



38 Its disadvantage is that the generation of body-fitted grid could be difficult. The  
39 quality of the grid directly determines the computational efficiency and accuracy.  
40 Especially for problems with moving boundary, the management of the moving grid  
41 is generally complex, which will increase the computational cost greatly. In addition,  
42 the governing equation needs to be changed during computation. The transformed  
43 PDE is often more complex than the original equation, which will also increase the  
44 computational cost.

45 For non body-fitted mesh methods, many scholars have also proposed some meth-  
46 ods, such as the embedded boundary method [1, 9–11, 21, 27], the immersed boundary  
47 method [4, 20, 22–24, 33], the ILW (inverse Lax-Wendroff) method [28–31] and so on.  
48 In this paper, the method we will introduce is a type of the ILW method.

49 The prototype of the earliest ILW method comes from the simulation of pedes-  
50 trian flow [7, 34]. The pedestrian walking direction can be determined by solving  
51 an Eikonal equation. They deal with the boundary conditions by transforming the  
52 normal derivative into the tangential derivative. Later, this method was extended  
53 to hyperbolic conservation law equations by Tan and Shu [28]. They transformed  
54 the normal derivative into time derivative and tangential derivative to deal with the  
55 corresponding inflow boundary conditions (different from the original Lax-Wendroff  
56 scheme, which transformed the time derivative into spatial derivative, hence the mean-  
57 ing of “inverse”), and applied this method to inviscid compressible fluids.

58 After the ILW method was proposed, many scholars have done a series of work,  
59 which have greatly developed this method. To deal with the heavy algebra of the  
60 original ILW method for nonlinear systems (especially in the high-dimensional case),  
61 the simplified ILW (SILW) method was proposed in [31], which greatly reduced the  
62 computational cost of the ILW method for solving systems. Lu et al. [19] proposed an  
63 ILW method to deal with sonic points by evaluating the flux values at ghost points,  
64 so it can deal with problems with changing wind direction. Ding et al. [5] redefined  
65 the concept of “conservation” for finite difference schemes, and gave an ILW method  
66 satisfying conservation in the new sense. In addition to hyperbolic conservation law  
67 equations, the ILW method was also applied to other types of equations, such as  
68 convection diffusion equation [13, 15, 18] and Boltzmann equation [6]. For the mov-  
69 ing boundary problem, Tan and Shu extended the ILW method to deal with the  
70 compressible inviscid fluid containing moving (translational) boundary in [29]. By  
71 redefining the material derivative on the boundary, in [3], Cheng et al. extended the  
72 method to deal with the arbitrary motion of the boundary, and used it to simulate  
73 the interaction between shock wave and rigid body. Liu et al. [17] extended this ILW  
74 method to convection-diffusion equations on moving domain, in which a unified algo-  
75 rithm was design for five cases: pure convection, convection-dominated, convection-  
76 diffusion, diffusion-dominated and pure diffusion cases. For the three-dimensional  
77 moving boundary problem, Liu et al. [16] extended the moving boundary treatment  
78 to the three-dimensional case, and simulated the interaction between inviscid / vis-  
79 cous fluid and three-dimensional rigid body. References [13–15, 32] have analyzed the  
80 linear stability of ILW and SILW methods, which provide guidelines for us to design  
81 stable ILW boundary treatments.

82 In this paper, we will design a new type of SILW method for conservation law  
83 equations. The new ILW method decomposes the construction of the ghost points  
84 into two steps: interpolation and extrapolation. At first, we approximate some special  
85 point values through an interpolation polynomial based on interior points near the  
86 boundary. Then, we will construct a Hermite extrapolation polynomial based on  
87 those special point values and spatial derivatives at the boundary obtained through

88 the ILW process. This extrapolation polynomial will give us the approximation of  
 89 the the ghost point values. Through an appropriate selection of the interpolation  
 90 points, high-order accuracy and stable results can be achieved. The eigenvalue analysis  
 91 method is used to help us select these interpolation points to ensure the stability of  
 92 the numerical scheme. The analysis results show that the new method can improve  
 93 the computational efficiency while maintaining accuracy and stability. Finally, we  
 94 apply our method to the simulation of inviscid compressible fluid.

