENO.

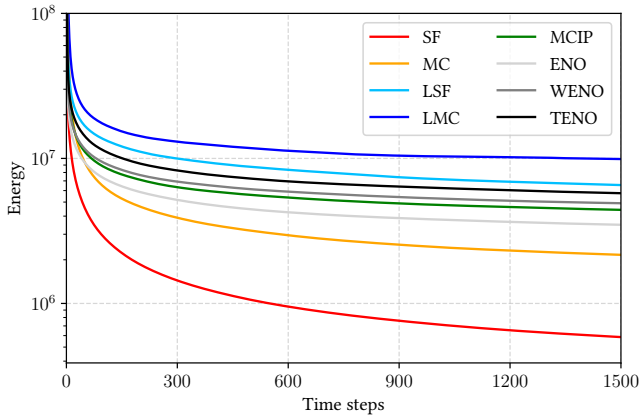


Fig. 9. The energy evolution for the 2D decaying turbulence simulations is shown. LMC presents the least energy decay from the initial kinetic energy, thanks to the operator's improved orthogonality.

whereas our Lanczos variants (LLFM and LNFM) maintain distinctly separated vortex cores and preserve the overall lattice structure much longer. This highlights the benefit of our extension in reducing diffusion while keeping the long-term coherent dynamics intact.

Fig. 15 examines a potential gain when our Lanczos filter is also applied to evaluate the backtrace velocity at each location $\mathbf{x}_i$. To this end, we set up a 2D Taylor–Green vortex example and perform the simulation using LMC and the fourth-order Runge-Kutta (RK4) integrator. As shown in the graph, the Lanczos version presents a marginally better energy decay curve over the traditional linear approximation. However, we find that the two variants produce visually indistinguishable results; thus, the improvement of the Lanczos filter applied to the backtrace velocity evaluation is negligible, as outlined in Section 6.

We evaluate the performance of our method in 3D examples involving knotted vortex tubes whose tightly self-linked filaments are highly sensitive to numerical diffusion. We initialize two torus knots: a trefoil knot with three crossings and a cinquefoil knot with

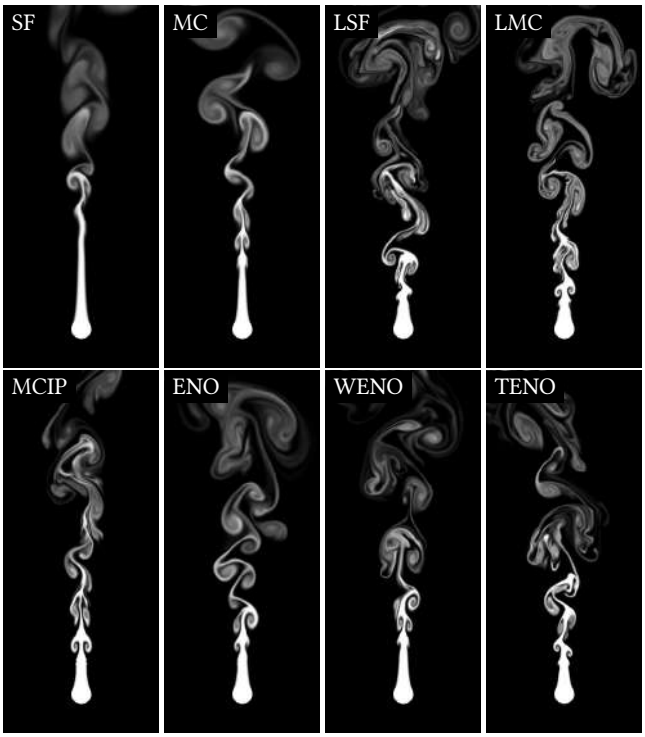


Fig. 10. 2D smoke plume. The marker density field at $t = 1500\Delta t$ is visualized. LSF and LMC preserve long-lived, fine-detail billows of smoke. WENO and TENO also preserve detail slightly longer than SF, MC, MCIP and ENO, but remain more diffusive than LSF and LMC.

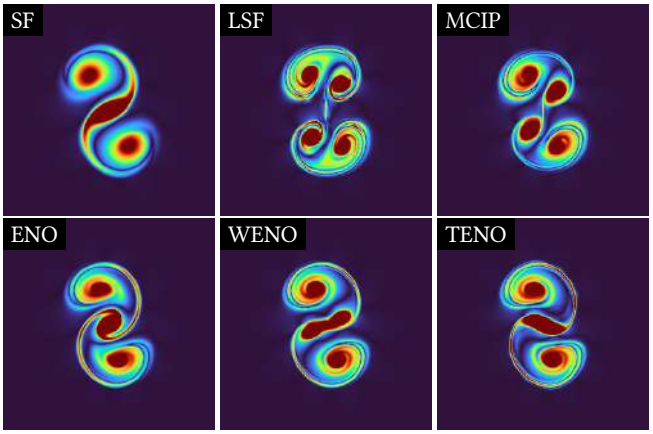


Fig. 11. Taylor vortex separation with a large CFL number. The marker density field at $t = 50\Delta t$ is visualized. Among the tested schemes, only LSF and MCIP preserve a clear, well-separated vortex pair, which matches the expected physical behavior; SF, ENO, WENO, and TENO exhibit noticeably higher numerical diffusion that smears the structures and reduces the separation.

five crossings. Figs. 16 and 17 compare the final-frame results of all sixteen variants in a 4×4 grid where the first and third rows show

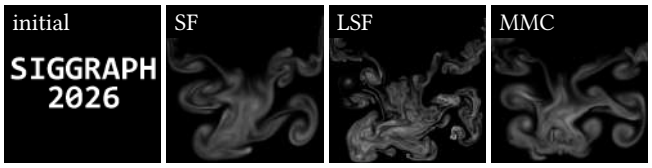


Fig. 12. SIGGRAPH label. We simulate SF, LSF and the mass-momentum conserving (MMC) method [Lentine et al. 2011]. SF and MMC exhibit weak flow detail. Our LSF preserves fine-scale vortical structures.

