A simple bound-preserving sweeping technique for conservative numerical approximations

Yuan Liu * Yingda Cheng † Chi-Wang Shu ‡

March 29, 2016

Abstract

In this paper, we propose a simple bound-preserving sweeping procedure for conservative numerical approximations. Conservative schemes are of importance in many applications, yet for high order methods, the numerical solutions do not necessarily satisfy maximum principle. This paper constructs a simple sweeping algorithm to enforce the bound of the solutions. It has a very general framework acting as a postprocessing step accommodating many point-based or cell average-based discretizations. The method is proven to preserve the bounds of the numerical solution while conserving a prescribed quantity designated as a weighted average of values from all points. The technique is demonstrated to work well with a spectral method, high order finite difference and finite volume methods for scalar conservation laws and incompressible flows. Extensive numerical tests in 1D and 2D are provided to verify the accuracy of the sweeping procedure.

*Department of Mathematics and Statistics, Mississippi State University, Mississippi State, 39762 U.S.A. yl686@msstate.edu. Research is supported by Mississippi State University startup grants.

†Department of Mathematics, Michigan State University, East Lansing, MI 48824 U.S.A. ycheng@math.msu.edu. Research is supported by NSF grants DMS-1318186 and DMS-1453661.

‡Division of Applied Mathematics, Brown University, Providence, RI 02912 U.S.A. shu@dam.brown.edu. Research is supported by ARO grant W911NF-15-1-0226 and NSF grant DMS-1418750.

1 Introduction

Conservative schemes are of importance in many applications such as hyperbolic conservation laws, yet for high order methods, the numerical solutions do not necessarily preserve the bounds as do the exact solutions of the differential equations. A naive truncation of the numerical values that go out of the bound will render the scheme non-conservative. Recent years have seen major developments in this area, particularly for high order schemes. In [20, 21], Zhang and Shu developed arbitrary high order finite finite volume and discontinuous Galerkin (DG) methods which preserve bound preserving property using a scaling limiter. The idea was generalized to unstructured meshes [22, 25] and finite difference framework [24]. They theoretically and numerically demonstrate that the schemes are able to the maintain designed order of accuracy under a CFL constraint. Thereafter, the limiter is widely applied to schemes for various partial differential equations (PDEs), see [12, 11, 23, 19, 26, 1]. Recently, flux-correction based limiters have been developed by Hu et al. [6] and Xu et al. [17, 8, 13, 15, 2, 3, 16, 14]. The parameterized limiting technique [17] can maintain the bounds of the solution without excessively restricting the CFL and its accuracy has been verified by numerical tests and theoretical proof for certain schemes. The design of all bound-preserving techniques mentioned above naturally depends on the underlying numerical schemes, and such connections are keys to establish accuracy and bound-preserving properties of the numerical solutions. On the other hand, it is generally understood that it is more challenging to enforce bound-preserving property for global schemes such as the spectral method without sacrificing accuracy [9]. Spectral method has superior performance approximating smooth solutions. However, they do not necessarily maintain the bounds of the solutions, and the issue is more prominent for approximation of discontinuous functions, where the Gibbs phenomenon comes into play.

In this paper, we propose a bound-preserving sweeping procedure for conservative numerical approximations. The main advantage of our method lies in its simplicity in implementation, consisting of only two sweeping steps independent of the mesh geometry. It has a very general framework acting as a postprocessing step accommodating many point-based or cell average-based discretizations. The method is proven to preserve the bounds of the numerical solution while conserving a prescribed quantity designated as a weighted average of values from all points. The technique is demonstrated to work well for scalar conservation laws and incompressible flows. Since our method is global in nature and does not depend on the underlying numerical schemes, we do not attempt to establish the theoretical convergence results but rely on extensive numerical studies to demonstrate its performance in accuracy preservation. In particular, we show the method works well for a Fourier spectral method, high order finite difference and finite volume methods in 1D and 2D.

The rest of the paper is organized as follows: in Section 2, we describe the sweeping technique and discuss its properties. Section 3 verifies the performance of the method for a Fourier spectral approximation. Section 4 contains the applications in solving PDEs, and we consider a semi-Lagrangian Fourier spectral method, high order finite difference schemes, and high order finite volume schemes in 1D and on 2D unstructured meshes. We conclude the paper with some remarks and future work in Section 5.

2 A conservative bound-preserving sweeping procedure

In this section, we describe our conservative bound-preserving sweeping procedure and discuss its properties. The method is designed to enforce bounds for a sequence of discrete point values $\{v_j\}_{j=1}^N$ while maintaining a weighted average $\sum_{j=1}^N w_j v_j = \bar{V}$ unchanged, where $\{w_j\}_{j=1}^N$ are given positive weights and $\sum_{j=1}^N w_j = 1$. The only assumption we need is that $\bar{V}$ is within the bounds.

Suppose we have a set of points $\{v_j\}_{j=1}^N$ and they do not necessarily belong to $[m, M]$, where m, M denote the lower and upper bound of the solution. However, $\bar{V} = \sum_{j=1}^N w_j v_j \in [m, M]$. The procedure we describe below will generate new $\{v_j\}_{j=1}^N$ values, such that $\bar{V}$ stays unchanged while the updated $v_j \in [m, M], \forall j$. In practice, $\{v_j\}_{j=1}^N$ can denote point values, cell averages in 1D or multi-D as long as they belong to the setting described above. The detailed procedure consists of a "right" sweep followed up by a "left" sweep. In particular, we follow

Algorithm 2.1 (A conservative bound-preserving sweeping procedure)

Step 1 *For* $j = 1 : N-1$,

- *if* $v_j < m$, *then we let* $v_{j+1} \leftarrow v_{j+1} + \frac{w_j}{w_{j+1}}(v_j - m), \quad v_j \leftarrow m.$
- *if* $v_j > M$, *then we let* $v_{j+1} \leftarrow v_{j+1} + \frac{w_j}{w_{j+1}}(v_j - M), \quad v_j \leftarrow M.$

Step 2 *For* $j = N : 2$,

- *if* $v_j < m$, *then we let* $v_{j-1} \leftarrow v_{j-1} + \frac{w_j}{w_{j-1}}(v_j - m), \quad v_j \leftarrow m.$
- *if* $v_j > M$, *then we let* $v_{j-1} \leftarrow v_{j-1} + \frac{w_j}{w_{j-1}}(v_j - M), \quad v_j \leftarrow M.$

The next theorem will establish the conservation and bound-preserving property of the values $\{v_j\}_{j=1}^N$ after the sweep.

Theorem 2.2 *Suppose* $\bar{V} = \sum_{j=1}^N w_j v_j \in [m, M]$ *with* $w_j > 0, \forall j$ *and* $\sum_{j=1}^N w_j = 1$, *Algorithm 2.1 will generate new values of* $\{v_j\}$, *such that* $v_j \in [m, M], \forall j$ *while the weighted average* $\bar{V}$ *is maintained.*

Proof: It is trivial to see that all the operations in Algorithm 2.1 do not change $\bar{V}$. For example, considering Step 1, since

$$w_j m + w_{j+1}\left(v_{j+1} + \frac{w_j}{w_{j+1}}(v_j - m)\right) = w_j v_j + w_{j+1} v_{j+1},$$

and

$$w_j M + w_{j+1}\left(v_{j+1} + \frac{w_j}{w_{j+1}}(v_j - M)\right) = w_j v_j + w_{j+1} v_{j+1},$$

the value of $w_j v_j + w_{j+1} v_{j+1}$ remains unchanged after both operations, and hence for $\bar{V}$. The situation is similar for Step 2.

Next, we will prove that all the new values of $\{v_j\}_{j=1}^N$ will stay in the bound $[m, M]$. After Step 1, all v_j for $j = 1, \ldots, N-1$ will be in $[m, M]$. If $v_N \in [m, M]$ after step 1, then no correction will be performed in Step 2 and we are done.

Otherwise, suppose the value of v_N after Step 1 is less than m, after we perform the correction in Step 2 for $j = N$, $v_N = m$ and this is the final value for v_N. This correction will also make v_{N-1} smaller. If the updated value of v_{N-1} is still within the bound $[m, M]$, then no further correction will be enacted, and we are done. Otherwise, v_{N-1} can only be smaller than m. Therefore, the next correction for $j = N - 1$ will make $v_{N-1} = m$ and v_{N-2} smaller. We can repeat the discussion hereon for smaller values of j until we reach a j_0, such that the correction for $j = j_0$ gives an updated $v_{j_0-1} \in [m, M]$. This number j_0 must exist and be bigger than or equal to 2, because otherwise will have $v_N = v_{N-1} = \cdots = v_2 = m$, $v_1 < m$ after all corrections are completed. This contradicts with $\bar{V} = \sum_{j=1}^{N} w_j v_j \in [m, M]$. The proof for the case of $v_N > M$ after Step 1 is similar and omitted. ■

We can easily see that Theorem 2.2 also holds when w_j are all negative. This case is essentially symmetric to the positive weight case, and is omitted in the discussion in this paper. Comparing with a naive truncation of the overshoot and undershoot values, the procedure has the advantage of being conservative, which is also the key to the assumption that $\bar{V} \in [m, M]$. In fact, we only need to satisfy this assumption when $t = 0$ for the PDE examples in Section 4 in order to use it for later time steps. The method is strikingly simple to implement, in particular, only $O(N)$ operations are involved. However, we do mention that such sweeping is not unique with respect to the ordering of $\{v_j\}_{j=1}^{N}$. Since for any one-to-one mapping from $\{v_j\}_{j=1}^{N}$ to $\{\tilde{v}_j\}_{j=1}^{N}$, we can apply the same procedure to $\{\tilde{v}_j\}_{j=1}^{N}$, and the updated values of $\{\tilde{v}_j\}_{j=1}^{N}$ will still be conservative and bound-preserving. Section 3 contains extensive numerical study on this subject. In fact, we find a natural ordering that maintains the neighboring relation of the points performs the best from the accuracy perspective.

Algorithm 2.1 is global in nature, involving all points in the procedure. Therefore, it is more naturally adapted to a global approximation. The main concern of this algorithm is its impact on accuracy. From the algorithm description, and the fact that the exact solution belongs to $[m, M]$, we can see the worst case scenario yields the errors enlarged by $O(N)$, i.e. loss of one order of the scheme in 1D case, and two orders in 2D case. However, the numerical tests in Sections 3 and 4 suggest that such worst case scenario does not happen for most benchmark test problems and we always recover similar order of convergence of the original method, perhaps either due to error cancellation between points or the fact that only isolated points need correction. We believe the improved convergence order can only be proven based on each individual scheme and we leave it to our future work.

3 Numerical tests for a spectral approximation

In this section, we illustrate the performance of the sweeping method when applied to the Fourier spectral approximation. Assuming periodic boundary conditions, Fourier spectral methods offer superb accuracy for smooth functions. However, discontinuity in the underlying function will cause the so-called Gibbs phenomenon [5], and bring numerical undershoot and overshoot. Filter and postprocessing techniques [4] have been invented to treat discontinuities to achieve accuracy enhancement, yet maintaining the bound of the solution still poses difficult numerical challenges. In this paper, we are interested in studying the performance of the Algorithm 2.1 for Fourier spectral approximation of smooth and non smooth functions paying particular attention to the impact of the sweeping methods to the accuracy of the scheme.

Our setting in one dimension is the following. Consider an unknown function $u(x)$ defined on the periodic domain $[0, 2\pi]$ within the bound $[m, M]$, we are given a set of points $\{u_j = u(x_j)\}_{j=1}^N$ at uniform points $x_j = \frac{2\pi}{N}j$, $j = 0, \ldots, N-1$, and are interested in finding the values of $\{v_j = u(x_j - a)\}_{j=1}^N$ at the shifted points $x_j - a$ by a Fourier spectral method, where a is a constant parameter. The approximation algorithms detail as follows:

1. Start with a set of points u_j, $j = 0, \ldots, N-1$.
2. Use fast Fourier transform (FFT) to reconstruct the interpolating function $u_N = \sum_{|k|\leq N/2} \hat{u}_k e^{ikx}$.
3. Evaluate $v_j = u_N(x_j - a) = \sum_{|k|\leq N/2} \hat{u}_k e^{-ika} e^{ikx_j}$ by FFT.

An interesting property of the Fourier approximation guarantees that $\sum_{j=0}^{N-1} v_j = \sum_{j=0}^{N-1} u_j$, which implies equal weights $w_j = 1/N$, $\forall j$. However, v_j may not stay in bound $[m, M]$. It is therefore to our interest to apply the conservative bound-preserving sweeping procedure to obtain conservative, bound-preserving approximations in this case. In particular, we will test the impact of the ordering of v_j on the quality of the solution. Our ordering includes

- regular sweep, i.e. apply the procedure to the sequence $\{v_1, v_2, v_3, ...\}$;
- odd/even sweep, i.e. apply the procedure to the sequence $\{v_1, v_3, v_5, ...v_2, v_4, ...\}$;
- random sweep, i.e. apply the procedure to a random ordering of $\{v_j\}_{j=1}^N$.

Naturally, the results of the random sweeps depend on the randomly generated array. The results for random sweep shown below will correspond to one particular random ordering with N cells, and may change upon each run.

Example 3.1 *We let $a = \pi/15$, and test the following functions of $u(x)$ with different degrees of smoothness.*

Case 1. $u(x) = e^{-5(x-\pi)^2}$. This is a smooth function with $m = 0, M = 1$. The results without and with different ordering of the sweeping method are reported in Table 1. We list the L^1, L^∞ errors as well as $v_{min} - m$ and $M - v_{max}$. Negative values of $v_{min} - m$ and $M - v_{max}$ imply the approximation has gone out of the bound $[m, M]$. We can see that for the smooth function the sweeping technique enforces the bound-preserving property while the spectral accuracy is still maintained. The results are not sensitive to the ordering of $\{v_j\}_{j=1}^N$, and we believe this is because only a few isolated points are actually out of bounds.

Table 1: Accuracy study for Example 3.1, Case 1 on the computational domain $[0, 2\pi]$, $a = \pi/15$. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	10	4.19e-02	–	1.46e-01	–	-3.92e-02	1.75e-01
	20	5.58e-04	6.23	1.38e-03	6.73	-3.50e-04	1.91e-02
	40	7.50e-11	22.83	1.38e-10	23.25	-3.13e-11	2.09e-03
	80	1.01e-16	19.50	5.55e-16	17.92	-5.55e-17	1.20e-03
Regular Sweep	10	2.63e-02	–	1.46e-01	–	0	1.75e-01
	20	4.40e-04	5.90	1.38e-03	6.73	0	1.91e-02
	40	7.01e-11	22.58	1.38e-10	23.25	0	2.09e-03
	80	9.86e-17	19.44	5.55e-16	17.92	0	1.20e-03
Odd/Even Sweep	10	3.49e-02	–	1.93e-01	–	0	2.22e-01
	20	5.54e-04	5.98	1.73e-03	6.80	0	1.91e-02
	40	7.27e-11	22.86	1.38e-10	23.58	0	2.09e-03
	80	9.87e-17	19.49	5.55e-16	17.92	0	1.20e-03
Random Sweep	10	2.74e-02	–	1.52e-01	–	0	1.81e-01
	20	5.24e-04	5.71	1.52e-03	6.64	0	1.93e-02
	40	7.25e-11	22.79	1.67e-10	23.12	0	2.09e-03
	80	9.98e-17	19.48	5.55e-16	18.20	0	1.20e-03

Case 2. We consider a discontinuous function

$$u(x) = \begin{cases} 1; & 1 \leq x \leq 2 \\ 0; & \text{Otherwise.} \end{cases} \tag{1}$$

Here $m = 0, M = 1$. The numerical results with different sweeping arrays on the whole computational domain are contained in Table 2 and Figure 1. The sweeping procedure successfully enforces the bounds of the approximation. However, since the spectral accuracy is lost due to the Gibbs phenomenon, we do not expect to gain more accuracy after sweeping, as can be seen from Table 2. On the other hand, from Figure 1, the impact of ordering in the sweeping is obvious. The best result is obtained when the regular sweep is used. This is because a change of ordering will break the neighboring relations of the point values, which is very important in the definition of the algorithm. For example, a randomly generated array will put point values in the $u = 0$ region next to point values in $u = 1$ region. When sweeping is applied, it can pollute the values even if it originally stays within bound without sweeping.

Next, we apply an exponential filter $\sigma(\eta) = e^{-\alpha\eta^p}$ with $p = 10$ and $\alpha = -\log(10^{-16})$ to the spectral approximation and study the sweeping method. It is well known that filters can enhance accuracy away from discontinuities. It is to our interest to see if such accuracy is maintained by the sweeping. The results with the exponential filter are presented in Table 3 and Figure 2. The errors in Table 3 are measured on the smooth region of the function and it is easy to see that the approximation maintains accuracy with the regular and odd/even sweep, but lose accuracy if we perform the sweep with random array. The regular sweep is still the best among all the orderings.

From this example, we can see that the regular sweep which maintains the neighboring relation of the point values is the best choice. We also performed numerical studies on sweeping with different starting points, e.g. applying the method to

$$\{v_s, v_{s+1}, \ldots, v_N, v_1, v_2, \ldots v_{s-1}\}$$

where s is different from 1. The results are similar to regular sweep and are omitted. For the rest cases of this example, we will only consider the regular sweep choice.