95 The organization of this paper is as follows. In Section 2, we will give the descrip-  
 96 tion of the new ILW method for one-dimensional scalar conservation law equations,  
 97 and use the eigenvalue analysis method to perform the linear stability analysis. In  
 98 Section 3, we will extend this algorithm to system and high-dimensional cases. The  
 99 high order accuracy and robustness of our algorithm will be shown through numerical  
 100 tests in Section 4. Conclusion remarks will be given in Section 5.

101 **2. The one-dimensional scalar conservation law case.** Consider the scalar  
 102 hyperbolic conservation law in the following form:

$$103 \quad (2.1) \quad \begin{cases} u_t + f(u)_x = 0, & x \in (-1, 1), t > 0, \\ u(-1, t) = g_l(t), & t > 0, \\ u(x, 0) = u_0(t), & x \in [-1, 1]. \end{cases}$$

104 We assume that  $f'(u(-1, t)) > 0$ , such that the left boundary  $x = -1$  is an in-  
 105 flow boundary, where a boundary condition needs to be given. We also assume that  
 106  $f'(u(1, t)) > 0$ . Hence the right boundary  $x = 1$  is an outflow boundary, where no  
 107 boundary condition is required.

108 Suppose the domain is divided by the uniform mesh:

$$109 \quad (2.2) \quad -1 + C_a \Delta x = x_0 < \dots < x_N = 1 - C_b \Delta x$$

110 with uniform mesh size  $\Delta x = 2/(C_a + C_b + N)$  and  $C_a, C_b \in [0, 1)$ . Note that we have  
 111 deliberately allowed the physical boundary  $x = \pm 1$  not coinciding with grid points.

112 We use the framework of method of lines (MOL) to construct a semi-discrete  
 113 scheme on the interior point  $x_j, j = 0, 1, 2, \dots, N$ :

$$114 \quad (2.3) \quad \frac{d}{dt} u_j = L_h(u)_j,$$

where,

$$L_h = -\frac{1}{\Delta x} (\hat{f}_{j+1/2} - \hat{f}_{j-1/2}) \approx -f(u)_x|_{x_j}$$

115 is the spatial discrete operator. Here,  $u_j(t)$  is the numerical approximation to the  
 116 exact solution  $u(x_j, t)$ , and  $\hat{f}_{j+1/2}$  is the numerical flux. In this paper, we will use an  
 117 upwind-biased finite difference conservative scheme to construct  $\hat{f}_{j+1/2}$ , such as the  
 118 WENO scheme [8].

119 After the spatial discretization, the semi-discrete scheme (2.3) is a system of  
 120 ordinary differential equations. For time discretization, we use the total variation  
 121 diminishing (TVD) Runge-Kutta (RK) scheme [26]. From time level  $t^n$  to  $t^{n+1}$ , the  
 122 third order TVD RK scheme is given as

$$123 \quad (2.4) \quad \begin{aligned} u_j^{(1)} &= u_j^n + \Delta t L_h(u^n)_j, \\ u_j^{(2)} &= \frac{3}{4} u_j^n + \frac{1}{4} u_j^{(1)} + \frac{1}{4} \Delta t L_h(u^{(1)})_j, \\ u_j^{n+1} &= \frac{1}{3} u_j^n + \frac{2}{3} u_j^{(2)} + \frac{2}{3} \Delta t L_h(u^{(2)})_j. \end{aligned}$$