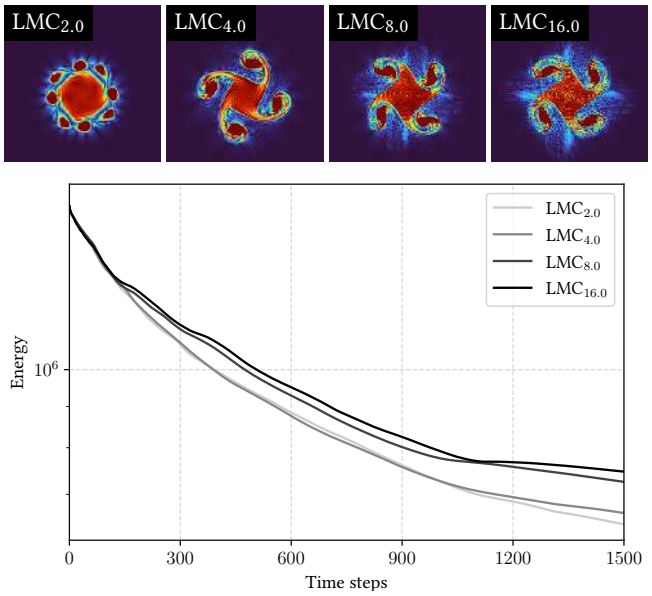


Fig. 13. Single vortex of LMC. Different Lanczos radii are examined without the limiter. Increasing the kernel radius from 2.0 to 16.0 improves energy retention and presents finer flow structure, while suffering from stronger dispersion artifacts.

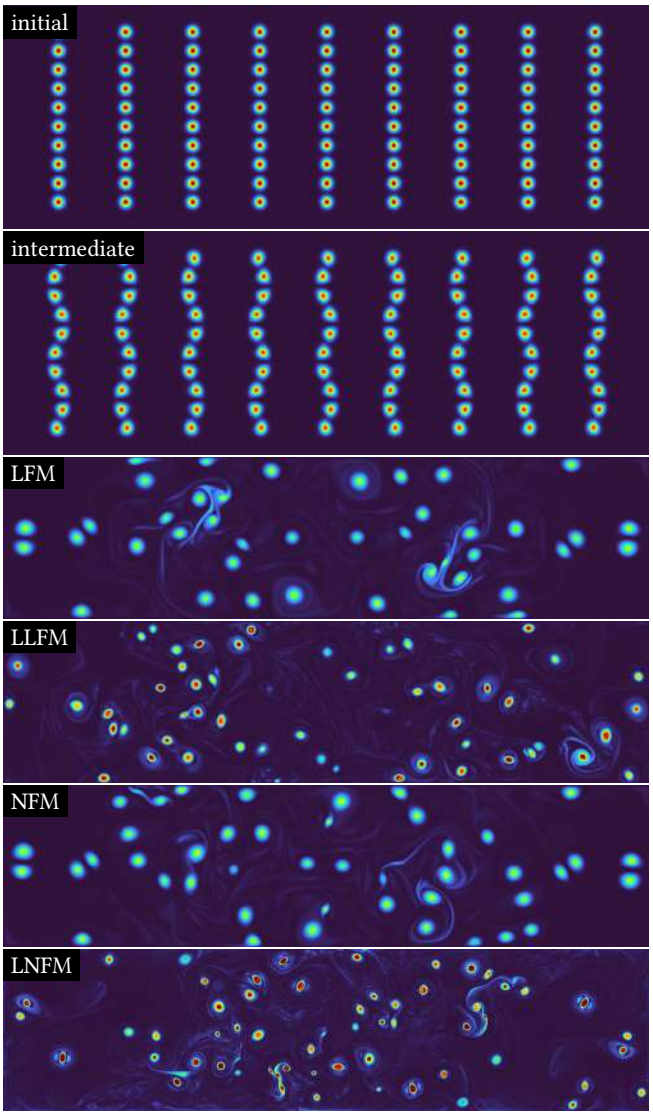


Fig. 14. 2D Vortex Lattice scene. The vorticity field at $t = 1000\,\Delta t$ is visualized. LFM and NFM with Lanczos (LLFM and LNFM) maintain vortex separation well, preventing premature merging compared to the original LFM/NFM.

eight different baselines, and the second and fourth rows show each of their Lanczos counterpart. In both scenes, our Lanczos counterparts retain crisper tube boundaries. The cinquefoil case, with its tighter self-linking, is more sensitive to numerical diffusion and accentuates the gap between baselines and our extension, confirming that vorticity preservation of our extension carries over to 3D knotted flows.

Fig. 18 presents further experiments of a 3D leapfrogging scene, in which two coaxial vortex rings repeatedly pass through each other. This mutual slip-through is highly sensitive to numerical diffusion, since even mild smearing of the vortex cores disrupts the delicate momentum exchange that sustains the repeated crossings. We compare all sixteen variants, arranging eight baselines in the odd rows and their Lanczos counterparts in the even rows. Among all the tested baselines, our Lanczos extension induces noticeable differences only with SF, MC, and their variants, yielding substantial improvement in secondary wake structure throughout the interaction. The other baselines, such as CF and its variants, whose advection routines are already known to be non-dissipative, present marginal visible change.

Fig. 19 shows the noticeable improvements of various baseline solvers with our extension in a 3D smoke plume example. As shown in each column, our extension consistently yields more vivid and finer details in smoke structures by effectively curbing numerical dissipation. Thanks to the improved advection operators, smoke plumes evolve with significantly richer vorticity fields, exhibiting energetic motion over time, compared to each baseline method.