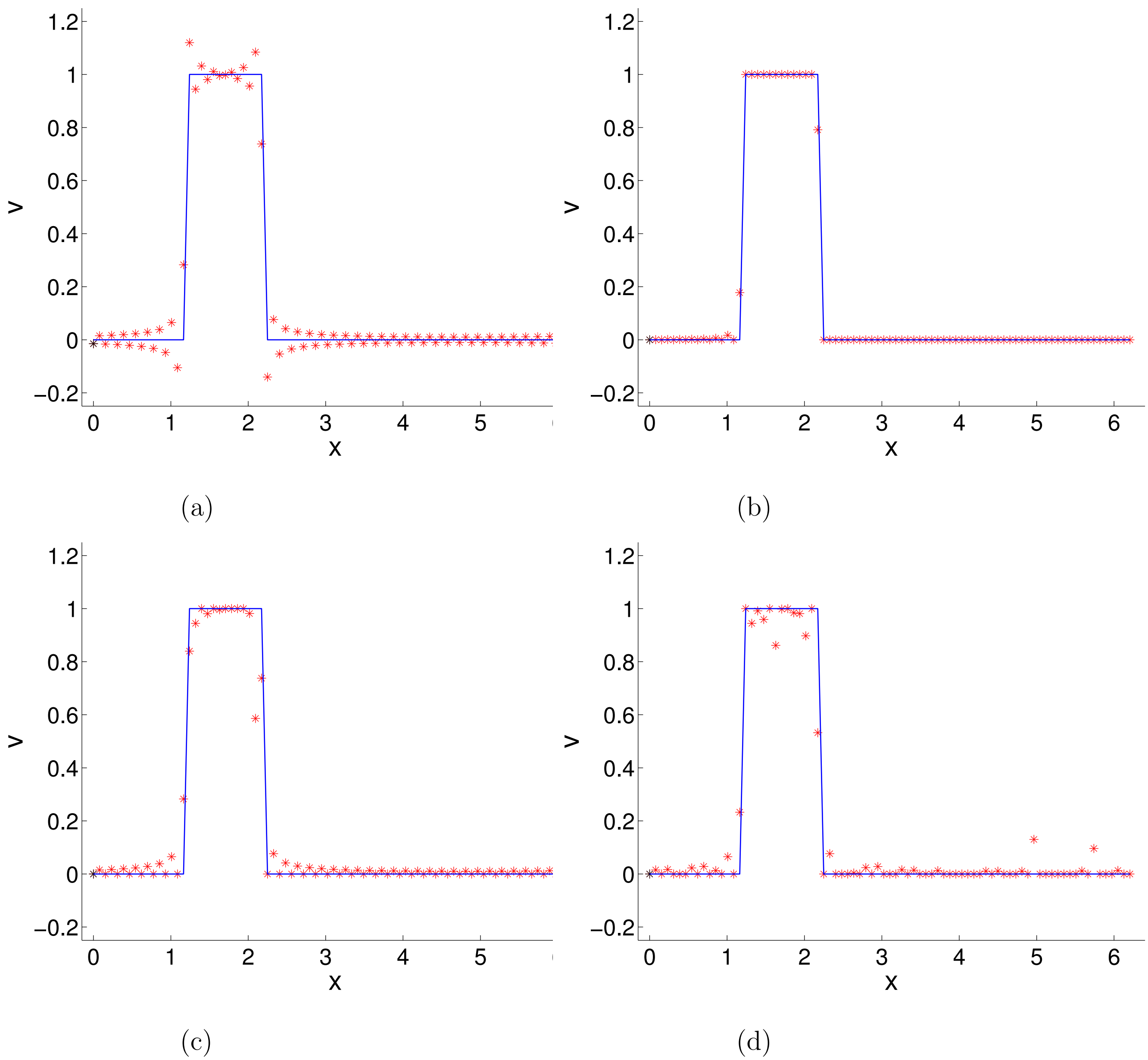
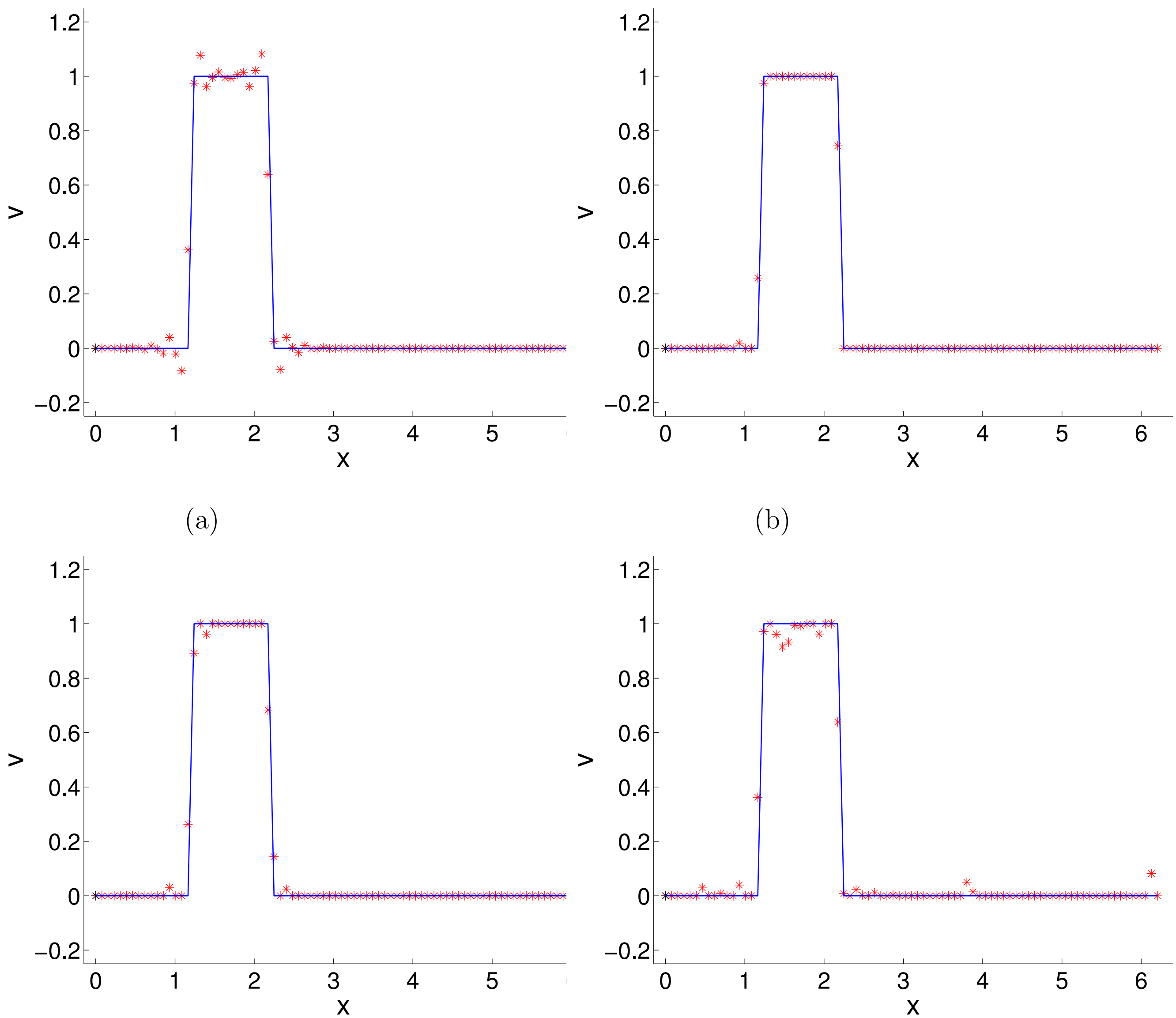


Figure 1: Numerical approximation without filter for Example 3.1, Case 2. (a) No sweeping technique; (b) regular sweep; (c) odd/even sweep; (d) random sweep. $N = 80$. Line: exact solution; dots: numerical solution.

Table 2: Accuracy study for Example 3.1, Case 2 on the computational domain $[0, 2\pi]$, $a = \pi/15$. No exponential filter with different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	3.00e-02	–	2.83e-01	–	-1.41e-01	-1.20e-01
	160	1.67e-02	0.85	3.52e-01	-0.31	-1.34e-01	-1.45e-01
	320	1.26e-02	0.41	7.30e-01	-1.05	-1.33e-01	-1.28e-01
	640	5.43e-03	1.21	3.48e-01	1.07	-1.38e-01	-1.41e-01
	1280	2.64e-03	1.04	2.73e-01	0.35	-1.30e-01	-1.31e-01
Regular Sweep	80	5.15e-03	–	2.09e-01	–	0	0
	160	6.03e-03	-0.23	4.86e-01	-1.22	0	0
	320	5.13e-03	0.23	8.23e-01	-0.76	0	0
	640	1.51e-03	1.76	4.85e-01	0.76	0	0
	1280	3.28e-04	2.20	2.10e-01	1.21	0	0
Odd/Even Sweep	80	2.30e-02	–	4.14e-01	–	0	0
	160	1.19e-02	0.95	3.93e-01	0.08	0	0
	320	1.02e-02	0.22	7.30e-01	-0.89	0	0
	640	3.97e-03	1.36	3.61e-01	1.02	0	0
	1280	1.81e-03	1.13	2.98e-01	0.28	0	0
Random Sweep	80	1.98e-02	–	5.24e-01	–	0	0
	160	1.42e-02	0.48	3.98e-01	0.40	0	0
	320	9.60e-03	0.56	7.30e-01	-0.89	0	0
	640	4.59e-03	1.06	3.48e-01	1.07	0	0
	1280	2.19e-03	1.07	2.73e-01	0.35	0	0



(a) (b)

(c) (d)

Figure 2: Numerical approximation with filter for Example 3.1, Case 2. (a) No sweeping technique; (b) regular sweep; (c) odd/even sweep; (d) random sweep. $N = 80$. Line: exact solution; dots: numerical solution.

Table 3: Accuracy study for Example 3.1, Case 2 on $[0, 2\pi]\backslash\{I_1 \cup I_2\}$, $I_1 = [1.2 - \epsilon, 1.2 + \epsilon]$, $I_2 = [2.2 - \epsilon, 2.2 + \epsilon]$, $\epsilon = 0.1$, $a = \pi/15$, with exponential filter and different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	8.45e-03	–	8.28e-02	–	-8.28e-02	-8.20e-02
	160	2.06e-03	2.04	4.26e-02	0.96	-7.33e-02	-7.33e-02
	320	3.72e-04	2.47	1.53e-02	1.48	-8.46e-02	-8.46e-02
	640	1.85e-05	4.33	1.50e-03	3.35	-7.45e-02	-7.45e-02
	1280	6.66e-08	8.12	2.33e-05	6.01	-8.51e-02	-8.51e-02
Regular Sweep	80	3.05e-04	–	1.87e-02	–	0	0
	160	2.20e-04	0.47	2.56e-02	-0.45	0	0
	320	1.36e-05	4.02	2.48e-03	3.37	0	0
	640	2.94e-07	5.53	1.58e-04	3.97	0	0
	1280	6.45e-10	8.83	4.28e-07	8.53	0	0
Odd/Even Sweep	80	1.24e-03	–	3.83e-02	–	0	0
	160	1.89e-04	2.71	2.59e-02	0.56	0	0
	320	2.71e-05	2.80	7.92e-03	1.71	0	0
	640	7.75e-07	5.13	2.50e-04	4.99	0	0
	1280	1.86e-09	8.70	1.53e-06	7.35	0	0
Random Sweep	80	4.97e-03	–	8.17e-02	–	0	0
	160	3.41e-03	0.54	7.33e-02	0.16	0	0
	320	7.31e-04	2.22	8.46e-02	-0.21	0	0
	640	4.59e-04	0.67	1.41e-01	-0.74	0	0
	1280	3.64e-04	0.33	8.51e-02	0.73	0	0

Case 3. We consider

$$u(x) = \begin{cases} 0; & x < 1 \\ 0.5; & 1 \leq x \leq 3 \\ 1; & \text{Otherwise.} \end{cases} \tag{2}$$

to see the behavior of the sweeping method if there are discontinuities in $u(x)$ that are not equal to the maximum and minimum values. Here, $m = 0$ and $M = 1$. We test this example with and without the exponential filter as in Case 2. The numerical solutions are shown in Figure 3. We can see the sweeping method can retain the bound, but it does not alleviate the oscillations around $u(x) = 0.5$. The error tables are reported in Tables 4 and 5. We can see that the sweeping procedure can retain the accuracy of the original scheme with or without the exponential filter.

Table 4: Accuracy study for Example 3.1, Case 3 on $[0, 2\pi]$, $a = \pi/15$. No exponential filter. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	2.71e-02	–	2.80e-01	–	-1.23e-01	-1.03e-01
	160	2.07e-02	0.39	6.52e-01	-1.22	-1.15e-01	-1.38e-01
	320	9.72e-03	1.09	3.63e-01	0.84	-1.33e-01	-9.36e-02
	640	6.34e-03	0.62	6.55e-01	-0.85	-1.12e-01	-1.40e-01
	1280	2.64e-03	1.26	2.72e-01	1.27	-1.31e-01	-9.50e-02
Regular Sweep	80	1.01e-02	–	2.03e-01	–	0	0
	160	1.02e-02	-0.01	5.14e-01	-1.34	0	0
	320	4.67e-03	1.13	4.84e-01	0.09	0	0
	640	2.63e-03	0.83	5.15e-01	-0.09	0	0
	1280	8.38e-04	1.65	1.82e-01	1.50	0	0

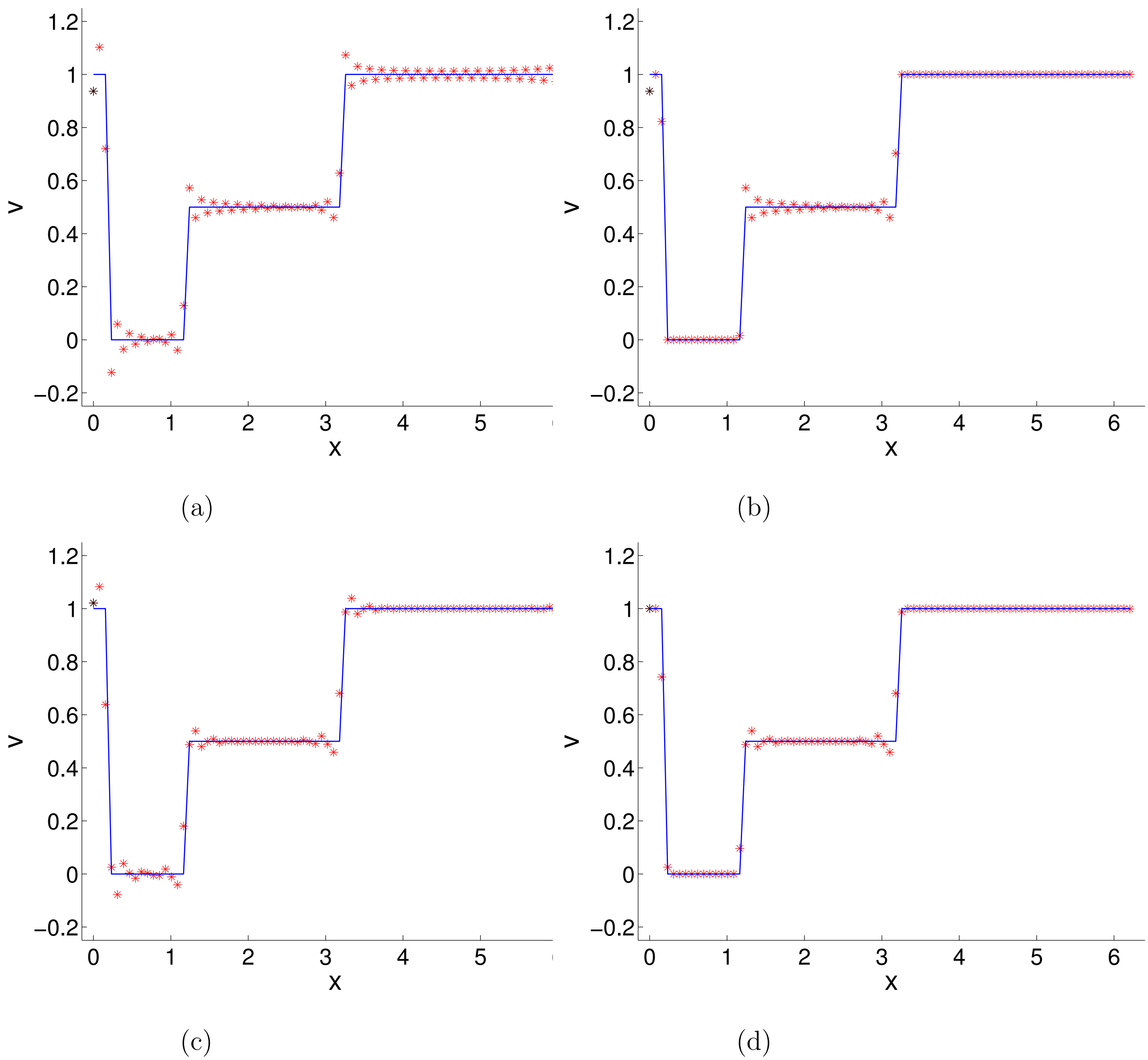


Figure 3: Numerical approximation for Example 3.1, Case 3. (a) No filter, no sweeping technique; (b) no filter, regular sweep; (c) with filter, no sweeping technique; (d) with filter and regular sweep. $N = 80$. Line: exact solution; dots: numerical solution.

Table 5: Accuracy study for Example 3.1, Case 3 on $[0, 2\pi]\backslash\{I_1 \cup I_2 \cup I_3\}$, $I_1 = [0.2 - \epsilon, 0.2 + \epsilon]$, $I_2 = [1.2 - \epsilon, 1.2 + \epsilon]$, $I_3 = [3.2 - \epsilon, 3.2 + \epsilon]$, $\epsilon = 0.1$, $a = \pi/15$, with exponential filter. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	8.35e-03	–	8.27e-02	–	-7.77e-02	-8.27e-02
	160	2.22e-03	1.91	4.26e-02	0.96	-7.33e-02	-6.83e-02
	320	3.37e-04	2.72	1.09e-02	1.97	-7.62e-02	-8.46e-02
	640	1.76e-05	4.26	1.50e-03	2.86	-7.45e-02	-6.72e-02
	1280	5.89e-08	8.22	1.17e-05	7.00	-7.57e-02	-8.51e-02
Regular Sweep	80	1.75e-03	–	3.90e-02	–	0	0
	160	7.33e-04	1.26	2.56e-02	0.61	0	0
	320	1.08e-04	2.76	7.64e-03	1.74	0	0
	640	5.56e-06	4.28	7.48e-04	3.35	0	0
	1280	1.29e-08	8.75	3.56e-06	7.72	0	0

Case 4. We consider a function which contains both smooth region and discontinuities

$$u(x) = \begin{cases} e^{-10(x-\frac{\pi}{2})^4} + 2.0; & \frac{1}{4}\pi \le x \le \frac{3}{4}\pi \\ 0; & \text{Otherwise.} \end{cases} \tag{3}$$

It is easy to see that $m = 0$, $M = 3$ here. We apply the exponential filter to the spectral method and compare the performance of the scheme in a smooth part of the domain $[1.2, 2.4]$ in Table 6. We can verify the good performance of the sweeping method, i.e. it does not affect the overall accuracy of the scheme, while being able to preserve the maximum principle for the point values.

Table 6: Accuracy study for Example 3.1, Case 4 on $[1.2, 2.4]$, $a = \pi/15$, with exponential filter and different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	2.14e-02	–	8.05e-02	–	-1.70e-01	-5.67e-03
	160	3.88e-03	2.46	2.76e-02	1.54	-1.49e-01	-3.64e-05
	320	1.87e-04	4.37	2.85e-03	3.28	-1.72e-01	9.82e-10
	640	1.63e-06	6.84	8.25e-05	5.11	-1.51e-01	1.99e-09
	1280	1.63e-10	13.29	1.69e-08	12.25	-1.72e-01	3.69e-14
Regular Sweep	80	2.05e-02	–	8.05e-02	–	0	0
	160	3.88e-03	2.40	2.76e-02	1.54	0	0
	320	1.87e-04	4.37	2.85e-03	3.28	0	9.82e-10
	640	1.63e-06	6.84	8.25e-05	5.11	0	1.99e-09
	1280	1.63e-10	13.29	1.69e-08	12.25	0	3.69e-14

Now we turn our attention to 2D case. Here, we want to obtain the approximation of $u(x-a, y-b)$, assuming periodic conditions on the domain $[0, 2\pi] \times [0, 2\pi]$. Without loss of generality, a uniform grid with $N \times N$ equally spaced points is used. Suppose we are given a set of points $\{u_{i,j} = u(x_i, y_j)\}_{i,j=1}^{N}$ at uniform points $x_i = \frac{2\pi}{N} i$, $y_j = \frac{2\pi}{N} j$, $\quad i, j = 0, \ldots, N-1$, and are interested in finding the values of $\{v_{i,j} = u(x_i - a, y_j - b)\}_{i,j=1}^{N}$ at the shifted points by a Fourier spectral method. The approximation algorithms detail as follows:

1. Start with a set of points $\{u_{i,j}\}_{i,j=1}^{N}$.
2. Use FFT to reconstruct the interpolating function $u_N = \sum_{|k_1| \leq N/2, |k_2| \leq N/2} \hat{u}_{k_1,k_2} e^{ik_1 x + ik_2 y}$.
3. Evaluate $v_{i,j} = u_N(x_i - a, y_j - b)$ by FFT.

In two dimensional examples, a conservative scheme maintains the conserved quantities on the whole computational domain, in particular here $\sum_{i,j=1}^{N} \frac{1}{N^2} v_{i,j}$ is a constant. We can then treat $\{v_{i,j}\}_{i,j=1}^{N}$ as a one dimensional matrix, and apply the sweeping technique. As shown in the 1D case, it is important to maintain the neighboring relations of the points. For 2D cases, a subtle point arises since there are multiple ways to rewrite a 2D matrix into 1D matrix while keeping the neighboring relations. It is to our interest to compare a few viable approaches in this section.

The ordering we consider here includes

- sweep I, i.e. apply the procedure to the sequence $\{v_{1,1}, \ldots v_{N,1}, v_{N,2}, \ldots, v_{1,2}, v_{1,3} \ldots\}$, which is made up of connected mesh cells column by column, i.e. for each column $j = 1, 2, ..., N$, we perform the sweep procedure with order $i = 1, 2, ..., N$ when j is odd while with order $i = N, N-1, ..., 1$ when j is even;
- sweep II, i.e. apply the procedure to the sequence $\{v_{1,1}, \ldots v_{1,N}, v_{2,N}, \ldots, v_{2,1}, v_{3,1} \ldots\}$, which is made up of the connected mesh cells row by row, i.e. for each row $i = 1, 2, ..., N$, we perform the sweep procedure with order $j = 1, 2, ..., N$ when i is odd while with order $j = N, N-1, ..., 1$ when i is even.

There are many other possible ordering of the points that are not considered in this paper. Similar issues also appear in Section 4 for unstructured meshes, for which we will just use the natural order from the mesh generator. We point out that a robust algorithm should not be overly sensitive to the ordering of the points/cells.

Example 3.2 *We let $a = \pi/15, b = \pi/25$, and test the following functions of $u(x,y)$ with different degrees of smoothness.*

Case 1. $u(x,y) = e^{-5[(x-\pi)^2+(y-\pi)^2]}$. This is a smooth function with $m = 0$, $M = 1$. We present the results without and with sweeping method in Table 7, where we can easily see that the spectral method with sweeping technique can achieve spectral accuracy and bound-preserving property simultaneously.