124 In particular, [2] pointed out that the boundary conditions in the intermediate stages  
 125 of the above RK scheme should be modified as follows to avoid order reduction:

$$\begin{aligned}
 & u^n \sim g_l(t_n), \\
 126 \quad (2.5) \quad & u^{(1)} \sim g_l(t_n) + \Delta t g_l'(t_n), \\
 & u^{(2)} \sim g_l(t_n) + \frac{1}{2} \Delta t g_l'(t_n) + \frac{1}{4} \Delta t^2 g_l''(t_n).
 \end{aligned}$$

Note that for a high order finite difference scheme, a wide computational stencil is generally required. Hence, it is inevitable that some points in the computational stencil are not in our computational domain,

$$x_{-p} = x_0 - p\Delta x, \quad x_{N+p} = x_N + p\Delta x, \quad p = 1, 2, \dots$$

127 Therefore, we can regard the boundary treatment problem as construction of the  
 128 ghost point values. In the following, we will first review the original (S)ILW method  
 129 proposed by Tan et al. [28, 31]. And then, a new SILW method will be proposed to  
 130 improve the computational efficiency while maintaining accuracy and stability. Linear  
 131 stability analysis will be given to demonstrate the advantage of the new proposed  
 132 method.

133 **2.1. Review of the original SILW method.** The main idea of the original  
 134 inverse Lax-Wendroff method for hyperbolic conservation law equations [28] is to con-  
 135 vert the spatial derivatives into the time derivatives through the PDE and boundary  
 136 conditions at the inflow boundary. At the outflow boundary, the spatial derivatives  
 137 of each order are approximated by extrapolation. After that, the values of the ghost  
 138 points outside the computational domain are obtained by Taylor expansion at the  
 139 boundary. More specifically, the ghost points near outflow boundaries, such as the  
 140 right boundary  $x = 1$  in our example problem (2.1), can be obtained by extrapolation  
 141 directly. We can choose the traditional Lagrange extrapolation with appropriate ac-  
 142 curacy when the solution is smooth near the boundary, or least square extrapolation /  
 143 WENO type extrapolation [19, 28, 31] when the solution contains discontinuities near  
 144 the boundary.

145 For the inflow boundary, such as the left boundary  $x = -1$  in our example problem  
 146 (2.1), to ensure our boundary treatment has  $d$ -th order accuracy, the value of the ghost  
 147 points near  $x = -1$  will be obtained by Taylor expansion:

$$148 \quad (2.6) \quad u_j = \sum_{k=0}^{d-1} \frac{(x_j + 1)^k}{k!} u^{*(k)}, \quad j = -1, -2, \dots$$

149 where,  $u^{*(k)}$  is the approximation of  $\partial_x^{(k)} u|_{x=-1}$  with at least  $(d-k)$ -th order accuracy.  
 150 Using PDE and boundary condition repeatedly, we have that

$$\begin{aligned}
 & \partial_x^{(0)} u|_{x=-1} = g_l(t), \\
 & \partial_x^{(1)} u|_{x=-1} = \frac{g_l'(t)}{-f(g_l(t))}, \\
 151 \quad (2.7) \quad & \partial_x^{(2)} u|_{x=-1} = \frac{f'(g_l(t))g_l''(t) - 2f''(g_l(t))g_l'(t)^2}{f'(g_l(t))^3}, \\
 & \dots
 \end{aligned}$$

**Table 1.** The table of  $(k_d)_{min}$  for original SILW method.

$d$	3	5	7	9	11	13
$(k_d)_{min}$	2	3	4	6	8	10

Thus, we can set

$$u^{*(k)} = \partial_x^{(k)} u|_{x=-1}.$$

152 To avoid the very heavy algebra of above original ILW method when calculating  
 153 the high order space derivatives, the simplified ILW (SILW) method was proposed  
 154 in [31]. Specifically,  $u^{*(0)}, u^{*(1)}$  are constructed by the original ILW procedure, i.e.,  
 155 converting the spatial derivatives into the time derivatives through the PDE and  
 156 boundary condition. The higher order spatial derivatives  $u^{*(k)}$ ,  $2 \leq k \leq d-1$ ,  
 157 are extrapolated from the interior points directly. This method can greatly improve  
 158 the computational efficiency, especially for high-dimensional systems. However, [14]  
 159 analyzed the linear stability of the SILW method through the eigenvalue method,  
 160 showing that the SILW method [31] is stable for any  $C_a \in [0, 1)$  only when  $d = 3$ , but  
 161 unstable for  $d > 3$ . In order to guarantee the stability, more ILW procedure need to  
 162 be used to construct higher order spatial derivatives at the boundary.

