Detail-preserving Smoke Simulation Using an Efficient High-order Numerical Scheme

Jian ZHU[1], Zhuo YANG[1], Hanqiu SUN[4], Enhua WU[2,3] , Ruichu CAI[1], Zhifeng HAO[1,5]

[1]School of Computer Science, Guangdong University of Technology

[2]State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences

[3]Faculty of Science and Technology, University of Macau

[4]Institute of Space and Earth Info. Science, The Chinese University of Hong Kong

[5]School of Mathematics and Big Data, Foshan University

Outline

1 Introduction

2 Related Work

3 Main Idea

4 Part-1: Approximation of The Cross Derivatives

5 Part-2: Taylor Expansion based CIP

6 Experimental Results

7 Conclusion & Future Work

01 Introduction

01Introduction

- Eulerian grid based fluid simulation method can create realistic fluid animation results by solving the Navier–Stokes (NS) equations on Grid.

$$\begin{cases} \nabla \cdot u = 0 \\ \dfrac{\partial u}{\partial t} = -(u \cdot \nabla)u + v\nabla^2 u - \dfrac{\nabla p}{\rho} + f \end{cases}$$

- However, it suffers from numerous numerical dissipations, and one source of the numerical dissipations is from smooth operations of linear interpolation at the advection step.

$$\frac{\partial u}{\partial t} + (u \cdot \nabla)u = 0$$

01Introduction

In this paper

We propose to alleviate the numerical dissipations by utilizing a novel CIP(Constrained Interpolation Profile) based advection scheme of third-order accuracy.

Features:

- ✓ Stable
- ✓ High accuracy
- ✓ Fast
- ✓ Low memory consumption
- ✓ Compact stencil

02 Related Work

02Related Work: Reducing advection dissipations

02Related Work: Reducing advection dissipations

High-order Interpolation Schemes

Cubic-spline[4], Essentially non-oscillatory (ENO) [5], Weighted ENO (WENO)[5], etc.

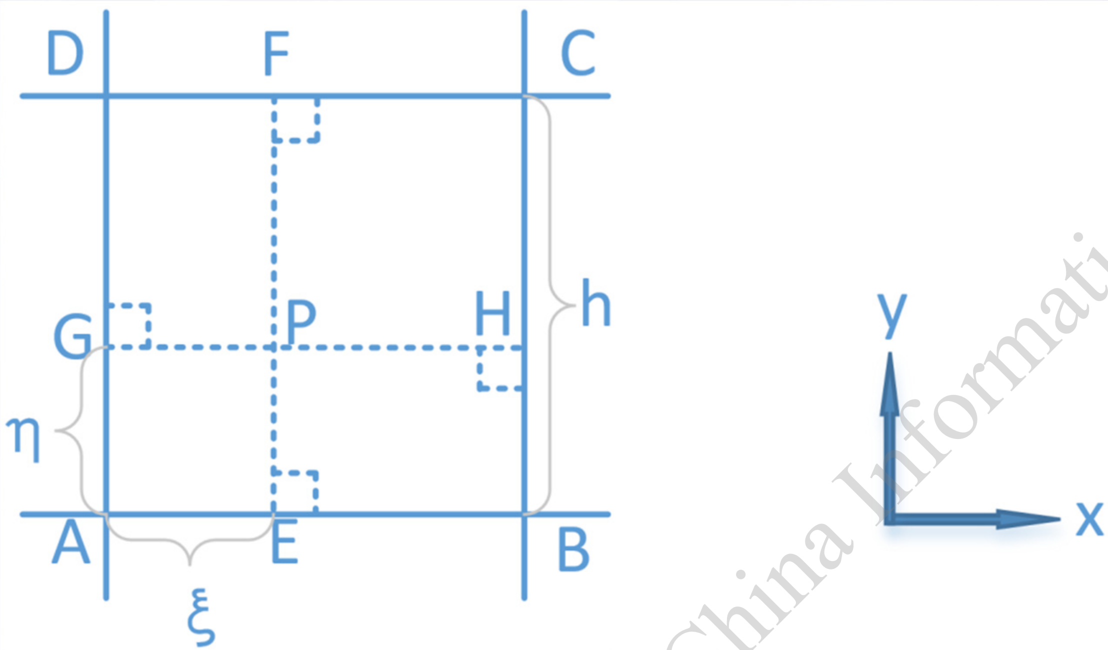
- ✓ Pros:
  - High order accuracy (3rd or higher order)
- ✓ Cons:
  - Computationally expensive, thus difficult to apply to high resolution simulations.
  - Wide Stencil, thus not applicable to non-uniform grids.