Table 7: Accuracy study for Example 3.2, Case 1 on the computational domain $[0,2\pi]\times[0,2\pi]$, $a = \pi/15$, $b = \pi/25$, No exponential filter; with different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	10	3.43e-03	–	2.66e-03	–	-3.06e-02	3.56e-01
	20	2.65e-05	7.02	3.13e-05	6.41	-3.94e-04	2.36e-02
	40	1.81e-12	23.80	1.17e-12	24.67	-3.09e-11	1.40e-02
	80	1.90e-18	19.86	1.35e-18	19.73	-1.41e-16	1.64e-03
Sweep I	10	2.38e-03	–	2.17e-03	–	0	3.56e-01
	20	2.12e-05	6.81	2.52e-05	6.43	0	2.36e-02
	40	1.69e-12	23.58	1.10e-12	24.45	0	1.40e-02
	80	1.91e-18	19.76	1.32e-18	19.67	0	1.64e-03
Sweep II	10	1.87e-03	–	2.24e-03	–	0	3.56e-01
	20	2.13e-05	6.46	2.55e-05	6.46	0	2.36e-02
	40	1.68e-12	23.60	1.10e-12	24.47	0	1.40e-02
	80	1.88e-18	19.77	1.39e-18	19.59	0	1.64e-03

Case 2. We test the discontinuous function

$$u(x,y) = \begin{cases} 1; & x \in [1,2]; \\ 0; & \text{Otherwise}, \end{cases} \tag{4}$$

where $m = 0$ and $M = 1$. This function only has $x-$dependence, and is used to compare the results from the two sweeping orderings. The numerical solutions without filters are presented in Table 8 and Figure 4. Because of the discontinuity, we do not observe the spectral accuracy with and without the sweeping methods. We then apply the exponential filter and the results are given in Table 9 and Figure 5, in which we can see high order of accuracy is retained in the smooth region. Sweep I offers better accuracy compared with sweep II, but the convergence order is comparable.

Table 8: Accuracy study for Example 3.2, Case 2 on the computational domain $[0, 2\pi]\times[0, 2\pi]$, $a = \pi/15$, $b = \pi/25$, without exponential filter; with different sweeping orderings. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	3.70e-04	–	2.83e-01	–	-1.41e-01	-1.20e-01
	160	1.04e-04	1.83	3.52e-01	-0.31	-1.34e-01	-1.45e-01
	320	3.93e-05	1.40	7.30e-01	-1.05	-1.33e-01	-1.28e-01
	640	8.48e-06	2.12	3.48e-01	1.07	-1.38e-01	-1.41e-01
Sweep I	80	1.31e-04	–	4.37e-01	–	0	0
	160	4.06e-05	1.69	5.37e-01	-0.30	0	0
	320	1.69e-05	1.26	8.79e-01	-0.71	0	0
	640	2.55e-06	2.73	5.34e-01	0.72	0	0
Sweep II	80	1.52e-04	–	1.0e-00	–	0	0
	160	7.72e-05	0.98	1.0e-00	0	0	0
	320	1.97e-05	1.97	1.0e-00	0	0	0
	640	4.87e-06	2.02	1.0e-00	0	0	0

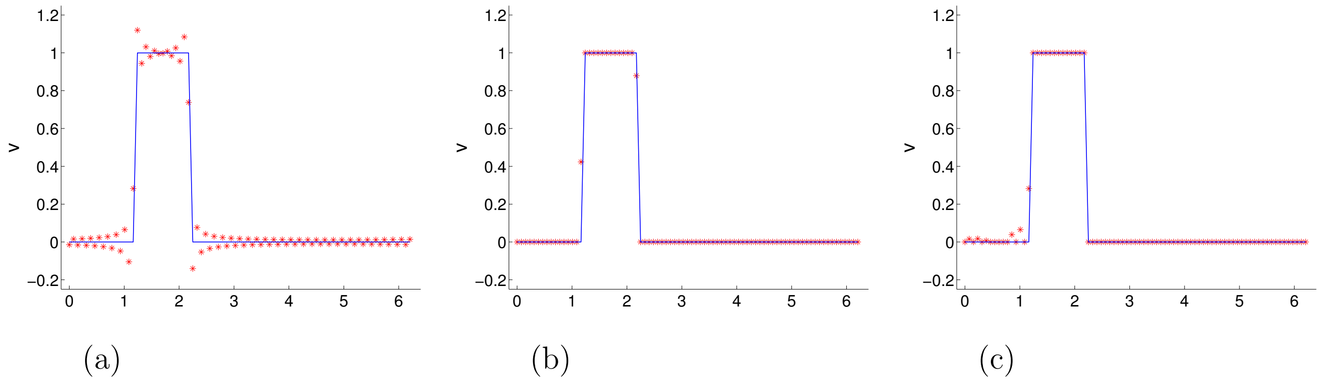


Figure 4: Numerical approximation along $y = x$ without filter for Case 2. (a) No sweeping technique; (b) with sweep array I; (c) with sweep array II. $N = 80$. Line: exact solution; dots: numerical solution.

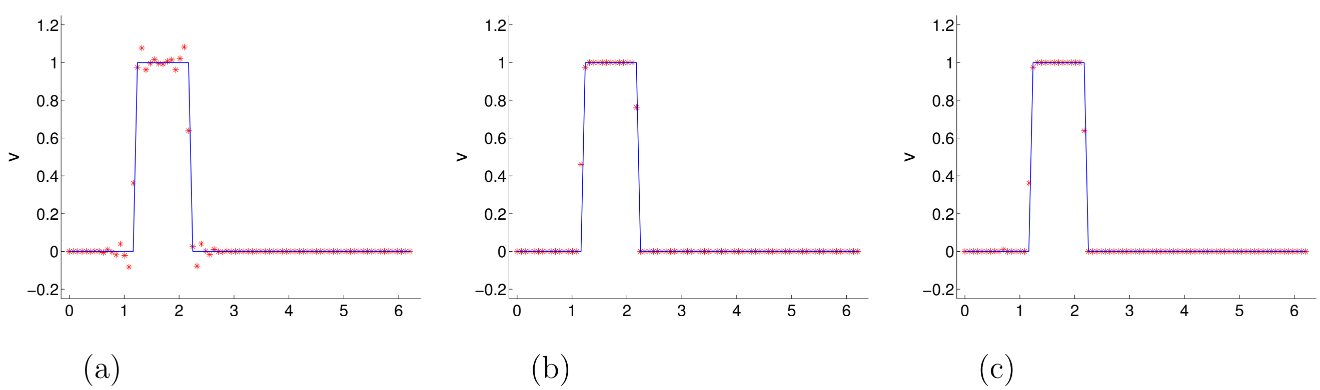


Figure 5: Numerical approximation along $y = x$ with filter for Case 2. (a) No sweeping technique; (b) with sweep array I; (c) with sweep array II. $N = 80$. Line: exact solution; dots: numerical solution.

Table 9: Accuracy study for Example 3.2, Case 2 on the computational domain $[0, 2\pi] \times [0, 2\pi] \backslash I,\ I = [1 + \pi/15 - \epsilon, 2 + \pi/15 + \epsilon] \times [0, 2\pi],\ \epsilon = 0.15,\ a = \pi/15,\ b = \pi/25$. With exponential filter; with different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	4.53e-03	–	3.99e-02	–	-8.28e-02	-8.20e-02
	160	1.05e-03	2.11	1.62e-02	1.30	-7.33e-02	-7.33e-02
	320	9.37e-05	3.49	2.80e-03	2.53	-8.46e-02	-8.46e-02
	640	1.40e-06	6.06	1.45e-04	4.27	-7.45e-02	-7.45e-02
Sweep I	80	3.97e-06	–	1.87e-02	–	0	0
	160	3.25e-07	3.61	6.10e-03	1.62	0	0
	320	1.74e-08	4.22	1.01e-03	2.59	0	0
	640	5.07e-11	8.42	1.51e-05	6.06	0	0
Sweep II	80	3.22e-04	–	3.93e-02	–	0	0
	160	5.23e-05	2.62	1.09e-02	1.85	0	0
	320	5.58e-06	3.23	1.41e-03	2.95	0	0
	640	3.25e-08	7.42	4.30e-05	5.04	0	0

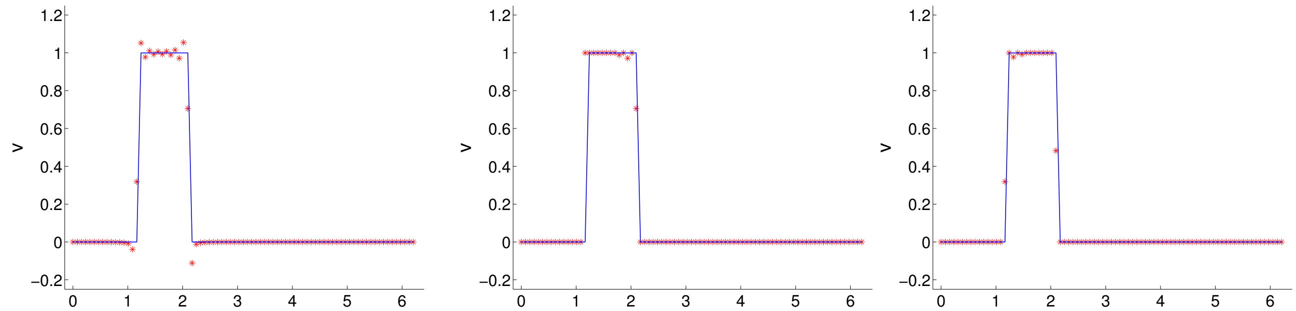
Case 3. We test the discontinuous function

$$u(x,y) = \begin{cases} 1; & (x,y) \in [1,2] \times [1,2] \\ 0; & \text{Otherwise,} \end{cases} \tag{5}$$

where $m = 0$ and $M = 1$. The numerical results with and without filters are presented in Tables 10-11 and Figures 6-7. Similar conclusions to case 2 can be made, except that here there is no particular winner between sweep I and II.sweeping techniques are comparable.

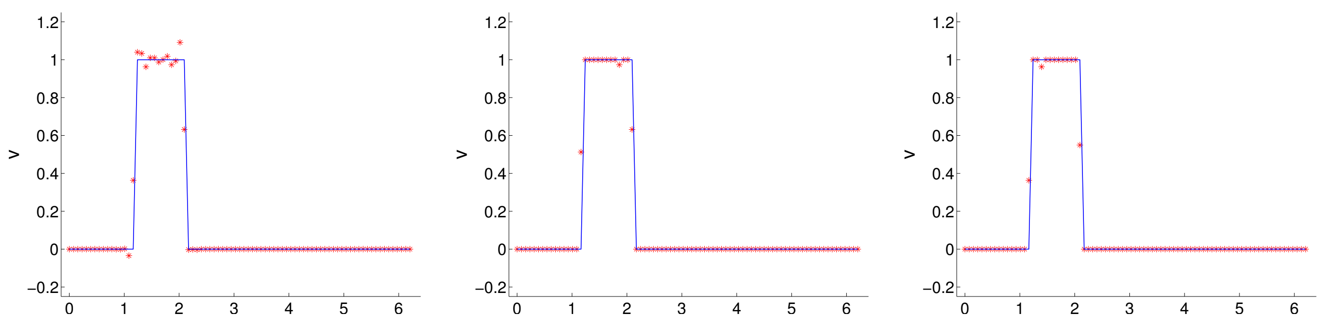
Table 10: Accuracy study for Example 3.2, Case 3 on the computational domain $[0, 2\pi] \times [0, 2\pi]$, $a = \pi/15$, $b = \pi/25$, No exponential filter; with different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	8.77e-04	–	5.20e-01	–	-1.69e-01	-2.63e-01
	160	2.39e-04	1.88	4.77e-01	0.12	-1.49e-01	-2.77e-01
	320	2.18e-04	0.13	8.36e-01	-0.81	-1.63e-01	-2.84e-01
	640	2.14e-04	0.03	9.69e-01	-0.21	-1.51e-01	-2.53e-01
Sweep I	80	9.15e-04	–	1.0	–	0	0
	160	2.99e-04	1.61	1.0	0	0	0
	320	2.79e-04	0.10	1.0	0	0	0
	640	2.26e-04	0.30	1.0	0	0	0
Sweep II	80	8.60e-04	–	1.0	–	0	0
	160	5.34e-04	0.69	1.0	0	0	0
	320	2.34e-04	1.19	1.0	0	0	0
	640	2.00e-04	0.23	1.0	0	0	0



(a) (b) (c)

Figure 6: Numerical approximation along $y = x$ without filter for Case 3. (a) No sweeping technique; (b) with sweep array I; (c) with sweep array II. $N = 80$. Line: exact solution; dots: numerical solution.



(a) (b) (c)

Figure 7: Numerical approximation along $y = x$ without filter for Case 3. (a) No sweeping technique; (b) with sweep I; (c) with sweep II. $N = 80$. Line: exact solution; dots: numerical solution.

Table 11: Accuracy study for Example 3.2, Case 3 on the computational domain $I_1 \cup I_2 \cup I_3 \cup I_4$, $I_1 = [0, 1 + \pi/15 - \epsilon] \times [0, 2\pi]$, $I_2 = [2 + \pi/15 + \epsilon, 2\pi] \times [0, 2\pi]$, $I_3 = [1 + \pi/15, 2 + \pi/15] \times [0, 1 + \pi/25 - \epsilon]$, $I_4 = [1 + \pi/15, 2 + \pi/15] \times [2 + \pi/25, 2\pi]$, $I_5 = [1 + \pi/15 + \epsilon, 2 + \pi/15 - \epsilon] \times [1 + \pi/25 + \epsilon, 2 + \pi/25 - \epsilon]$, $\epsilon = 0.15$, $a = \pi/15$, $b = \pi/25$, with exponential filter; with different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	1.15e-03	–	7.75e-02	–	-8.85e-02	-1.56e-01
	160	2.67e-04	2.11	3.65e-02	1.09	-9.99e-02	-1.73e-01
	320	2.09e-05	3.68	5.19e-03	2.81	-9.00e-02	-1.54e-01
	640	3.38e-07	5.95	2.01e-04	4.69	-1.04e-01	-1.79e-01
Sweep I	80	1.87e-04	–	7.75e-02	–	0	0
	160	2.02e-05	3.21	1.94e-02	2.00	0	0
	320	1.30e-06	3.96	3.80e-03	2.35	0	0
	640	1.45e-08	6.49	8.22e-05	5.53	0	0
Sweep II	80	1.79e-04	–	7.73e-02	–	0	0
	160	2.12e-05	3.08	1.80e-02	2.10	0	0
	320	1.76e-06	3.59	4.45e-03	2.02	0	0
	640	5.04e-09	8.45	4.68e-05	6.57	0	0

Case 4. In this case, we test the discontinuous function

$$u(x) = \begin{cases} 1; & (x, y) \in (0, 3) \times (0, 3); \\ 0.5; & (x, y) \in [3, 5) \times [3, 5); \\ 0; & \text{Otherwise.} \end{cases} \tag{6}$$

where $m = 0$ and $M = 1$. Similarly as the previous cases, we test the method with and without exponential filter. The numerical results without filters are given in Table 12 and Figure 8, in which no accuracy is attained. The numerical results with exponential filter are presented in Table 13 and Figure 9.

Table 12: Accuracy study for Example 3.2, Case 4 on the domain $[0, 2\pi] \times [0, 2\pi]$, $a = \pi/15$, $b = \pi/25$, No exponential filter; With different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	3.92e-03	–	9.05e-01	–	-2.03e-01	-3.02e-01
	160	3.22e-03	0.28	8.79e-01	0.04	-1.83e-01	-2.79e-01
	320	8.13e-04	1.99	8.37e-01	0.07	-2.29e-01	-2.87e-01
	640	8.45e-04	-0.06	9.59e-01	-0.20	-1.86e-01	-2.51e-01
Sweep I	80	4.52e-03	–	1.0	–	0	0
	160	3.29e-03	0.46	1.0	0	0	0
	320	8.93e-04	1.88	1.0	0	0	0
	640	8.56e-04	0.06	1.0	0	0	0
Sweep II	80	3.90e-03	–	1.0	–	0	0
	160	3.14e-03	0.31	1.0	0	0	0
	320	9.44e-04	1.73	1.0	0	0	0
	640	7.94e-04	0.25	1.0	0	0	0

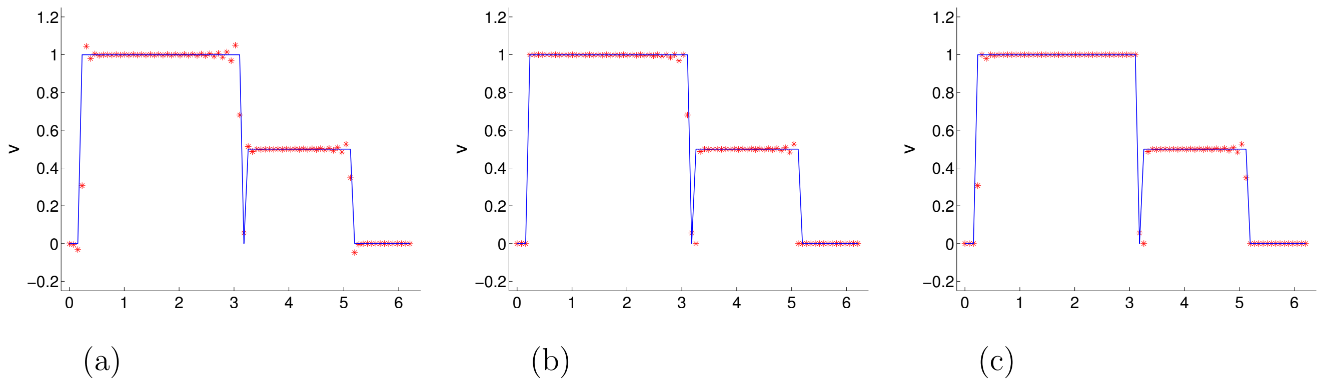


Figure 8: Numerical approximation without filter for Case 4. (a) No sweeping technique; (b) with sweep I; (c) with sweep II. $N = 80$. Line: exact solution; dots: numerical solution.

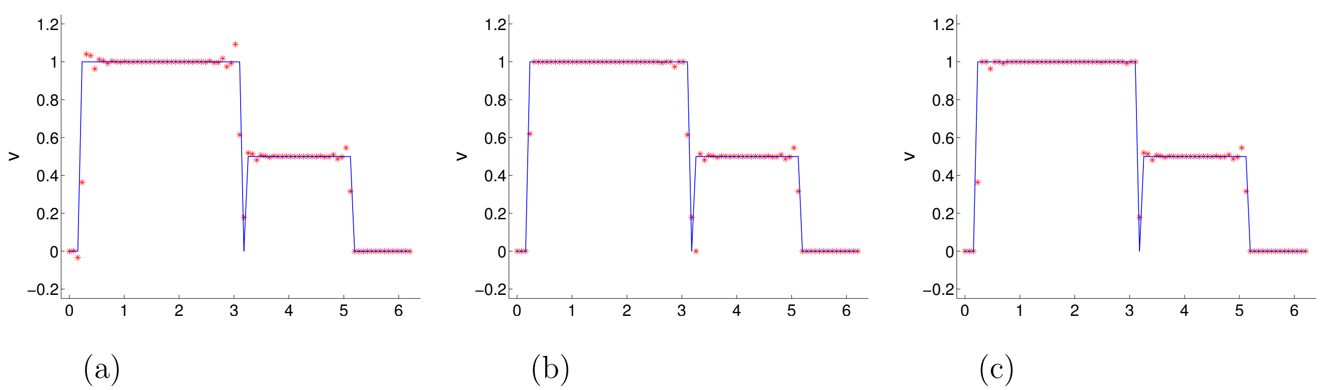


Figure 9: Numerical approximation along $y = x$ with filter for Case 4. (a) No sweeping technique; (b) with sweep I; (c) with sweep II. $N = 80$. Line: exact solution; dots: numerical solution.