We further examine our extension with a large time-step size of 40 ms in the 3D smoke plume example. Fig. 20 shows results from a set of selective methods, including LSF, LSF+R, LMC, LMC+R,

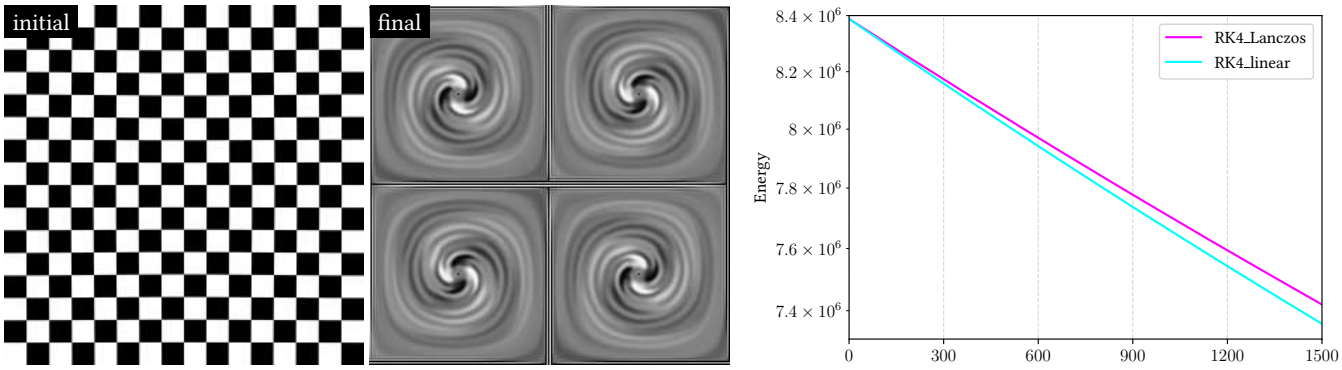


Fig. 15. Taylor–Green vortex. An additional application of our Lanczos interpolation (under LMC) to the back-tracing velocity retrieval is examined. Using the additional Lanczos interpolation with the RK4 integrator yields only a negligible improvement in energy over the conventional linear interpolation; both present nearly identical visual results.

and LBiMocq, from which we can observe a good balance between runtime cost and visual quality. LSF presents the fastest runtime performance with appreciable small-scale detail. LSF+R, LMC and LMC+R present slightly more vorticity at a marginally higher cost. LBiMocq retains the greatest energy overall, albeit at the greatest per-step expense. These five methods will be a practical set for different applications, depending on the efficiency and visual quality required.

Table 1 summarizes the simulation parameters of our experiments, presenting the time-step sizes and grid resolutions. In particular, Table 2 shows that the elapsed time of our LSF remains close to its corresponding baselines (ENO, WENO and TENO), with a consistently small gap across all tested scenes. This indicates that the proposed extension preserves the practical efficiency of the underlying schemes while providing the improved behavior reported in the experiments.

8 Discussion and Limitations

Similar to many advanced interpolation methods adopted in different advection schemes, our interpolation limiter is crucial for stable simulation. We observed that, without the limiter, several methods (e.g., LMC and LMC+R) suffered from overshoot as shown in Fig. 21. Beyond this evidence, our limiter satisfies a provable local boundedness property in which every reconstructed value remains inside the trilinear source interval while the accepted Lanczos correction is bounded by $\lambda(q_{\max} - q_{\min})$. This yields a CFL-independent local maximum-principle-type guarantee and an L^∞-nonexpansive interpolation step. The guarantee applies to the sampling stage in isolation while a global energy- or stability-theorem for the full coupled solver, together with stability criteria that avoid both under- and over-stabilization, remains future work because the projection and boundary treatment depend nonlinearly on the limited samples.

As described in Section 4.3, our implementation for the advection operator performs row normalization only, omitting column normalization. This implementation decision comes from our cost-benefit analysis on both normalizations. Row normalization incurs computationally negligible cost while leaving the near-orthogonal structure intact. In our extended analysis on the experiment of Fig. 3, relative to the unmodified Lanczos operator ($\mathrm{LSF}_{2.0}$), it increases the deviation in orthogonality $\sigma(\mathbf{A})$ of Eq. (9) by only 2.69×10^{-7}%. The additional column normalization even slightly increases the deviation by 0.019%. When row and column sums are enforced jointly via Sinkhorn iterations, the deviation is increased by 0.042%. Moreover, we note that Sinkhorn balancing incurs the computational overhead of an iterative procedure at every step. Most importantly, enforcing both row and column normalizations produced no discernible improvement in visual quality over row-only normalization. Therefore, we adopt row normalization alone. Nonetheless, enforcing exact column sums may become more relevant in momentum-sensitive settings, which we leave as future work.

9 Conclusion

This paper introduced a novel extension realized as a bounded, near-orthogonal semi-Lagrangian reconstruction with energy-retention benefits, by leveraging a windowed sinc (i.e., Lanczos) kernel. Without a significant modification of the popular, conventional semi-Lagrangian framework, we replaced the interpolation with our Lanczos-based weight sum, which requires a moderate amount of additional cost. Our method successfully improved the visual details of fluid simulations, presenting richer vortices and spatial structures of fluids, thanks to the bounded near-orthogonal advection operator and the energy-retention benefits it provides. An important aspect of the improvement was studied in terms of the orthogonality of the discrete advection operator, investigating different advection methods such as SF, MC, and MCIP. Our experiments demonstrated the benefit of our improved advection operator for different scenarios in both 2D and 3D. Our Taichi-based reference implementation will be made publicly available upon publication to facilitate reproducibility and follow-up research.

References

Rodolfo Bermejo. 1995. A Galerkin-characteristic algorithm for transport-diffusion equations. *SIAM journal on numerical analysis* 32, 2 (1995), 425–454.

Stefan Bilbao. 2021. 3D interpolation in wave-based acoustic simulation. *IEEE Signal Processing Letters* 29 (2021), 384–388.