163 Suppose that for a  $d$ -th order scheme,  $u^{*(k)}$  is obtained through ILW procedure if  
 164  $k \leq k_d - 1$ , or by extrapolation if  $k_d \leq k \leq d - 1$ . For different high order schemes, [14]  
 165 used the eigenvalue analysis to find out the minimum  $k_d$ , denoted by  $(k_d)_{min}$ , to make  
 166 sure the scheme is stable for all  $C_a \in [0, 1)$ . The values of  $(k_d)_{min}$  for a variety of  
 167  $d$  are shown in Table 1. It can be seen that the  $(k_d)_{min}$  is still large for high order  
 168 scheme. This results in difficulty in writing codes and affects computational efficiency  
 169 for high dimensional systems.

170 In summary, the above SILW method can be divided into the following two steps,  
 171 i.e., “interpolation” and “extrapolation”:

Step 1. Construct an interpolation polynomial  $p(x)$  of degree  $d-1$  with interior points  
 $\{x_0, \dots, x_{d-1}\}$ , and obtain the approximation of spatial derivatives of each  
 order on the boundary

$$u^{*(k)} \approx \partial_x^k p|_{x=-1}, \quad k = k_d, \dots, d-1.$$

Step 2. Construct the extrapolation polynomial  $q(x)$  of degree  $d-1$  satisfying

$$q^{(k)}(-1) = u^{*(k)}, \quad k = 0, \dots, d-1,$$

where,  $u^{*(k)}$  for  $k < k_d$  are obtained by the ILW procedure (2.7), and the else  
 are obtained by Step 1. Actually, in this case,  $q(x)$  is the Taylor expansion  
 polynomial. Then, we can get the ghost points values

$$u_j = q(x_j), \quad j = -1, -2, \dots$$

172 The above (S)ILW method is based on Taylor expansion. When assembling the  
 173 extrapolation polynomial  $q(x)$  in the second step, all information used is on the bound-  
 174 ary. In fact, the information we can use to construct the extrapolation polynomial  
 175  $q(x)$  mainly consists of two parts. One is the spatial derivatives on the boundary ob-  
 176 tained by the ILW procedure, and the other part is the polynomial  $p(x)$  constructed  
 177 by interior points. In order to make the algorithm more efficient, we hope to use the  
 178 first part information as less as possible under the premise of ensuring the stability of  
 179 the scheme.

180 **2.2. A new SILW method.** In the following, we will describe our new SILW  
 181 method for the scalar conservation law equation in (2.1), hoping  $(k_d)_{min}$  would be  
 182 smaller for the same  $d$ . The key difference between the new method and the original  
 183 one is that the extrapolation polynomial  $q(x)$  will employ the point values on some  
 184 special points in computational domain,

$$185 \quad u(-1 + k\alpha\Delta x, t) \approx p(-1 + k\alpha\Delta x), \quad k = 1, 2 \dots$$

187 instead of using the high order derivatives of the interpolation polynomial  $p(x)$  at the  
 188 boundary. Here,  $\alpha \geq 0$  is a parameter to be determined such that the  $(k_d)_{min}$  would  
 189 be smaller.