Table 13: Accuracy study for Example 3.2, Case 4 on the smooth part of the computational domain, i.e. $\{I_1 \cup I_2 \cup I_3 \cup I_4 \cup I_5 \cup I_6\}$, where $I_1 = [0, 3+\pi/15-\epsilon] \times [3+\pi/25+\epsilon, 2\pi]$, $I_2 = [3.2+\pi/15+\epsilon, 2\pi] \times [0, 3+\pi/25-\epsilon]$, $I_3 = [5+\pi/15+\epsilon, 2\pi] \times [3+\pi/25-\epsilon, 5+\pi/25+\epsilon]$, $I_4 = [3+\pi/15-\epsilon, 2\pi] \times [5+\pi/25+\epsilon, 2\pi]$, $I_5 = [\pi/15+\epsilon, 3+\pi/15-\epsilon] \times [\pi/25+\epsilon, 3+\pi/25-\epsilon]$, $I_6 = [3+\pi/15+\epsilon, 5+\pi/15-\epsilon] \times [3+\pi/25+\epsilon, 5+\pi/25-\epsilon]$, $\epsilon = 0.15$, $a = \pi/15$, $b = \pi/25$, with exponential filter; with different sweeping orderings. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	6.12e-03	–	1.51e-01	–	-1.21e-01	-1.60e-01
	160	1.12e-03	2.45	3.65e-02	2.05	-1.37e-01	-1.73e-01
	320	8.86e-05	3.66	5.19e-03	2.81	-1.14e-01	-1.54e-01
	640	1.58e-06	5.81	2.01e-04	4.69	-1.34e-01	-1.79e-01
Sweep I	80	1.75e-03	–	8.05e-02	–	0	0
	160	2.50e-04	2.81	2.10e-02	1.94	0	0
	320	2.18e-05	3.52	5.19e-03	2.02	0	0
	640	1.90e-07	6.84	1.00e-04	5.70	0	0
Sweep II	80	1.66e-03	–	3.32e-01	–	0	0
	160	2.08e-04	3.00	2.10e-02	3.98	0	0
	320	1.99e-05	3.39	5.19e-03	2.02	0	0
	640	1.90e-07	6.71	1.00e-04	5.70	0	0

Case 5. We test the function

$$u(x) = \begin{cases} e^{-10((x-\frac{\pi}{2})^4+(y-\frac{\pi}{2})^4)} + 2.0; & (x,y) \in (\frac{1}{4}\pi, \frac{3}{4}\pi) \times (\frac{1}{4}\pi, \frac{3}{4}\pi); \\ 0; & \text{Otherwise}, \end{cases} \tag{7}$$

where $m = 0$ and $M = 3$. The numerical results without any special treatment are give in Table 14 and Figure 10, in which no accuracy is attained. The numerical results with exponential filter are presented in Table 15 and Figure 11, and we recover the high order convergence rate.

Table 14: Accuracy study for Example 3.2, Case 5 on the domain $[0, 2\pi] \times [0, 2\pi]$, $a = \pi/15$, $b = \pi/25$, with exponential filter; with different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	4.34e-03	–	1.81e-00	–	-3.01e-01	-1.17e-01
	160	6.90e-04	2.65	9.50e-01	0.93	-3.02e-01	- 5.16e-02
	320	6.70e-04	0.04	1.68e-00	-0.82	-3.15e-01	-2.95e-02
	640	6.69e-04	0.00	1.81e-00	-0.11	-3.04e-01	-1.20e-02
Sweep I	80	4.94e-03	–	2.03e-00	–	0	0
	160	9.73e-04	2.34	2.01e-00	0.01	0	0
	320	8.77e-04	0.15	2.02e-00	-0.01	0	0
	640	7.13e-04	0.30	2.02e-00	0	0	0
Sweep II	80	4.77e-03	–	2.03e-00	–	0	0
	160	8.82e-04	2.44	2.02e-00	0.01	0	0
	320	8.16e-04	0.11	2.02e-00	0	0	0
	640	6.97e-04	0.23	2.02e-00	0	0	0

Table 15: Accuracy study for Example 3.2, Case 3 on the domain $\{I_1 \cup I_2 \cup I_3 \cup I_4\}$, $I_1 = [0, \frac{19\pi}{60} - \epsilon] \times [0, 2\pi]$, $I_2 = [\frac{49\pi}{60} + \epsilon, 2\pi] \times [0, 2\pi]$, $I_3 = [\frac{19\pi}{60}, \frac{49\pi}{60}] \times [0, \frac{29\pi}{100} - \epsilon]$, $I_4 = [\frac{19\pi}{60}, \frac{49\pi}{60}] \times [\frac{79\pi}{100} + \epsilon, 2\pi]$, $\epsilon = 0.15$, $a = \pi/15$, $b = \pi/25$, with exponential filter; with different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	2.78e-03	–	1.47e-01	–	-1.77e-01	-1.18e-02
	160	5.44e-04	2.35	4.31e-02	1.77	-2.00e-01	-6.11e-05
	320	4.35e-05	3.64	5.95e-03	2.86	-1.80e-01	1.97e-08
	640	5.53e-07	6.30	1.65e-04	5.17	-2.08e-01	2.00e-09
Sweep I	80	1.97e-04	–	9.06e-02	–	0	0
	160	2.84e-05	2.79	2.19e-02	2.05	0	0
	320	2.26e-06	3.65	5.20e-03	2.07	0	1.97e-08
	640	4.88e-08	5.53	1.65e-04	4.98	0	2.00e-09
Sweep II	80	1.65e-04	–	8.43e-02	–	0	0
	160	2.73e-05	2.60	2.38e-02	1.82	0	0
	320	2.89e-06	3.24	3.00e-03	2.99	0	1.97e-08
	640	1.69e-08	7.42	9.37e-05	5.00	0	2.00e-09

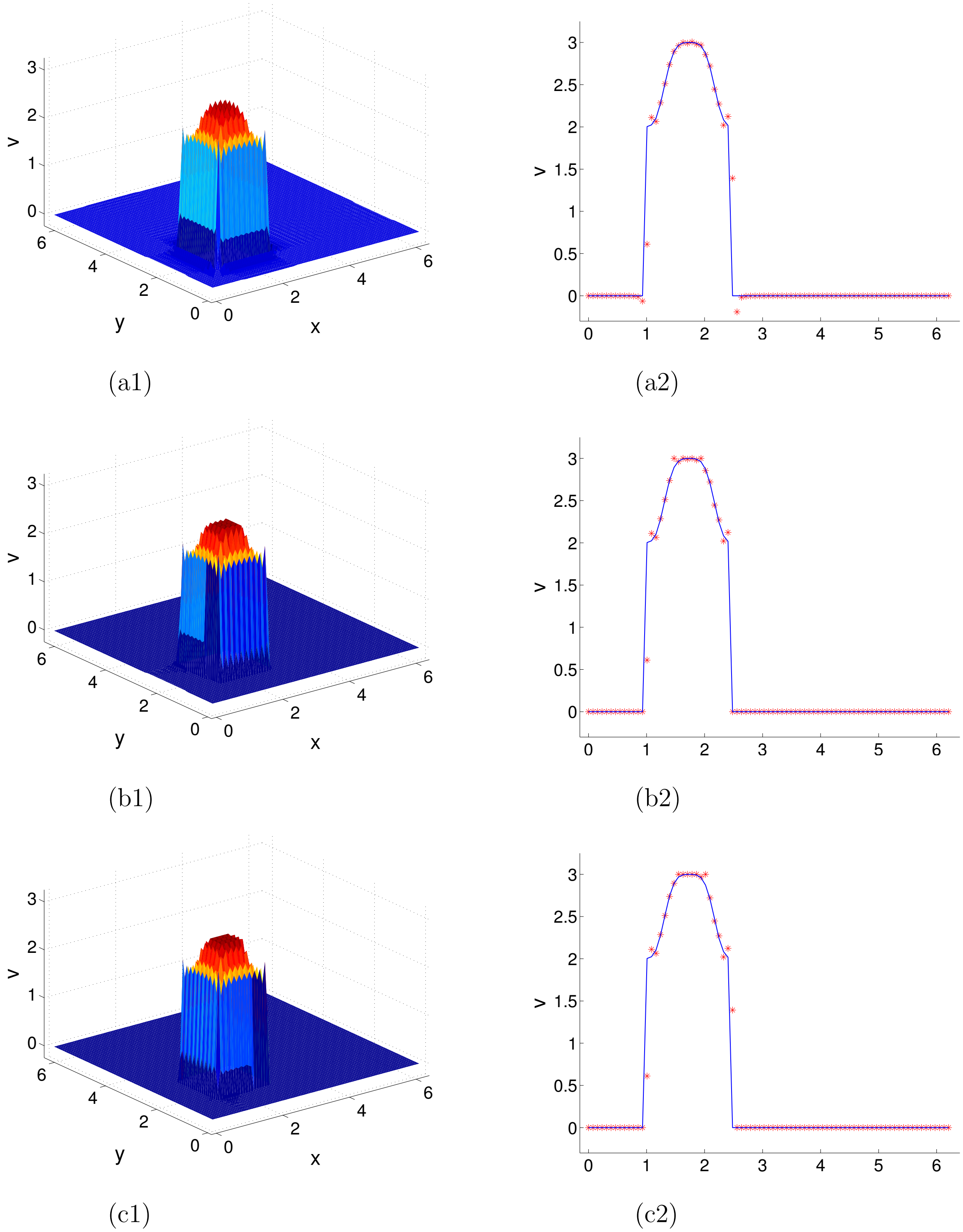


Figure 10: Numerical approximation without filter for Case 4. (a1-c1) Numerical solution on the computational domain; (a2-c2) Numerical solution along the line $y = x$; Line: exact solution; dots: numerical solution. (a1-a2) No sweeping technique; (b1-b2) with sweep I; (c1-c2) with sweep II. $N = 80$.

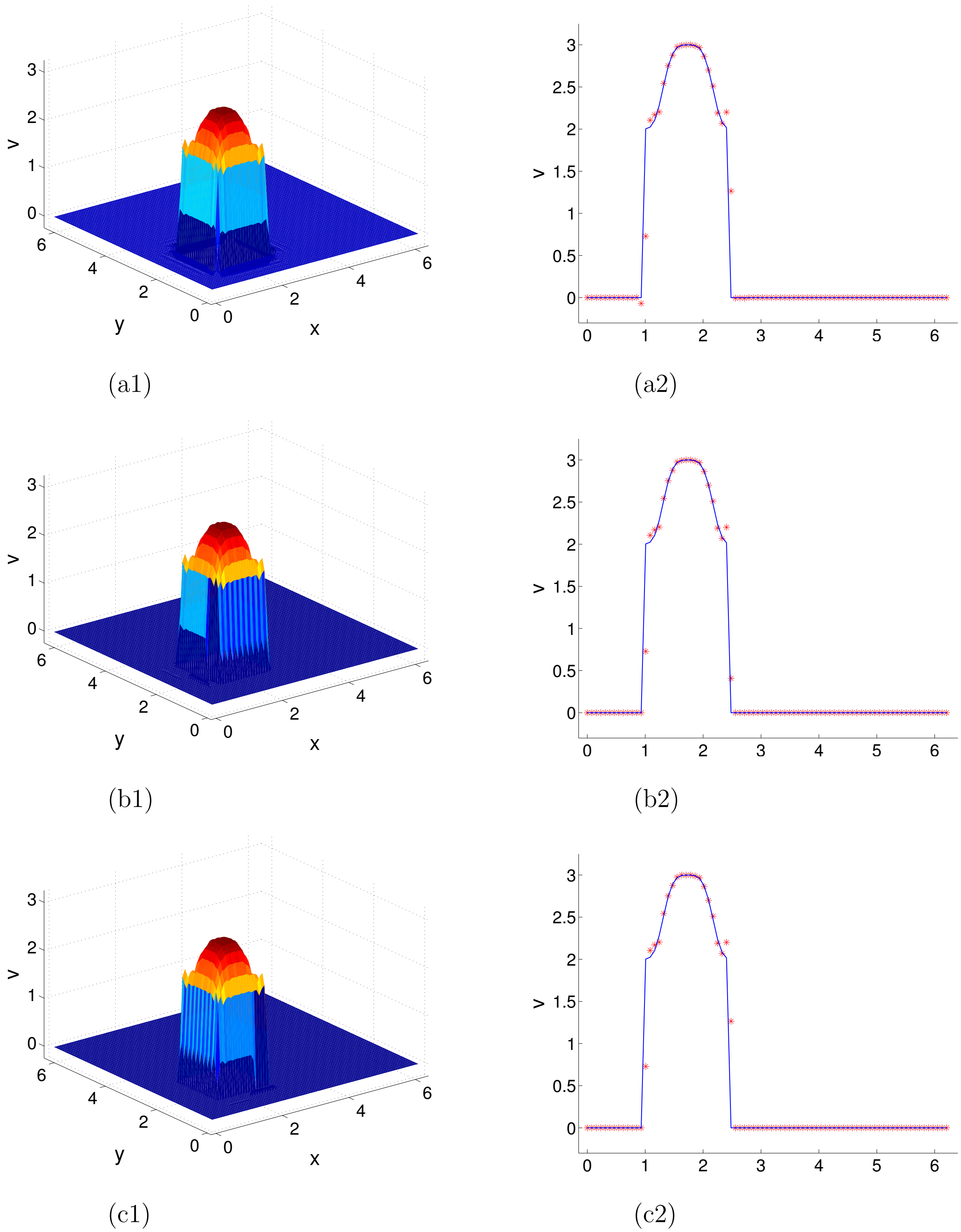


Figure 11: Numerical approximation with filter for Case 4. (a1-c1) Numerical solution on the computational domain; (a2-c2) Numerical solution along the line $y = x$; Line: exact solution; dots: numerical solution. (a1-a2) No sweeping technique; (b1-b2) with sweep I; (c1-c2) with sweep II. $N = 80$.

4 PDE examples

In this section, we provide computational results for solving linear and nonlinear PDEs. We first consider a semi-Lagrangian spectral method in Section 4.1, and then high order finite difference and finite volume schemes in Section 4.2. There is no CFL constraint for semi-Lagrangian spectral method. As for finite difference and finite volume schemes, CFL numbers are presented in the corresponding examples and we do not observe any CFL sacrifice compared with the original high order schemes without the sweeping technique in all examples.

4.1 A semi-Lagrangian spectral method for linear transport equations

In this subsection, we employ a semi-Lagrangian spectral method to solve 1D and 2D linear transport equations. The algorithm is the same as the approximation framework described in Section 3 but with many steps in time evolution instead of one-step approximation. Below, we review the semi-Lagrangian spectral method for the 1D transport equation

$$u_t + cu_x = 0 \tag{8}$$

with periodic boundary conditions. The computation domain $[0, 2\pi]$ is again uniformly partitioned into N points, and then in each time step update, we use the fact that

$$\begin{aligned} u(x, t^{n+1}) &= u(x, t^n + \Delta t_n), \\ &= u(x - \Delta t, t^n), \\ &= u(x - c\Delta x, t^n), \end{aligned}$$

and by the method in Section 3 with $a = c\Delta x$ to obtain the values of $u(x_j, t^{n+1})$. The discussion is similar for 2D transport equation $u_t + c_1 u_x + c_2 u_y = 0$. In the following examples, we will apply the sweeping technique during each time step and test the performance of the method under the influence of such repeated sweepings.

Example 4.1 $u_t + c\, u_x = 0$ *on* $[0, 2\pi]$ *with* $c = 2$ *and periodic boundary condition.*

We check the proposed method with different initial data. The final time is $T = 0.5$, the time step is chosen as $\Delta t = 0.0025$. This means 200 sweeps will be used when t reaches T.

Case 1. $u_0(x) = e^{-a(x-\pi)^2}$ with $a = 5$. The setup is identical to Case 1 in Example 3.1. $m = 0$ and $M = 1$. The numerical error without and with sweeping method are presented in Table 16. We can clearly observe that the spectral method with sweeping technique can achieve spectral accuracy and bound-preserving property simultaneously.

Table 16: Accuracy study for Example 4.1, Case 1 on the computational domain $[0, 2\pi]$, $T = 0.5$. No exponential filter. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	10	3.24e-02	–	1.10e-01	–	-3.00e-02	2.08e-01
	20	6.06e-04	5.74	1.50e-03	6.20	-3.75e-04	1.26e-02
	40	8.19e-11	22.82	1.51e-10	23.24	-3.34e-11	7.55e-05
	80	3.73e-15	14.42	2.10e-14	12.81	-1.01e-15	4.60e-03
Regular Sweep	10	2.73e-02	–	1.47e-01	–	0	2.44e-01
	20	5.74e-04	5.57	1.79e-03	6.36	0	1.29e-02
	40	8.46e-11	22.69	1.60e-10	23.42	0	7.55e-05
	80	3.75e-15	14.46	2.09e-14	12.90	0	4.60e-03

Case 2. We consider a discontinuous function

$$u_0(x) = \begin{cases} 1; & 1 \le x \le 2 \\ 0; & \text{Otherwise.} \end{cases} \tag{9}$$

which corresponds to the one-step problem as Example 3.1 Case 2. $m = 0$ and $M = 1$. The numerical results at $T = 0.5$ on the entire domain $[0, 2\pi]$ are given in Table 17 and Figure 18. As expected, we cannot obtain spectral accuracy because of the discontinuity in the initial condition. We then apply the exponential filter and study the accuracy on the region which exclude the neighborhood of discontinuity. In Table 18, we can see that with the sweeping procedure, the scheme maintains accuracy.

Table 17: Accuracy study for Example 4.1, Case 2 on the computational domain $[0, 2\pi]$, $T = 0.5$. No exponential filter. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	1.19e-02	–	8.86e-02	–	-6.93e-02	-6.08e-02
	160	1.87e-02	-0.65	6.37e-01	-2.85	-1.34e-01	-1.46e-01
	320	3.22e-03	2.54	6.87e-02	3.21	-5.58e-02	-5.40e-02
	640	3.24e-04	3.31	1.31e-02	2.39	-1.23e-02	-1.25e-02
	1280	1.19e-03	-1.88	9.70e-02	-2.89	-7.19e-02	-7.25e-02
Regular Sweep	80	1.22e-02	–	2.35e-01	–	0	0
	160	8.58e-03	0.51	6.59e-01	-1.49	0	0
	320	3.44e-03	1.32	3.58e-01	0.88	0	0
	640	1.67e-03	1.04	3.49e-01	0.04	0	0
	1280	7.00e-04	1.25	2.16e-01	0.69	0	0

Table 18: Accuracy study for Example 4.1, Case 2 on $[0, 2\pi]\backslash\{I_1 \cup I_2\}$, $I_1 = [2.0 - \epsilon, 2.0 + \epsilon]$, $I_2 = [3.0 - \epsilon, 3.0 + \epsilon]$, $\epsilon = 0.15$, $T = 0.5$. with exponential filter. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	1.09e-02	–	8.92e-02	–	-8.92e-02	-7.90e-02
	160	3.22e-03	1.76	3.82e-02	1.22	-8.04e-02	-8.02e-02
	320	6.46e-04	2.32	1.83e-02	1.06	-8.79e-02	-8.79e-02
	640	3.44e-05	4.23	1.32e-03	3.79	-8.82e-02	-8.82e-02
	1280	2.58e-07	7.06	3.37e-05	5.29	-8.74e-02	-8.74e-02
Regular Sweep	80	2.17e-03	–	3.40e-02	–	0	0
	160	6.62e-04	1.71	3.38e-02	0.01	0	0
	320	4.90e-05	3.76	6.38e-03	2.41	0	0
	640	1.19e-06	5.36	2.64e-04	4.59	0	0
	1280	5.48e-09	7.76	3.76e-06	6.13	0	0

Case 3. We consider a function which contains both smooth region and discontinuities

$$u_0(x) = \begin{cases} e^{-10(x-\frac{\pi}{2})^4} + 2.0; & \frac{1}{4}\pi \le x \le \frac{3}{4}\pi \\ 0; & \text{Otherwise} \end{cases} \tag{10}$$

corresponding to the setup in Example 3.1 Case 3. Here, $m = 0$ and $M = 1$. The numerical results without exponential filter on the entire domain $[0, 2\pi]$ are presented in Table 19 and Figure 13. With the exponential filter, the numerical errors on smooth region are reported in Table 20. Similar conclusion as Example 4.1 Case 2 can be made that the sweeping procedure can retain the accuracy of the original scheme while enjoy the bound-preserving property.