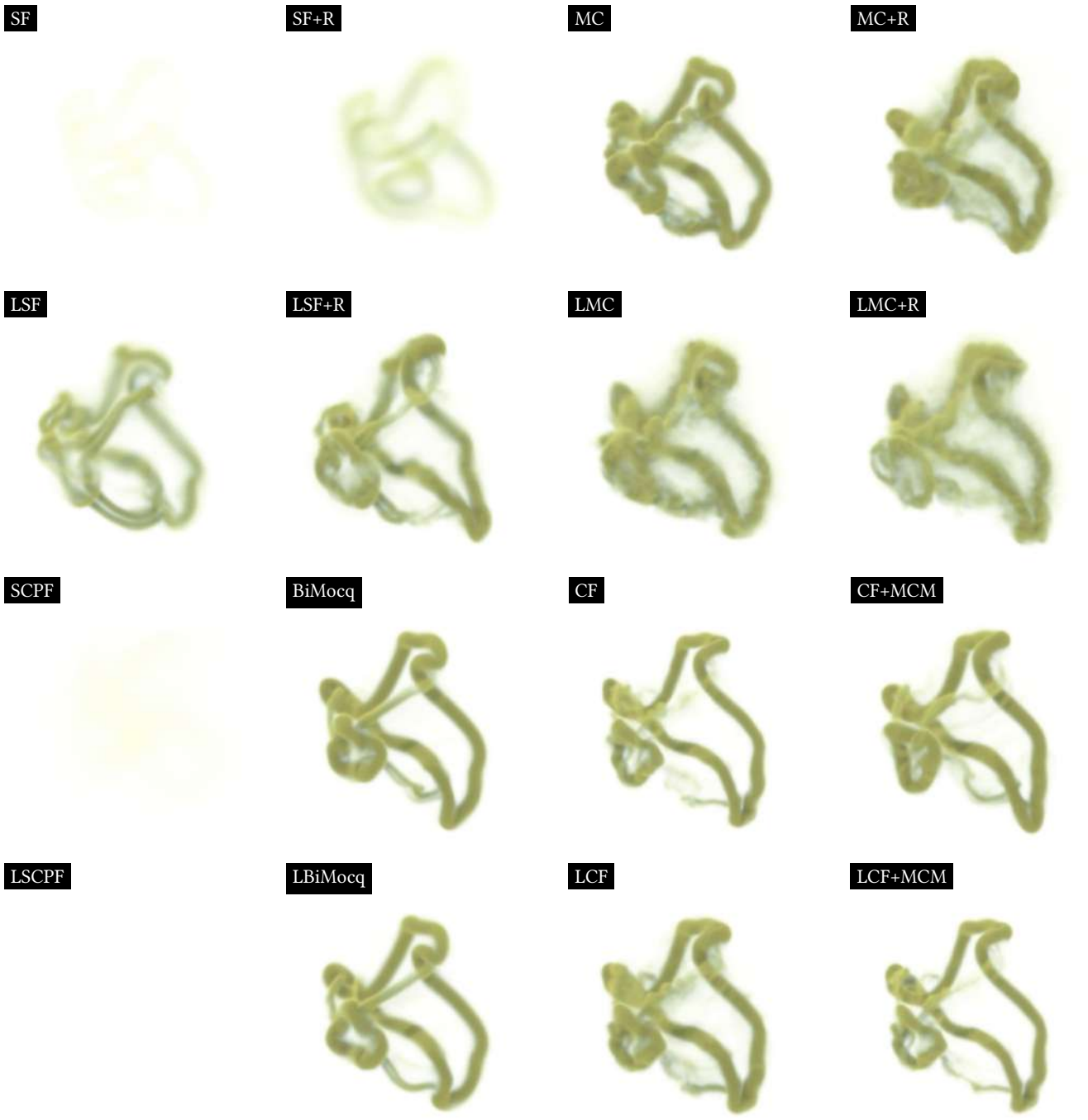


Fig. 16. 3D trefoil knot. Snapshots show a knotted vortex tube initialized as a trefoil knot and evolved under each advection scheme. Eight baselines are shown in the odd rows, while their Lanczos counterparts are placed in the even rows. Across all baselines, the Lanczos counterparts present substantially more dynamic knot structures.

Yadi Cao, Yunuo Chen, Minchen Li, Yin Yang, Xinxin Zhang, Mridul Aanjaneya, and Chenfanfu Jiang. 2022. An efficient B-spline Lagrangian/Eulerian method for compressible flow, shock waves, and fracturing solids. *ACM Transactions on Graphics* 41, 5 (2022), 1–13. doi:10.1145/3519595

Alexandre Joel Chorin. 1967. The numerical solution of the Navier-Stokes equations for an incompressible fluid. *Bull. Amer. Math. Soc.* 73, 6 (1967), 928–931.

Jonathan M Cohen, Sarah Tariq, and Simon Green. 2010. Interactive fluid-particle simulation using translating Eulerian grids. In *Proceedings of the 2010 ACM SIGGRAPH*