190 Specifically, we summarize the procedure of our new SILW method with  $d$ -th  
 191 order accuracy in the following:

Step 1. Obtain the interpolation polynomial  $p(x)$  of degree  $d - 1$  based on the points  
 $\{x_0, \dots, x_{d-1}\}$ . Let

$$u_{k*} = p(-1 + k\alpha\Delta x), \quad 1 \leq k \leq d - k_d.$$

192 Step 2. Construct the extrapolation polynomial  $q(x)$  of degree  $d - 1$  to satisfy the  
 193 following conditions:

$$194 \quad \begin{aligned} q^{(k)}(-1) &= \partial_x^{(k)} u|_{x=-1}, \quad 0 \leq k \leq k_d - 1, \\ q(-1 + k\alpha\Delta x) &= u_{k*}, \quad 1 \leq k \leq d - k_d, \end{aligned}$$

where,  $\partial_x^{(k)} u|_{x=-1}$  is obtained by the ILW procedure. Let the ghost point  
 values be the values of the extrapolation polynomial  $q(x)$  at the corresponding  
 points:

$$u_j = q(x_j) \quad j = -1, -2, \dots$$

196 In the next subsection, we will show that through adjusting the value of  $\alpha$ , our new  
 197 SILW method is better than the original SILW method in computational efficiency  
 198 and stability.

199 **2.3. Linear stability analysis.** Here, we will give the stability analysis of the  
 200 fully discrete schemes using the eigenvalue spectrum visualization. We consider the  
 201 case of  $d = 2k - 1$  ( $k = 2, 3, 4, 5, 6, 7$ ) and assume that  $f'(u) > 0$ . The conservative  
 202 linear upwind scheme is used for spatial discretization. That is,  $L_h$  in the scheme  
 203 (2.3) is in the following form:

$d = 3:$

$$L_h(u)_j = -\frac{1}{\Delta x} \left( \frac{1}{6} f_{j-2} - f_{j-1} + \frac{1}{2} f_j + \frac{1}{3} f_{j+1} \right),$$

$d = 5:$

$$L_h(u)_j = -\frac{1}{\Delta x} \left( -\frac{1}{30} f_{j-3} + \frac{1}{4} f_{j-2} - f_{j-1} + \frac{1}{3} f_j + \frac{1}{2} f_{j+1} - \frac{1}{20} f_{j+2} \right),$$

$d = 7:$

$$204 \quad \begin{aligned} L_h(u)_j &= -\frac{1}{\Delta x} \left( \frac{1}{140} f_{j-4} - \frac{7}{105} f_{j-3} + \frac{3}{10} f_{j-2} - f_{j-1} + \frac{1}{4} f_j + \frac{3}{5} f_{j+1} \right. \\ &\quad \left. - \frac{1}{10} f_{j+2} + \frac{1}{105} f_{j+3} \right), \end{aligned}$$

205

$d = 9$ :

$$L_h(u)_j = -\frac{1}{\Delta x} \left( -\frac{1}{630}f_{j-5} + \frac{1}{56}f_{j-4} - \frac{2}{21}f_{j-3} + \frac{1}{3}f_{j-2} - f_{j-1} + \frac{1}{5}f_j \right. \\ \left. + \frac{2}{3}f_{j+1} - \frac{1}{7}f_{j+2} + \frac{1}{42}f_{j+3} - \frac{1}{504}f_{j+4} \right),$$

$d = 11$ :

$$L_h(u)_j = -\frac{1}{\Delta x} \left( \frac{1}{2772}f_{j-6} - \frac{1}{210}f_{j-5} + \frac{5}{168}f_{j-4} - \frac{5}{42}f_{j-3} + \frac{5}{14}f_{j-2} \right. \\ \left. - f_{j-1} + \frac{1}{6}f_j + \frac{5}{7}f_{j+1} - \frac{5}{28}f_{j+2} + \frac{5}{126}f_{j+3} - \frac{1}{168}f_{j+4} \right. \\ \left. + \frac{1}{2310}f_{j+5} \right),$$