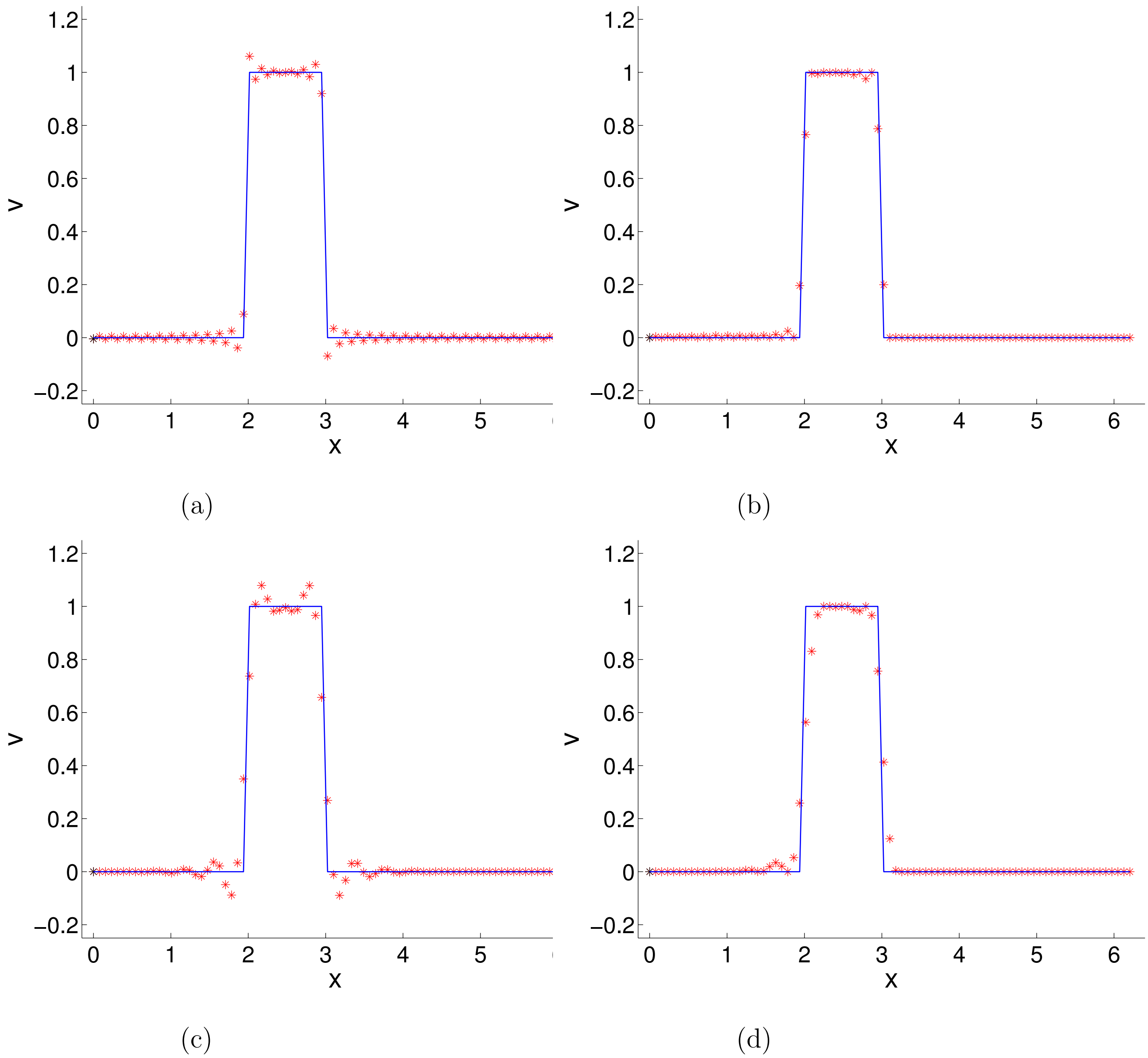


Figure 12: Numerical approximation for Example 4.1, Case 2. (a) No sweeping technique; no filter (b) regular sweep; no filter (c) no sweeping technique; with filter ; (d) with regular sweeping technique and filter. $N = 80$. Line: exact solution; dots: numerical solution.

Example 4.2 $u_t + c_1\, u_x + c_2\, u_y = 0$ *on* $[0, 2\pi]^2$ *with* $c_1 = 2$*,* $c_2 = 3$ *and periodic boundary condition. Similarly, we test our proposed sweeping method within semi-Lagrangian framework using different initial data. The final time is taken as* $T = 0.5$ *and the time step* $\Delta t = 0.0025$*.*

Table 19: Accuracy study for Example 4.1, Case 3 on $[0, 2\pi]$, $T = 0.5$, without exponential filter. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	2.10e-02	–	1.76e-01	–	-1.26e-01	-2.19e-02
	160	4.39e-02	-1.06	1.31e-00	-2.90	-2.75e-01	-3.06e-02
	320	5.87e-03	2.90	1.38e-01	3.25	-1.10e-01	-4.55e-03
	640	6.93e-04	3.08	2.64e-02	2.39	-2.50e-02	-4.75e-04
	1280	2.54e-03	-1.87	1.96e-01	-2.89	-1.46e-01	-1.55e-03
Regular Sweep	80	3.84e-02	–	6.27e-01	–	0	2.14e-02
	160	2.74e-02	0.49	1.39e-00	-1.15	0	0
	320	1.08e-02	1.34	5.85e-01	1.25	0	0
	640	5.55e-03	0.96	4.99e-01	0.23	0	2.62e-04
	1280	5.08e-03	0.13	5.10e-01	-0.03	0	0

Case 1. $u_0(x, y) = e^{-5[(x-\pi)^2+(y-\pi)^2]}$.

The corresponding one-step problem is Example 3.2 Case 1. $m = 0$ and $M = 1$. The numerical error at $T = 0.5$ is given in Table 21 in which we can easily conclude that the spectral method with sweeping technique can achieve spectral accuracy and bound-preserving property at the same time.

Table 20: Accuracy study for Example 4.1, Case 3 on $[0, 2\pi]\backslash\{I_1 \cup I_2\}$, $I_1 = [\frac{\pi}{4}+1-\epsilon, \frac{\pi}{4}+1+\epsilon]$, $I_2 = [\frac{3\pi}{4}+1-\epsilon, \frac{3\pi}{4}+1+\epsilon]$, $\epsilon = 0.15$, $T = 0.5$, with exponential filter and different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain $[0, 2\pi]$.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	2.71e-02	–	1.80e-01	–	-1.74e-01	-4.06e-02
	160	6.70e-03	2.02	7.74e-02	1.22	-1.62e-01	-1.60e-03
	320	1.19e-03	2.49	3.69e-02	1.07	-1.78e-01	-1.22e-06
	640	7.36e-05	4.02	2.67e-03	3.79	-1.79e-01	5.31e-10
	1280	4.98e-07	7.21	5.42e-05	5.62	-1.77e-01	1.61e-12
Regular Sweep	80	1.70e-02	–	1.97e-01	–	0	0
	160	4.76e-03	1.84	7.98e-02	1.30	0	0
	320	8.65e-04	2.46	3.50e-02	1.19	0	2.27e-06
	640	4.85e-05	4.16	4.33e-03	3.01	0	5.39e-10
	1280	3.18e-07	7.25	8.31e-05	5.70	0	1.60e-12

Case 2. We test the discontinuous function

$$u_0(x, y) = \begin{cases} 1; & (x, y) \in [1, 2] \times [1, 2] \\ 0; & \text{Otherwise,} \end{cases} \tag{11}$$

corresponding to Example 3.2 Case 3. Here $m = 0$ and $M = 1$. The numerical results without exponential filter on the whole domain is presented in Table 22 and Figure 14. With the exponential filter, we can see the spectral method can maintain the order of accuracy in the smooth region, see Table 23.

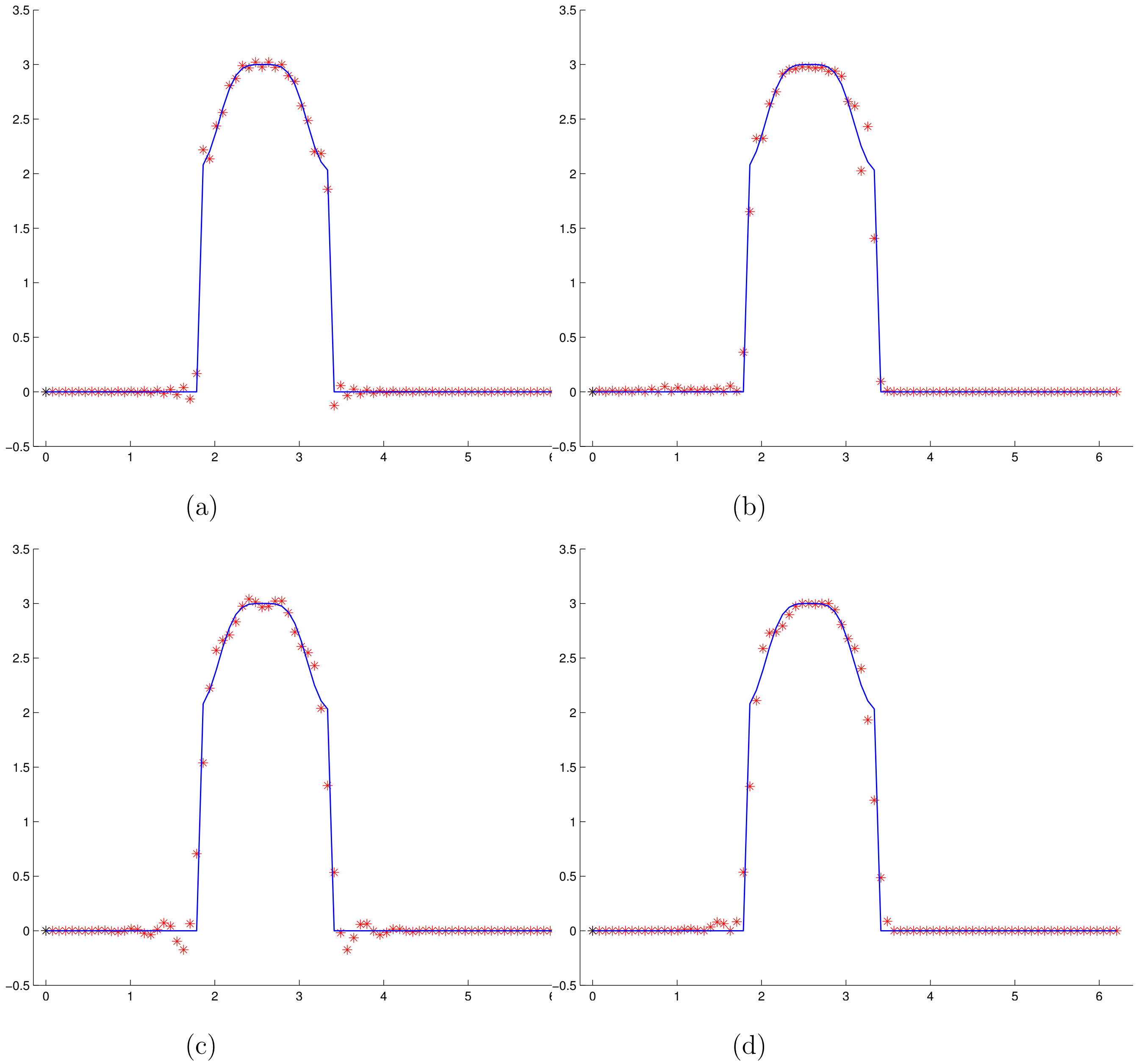


Figure 13: Numerical approximation for Example 4.1, Case 3. (a) No sweeping technique; no filter; (b) regular sweep; no filter; (c) no sweeping technique; with filter; (d) with regular sweeping technique and filter. $N = 80$. Line: exact solution; dots: numerical solution.

Case 3. We test the function

$$u_0(x) = \begin{cases} e^{-10((x-\frac{\pi}{2})^4+(y-\frac{\pi}{2})^4)} + 2.0; & (x,y) \in (\frac{1}{4}\pi, \frac{3}{4}\pi) \times (\frac{1}{4}\pi, \frac{3}{4}\pi); \\ 0; & \text{Otherwise,} \end{cases} \tag{12}$$

Table 21: Accuracy study for Example 4.2, Case 1 on the computational domain $[0, 2\pi]\times[0, 2\pi]$, $T = 0.5$. No exponential filter; with different sweeping arrays. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	10	2.96e-03	–	3.56e-03	–	-3.13e-02	3.45e-01
	20	2.60e-05	6.83	9.53e-06	8.55	-3.37e-04	1.12e-01
	40	1.98e-12	23.65	1.33e-12	22.77	-3.30e-11	9.77e-03
	80	6.51e-17	14.89	6.27e-17	14.37	-1.82e-15	5.39e-03
Sweep I	10	2.98e-03	–	3.88e-03	–	0	3.77e-01
	20	2.50e-05	6.90	1.81e-05	7.74	0	1.13e-01
	40	2.03e-12	23.55	1.37e-12	23.66	0	9.77e-03
	80	6.53e-17	14.92	5.91e-17	14.50	0	5.39e-03

The corresponding one-step problem is given as Example 3.2 Case 5. We test the method with and without exponential filter. The numerical results without filter on the entire computational domain is given in Table 24 and Figure 15. The numerical results with exponential filter on smooth region are reported in Table 25 in which the effectiveness of the sweeping procedure can be observed.

Table 22: Accuracy study for Example 4.2, Case 2 on the computational domain $[0, 2\pi]\times[0, 2\pi]$, $T = 0.5$. No exponential filter. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	1.46e-03	–	7.40e-01	–	-1.56e-01	-1.93e-01
	160	5.94e-04	1.30	7.31e-01	0.02	-1.55e-01	-3.15e-01
	320	3.29e-04	0.85	6.92e-01	0.08	-1.50e-01	-1.98e-01
	640	5.15e-06	6.00	3.26e-02	4.41	-1.85e-02	-3.13e-02
Sweep I	80	1.56e-03	–	9.75e-01	–	0	0
	160	4.16e-04	1.91	9.98e-01	-0.03	0	0
	320	3.74e-04	0.16	9.65e-01	0.05	0	0
	640	9.95e-05	1.91	9.98e-01	-0.05	0	0

Table 23: Accuracy study for Example 4.2, Case 2 on the computational domain $\{I_1 \cup I_2 \cup I_3 \cup I_4 \cup I_5\}$, $I_1 = [0, 2-\epsilon] \times [0, 2\pi]$, $I_2 = [3+\epsilon, 2\pi] \times [0, 2\pi]$, $I_3 = [2, 3] \times [0, 2.5-\epsilon]$, $I_4 = [2, 3]\times[3.5+\epsilon, 2\pi]$, $I_5 = [2+\epsilon, 3-\epsilon]\times[2.5+\epsilon, 3.5-\epsilon]$, $\epsilon = 0.15$, $T = 0.5$. With exponential filter. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	2.68e-03	–	1.58e-01	–	-9.57e-02	-1.58e-01
	160	7.66e-04	1.81	7.44e-02	1.09	-8.66e-02	-1.64e-01
	320	1.57e-04	2.29	3.60e-02	1.05	-9.49e-02	-1.75e-01
	640	8.69e-06	4.18	2.65e-03	3.76	-9.59e-02	-1.84e-01
Sweep I	80	6.29e-04	–	1.17e-01	–	0	0
	160	2.65e-04	1.25	6.55e-02	0.84	0	0
	320	2.54e-05	3.38	1.13e-02	2.54	0	0
	640	4.96e-07	5.68	1.37e-03	3.04	0	0

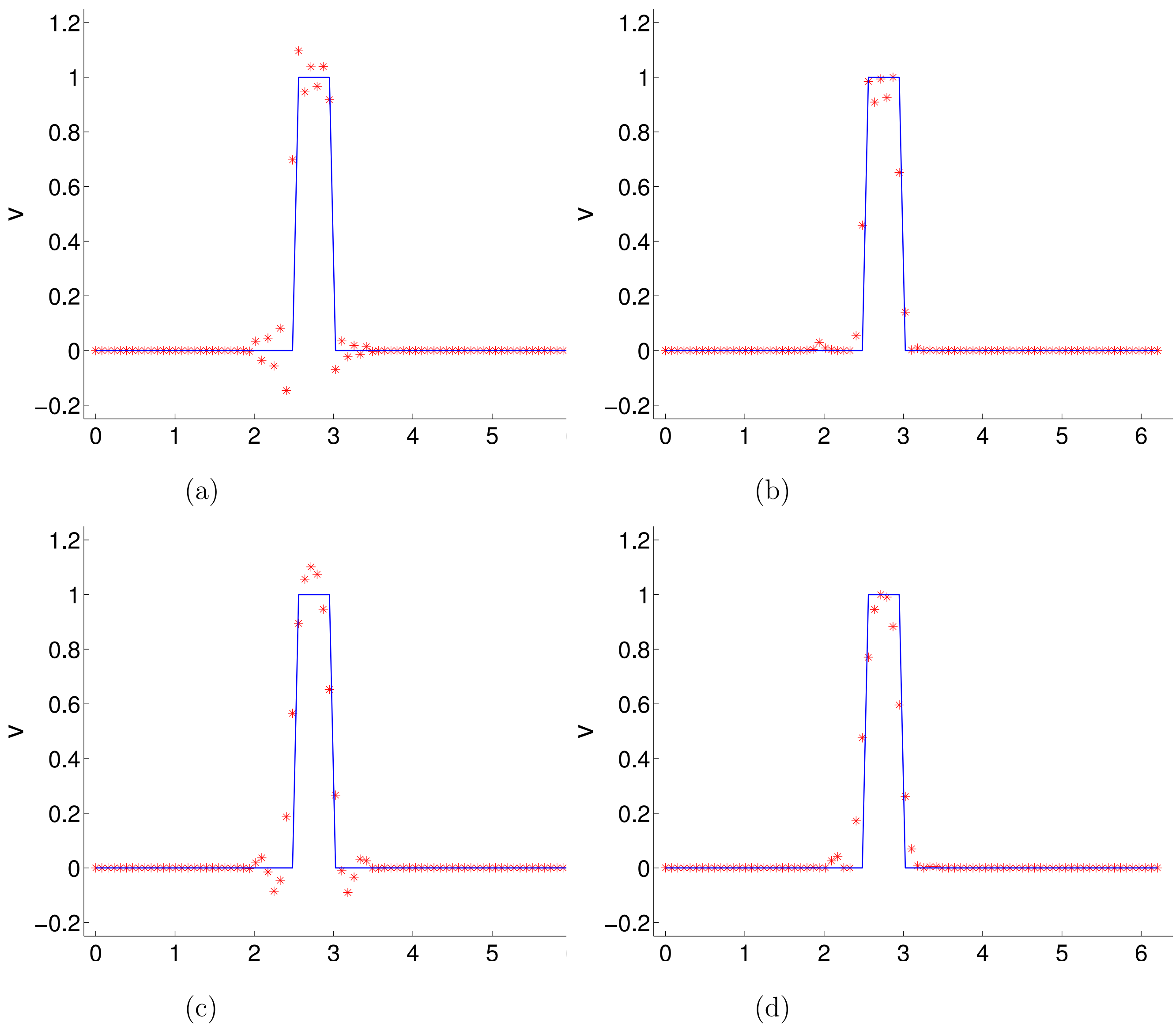


Figure 14: Numerical approximation for Example 4.2, Case 2 along $y = x$. (a) No sweeping technique; no filter (b) regular sweep; no filter (c) no sweeping technique; with filter ; (d) with regular sweeping technique and filter. $N = 80$. Line: exact solution; dots: numerical solution.

Table 24: Accuracy study for Example 4.2, Case 3 on the computational domain $[0, 2\pi]\times[0, 2\pi]$, $T = 0.5$. No exponential filter. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	2.05e-03	–	7.61e-01	–	-2.76e-01	-8.03e-02
	160	1.43e-03	0.52	1.60e-00	-1.07	-3.13e-01	-6.35e-02
	320	1.03e-03	0.47	1.36e-00	0.23	-2.90e-01	-1.97e-02
	640	1.60e-05	6.01	6.51e-02	4.38	-3.71e-02	-1.19e-03
Sweep I	80	3.19e-03	–	2.00e-00	–	0	0
	160	1.84e-03	0.79	2.00e-00	0	0	0
	320	1.15e-03	0.68	2.00e-00	0	0	0
	640	2.00e-04	2.52	2.00e-00	0	0	1.42e-03

Table 25: Accuracy study for Example 4.2, Case 3 on the computational domain $\{I_1 \cup I_2 \cup I_3 \cup I_4\}$, $I_1 = [0, \frac{\pi}{4}+1-\epsilon]\times[0, 2\pi]$, $I_2 = [\frac{3\pi}{4}+1+\epsilon, 2\pi]\times[0, 2\pi]$, $I_3 = [\frac{\pi}{4}+1, \frac{3\pi}{4}+1]\times[0, \frac{\pi}{4}+1.5-\epsilon]$, $I_4 = [\frac{\pi}{4}+1, \frac{3\pi}{4}+1]\times[\frac{3\pi}{4}+1.5+\epsilon, 2\pi]$, $\epsilon = 0.15$, $T = 0.5$. With exponential filter. The minimum and maximum values are evaluated on the whole computational domain.