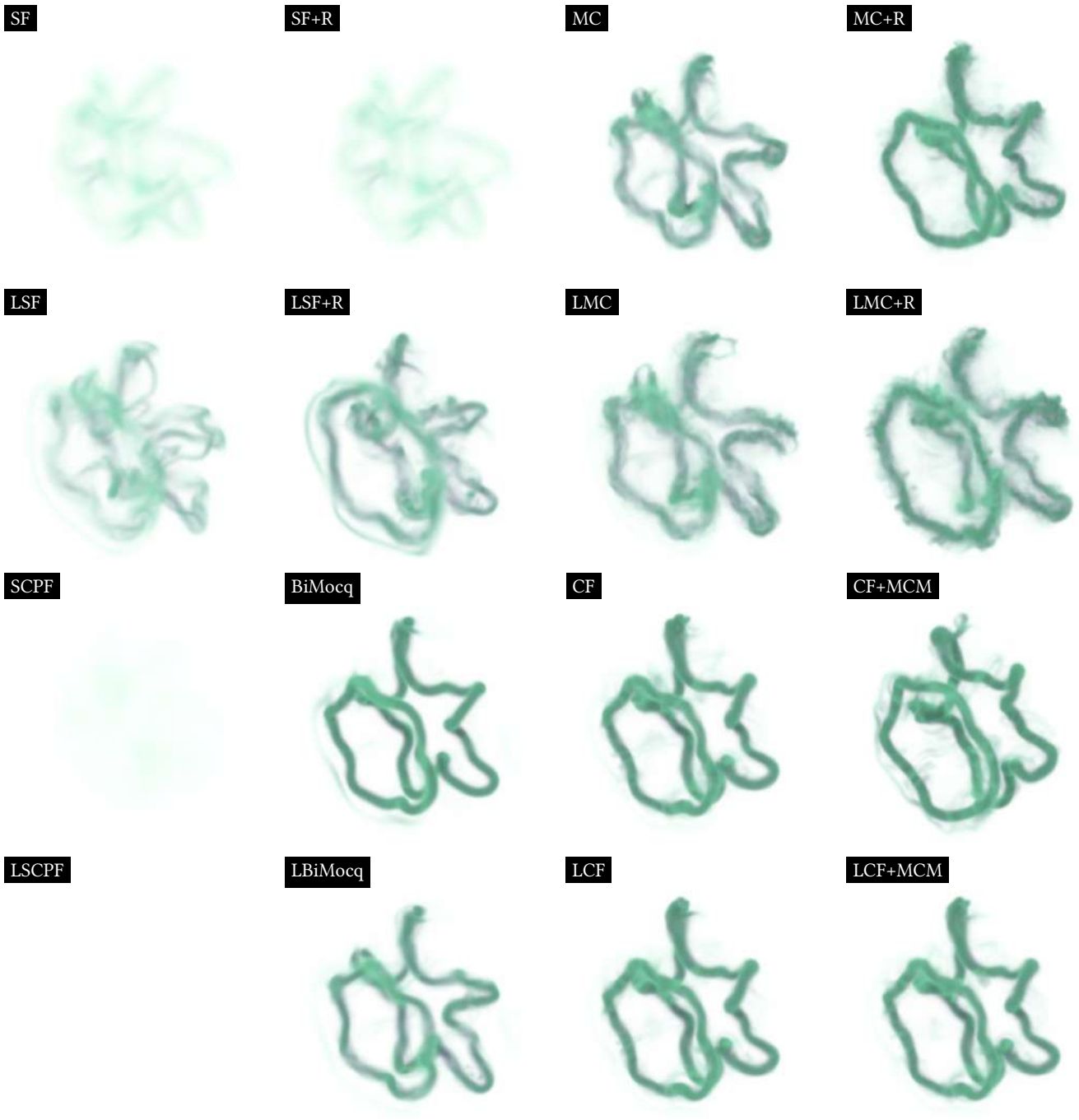


Fig. 17. 3D Cinquefoil-knot. Snapshots of a knotted vortex tube initialized as a cinquefoil knot and evolved under each advection scheme. Eight baselines are shown in the odd rows, while their Lanczos counterparts are placed in the even rows, so each Lanczos cell sits directly below its baseline. The tight, five-fold self-linking structure leads to a high sensitivity to numerical diffusion, and the Lanczos counterparts result in more dynamic knot structure across all baselines.

symposium on Interactive 3D Graphics and Games. 15–22.

Yitong Deng, Hong-Xing Yu, Diyang Zhang, Jiajun Wu, and Bo Zhu. 2023. Fluid simulation on neural flow maps. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–21.

Claude E Duchon. 1979. Lanczos filtering in one and two dimensions. *Journal of Applied Meteorology (1962-1982)* (1979), 1016–1022.

Sharif Elcott, Yiying Tong, Eva Kanso, Peter Schröder, and Mathieu Desbrun. 2007. Stable, circulation-preserving, simplicial fluids. *ACM Transactions on Graphics (TOG)*

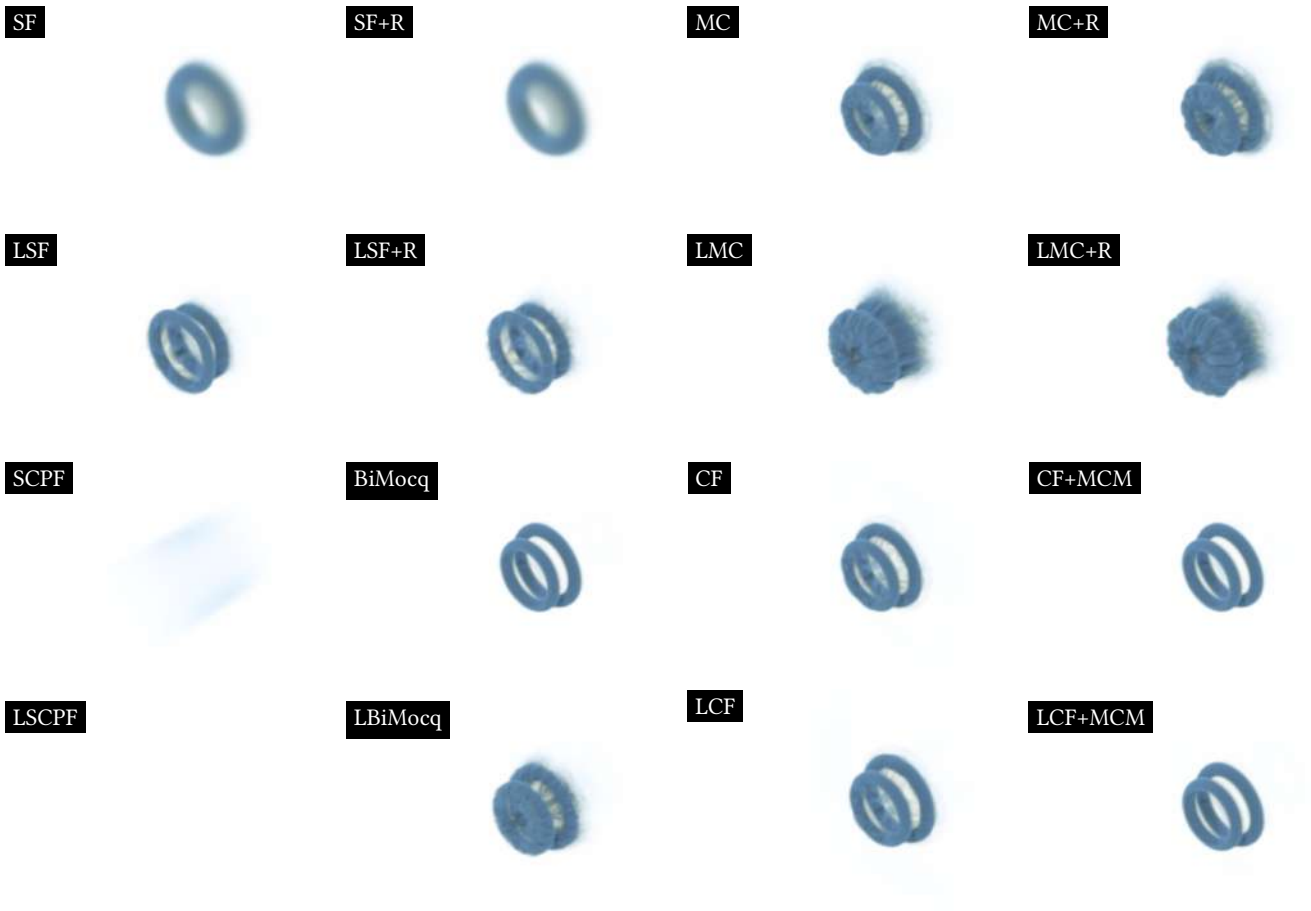


Fig. 18. 3D Leapfrogging vortex rings. Snapshots of two coaxial vortex rings evolved under each advection scheme. Eight baselines are shown in the odd rows, while their Lanczos counterparts are placed in the even rows, so that each Lanczos cell sits directly below its baseline. The repeated mutual slip-through of the two rings is highly sensitive to numerical diffusion. Our Lanczos extension induces substantial improvement on SF, MC, and their variants, while presenting marginal change on CF and its variants.