Constrained Interpolation Profile (CIP) [6] based methods: MCIP[7], USCIP[8], etc.

- ✓ Pros:
  - Compact Stencil, thus applicable to non-uniform grids
  - Third order accuracy (Higher than most of the advection schemes)
- ✓ Cons:
  - Problems exists when being extended to high-dimensional simulations:
    - Unstable
    - Computational expensive
    - High memory cost
    - etc.

03 Main Idea

03: Problem Description



A, *B*, *C*, *D* are grid points, *P* is the backtracked point, and *E*, *F*, *G* and *H* are projections of *P* to the four cell sides *AB*, *DC*, *AD* and *BC*, respectively. The cell length is *h*.

- *Question:* How to obtain physical quantities at *P* from its neighboring four grid points with low dissipations ?
- CIP advects not only the physical quantity but also its derivatives, and constructs high order polynomials for interpolations with 3rd order accuracy.
- To solve the aforementioned problems of previous CIP-based methods, we propose an efficient CIP-based method by using local Taylor expansions.

03 Main Idea

Key idea

- Only store the physical quantity and its first-order derivatives on grid.
- Approximate the high-order derivatives on demand by using local Taylor expansions.

Traditional CIP

1. Store the physical quantity ϕ to be advected and its derivatives $[\partial_x\phi, \partial_y\phi, \partial_{xy}\phi]$ on grid.
2. Interpolations with high order polynomials.

VS

Our method

1. Store the physical quantity ϕ to be advected and its first-order derivatives $[\partial_x\phi, \partial_y\phi]$ on grid.
2. Approximation of the cross derivatives $\partial_{xy}\phi$ with third-order accuracy.
3. Iterative interpolations with 1D CIP.

04 Part-1: Approximation of The Cross Derivatives

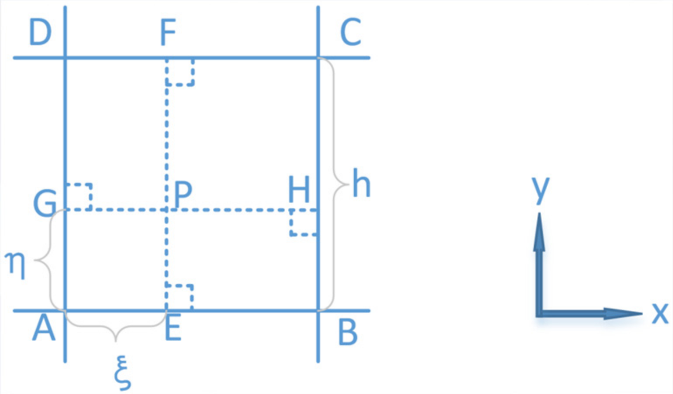
Approximation of The Cross Derivatives

- As proved in [9], for a bivariate function *f(x, y)* that has continuous derivatives on the neighborhood of *(x, y)*, we can approximate $\partial_{xy} f$ with the following equation based on Taylor expansion.

$$\partial_{xy} f(x,y) = (\partial_y f(x+\Delta x, y) - \partial_y f(x,y))/\Delta x + (\partial_x f(x, y+\Delta y) - \partial_x f(x,y))/\Delta y \\ -(f(x+\Delta x, y+\Delta y) + f(x,y) - f(x+\Delta x, y) - f(x, y+\Delta y))/\Delta x \Delta y \quad (1)$$

- $\partial_{xy}\phi$ at grid point *A* can be computed as below using Eq. (1) by setting $\Delta x = \Delta y = h$.

$$\partial_{xy}\phi_A = (\partial_y \phi_B - \partial_y \phi_A)/h + (\partial_x \phi_D - \partial_x \phi_A)/h - (\phi_C + \phi_A - \phi_B - \phi_D)/h^2 \quad (2)$$