	N	L^1 error	order	L^∞ error	order	$v_{min} - m$	$M - v_{max}$
W/O Sweep	80	6.80e-03	–	1.90e-01	–	-1.90e-01	-7.60e-02
	160	1.68e-03	2.02	8.26e-02	1.20	-1.73e-01	-3.06e-03
	320	3.00e-04	2.49	3.78e-02	1.13	-1.90e-01	-3.81e-06
	640	1.78e-05	4.08	2.92e-03	3.69	-1.92e-01	1.13e-09
Sweep I	80	8.00e-04	–	1.07e-01	–	0	0
	160	1.95e-04	2.04	7.31e-02	0.55	0	0
	320	3.23e-05	2.59	1.87e-02	1.97	0	0
	640	5.29e-07	5.93	1.13e-03	4.05	0	1.13e-09

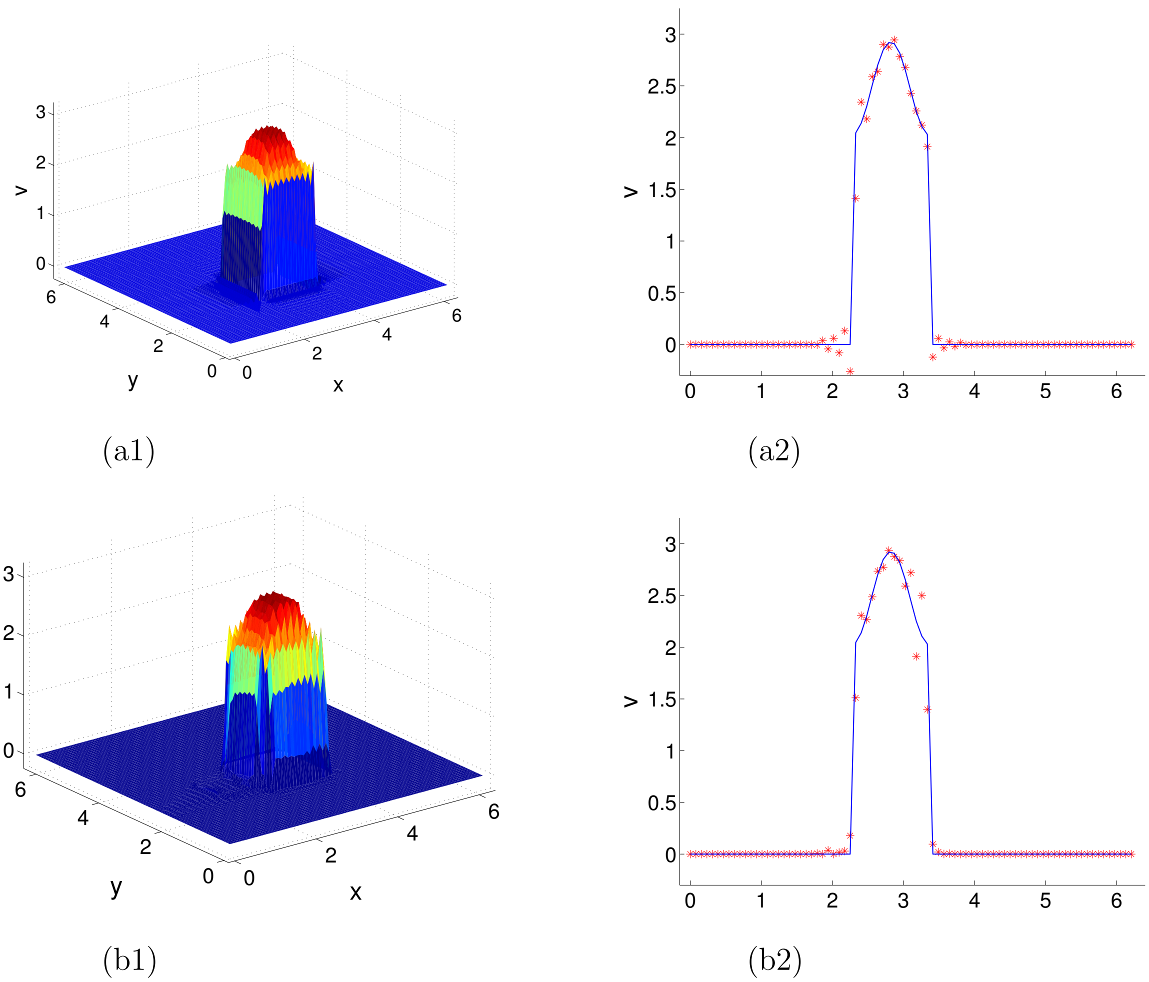


Figure 15: Numerical approximation without filter for Example 4.2, Case 3. (a1-b1) Numerical solution on the computational domain; (a2-b2) Numerical solution along the line $y = x$; Line: exact solution; dots: numerical solution. $N = 80$.

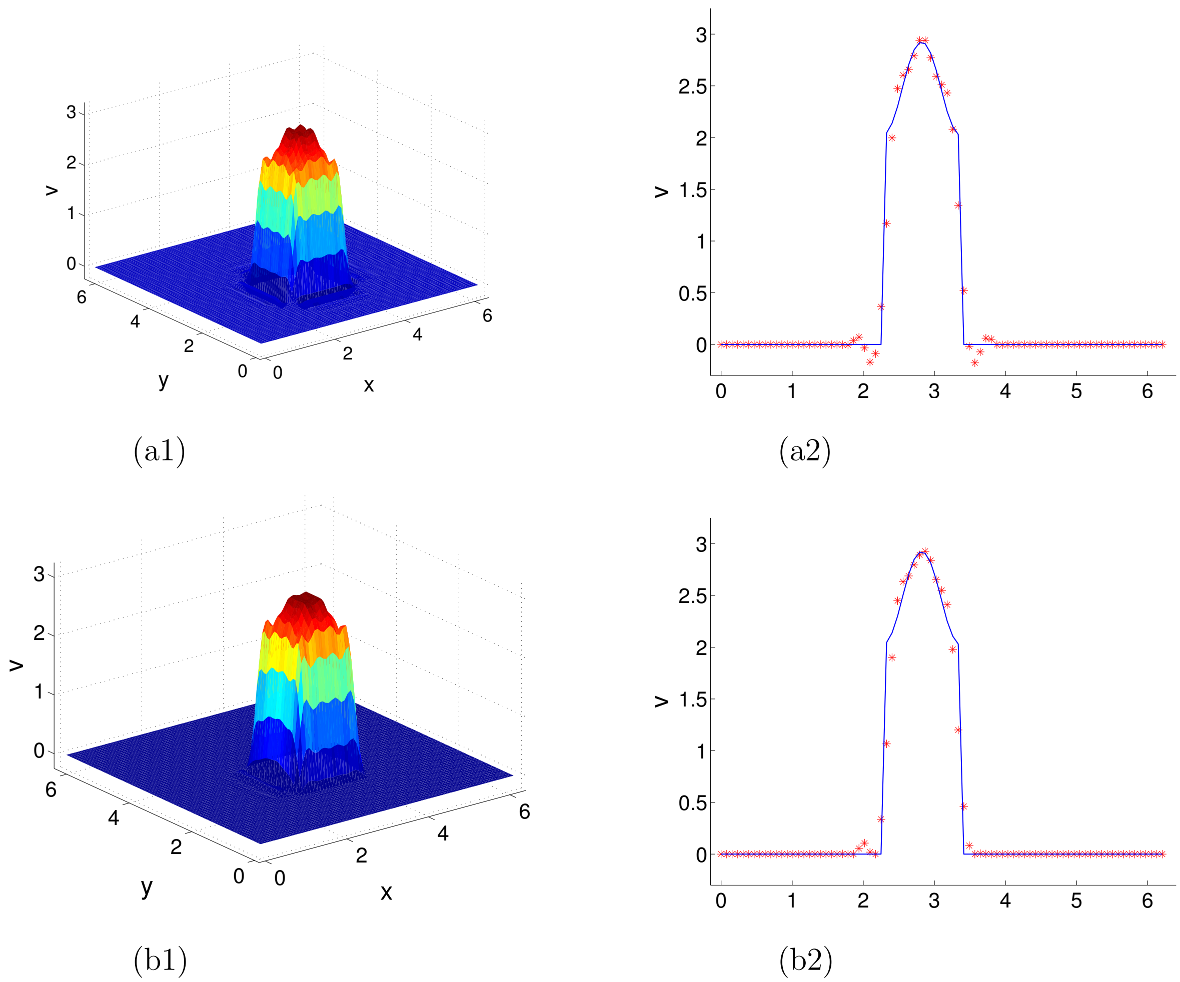


Figure 16: Numerical approximation with filter for Example 4.2, Case 3. (a1-b1) Numerical solution on the computational domain; (a2-b2) Numerical solution along the line $y = x$; Line: exact solution; dots: numerical solution. $N = 80$.

4.2 High order finite difference and finite volume schemes

In this subsection, we present numerical examples for the proposed sweeping technique with high order finite difference and finite volume schemes. We consider high order linear and WENO schemes [7] applied to scalar conservation laws and incompressible Euler equations. For all our examples, the 3^{rd} order TVD Runge-Kutta time discretization [10] is used. The time step size Δt is chosen for the purpose of verifying the the spatial accuracy. For example, $\Delta t = \Delta x, \Delta x^{5/3}, \Delta x^3$ for 3^{rd}, 5^{th} and 9^{th} order spatial discretizations respectively. For finite difference methods, a uniform Cartesian mesh is employed; while for finite volume scheme, we use uniform and nonuniform meshes for 1D simulations, and Cartesian and unstructured mesh for 2D simulations. We consider conservative schemes in our simulations, which means the sweeping procedure will be based on conserving $\sum w_j u_j$, with w_j as uniform weights for finite difference methods, and w_j as the weighted cell sizes for the finite volume schemes.

Example 4.3 *Consider the 1D linear equation*

$$\begin{cases} u_t + u_x = 0; \\ u(x,0) = u_0(x). \end{cases} \tag{13}$$

with periodic boundary conditions. The computational domain is $[0, 2\pi]$ *and* $u_0(x) = \sin^4(x)$ *and final time* $T = 2.0$.

The numerical results obtained by 3^{rd} and 9^{th} order finite difference and WENO schemes are presented in Tables 26 and 27. Here and for all error tables in this subsection, the maximum values of the numerical solution are not listed because no simulation exceeds the upper bound M. Clearly, the original accuracy of the finite difference linear scheme and WENO scheme is retained while the lower bound is enforced by the sweeping method.

We also test the finite volume schemes on non-uniform meshes. The mesh is chosen so that the maximum ratio of the size between neighboring cells is 10. In particular, for the coarsest level mesh, the odd number cells are 10 times larger than the even number cells. The numerical results for the 5^{th} order schemes are reported in Table 28 and similar conclusions as finite difference schemes can be obtained even with nonuniform weights in the sweeping procedure.

Table 26: Accuracy study of finite difference schemes for Example 4.3 with $u_0(x) = \sin^4(x)$ on $[0, 2\pi]$; CFL=1.0; T=2.0.

3^{rd} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	1.28E-01	–	3.06E-01	–	-7.92E-02	1.03E-01	–	3.44E-01	–	0
40	5.58E-02	1.20	2.23E-01	0.46	-4.37E-02	4.05E-02	1.35	2.23E-01	0.63	0
80	1.33E-02	2.07	6.76E-02	1.72	-8.38E-03	1.23E-02	1.72	6.77E-02	1.72	0
160	2.06E-03	2.69	1.20E-02	2.49	-2.67E-04	2.06E-03	2.57	1.20E-02	2.49	0
320	2.72E-04	2.92	1.61E-03	2.90	-3.74E-07	2.72E-04	2.92	1.61E-03	2.90	0
640	3.42E-05	2.99	2.04E-04	2.98	-5.03E-11	3.42E-05	2.99	2.04E-04	2.98	0
3^{rd} order WENO Scheme										
	WO Sweep					W/ Sweep				
20	1.14E-01	–	3.69E-01	–	-6.86E-02	1.03E-01	–	4.26E-01	–	0
40	4.95E-02	1.20	2.32E-01	0.67	-3.08E-02	4.47E-02	1.20	2.30E-01	0.87	0
80	1.50E-02	1.72	6.98E-02	1.73	-1.87E-02	1.42E-02	1.65	6.96E-02	1.73	0
160	2.40E-03	2.64	1.26E-02	2.47	-1.12E-03	2.40E-03	2.57	1.26E-02	2.46	0
320	2.88E-04	3.06	1.64E-03	2.94	-2.71E-07	2.88E-04	3.06	1.64E-03	2.94	0
640	3.52E-05	3.03	2.05E-04	3.00	-5.03E-11	3.52E-05	3.03	2.05E-04	3.00	0

Table 27: Accuracy study of finite difference scheme for Example 4.3 with $u_0(x) = \sin^4(x)$ on $[0, 2\pi]$; CFL=1.0; T=2.0.

9^{th} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	6.93E-02	–	1.72E-01	–	-9.47E-02	5.33E-02	–	1.89E-01	–	0
40	7.05E-03	3.30	2.63E-02	2.71	-5.31E-03	5.99E-03	3.15	2.63E-02	2.85	0
80	5.13E-05	7.10	2.66E-04	6.62	-4.32E-06	5.12E-05	6.87	2.66E-04	6.62	0
160	1.27E-07	8.66	6.82E-07	8.61	-1.07E-13	1.27E-07	8.65	6.82E-07	8.61	0
320	2.65E-10	8.91	1.46E-09	8.87	-2.02E-19	2.65E-10	8.91	1.46E-09	8.87	6.57E-20
9^{th} order WENO Scheme										
	WO Sweep					W/ Sweep				
20	8.59E-02	–	2.47E-01	–	-1.09E-02	8.23E-02	–	2.38E-01	–	0
40	8.85E-03	3.28	5.88E-02	2.07	-5.06E-06	8.86E-03	3.22	5.87E-02	2.02	0
80	9.09E-05	6.61	4.59E-04	7.00	-2.34E-07	9.07E-05	6.61	4.58E-04	7.00	0
160	2.70E-07	8.40	2.15E-06	7.74	-4.85E-08	2.96E-07	8.26	2.77E-06	7.37	0
320	6.00E-10	8.81	5.64E-09	8.57	-5.34E-14	6.39E-10	8.85	8.15E-09	8.41	0

Table 28: Accuracy study of finite volume scheme for Example 4.3 on non-uniform meshes with $u_0(x) = \sin^4(x)$ on $[0, 2\pi]$; The maximum ratio of the cell size is 10. CFL=1.0; T=2.0.

5^{th} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	1.77E-02	–	2.93E-02	–	-2.89E-02	1.26E-02	–	2.80E-02	–	0
40	1.28E-03	3.79	2.88E-03	3.35	-5.62E-04	1.23E-03	3.35	2.70E-03	3.37	0
80	2.09E-04	2.62	7.36E-04	1.97	-9.80E-05	2.11E-04	2.54	4.00E-04	2.75	5.16E-09
160	1.36E-05	3.94	5.37E-05	3.78	-4.22E-06	1.21E-05	4.12	3.06E-05	3.71	0
320	5.25E-07	4.70	2.28E-06	4.56	-7.51E-08	4.72E-07	4.68	1.32E-06	4.54	9.83E-08
640	1.65E-08	4.99	5.19E-08	5.46	-7.78E-09	1.71E-08	4.79	5.59E-08	4.56	3.58E-10
5^{th} order WENO Scheme										
	WO Sweep					W/ Sweep				
20	3.30E-02	–	1.42E-01	–	5.97E-04	2.17E-02	–	5.73E-02	–	1.31E-03
40	5.75E-03	2.52	1.30E-02	3.46	-7.61E-04	4.50E-03	2.27	1.75E-02	1.71	0
80	2.36E-04	4.61	7.41E-04	4.13	-9.57E-05	2.37E-04	4.25	4.36E-04	5.33	6.15E-09
160	1.37E-05	4.11	5.37E-05	3.79	-4.22E-06	1.22E-05	4.29	3.06E-05	3.83	0
320	5.25E-07	4.70	2.28E-06	4.56	-7.51E-08	4.72E-07	4.69	1.32E-06	4.54	9.83E-08
640	1.65E-08	4.99	5.19E-08	5.46	-7.78E-09	1.71E-08	4.79	5.59E-08	4.56	3.58E-10

Example 4.4 *Consider the Burgers' equation*

$$\begin{cases} u_t + (\frac{u^2}{2})_x = 0; \\ u(x,0) = u_0(x). \end{cases} \tag{14}$$

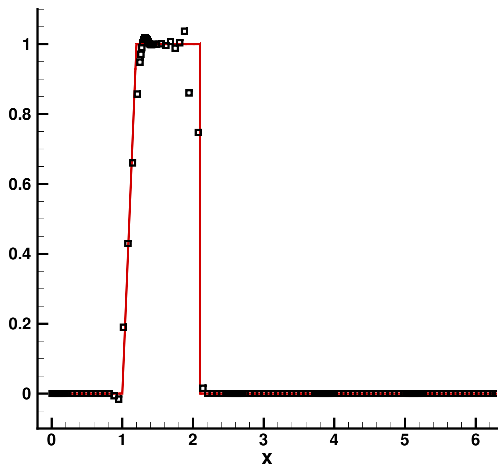
with periodic boundary conditions and computational domain $[0, 2\pi]$.

We first take the initial condition to be $u_0(x) = \sin^4(x)$. The numerical results with finite difference schemes are given in Table 29 and Table 30 when the solution is still smooth at $T = 0.2$. We can easily observe that the numerical solutions enjoy bound-preserving property while maintaining the overall accuracy.

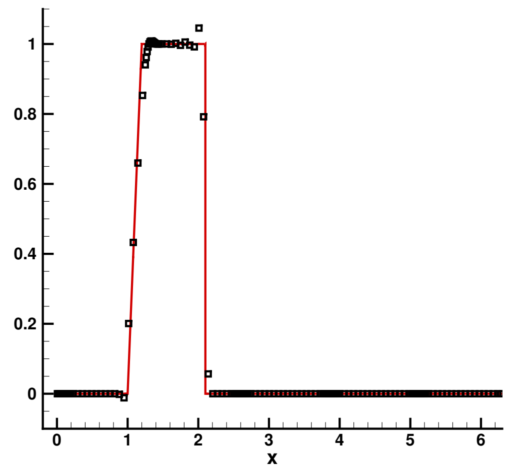
Similar as in [18], we also consider a discontinuous initial condition

$$u(x) = \begin{cases} 1; & 1 \le x \le 2 \\ 0; & \text{Otherwise.} \end{cases} \tag{15}$$

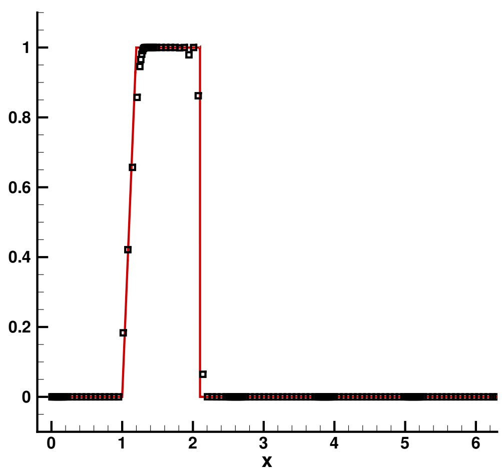
and solve the equation by finite volume methods. The numerical results are presented in Table 31 and Figure 17. It clearly shows that the numerical solutions via 5^{th} order scheme go beyond the bounds without any special treatment while stay within bounds with sweeping procedure.



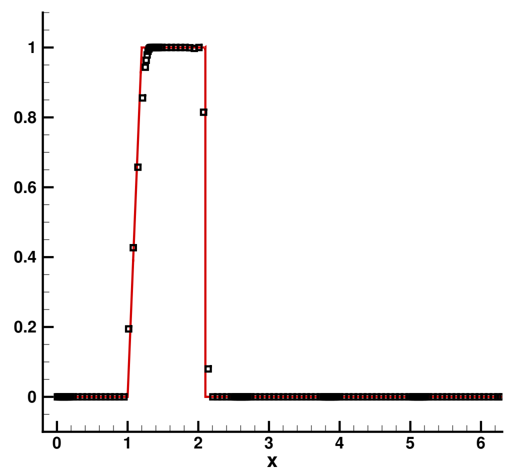
(a)



(b)



(c)



(d)

Figure 17: Numerical solutions via 5^{th} order finite volume schemes. (a-b) without sweeping technique; (c-d) with sweeping technique. Left: linear scheme. Right: WENO scheme. Solid line: exact solutions; Symbols: numerical solutions. $N = 160$. CFL$= 1.0$.

Table 29: Accuracy study of finite difference scheme for Example 4.4 with $u_0(x) = \sin^4(x)$ on $[0, 2\pi]$. CFL=1.0. T=0.2.