26, 1 (2007), 4–es.
Ronald Fedkiw, Jos Stam, and Henrik Wann Jensen. 2001. Visual simulation of smoke. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. 15–22.
Bryan E Feldman, James F O'brien, and Bryan M Klingner. 2005. Animating gases with hybrid meshes. *ACM Transactions on Graphics (TOG)* 24, 3 (2005), 904–909.
Lin Fu, Xiangyu Y Hu, and Nikolaus A Adams. 2016. A family of high-order targeted ENO schemes for compressible-fluid simulations. *J. Comput. Phys.* 305 (2016), 333–359. doi:10.1016/j.jcp.2015.10.037
Lin Fu, Xiangyu Y Hu, and Nikolaus A Adams. 2017. Targeted ENO schemes with tailored resolution property for hyperbolic conservation laws. *J. Comput. Phys.* 349 (2017), 97–121. doi:10.1016/j.jcp.2017.07.054
Steven W Gagniere, David AB Hyde, Alan Marquez-Razon, Chenfanfu Jiang, Ziheng Ge, Xuchen Han, Qi Guo, and Joseph Teran. 2020. A hybrid Lagrangian/Eulerian collocated advection and projection method for fluid simulation. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 1–14. doi:10.1111/cgf.14096
Arash Hamzehloo, David J Lusher, Sylvain Laizet, and Neil D Sandham. 2021. On the performance of WENO/TENO schemes to resolve turbulence in DNS/LES of high-speed compressible flows. *International Journal for Numerical Methods in Fluids* 93, 1 (2021), 176–196. doi:10.1002/fld.4879
Francis H. Harlow and J. Eddie Welch. 1965. Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface. *Physics of Fluids* 8, 12 (Dec. 1965), 2182–2189. doi:doi:10.1063/1.1761178
Ami Harten, Bjorn Engquist, Stanley Osher, and Sukumar R Chakravarthy. 1987. Uniformly high order accurate essentially non-oscillatory schemes, III. *J. Comput. Phys.* 71, 2 (1987), 231–303. doi:10.1016/0021-9991(87)90031-3
AC Haza, TM Özgökmen, and P Hogan. 2016. Impact of submesoscales on surface material distribution in a gulf of Mexico mesoscale eddy. *Ocean Modelling* 107 (2016), 28–47.
Shengfeng He and Rynson WH Lau. 2013. Synthetic controllable turbulence using robust second vorticity confinement. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library, 27–35.
Shengfeng He, Hon-Cheng Wong, Wai-Man Pang, and Un-Hong Wong. 2011. Real-time smoke simulation with improved turbulence by spatial adaptive vorticity confinement. *Computer Animation and Virtual Worlds* 22, 2-3 (2011), 107–114.
Yuanming Hu, Tzu-Mao Li, Luke Anderson, Jonathan Ragan-Kelley, and Frédo Durand. 2019. Taichi: a language for high-performance computation on spatially sparse data structures. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 201.
Guang-Shan Jiang and Chi-Wang Shu. 1996. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* 126, 1 (1996), 202–228. doi:10.1006/jcph.1996.0130
ByungMoon Kim, Yingjie Liu, Ignacio Llamas, and Jarek Rossignac. 2005. FlowFixer: Using BFECC for Fluid Simulation.. In *NPH*. 51–56.
Doyub Kim, Oh-young Song, and Hyeong-Seok Ko. 2008a. A Semi-Lagrangian CIP Fluid Solver without Dimensional Splitting. *Computer Graphics Forum* 27, 2 (2008), 467–475. doi:10.1111/j.1467-8659.2008.01144.x
Sun-Tae Kim and Jeong-Mo Hong. 2013. Visual Simulation of Turbulent Fluids Using MLS Interpolation Profiles. *The Visual Computer* 29, 12 (Dec. 2013), 1293–1302. doi:10.1007/s00371-012-0770-4
Theodore Kim, Nils Thürey, Doug James, and Markus Gross. 2008b. Wavelet turbulence for fluid simulation. *ACM Transactions on Graphics (TOG)* 27, 3 (2008), 1–6.