$d = 13$ :

$$L_h(u)_j = -\frac{1}{\Delta x} \left( -\frac{1}{12012}f_{j-7} + \frac{1}{792}f_{j-6} - \frac{1}{110}f_{j-5} + \frac{1}{24}f_{j-4} - \frac{5}{36}f_{j-3} \right. \\ \left. + \frac{3}{8}f_{j-2} - f_{j-1} + \frac{1}{7}f_j + \frac{3}{4}f_{j+1} - \frac{5}{24}f_{j+2} + \frac{1}{18}f_{j+3} \right. \\ \left. - \frac{1}{188}f_{j+4} + \frac{1}{660}f_{j+5} - \frac{1}{10296}f_{j+6} \right).$$

In particular, for the linear case  $f(u) = u$ , the semi-discrete scheme (2.3) can be written in the matrix-vector form,

$$\frac{d\mathbf{U}}{dt} = \frac{1}{\Delta x}\mathbf{Q}\mathbf{U},$$

where,  $\mathbf{U} = (u_0, u_2, \dots, u_N)^T$  and  $\mathbf{Q}$  is the coefficient matrix of the spatial discretization.

References [14, 32] pointed out that we only need to care about the fixed eigenvalues of the matrix  $\mathbf{Q}$  with the increase of grid points for stability analysis. If we use the third-order TVD RK time discretization (2.4), the stability region can be expressed as

$$(2.8) \quad |z(\mu)| \leq 1, \quad z(\mu) = 1 + \mu + \frac{\mu^2}{2} + \frac{\mu^2}{6},$$

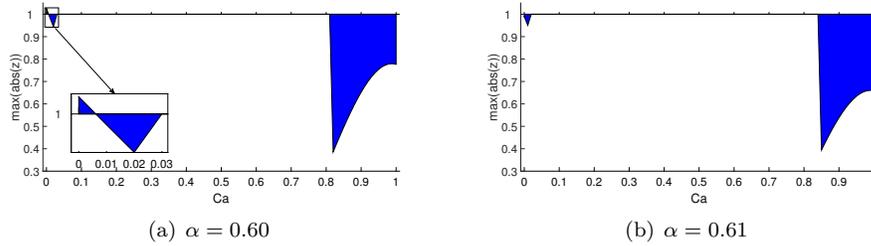
where,  $\mu = s\frac{\Delta t}{\Delta x}$ , and  $s$  is the fixed eigenvalue of  $\mathbf{Q}$  we are concerned about. Notice that  $z$  may not exist or there may be more than one. If there is more than one  $z$ , we consider the largest  $|z(\mu)|$ . Let the CFL number be  $\lambda_{cfl} = \frac{\Delta t}{\Delta x}$ . We hope that the CFL number is independent of  $C_a$ . Hence, we discuss stability on the premise of maximum CFL number  $(\lambda_{cfl})_{max}$ , where,  $(\lambda_{cfl})_{max}$  is the maximum CFL number for the corresponding Cauchy problem, and their specific values are shown in the Table 2.

We select several groups of different  $\alpha$  and  $k_d$  for linear stability analysis for different schemes, and the range of  $\alpha$  is given as  $[0, 10]$ . We compute the largest  $|z(\mu)|$  for all the eigenvalues  $s$ . By using the software Matlab, we show the max  $|z(\mu)|$  for  $C_a \in [0, 1)$  with different  $\alpha$ . For instance, we get the results of the third order

**Table 2.** The maximum CFL number for Cauchy problem.

$d$	3	5	7	9	11	13
$(\lambda_{cfl})_{max}$	1.62	1.43	1.24	1.12	1.04	0.99

233 scheme with the new SILW procedure with  $k_d = 2$  as in Figure 1. When  $\alpha = 0.60$ ,  
 234  $\max |z(\mu)| > 1$  when  $C_a$  approaches 0. However,  $\max |z(\mu)| \leq 1$  for all  $C_a \in [0, 1]$   
 235 if  $\alpha = 0.61$ . This indicates that we should take  $\alpha \geq 0.61$  to guarantee the scheme  
 236 is stable with  $k_d = 2$ . More cases are placed in Appendix A. A numerical test is  
 237 also given to verify the stability analysis results. Finally, the minimum  $k_d$  and the  
 238 corresponding appropriate  $\alpha$  range are shown in Table 3.