3^{rd} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	9.95E-03	–	3.40E-02	–	-8.58E-03	9.60E-03	–	3.40E-02	–	0
40	1.66E-03	2.58	7.09E-03	2.26	-1.42E-03	1.63E-03	2.56	7.09E-03	2.26	0
80	2.21E-04	2.91	8.93E-04	2.99	-1.91E-04	2.21E-04	2.88	8.93E-04	2.99	0
160	2.71E-05	3.03	1.05E-04	3.09	-2.43E-05	2.71E-05	3.03	1.05E-04	3.09	0
320	3.32E-06	3.03	1.26E-05	3.07	-3.05E-06	3.328E-06	3.03	1.26E-05	3.07	0
640	4.11E-07	3.02	1.52E-06	3.05	-3.82E-07	4.11E-07	3.02	1.52E-06	3.05	0
3^{rd} order WENO Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	1.35E-02	–	3.95E-02	–	-1.95E-02	1.19E-02	–	3.95E-02	–	0
40	1.85E-03	2.87	4.64E-03	3.09	-3.31E-03	1.59E-03	2.90	4.64E-03	3.09	0
80	2.02E-04	3.19	5.94E-04	2.96	-2.26E-04	1.92E-04	3.05	5.94E-04	2.96	0
160	2.45E-05	3.04	8.34E-05	2.83	-2.49E-05	2.45E-05	2.97	8.34E-05	2.83	0
320	3.16E-06	2.96	1.11E-05	2.91	-3.05E-06	3.16E-06	2.96	1.11E-05	2.91	0
640	4.02E-07	2.97	1.43E-06	2.96	-3.82E-07	4.02E-07	2.97	1.43E-06	2.96	0

Table 30: Accuracy study of finite difference WENO scheme for Example 4.4 with $u_0(x) = \sin^4(x)$ on $[0, 2\pi]$. CFL=1.0. T=0.2.

9^{th} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	2.08E-03	–	5.84E-03	–	6.78E-04	2.09E-03	–	5.84E-03	–	7.07E-04
40	8.99E-05	4.53	4.96E-04	3.56	-6.65E-06	9.26E-05	4.49	4.77E-04	3.61	0
80	6.63E-07	7.08	7.48E-06	6.05	7.50E-09	6.63E-07	7.13	7.48E-06	5.99	7.65E-09
160	2.07E-09	8.33	2.55E-08	8.20	3.32E-11	2.07E-09	8.33	2.55E-08	8.20	3.34E-11
320	4.35E-12	8.89	5.30E-11	8.91	7.00E-14	4.35E-12	8.89	5.30E-11	8.91	7.04E-14
9^{th} order WENO Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	4.44E-03	–	1.25E-02	–	9.33E-05	4.45E-03	–	1.25E-02	–	1.59E-004
40	1.29E-04	5.11	7.40E-04	4.08	-1.91E-05	1.28E-04	5.11	7.39E-04	4.08	0
80	1.50E-06	6.43	1.18E-05	5.97	6.45E-07	1.50E-06	6.42	1.18E-05	5.97	6.45E-07
160	4.42E-09	8.40	7.93E-08	7.21	2.98E-09	4.49E-09	8.38	8.05E-08	7.19	2.98E-09
320	7.15E-12	9.27	1.26E-10	9.30	8.13E-13	7.17E-12	9.29	1.27E-10	9.30	8.13E-013

Table 31: The maximum and minimum values of the numerical solution computed from the indicated finite volume schemes for Example 4.4 on non-uniform meshes with initial condition (15); CFL=1.0; T=0.2.

5^{th} order Linear Scheme				
	WO Sweep		W/ Sweep	
N	$u_{\max}$	$u_{\min}$	$u_{\max}$	$u_{\min}$
20	1.19	-1.91E-03	1	0
40	1.04	-0.34	1	0
80	1.03	-0.26	1	0
160	1.06	-1.59E-02	1	0
320	1.29	-1.60E-02	1	0
5^{th} order WENO Scheme				
20	1.12	-1.95E-03	1	0
40	1.02	-1.72E-04	1	0
80	1.02	-0.24	1	0
160	1.01	-1.17E-02	1	0
320	1.05	-1.31E-02	1	0

Example 4.5 *Consider the 2D linear equation*

$$\begin{cases} u_t + u_x + u_y = 0; \\ u(x,y,0) = u_0(x,y). \end{cases} \tag{16}$$

with periodic boundary conditions.

We first solve (16) by finite difference schemes on the domain $[0, 2\pi] \times [0, 2\pi]$ with initial condition as $u_0(x) = \sin^4(x + y)$. The final time $T = 2.0$. The numerical results with and without sweeping technique are reported in Table 32 and Table 33, in which we can observe that the sweeping technique does not affect the accuracy of the original schemes while enjoys bound-preserving property.

We then solve (16) by finite volume and WENO schemes with non-uniform triangular meshes on domain $[-2, 2] \times [-2, 2]$. $u_0(x) = \sin^4(\frac{\pi}{2}(x + y))$ and $T = 2.0$. The numerical results are presented in Table 34, which confirms the good properties of the method for such nonuniform weight simulations.

We also test the 3^{rd} order linear finite difference method for Example 4.5 with discontinuous initial condition

$$u_0(x,y) = \begin{cases} 1; & y \geq x \\ -1; & y < x. \end{cases} \tag{17}$$

and final time $T = 0.5$. The numerical results along $y = x$ are given in Figure 18, from which we can conclude that the finite difference scheme with sweeping technique can effectively produce non-oscillatory numerical solutions in this example.

Table 32: Accuracy study of finite difference schemes for Example 4.5 on $[0, 2\pi] \times [0, 2\pi]$ with $u_0(x) = \sin^4(x + y)$. CFL=1.0. $T = 2.0$.

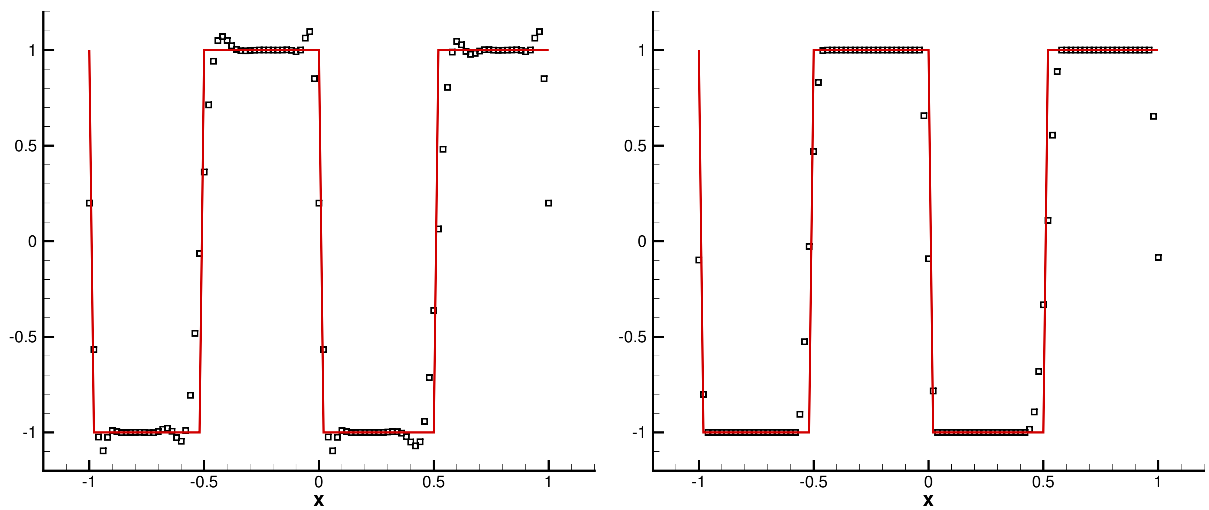
3^{rd} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	7.44E-02	–	1.82E-01	–	-4.83E-02	6.66E-02	–	1.85E-01	–	0
40	2.08E-02	1.84	4.36E-02	2.06	-2.36E-02	1.50E-02	2.15	4.40E-02	2.07	0
80	3.12E-03	2.73	6.32E-03	2.79	-3.73E-03	2.84E-03	2.40	6.37E-03	2.79	0
160	4.01E-04	2.96	8.11E-04	2.96	-4.86E-04	4.05E-04	2.81	1.12E-03	2.51	0
320	5.04E-05	2.99	1.02E-04	2.99	-6.11E-05	5.04E-05	3.01	1.49E-04	2.90	0
3^{rd} order WENO Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	9.15E-02	–	1.89E-01	–	-4.51E-002	7.68E-02	–	2.02E-01	–	0
40	2.33E-02	1.98	4.83E-02	1.97	-2.28E-02	1.85E-02	2.06	5.41E-02	1.90	0
80	4.20E-03	2.47	1.29E-02	1.90	-4.55E-03	3.71E-03	2.32	1.61E-02	1.75	0
160	5.20E-04	3.01	1.54E-03	3.07	-9.67E-04	4.70E-04	2.98	2.62E-03	2.62	0
320	5.65E-05	3.20	1.53E-04	3.32	-5.91E-05	5.56E-05	3.08	2.42E-04	3.44	0

Table 33: Accuracy study of finite difference schemes for Example 4.5 on $[0, 2\pi] \times [0, 2\pi]$ with $u_0(x) = \sin^4(x + y)$. CFL=1.0. $T = 2.0$.

9^{th} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
10	1.04E-01	–	2.18E-01	–	4.84E-02	1.24E-01	–	3.01E-01	–	5.22E-02
20	5.11E-03	4.35	8.43E-03	4.69	-7.69E-03	3.32E-03	5.23	9.08E-03	5.05	4.56E-05
40	1.49E-05	8.42	2.43E-05	8.44	8.76E-06	1.53E-05	7.76	3.56E-05	7.99	2.90E-05
80	3.16E-08	8.88	5.17E-08	8.88	-4.80E-08	3.25E-08	8.88	9.01E-08	8.63	0
160	6.30E-11	8.97	1.05E-10	8.95	8.37E-10	6.45E-11	8.98	2.00E-10	8.81	8.97E-10
9^{th} order WENO Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
10	1.09E-01	–	2.96E-01	–	7.98E-02	1.17E-01	–	3.33E-01	–	8.46E-02
20	1.33E-02	3.05	2.95E-02	3.33	-1.29E-02	9.83E-03	3.57	2.79E-02	3.58	0
40	3.22E-04	5.36	1.64E-03	4.17	1.97E-04	2.96E-04	5.05	1.53E-03	4.19	2.66E-04
80	1.31E-06	7.94	9.21E-06	7.47	9.21E-06	1.31E-06	7.82	9.21E-06	7.38	9.21E-06
160	3.11E-09	8.72	3.41E-08	8.07	3.51E-08	3.10E-09	8.72	3.41E-08	8.07	3.51E-08

Table 34: Accuracy study of finite volume schemes on 2D unstructured meshes for Example 4.5 on $[-2,2]\times[-2,2]$ with $u_0(x)=\sin^4(\frac{\pi}{2}(x+y))$. CFL=0.6. $T=2.0$.

3^{rd} order Linear scheme										
	WO Sweep					W/ Sweep				
cells	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
722	1.78E-01	–	3.64E-01	–	1.48E-01	1.79E-01	–	3.69E-01	–	1.52E-01
3042	6.30E-02	1.50	1.39E-01	1.39	-4.76E-02	5.40E-02	1.73	1.37E-01	1.43	0
12482	1.36E-02	2.21	2.67E-02	2.38	-1.51E-02	1.15E-02	2.23	2.68E-02	2.35	0
50562	1.84E-03	2.89	3.56E-03	2.91	-2.12E-03	1.79E-03	2.68	4.86E-03	2.46	0
203522	2.31E-04	2.99	4.47E-04	2.99	-2.68E-04	2.33E-04	2.94	7.75E-04	2.65	0
816642	2.88E-05	3.00	5.58E-05	3.00	-3.34E-05	2.88E-05	3.02	1.04E-04	2.90	0
3^{rd} order WENO scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
722	2.38E-01	–	4.60E-01	–	2.52E-01	2.38E-01	–	4.60E-01	–	2.52E-01
3042	1.04E-01	1.19	2.66E-01	0.79	4.72E-02	1.02E-01	1.22	2.57E-01	0.84	4.06E-02
12482	2.95E-02	1.82	1.65E-01	0.69	-8.54E-03	8.82E-02	0.21	3.91E-01	-0.61	0
50562	3.46E-03	3.09	1.56E-02	3.40	-2.28E-03	3.29E-03	4.74	1.56E-02	4.65	0
203522	2.90E-04	3.58	1.21E-03	3.69	-2.68E-04	2.92E-04	3.49	1.21E-03	3.69	0
816642	2.98E-05	3.28	8.20E-05	3.88	-3.34E-05	2.98E-05	3.29	1.04E-04	3.54	0



(a) (b)

Figure 18: Numerical solutions, cuts along $x + y = 0$ via 3^{rd} order linear finite difference scheme. (a) without sweeping technique. (b)with sweeping technique. $N_x = N_y = 100$. CFL= 1.0

Example 4.6 *Consider the 2D Burgers' equation*

$$\begin{cases} u_t + (\frac{u^2}{2})_x + (\frac{u^2}{2})_y = 0; \\ u(x,y,0) = u_0(x,y). \end{cases} \tag{18}$$

with periodic boundary conditions. We take the initial condition to be $u_0(x) = \sin^4(x+y)$ *and final time* $T = 0.2$ *when the solution is still smooth. The computation domain is* $[0,2\pi]\times[0,2\pi]$.

The results with finite difference and finite volume schemes are collected in Tables 35, 36 and 37. Similar conclusions can be made. With the sweeping technique, the numerical solutions maintain the accuracy and bound-preserving property at the same time.

Table 35: Accuracy study of finite difference schemes for Example 4.6 with $u_0(x) = \sin^4(x+y)$ on $[0, 2\pi] \times [0, 2\pi]$; CFL=1.0; $T = 0.2$.

3^{rd} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	2.52E-02	–	1.07E-01	–	-1.82E-02	2.33E-02	–	1.08E-01	–	0
40	5.19E-03	2.28	3.59E-02	1.58	-2.78E-03	5.06E-03	2.21	3.57E-02	1.59	0
80	8.58E-04	2.60	6.54E-03	2.46	-3.81E-04	8.53E-04	2.57	6.55E-03	2.45	0
160	1.06E-04	3.02	1.17E-03	2.48	-4.86E-05	1.06E-04	3.01	1.17E-03	2.48	0
320	1.33E-05	2.99	1.51E-04	2.95	-6.11E-06	1.33E-05	2.99	1.51E-04	2.95	0
3^{rd} order WENO Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	3.05E-02	–	1.00E-01	–	-3.10E-02	2.91E-02	–	1.00E-01	–	0
40	4.77E-03	2.68	3.22E-02	1.63	-4.18E-03	4.51E-03	2.69	3.22E-02	1.64	0
80	7.05E-04	2.76	5.65E-03	2.51	-4.15E-04	6.88E-04	2.17	5.66E-03	2.51	0
160	9.44E-05	2.90	1.05E-03	2.43	-5.11E-05	9.42E-05	2.87	1.05E-03	2.43	0
320	1.24E-05	2.92	1.42E-04	2.89	-6.11E-06	1.24E-05	2.92	1.42E-04	2.89	0

Table 36: Accuracy study of finite difference schemes for Example 4.6 with $u_0(x) = \sin^4(x+y)$ on $[0, 2\pi] \times [0, 2\pi]$; CFL=1.0; $T = 0.2$.

9^{th} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	1.64E-02	–	5.87E-02	–	-8.23E-03	1.29E-02	–	6.41E-02	–	0
40	1.39E-03	3.56	7.62E-03	2.94	-7.13E-05	1.40E-03	3.21	7.54E-03	3.09	0
80	6.19E-05	4.48	7.71E-04	3.30	6.01E-08	6.12E-05	4.51	7.71E-04	3.29	6.02E-08
160	6.84E-07	6.50	1.57E-05	5.61	2.59E-10	6.84E-07	6.48	1.57E-05	5.61	2.59E-10
320	2.85E-09	7.91	8.37E-08	7.55	5.72E-13	2.85E-09	7.91	8.37E-08	7.55	5.72E-13
9^{th} order WENO Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
20	1.60E-02	–	7.07E-02	–	-3.10E-04	1.59E-02	–	7.08E-02	–	0
40	1.35E-03	3.57	1.05E-02	2.75	-5.71E-05	1.36E-03	3.54	1.03E-02	2.78	0
80	6.34E-05	4.41	9.15E-04	3.52	1.38E-06	6.34E-05	4.43	9.15E-04	3.50	1.38E-06
160	7.25E-07	6.45	1.64E-05	5.80	6.68E-09	7.25E-07	6.45	1.64E-05	5.80	6.68E-09
320	2.88E-09	7.97	8.55E-08	7.59	1.88E-12	2.88E-09	7.97	8.55E-08	7.59	1.88E-12

Table 37: Accuracy study of finite volume schemes on 2D unstructured meshes for Example 4.6 with $u_0(x) = \sin^4(\pi(x+y))$ on $[-2,2] \times [-2,2]$; CFL=0.6; $T = 0.2$.

3^{rd} order Linear Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
722	1.27E-02	–	3.22E-02	–	-9.33E-03	1.23E-02	–	3.22E-02	–	0
3042	2.71E-03	2.23	7.58E-03	2.09	-2.78E-03	2.65E-03	2.21	7.58E-03	2.09	0
12482	3.92E-04	2.79	1.14E-03	2.73	-4.05E-04	3.92E-04	2.76	1.68E-03	2.17	0
50562	5.03E-05	2.96	1.52E-04	2.91	-5.20E-05	5.03E-05	2.96	2.40E-04	2.81	0
203522	6.31E-06	2.99	1.92E-05	2.98	-6.52E-06	6.31E-06	2.99	3.64E-05	2.72	0
816642	7.87E-07	3.00	2.39E-06	3.01	-8.13E-07	7.87E-07	3.00	4.49E-06	3.02	0
3^{rd} order WENO Scheme										
	WO Sweep					W/ Sweep				
N	L^1 error	order	L^∞ error	order	min	L^1 error	order	L^∞ error	order	min
722	1.26E-02	–	3.65E-02	–	8.91E-03	1.26E-02	–	3.65E-02	–	8.91E-03
3042	4.01E-03	1.65	1.90E-02	0.94	-1.02E-03	3.96E-03	1.67	1.90E-02	0.94	0
12482	1.05E-03	1.93	6.33E-03	1.59	-4.18E-04	1.04E-03	1.93	6.33E-03	1.59	0
50562	1.00E-04	3.39	6.93E-04	3.19	-5.20E-05	1.01E-04	3.36	6.93E-04	3.19	0
203522	7.85E-06	3.67	3.70E-05	4.23	-6.52E-06	7.85E-06	3.69	3.70E-05	4.23	0
816642	8.22E-07	3.26	2.72E-06	3.77	-8.13E-07	8.22E-07	3.26	4.49E-06	3.04	0

Example 4.7 *Consider the rigid body rotation*

$$u_t - (yu)_x + (xu)_y = 0, \qquad (x,y) \in [-\pi,\pi] \times [-\pi,\pi]. \tag{19}$$

The initial condition includes a slotted disk, a cone and a smooth hump, as shown in Figure 19. The final time $T = 12\pi$.

The numerical solutions along the line $x = 0$, $y = 0.8$ and $y = -2$ by the 3^{rd} order linear finite difference method are given in Figure 20, from which we can clearly see that the numerical solutions with sweeping technique stays within the range $[0,1]$.

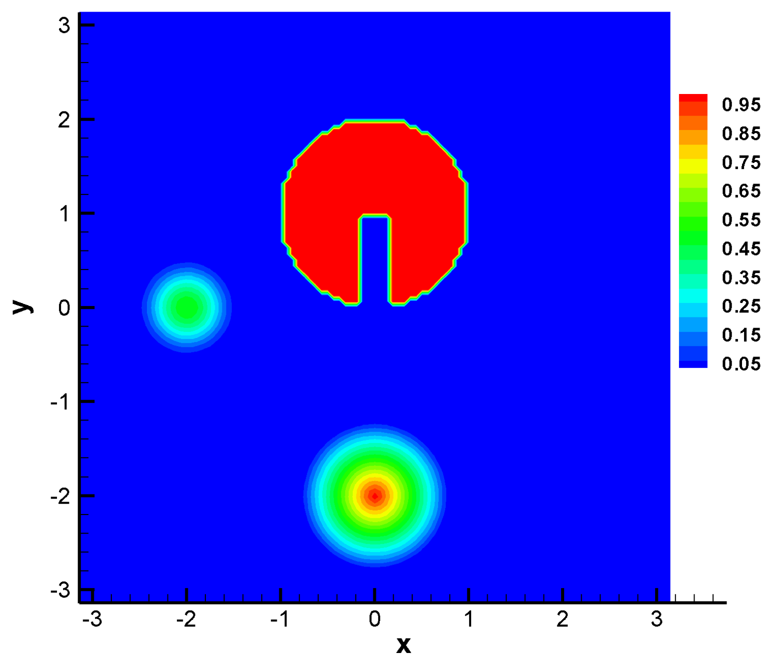


Figure 19: Initial condition for Example 4.7 and Example 4.8.