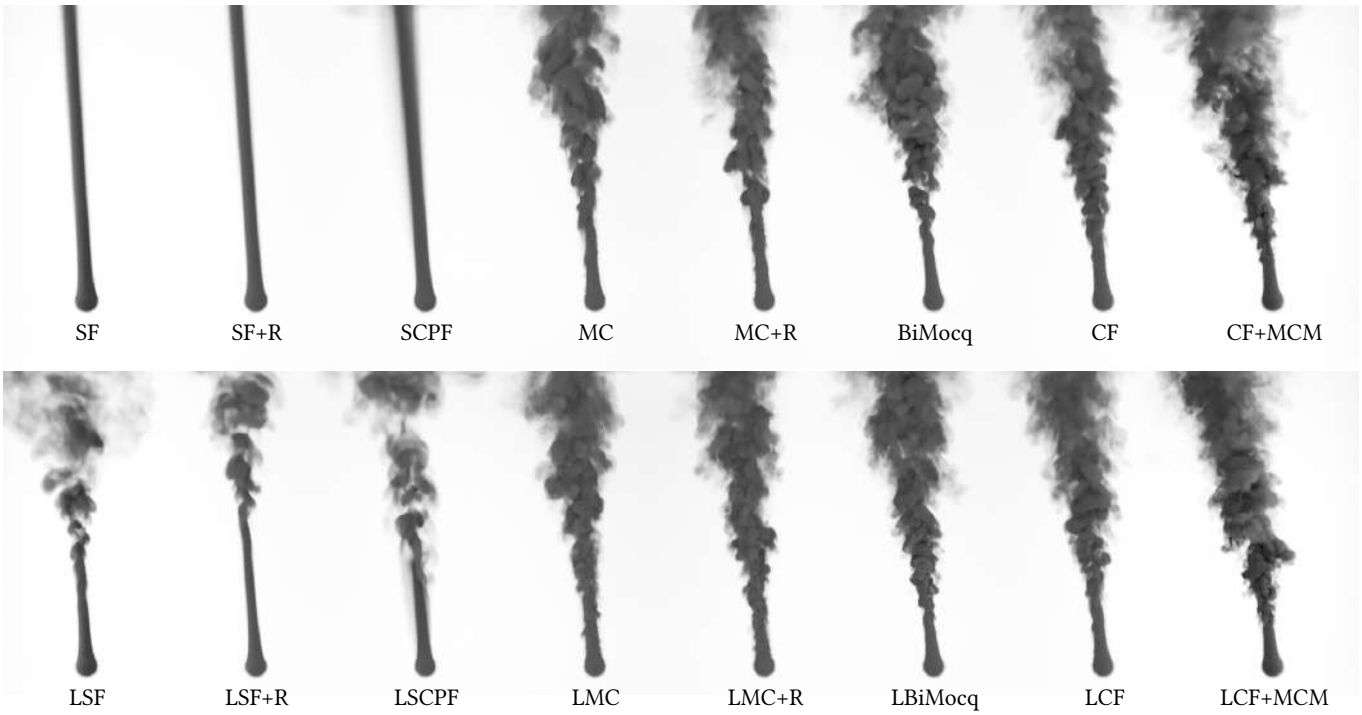


Fig. 19. 3D smoke plume. Each baseline solver (top) is compared to its extended version with our Lanczos filtering (bottom). Our Lanczos operator improves each baseline method, presenting visibly richer small-scale vortex motion.

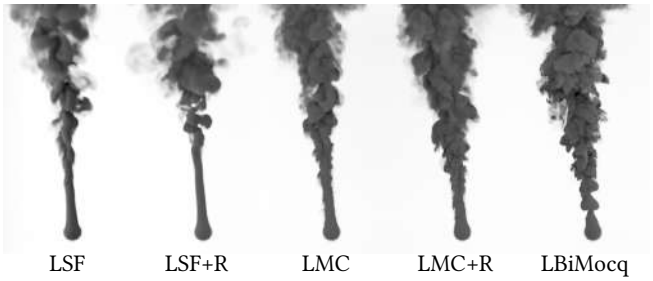


Fig. 20. Our extension is examined with a large time-step (i.e., $\Delta t = 40$ms) for the 3D smoke plume example. Among different variants, LSF and LSF+R offer a good trade-off between efficiency and effectiveness; LMC and LMC+R show better quality at a higher cost; and LBiMocq attains the highest fidelity at the greatest computational expense.

Table 1. Summary of the simulation parameters.

Scene		Δt [ms]	Grid resolution
Meteor	Fig. 1	33	$512 \times 256 \times 256$
2D Kármán vortex street	Fig. 4	100	512×256
2D Taylor vortex	Fig. 5	50	512×512
2D Leapfrogging	Fig. 6	200	256×256
2D SIGGRAPH Logo	Fig. 7	10	420×520
2D Decaying turbulence	Fig. 8	33	256×256
2D Smoke plume	Fig. 10	33	256×768
2D Single vortex	Fig. 13	33	256×256
2D Taylor-Green vortex	Fig. 15	33	256×256
3D Trefoil knot	Fig. 16	1	$256 \times 256 \times 256$
3D Cinquefoil knot	Fig. 17	2	$256 \times 256 \times 256$
3D Leapfrogging	Fig. 18	2	$256 \times 128 \times 128$
3D Smoke plume	Fig. 19	5	$256 \times 128 \times 128$
3D Smoke plume (best)	Fig. 20	40	$256 \times 128 \times 128$

Jozef Kristek, Peter Moczo, and Martin Galis. 2010. Stable discontinuous staggered grid in the finite-difference modelling of seismic motion. *Geophysical Journal International* 183, 3 (2010), 1401–1407.

Jens Krüger, Jens Schneider, and Rüdiger Westermann. 2006. Compression and rendering of iso-surfaces and point sampled geometry. *The Visual Computer* 22, 8 (2006), 517–530.

Cornelius Lanczos. 1988. *Applied analysis*. Courier Corporation.

Michael Lentine, Mridul Aanjaneya, and Ronald Fedkiw. 2011. Mass and momentum conservation for fluid simulation. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 91–100.

Xu-Dong Liu, Stanley Osher, and Tony Chan. 1994. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.* 115, 1 (1994), 200–212. doi:10.1006/jcph.1994.1187

Frank Losasso, Frédéric Gibou, and Ron Fedkiw. 2004. Simulating water and smoke with an octree data structure. In *Acm siggraph 2004 papers*. 457–462.

Jeroen Molemaker, Jonathan M Cohen, Sanjit Patel, Jonyong Noh, et al. 2008. Low Viscosity Flow Simulations for Animation.. In *Symposium on Computer Animation*, Vol. 9. 18.

Mohammad Sina Nabizadeh, Stephanie Wang, Ravi Ramamoorthi, and Albert Chern. 2022. Covector fluids. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–16.

Rahul Narain, Jason Sewall, Mark Carlson, and Ming C Lin. 2008. Fast animation of turbulence using energy transport and procedural synthesis. *ACM Transactions on Graphics (TOG)* 27, 5 (2008), 1–8.

Ziyin Qu, Xinxin Zhang, Ming Gao, Chenfanfu Jiang, and Baoquan Chen. 2019. Efficient and conservative fluids using bidirectional mapping. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–12.

Table 2. Per-frame elapsed time for each method (in milliseconds).