**Fig. 1.** The third order scheme with the new SILW procedure with  $k_d = 2$ . The horizontal axis represents  $C_a$  and the vertical axis represents the largest  $|z(\mu)|$ .

239 As can be seen from the Table 3, compared with the original SILW method,  
 240 we can construct the stable boundary treatments with smaller  $k_d$  by adjusting  $\alpha$ .  
 241 Here, we plot  $C_a$  versus  $\max |z(\mu)|$  for both boundary treatments with the same order  
 242 of accuracy and the same CFL number to compare the original and the new SILW  
 243 methods. The fifth order schemes with  $k_d = 2$  are taken as an example, see Figure  
 244 2. It is observed that with the original SILW method, the fifth order upwind scheme  
 245 is unstable when the first grid point is either close to or far from the left boundary,  
 246 while the new SILW method is stable for all the  $C_a \in [0, 1]$ .

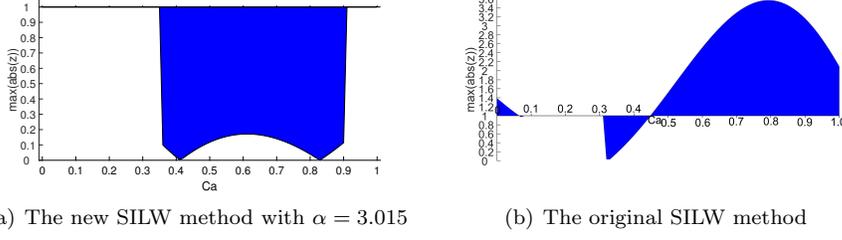
**Table 3.** Linear stability analysis results of the new SILW method

$d$	3	5	7	9	11	13
$(k_d)_{min}$	2	2	2	3	3	4
$\alpha$	[0.61,10]	[0.92,5.11]	[1.34,1.99]	[1.29,2.43]	[1.42,1.70]	[1.49,2.08]

### 247 3. The new SILW method for one- and two-dimensional systems.

248 **3.1. The new SILW method for one-dimensional Euler equation.** We  
 249 consider the following one-dimensional compressible Euler equation:

250 (3.1) 
$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = \mathbf{0}, \quad x \in (-1, 1), t > 0,$$



**Fig. 2.** The stability analysis results of the fifth order scheme with  $k_d = 2$ , the horizontal axis represents  $C_a$  and the vertical axis represents the largest  $|z(\mu)|$ .

251 where

$$252 \quad \mathbf{U} = \begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix} = \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(E + p) \end{pmatrix}.$$

253 Here,  $\rho$ ,  $u$ ,  $p$  and  $E$  represent the density, velocity, pressure and total energy per  
 254 volume, respectively. In order to close the system, we give the following equation of  
 255 state of ideal gas:

$$256 \quad E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho u^2.$$

257 Here,  $\gamma$  is the adiabatic constant, which equals to 1.4 for an ideal polytropic gas.

258 We consider the boundary treatment of left boundary  $x = -1$  as an example.  
 259 The original Euler equation (3.1) can be rewritten into the following nonconservative  
 260 form:

$$261 \quad \mathbf{U}_t + \mathbf{A}(\mathbf{U})\mathbf{U}_x = \mathbf{0},$$

262 where,  $\mathbf{A}(\mathbf{U}) = \mathbf{F}'(\mathbf{U})$  is the Jacobi matrix,

$$263 \quad (3.2) \quad \mathbf{A}(\mathbf{U}) = \begin{pmatrix} 0 & 1 & 0 \\ \frac{1}{2}(\gamma - 3)u^2 & (3 - \gamma)u & \gamma - 1 \\ \frac{1}{2}(\gamma - 1)u^3 - uH & H - (\gamma - 1)u^2 & \gamma u \end{pmatrix} = \begin{pmatrix} \mathbf{a}_1(\mathbf{U}) \\ \mathbf{a}_2(\mathbf{U}) \\ \mathbf{a}_3(\mathbf{U}) \end{pmatrix},$$