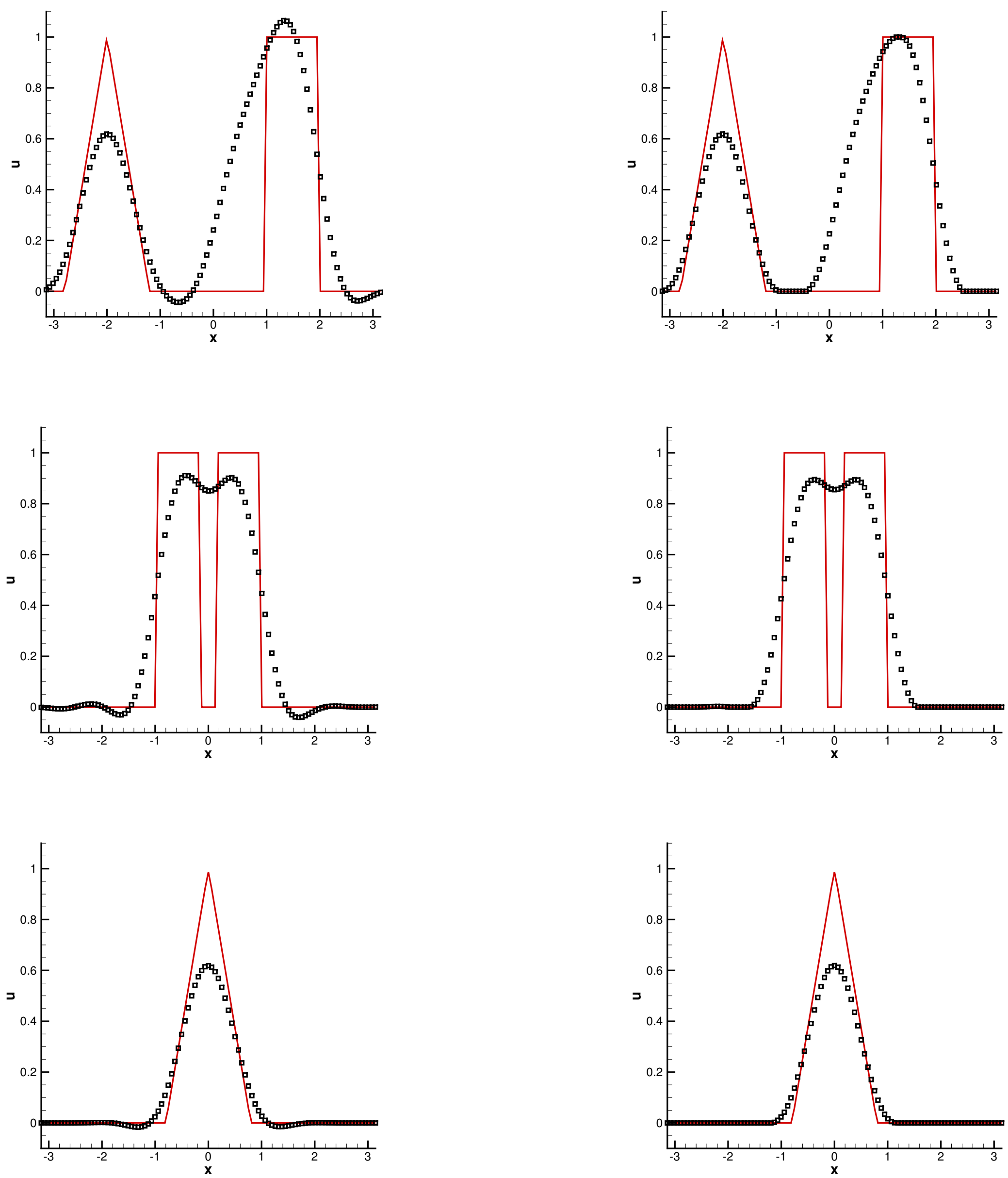


Figure 20: Numerical solutions with 3^{rd} order linear finite difference scheme. Left: without sweeping technique. Right: with sweeping technique. Cuts along $x = 0$, $y = 0.8$ and $y = -2$ from top to bottom. CFL=1. $N_x = N_y = 100$.

Example 4.8 *Consider swirling deformation flow*

$$u_t - (\cos^2(\frac{x}{2})\sin(y)g(t)u)_x + (\sin(x)\cos^2(\frac{y}{2})g(t)u)_y = 0, \qquad (x,y) \in [-\pi,\pi] \times [-\pi,\pi] \quad (20)$$

with $g(t) = \cos(\frac{\pi t}{T})\pi$. *The initial condition is the same as Example 4.7. The final time* $T = 1.5$.

We also use 3^{rd} order linear finite difference method to compute this example and similar conclusion from the previous example can be made from Figure 21. The method with sweeping technique can successfully control the overshoots and undershoots to the theoretical range.

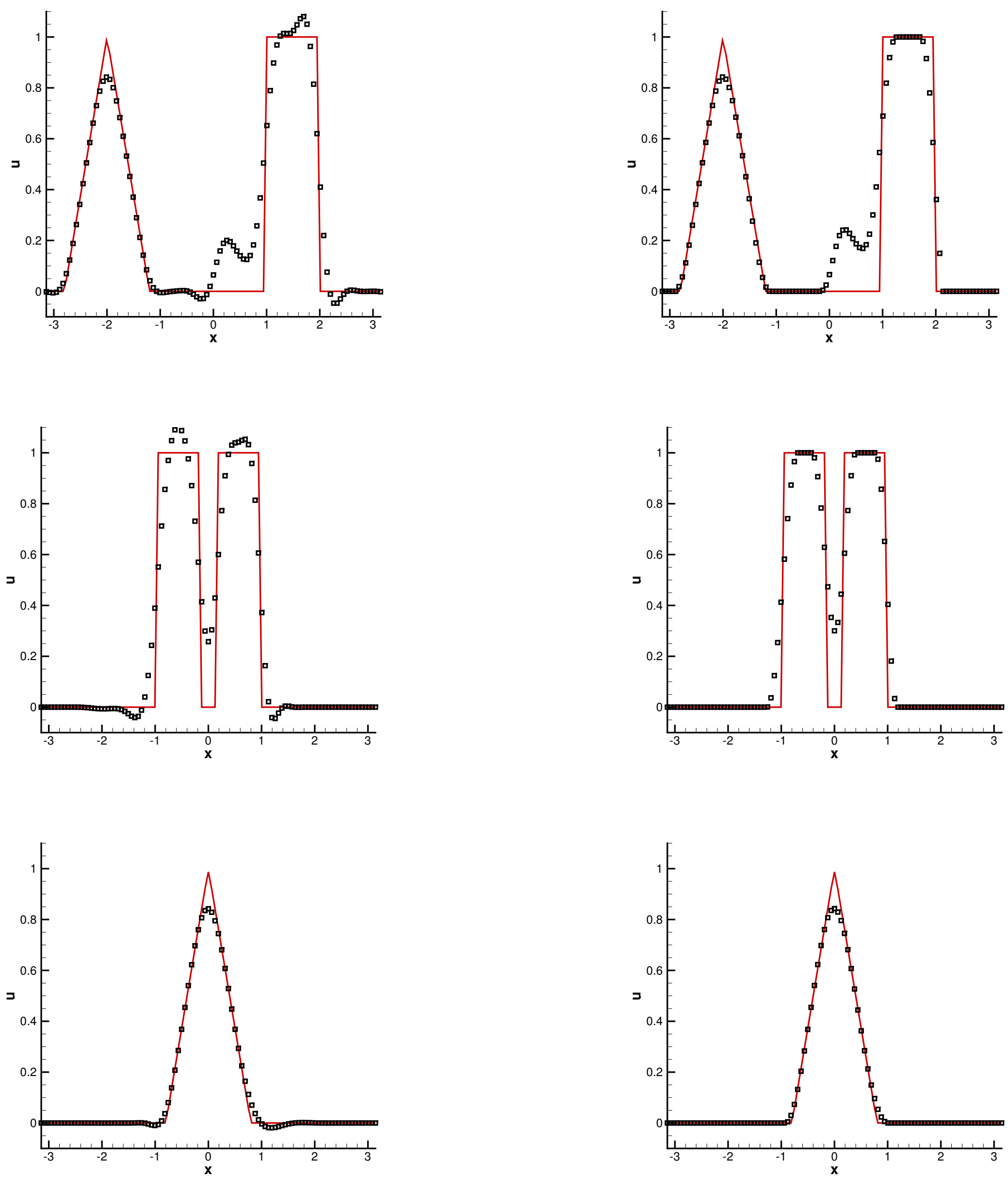


Figure 21: Numerical solutions with 3^{rd} order linear finite difference scheme. Left: without sweeping technique. Right: with sweeping technique. Cuts along $x = 0$, $y = 0.8$ and $y = -2$ from top to bottom. CFL=1. $N_x = N_y = 100$.

Example 4.9 *Consider the incompressible Euler equation*

$$\begin{cases} \omega_t + (u\omega)_x + (v\omega)_y = 0, \\ \Delta\psi = \omega, \quad \langle u, v\rangle = \langle -\psi_y, \psi_x\rangle, \\ \omega(x, y, 0) = \omega_0(x, y). \end{cases}$$

$\langle u, v\rangle \cdot \boldsymbol{n}$ *is given on the boundary of the domain. The computational domain is* $[0, 2\pi] \times [0, 2\pi]$ *and periodic boundary condition is employed. The initial condition is taken as* $\omega_0(x, y) = -2\sin(x)\sin(y)$*. The final time* $T = 0.1$*. The exact solution stays stationary with* $\omega_0(x, y) = -2\sin(x)\sin(y)$*.*

We denote the numerical undershoot as $u^d_{\min} = v_{\min} - u_{\min}$, where v is the numerical solution. The numerical results via finite difference WENO schemes are listed in Table 38, where we can easily observe that the results with sweeping technique keep within the range $[-2, 2]$ without affecting the accuracy.

Table 38: Accuracy study of 5^{th} order finite difference WENO schemes for Example 4.9 with $\omega_0(x, y) = -2\sin(x)\sin(y)$ on $[0, 2\pi] \times [0, 2\pi]$; CFL=1.0; $T = 0.1$.

N	L^1 error	order	L^∞ error	order	$u^d_{\min}$	L^1 error	order	L^∞ error	order	$u^d_{\min}$
40	4.49E-05	–	3.76E-04	–	-1.69E-06	4.49E-05	–	3.76E-04	–	0
80	2.15E-06	4.38	3.60E-05	3.38	-6.77E-08	2.15E-06	4.38	3.60E-05	3.38	0
160	7.88E-08	4.77	1.60E-06	4.49	-1.57E-09	7.88E-08	4.77	1.60E-06	4.49	0
320	2.56E-09	4.94	7.58E-08	4.40	-2.60E-11	2.56E-09	4.94	7.58E-08	4.40	0
640	6.54E-11	5.29	1.42E-09	5.74	4.73E-13	6.54E-11	5.29	1.42E-09	5.74	4.73E-13

Example 4.10 *Consider the incompressible Euler system in Example 4.9, with the following initial condition*

$$\omega_0(x) = \begin{cases} -1; & \frac{\pi}{2} \le x \le \frac{3\pi}{2}, \frac{\pi}{4} \le y \le \frac{3\pi}{4} \\ 1; & \frac{\pi}{2} \le x \le \frac{3\pi}{2}, \frac{5\pi}{4} \le y \le \frac{7\pi}{4} \\ 0; & \textit{Otherwise.} \end{cases} \tag{21}$$

We solve this problem with 3^{rd} order linear finite difference scheme and the numerical solutions at $T = 5$ are given in Figure 22. It is easy to observe that the numerical solutions with sweeping technique are all in $[-1, 1]$.

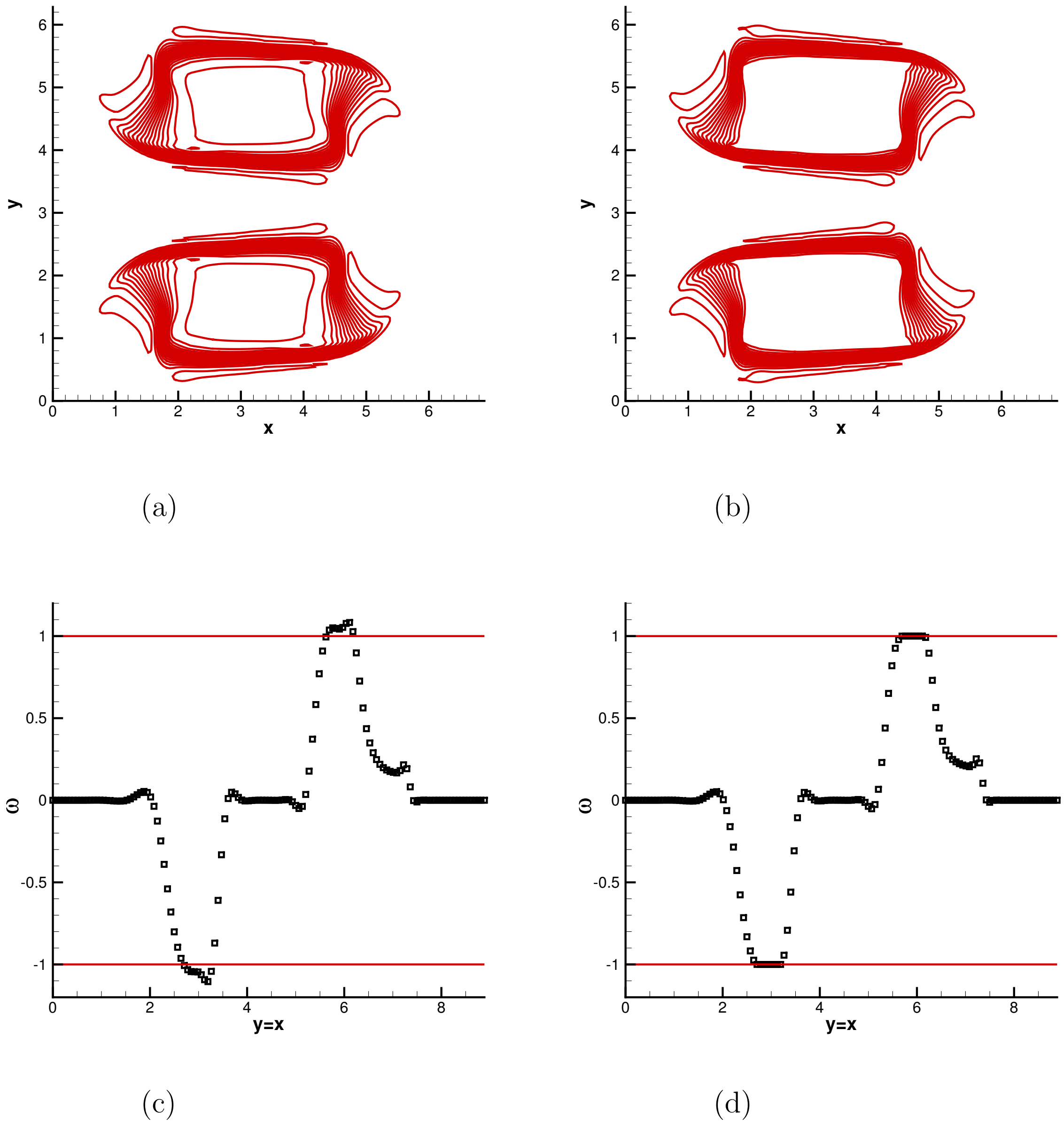


Figure 22: Numerical solutions via 3^{rd} order linear finite difference scheme. (a-b) Numerical vorticity with 30 equally spaced contours in $[-1.1, 1.1]$; (c-d) Numerical vorticity along $y = x$. Left: without sweeping technique; Right: with sweeping technique. $N_x = N_y = 128$. CFL= 1.0.

5 Conclusions and future work

In this paper, we propose a simple bound-preserving sweeping technique for conservative approximations. The method can be proven to enforce the maximum principle of the numerical solution while preserving conservation of the prescribed quantity. Extensive numerical tests demonstrate that our method is able to retain the accuracy of the underlying discretization for most benchmark problems by spectral, high order finite difference and finite volume schemes. Although the scheme is very simple to implement, it is effective in limiting the solutions and provides comparable numerical results with other popular techniques in the literature. In future, we plan to provide a theoretical framework to analyze the accuracy impact of the sweeping technique.

References

[1] X. Cai, X. Zhang, and J. Qiu. Positivity-preserving high order finite volume HWENO schemes for compressible Euler equations. *Journal of Scientific Computing*, to appear, 2016.

[2] A. J. Christlieb, Y. Liu, Q. Tang, and Z. Xu. High order parametrized maximum-principle-preserving and positivity-preserving WENO schemes on unstructured meshes. *Journal of Computational Physics*, 281:334–351, 2015.

[3] A. J. Christlieb, Y. Liu, Q. Tang, and Z. Xu. Positivity-preserving finite difference weighted ENO schemes with constrained transport for ideal magnetohydrodynamic equations. *SIAM Journal on Scientific Computing*, 37(4):A1825–A1845, 2015.

[4] D. Gottlieb and C.-W. Shu. On the gibbs phenomenon and its resolution. *SIAM review*, 39(4):644–668, 1997.

[5] E. Hewitt and R. E. Hewitt. The Gibbs-Wilbraham phenomenon: an episode in Fourier analysis. *Archive for History of Exact Sciences*, 21(2):129–160, 1979.

[6] X. Hu, N. Adams, and C.-W. Shu. Positivity-preserving method for high-order conservative schemes solving compressible Euler equations. *Journal of Computational Physics*, 242:169–180, 2013.

[7] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126:202–228, 1996.

[8] C. Liang and Z. Xu. Parametrized maximum principle preserving flux limiters for high order schemes solving multi-dimensional scalar hyperbolic conservation laws. *Journal of Scientific Computing*, 58(1):41–60, 2014.

[9] L. Pareschi and G. Russo. On the stability of spectral methods for the homogeneous boltzmann equation. *Transport Theory and Statistical Physics*, 29(3-5):431–447, 2000.

[10] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77:439–471, 1988.

[11] Y. Xing and X. Zhang. Positivity-preserving well-balanced discontinuous Galerkin methods for the shallow water equations on unstructured triangular meshes. *Journal of Scientific Computing*, 57(1):19–41, 2013.

[12] Y. Xing, X. Zhang, and C.-W. Shu. Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations. *Advances in Water Resources*, 33(12):1476–1493, 2010.

[13] T. Xiong, J.-M. Qiu, and Z. Xu. A parametrized maximum principle preserving flux limiter for finite difference RK-WENO schemes with applications in incompressible flows. *Journal of Computational Physics*, 252:310–331, 2013.

[14] T. Xiong, J.-M. Qiu, and Z. Xu. High order maximum-principle-preserving discontinuous galerkin method for convection-diffusion equations. *SIAM Journal on Scientific Computing*, 37(2):A583–A608, 2015.

[15] T. Xiong, J.-M. Qiu, and Z. Xu. Parametrized positivity preserving flux limiters for high order finite difference WENO scheme solving compressible Euler equations. *Journal of Scientific Computing*, to appear, 2016.

[16] T. Xiong, J.-M. Qiu, Z. Xu, and A. Christlieb. High order maximum principle preserving semi-lagrangian finite difference WENO schemes for the Vlasov equation. *Journal of Computational Physics*, 273:618–639, 2014.

[17] Z. Xu. Parametrized maximum principle preserving flux limiters for high order schemes solving hyperbolic conservation laws: one-dimensional scalar problem. *Mathematics of Computation*, 83(289):2213–2238, 2014.

[18] P. Yang, T. Xiong, J.-M. Qiu, and Z. Xu. High order maximum principle preserving finite volume method for convection dominated problems. *Journal of Scientific Computing*, to appear, 2016.

[19] X. Zhang, Y. Liu, and C.-W. Shu. Maximum-principle-satisfying high order finite volume weighted essentially nonoscillatory schemes for convection-diffusion equations. *SIAM Journal on Scientific Computing*, 34(2):A627–A658, 2012.

[20] X. Zhang and C.-W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229(9):3091–3120, 2010.

[21] X. Zhang and C.-W. Shu. On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *Journal of Computational Physics*, 229(23):8918–8934, 2010.

[22] X. Zhang and C.-W. Shu. Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: Survey and new developments. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 467:2752–2776, 2011.

[23] X. Zhang and C.-W. Shu. Positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations with source terms. *Journal of Computational Physics*, 230(4):1238–1248, 2011.

[24] X. Zhang and C.-W. Shu. Positivity-preserving high order finite difference WENO schemes for compressible Euler equations. *Journal of Computational Physics*, 231(5):2245–2258, 2012.

[25] X. Zhang, Y. Xia, and C.-W. Shu. Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes. *Journal of Scientific Computing*, 50(1):29–62, 2012.

[26] Y. Zhang, X. Zhang, and C.-W. Shu. Maximum-principle-satisfying second order discontinuous Galerkin schemes for convection–diffusion equations on triangular meshes. *Journal of Computational Physics*, 234:295–316, 2013.