- Similarly, $\partial_{xy}\phi$ at grid point *B, C* and *D* can be computed as

$$\partial_{xy}\phi_B = (\partial_y \phi_B - \partial_y \phi_A)/h + (\partial_x \phi_C - \partial_x \phi_B)/h - (\phi_C + \phi_A - \phi_B - \phi_D)/h^2 \quad (3)$$

$$\partial_{xy}\phi_C = (\partial_y \phi_C - \partial_y \phi_D)/h + (\partial_x \phi_C - \partial_x \phi_B)/h - (\phi_C + \phi_A - \phi_B - \phi_D)/h^2 \quad (4)$$

$$\partial_{xy}\phi_D = (\partial_y \phi_C - \partial_y \phi_D)/h + (\partial_x \phi_D - \partial_x \phi_A)/h - (\phi_C + \phi_A - \phi_B - \phi_D)/h^2 \quad (5)$$

05 Part-2: Taylor Expansion based CIP

Algorithm Details

I. $\partial_{xy}\phi$ at the grid points A, B, C and D are computed with Eq. (2) to (5), respectively.

II. $[\phi, \partial_x\phi]$ and $[\partial_y\phi, \partial_{xy}\phi]$ at point E are computed by 1D CIP interpolations between A and B.

III. $[\phi, \partial_x\phi]$ and $[\partial_y\phi, \partial_{xy}\phi]$ at point F are computed by 1D CIP interpolations between D and C.

IV. $[\phi, \partial_y\phi]$ and $[\partial_x\phi, \partial_{xy}\phi]$ at point P are computed by 1D CIP interpolations between E and F.

Algorithm 1 Taylor expansion based CIP in 2D

Input: $[\phi, \partial_x\phi, \partial_y\phi]$ at A, B, C, D;
Output: $[\phi, \partial_x\phi, \partial_y\phi]$ at P;
1: Compute $\partial_{xy}\phi$ at A, B, C, D with Eq. (2) to (5);
2: Compute $[\phi, \partial_x\phi]_E$ from $[\phi, \partial_x\phi]_A$ and $[\phi, \partial_x\phi]_B$ by 1D CIP interpolation;
3: Compute $[\partial_y\phi, \partial_{xy}\phi]_E$ from $[\partial_y\phi, \partial_{xy}\phi]_A$ and $[\partial_y\phi, \partial_{xy}\phi]_B$ by 1D CIP interpolation;
4: Compute $[\phi, \partial_x\phi]_F$ from $[\phi, \partial_x\phi]_D$ and $[\phi, \partial_x\phi]_C$ by 1D CIP interpolation;
5: Compute $[\partial_y\phi, \partial_{xy}\phi]_F$ from $[\partial_y\phi, \partial_{xy}\phi]_D$ and $[\partial_y\phi, \partial_{xy}\phi]_C$ by 1D CIP interpolation;
6: Compute $[\phi, \partial_y\phi]_P$ from $[\phi, \partial_y\phi]_E$ and $[\phi, \partial_y\phi]_F$ by 1D CIP interpolation;
7: Compute $[\partial_x\phi, \partial_{xy}\phi]_P$ from $[\partial_x\phi, \partial_{xy}\phi]_E$ and $[\partial_x\phi, \partial_{xy}\phi]_F$ by 1D CIP interpolation;

06 Experimental Results

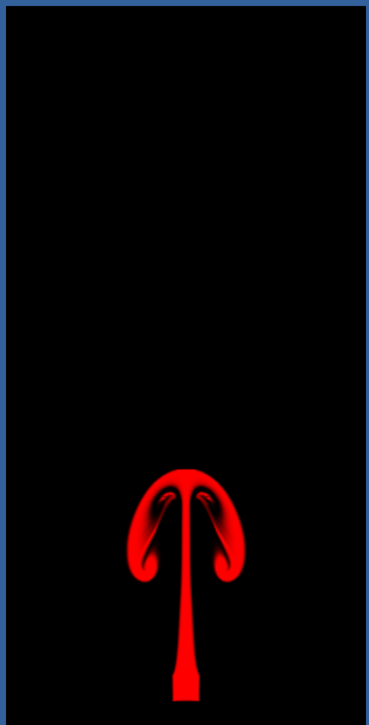
06 Experimental Results

- **Platform:**
  - Intel i5 3.30 GHz CPU and 12.0 GB RAM
- **Configuration:**
  - Resolution: 256×512
  - Uniform Grid
  - No Vorticity Confinement
- **Comparisons:**
  - linear semi-Lagrangian [4]
  - BFECC [2]
  - MCIP [7]
  - USCIP [8]