Scene	SF	MC	LSF	LMC	MCIP	ENO	WENO	TENO
2D Decaying turbulence	0.272	0.490	0.362	0.692	0.357	0.380	0.385	0.523
2D Taylor vortex	0.301	0.603	0.420	0.769	0.320	0.365	0.415	0.575
2D Smoke plume	0.541	1.045	0.701	1.502	0.591	0.687	0.790	0.943

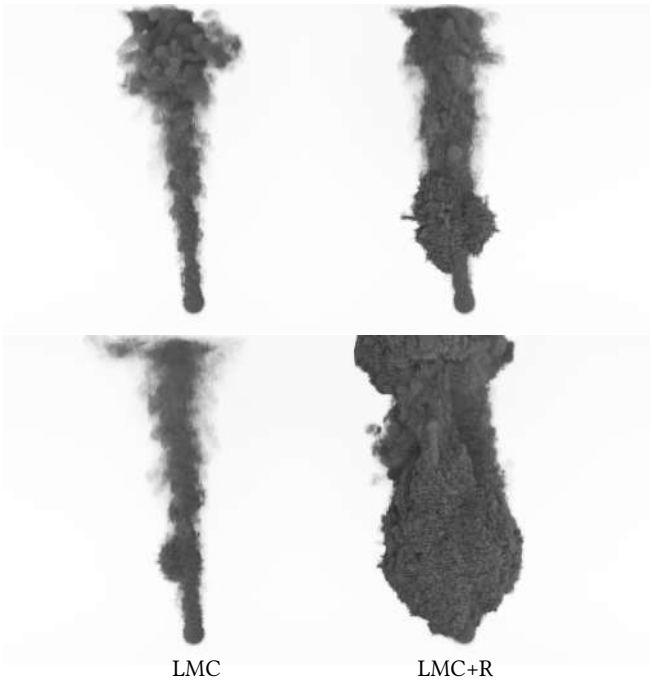


Fig. 21. Ablation study for our limiter. Without the limiter, the Lanczos-based MC variants exhibit overshoot, leading to instability around $t = 50\Delta t$.

LP Riishøjgaard, SE Cohn, Y Li, and R Ménard. 1998. The use of spline interpolation in semi-Lagrangian transport models. *Monthly weather review* 126, 7 (1998), 2008–2016.

Takahiro Sato, Christopher Batty, Takeo Igarashi, and Ryoichi Ando. 2018. Spatially adaptive long-term semi-Lagrangian method for accurate velocity advection. *Computational Visual Media* 4 (2018), 223–230.

Takahiro Sato, Takeo Igarashi, Christopher Batty, and Ryoichi Ando. 2017. A long-term semi-lagrangian method for accurate velocity advection. In *SIGGRAPH Asia 2017 Technical Briefs*. 1–4.

Andrew Selle, Ronald Fedkiw, Byungmoon Kim, Yingjie Liu, and Jarek Rossignac. 2008. An unconditionally stable MacCormack method. *Journal of Scientific Computing* 35 (2008), 350–371.

Andrew Selle, Nick Rasmussen, and Ronald Fedkiw. 2005. A Vortex Particle Method for Smoke, Water and Explosions. *ACM Trans. Graph.* 24, 3 (July 2005), 910–914. doi:10.1145/1073204.1073282

Maurya Shah, Jonathan M Cohen, Sanjit Patel, Penne Lee, and Frédéric Pighin. 2004. Extended galilean invariance for adaptive fluid simulation. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*. 213–221.

CE Shannon. 1949. Communication in the Presence of Noise. *Proceedings of the IRE* 37, 1 (1949), 10–21.

Chi-Wang Shu. 1998. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*. Springer, Berlin, Heidelberg, 325–432.

Chi-Wang Shu and Stanley Osher. 1988. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.* 77, 2 (1988), 439–471. doi:10.1016/0021-9991(88)90177-5

Oh-Young Song, Hyuncheol Shin, and Hyeong-Seok Ko. 2005. Stable but nondissipative water. *ACM Transactions on Graphics (TOG)* 24, 1 (2005), 81–97.

Jos Stam. 1999. Stable Fluids. In *Proceedings of ACM SIGGRAPH 99*. 121–128.

Andrew Staniforth and Jean Côté. 1991. Semi-Lagrangian integration schemes for atmospheric models—a review. *Monthly Weather Review* 119, 9 (1991), 2206–2223. doi:10.1175/1520-0493(1991)119<2206:SLISFA>2.0.CO;2

Yuchen Sun, Junlin Li, Ruicheng Wang, Sinan Wang, Zhiqi Li, Bart G van Bloemen Waanders, and Bo Zhu. 2025. Leapfrog Flow Maps for Real-Time Fluid Simulation. *ACM Transactions on Graphics (TOG)* 44, 4 (2025), 1–12.

Jerry Tessendorf. 2015. Advection Solver Performance with Long Time Steps, and Strategies for Fast and Accurate Numerical Implementation. (2015).

Jerry Tessendorf and Brandon Pelfrey. 2011. The characteristic map for fast and efficient vfx fluid simulations. In *Computer Graphics International Workshop on VFX, Computer Animation, and Stereo Movies. Ottawa, Canada.*

Ken Turkowski. 1990. Filters for common resampling tasks. In *Graphics gems*. 147–165.

Michael Unser. 2000. Sampling—50 Years After Shannon. *PROCEEDINGS OF THE IEEE* 88, 4 (2000), 569.

Xiaokun Wang, Yanrui Xu, Sinuo Liu, Bo Ren, Jirí Kosinka, Alexandru C Telea, Jiamin Wang, Chongming Song, Jian Chang, Chenfeng Li, et al. 2024. Physics-based fluid simulation in computer graphics: Survey, research trends, and challenges. *Computational Visual Media* 10, 5 (2024), 803–858.

Xi-Yuan Yin, Kai Schneider, and Jean-Christophe Nave. 2023. A Characteristic Mapping Method for the three-dimensional incompressible Euler equations. *J. Comput. Phys.* 477 (2023), 111876.

Jong-Chul Yoon, Hyeong Ryeol Kam, Jeong-Mo Hong, Shin Jin Kang, and Chang-Hun Kim. 2009. Procedural Synthesis Using Vortex Particle Method for Fluid Simulation. *Computer Graphics Forum* 28, 7 (Oct. 2009), 1853–1859. doi:10.1111/j.1467-8659.2009.01563.x

Jonas Zehnder, Rahul Narain, and Bernhard Thomaszewski. 2018. An advection-reflection solver for detail-preserving fluid simulation. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–8.