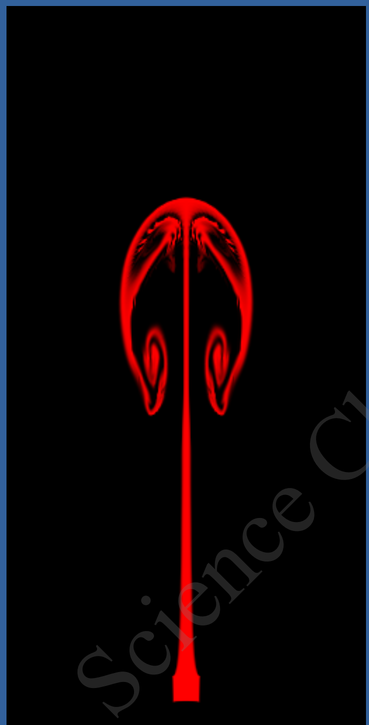
06 Rising Smoke: Case I

Case I: The density field is advected with different schemes, and the velocity field is advected with BFECC

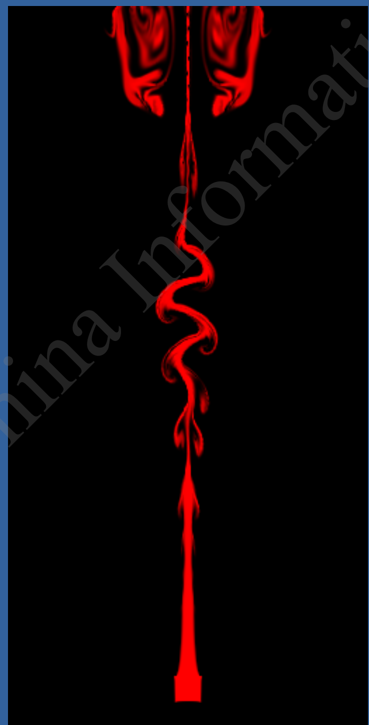
I-A: Our result



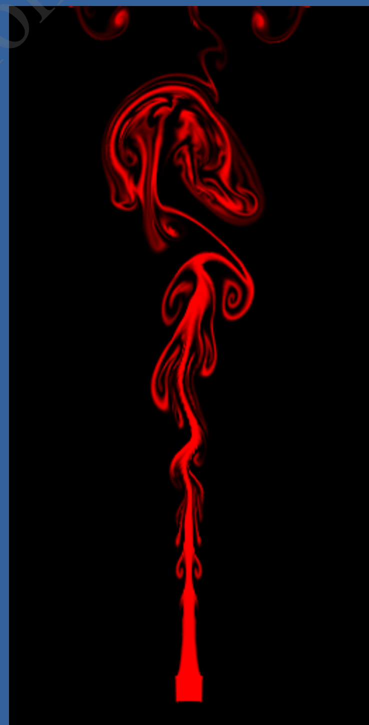
80th frame



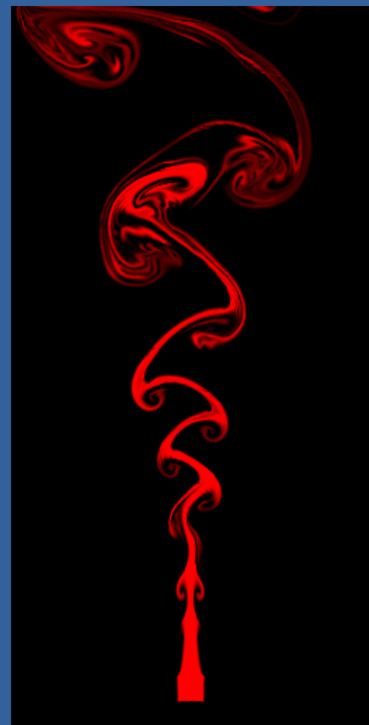
130th frame



180th frame



230th frame

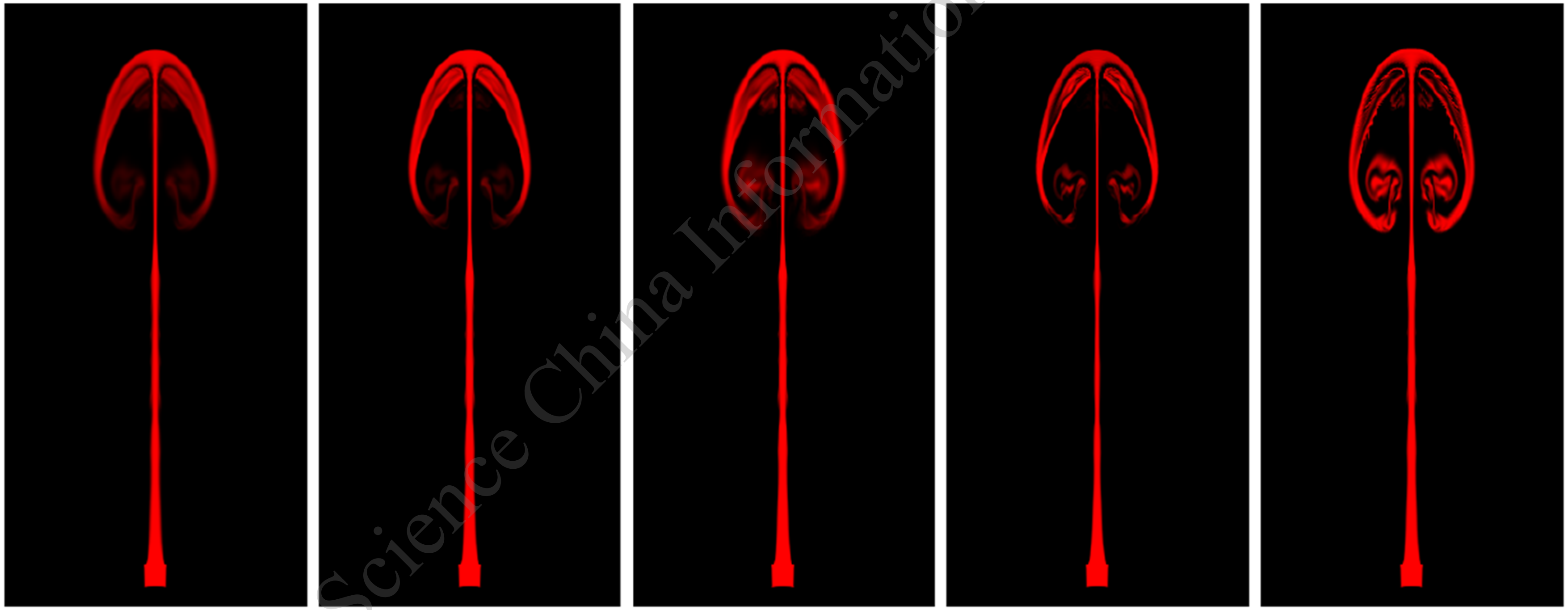


280th frame

06 Rising Smoke: Case I

Case I: The density field is advected with different schemes, and the velocity field is advected with BFECC

I-B: Comparisons



(a) Linear (b) BFECC (c) MCIP (d) USCIP (e) Ours

06 Rising Smoke: Case II

Case II: The density field is advected with BFECC, and the velocity field is advected with different schemes.

II-A: Our result

80th frame | 130th frame | 180th frame | 230th frame | 280th frame

06 Rising Smoke: Case II

Case II: The density field is advected with BFECC, and the velocity field is advected with different schemes.

II-B: Comparisons on the visual quality

(a) Linear (b) BFECC (c) MCIP (d) USCIP (e) Ours

Case II: The density field is advected with BFECC, and the velocity field is advected with different schemes.

II-C: Comparisons on the computation time and memory consumption

Table 1 Computation time and memory requirement for the smoke simulations in Case II-B

	Linear	BFECC	MCIP	USCIP	Ours
Time for advection only (sec/frame)	0.008	0.028	0.015	0.021	0.020
Time for whole simulation (sec/frame)	0.58	0.63	0.60	0.62	0.61
Number of advection	1	3	1	1	1
Number of variables on each grid point for each physical field	1	2	4	3	3

07 Conclusion and Future Work

In this paper:

- A novel CIP-based advection method of third-order accuracy is proposed to achieve better advection results for visually convincing smoke simulation.
- The method only stores the physical quantity and its first-order derivatives on the grid, and efficiently computes the high order derivatives on the fly based on Taylor expansions.
- Overall, it is stable, fast and accurate with low memory cost.

Future Work:

- Use the proposed method to accurately track free surfaces to simulate highly detailed water.
- Combine our method with adaptive techniques to achieve better simulation results on non-uniform grids.

Thank You