264 with  $H = (E + p)/\rho$ . And the Jacobi matrix is diagonalizable:

$$\mathbf{A}(\mathbf{U}) = \mathbf{F}'(\mathbf{U}) = \mathbf{R}\mathbf{\Lambda}\mathbf{L}.$$

265 Here,  $\mathbf{\Lambda} = \text{diag}(u - c, u, u + c)$ ,  $c = \sqrt{\gamma p/\rho}$  is the speed of sound,  $\mathbf{R}$  and  $\mathbf{L}$  are  
 266 matrices as follows:

$$267 \quad \mathbf{R}(\mathbf{U}) = \begin{pmatrix} 1 & 1 & 1 \\ u - c & u & u + c \\ H - uc & \frac{1}{2}u^2 & H + uc \end{pmatrix},$$

269

$$270 \quad \mathbf{L}(\mathbf{U}) = \frac{1}{c^2} \begin{pmatrix} \frac{1}{2}uc + \frac{1}{4}(\gamma-1)u^2 & -\frac{1}{2}(\gamma-1)u - \frac{1}{2}c & \frac{1}{2}(\gamma-1) \\ c^2 - \frac{1}{2}(\gamma-1)u^2 & (\gamma-1)u & 1-\gamma \\ -\frac{1}{2}uc + \frac{1}{4}(\gamma-1)u^2 & -\frac{1}{2}(\gamma-1)u + \frac{1}{2}c & \frac{1}{2}(\gamma-1) \end{pmatrix} = \begin{pmatrix} \mathbf{l}_1(\mathbf{U}) \\ \mathbf{l}_2(\mathbf{U}) \\ \mathbf{l}_3(\mathbf{U}) \end{pmatrix}.$$

271

272 The number of boundary conditions we need to give is determined by the sign  
273 of the eigenvalues of Jacobi matrix  $\mathbf{A}(\mathbf{U})$  at the boundary. Specifically, on the left  
274 boundary  $x = -1$ , it can be divided into the following cases:

275 Case 1.  $u - c > 0$ : three boundary conditions need to be given;

276 Case 2.  $u - c \leq 0$ ,  $u > 0$ : two boundary conditions need to be given;

277 Case 3.  $u \leq 0$ ,  $u + c > 0$ : only one boundary condition need to be given;

278 Case 4.  $u + c \leq 0$ : no boundary conditions are required.

279 In particular, for case 3, when the given boundary condition is  $U_2(-1, t) = 0$ , we get  
280 the free slip solid wall boundary condition.

Here, we take the case 2 as an example to describe our algorithm. Suppose two  
boundary conditions are given at the left boundary,

$$U_1(-1, t) = g_1(t), \quad \text{and} \quad U_2(-1, t) = g_2(t).$$

281 Again, we perform uniform mesh generation on the computational domain with mesh  
282 size  $\Delta x$ ,

283

$$-1 + C_a \Delta x = x_0 < \dots < x_N = 1 - C_b \Delta x.$$

284 Employ the finite difference methods to get the semi-discrete scheme:

$$285 \quad (3.3) \quad \frac{d}{dt} \mathbf{U}_j = -\frac{1}{\Delta x} (\hat{\mathbf{F}}_{j+1/2} - \hat{\mathbf{F}}_{j-1/2}), \quad j = 0, \dots, N.$$

286 Here,  $\mathbf{U}_j(t)$  is the approximation of  $\mathbf{U}(x_j, t)$ , and numerical flux  $\hat{\mathbf{F}}_{j+1/2}$  can be ob-  
287 tained by the WENO reconstruction. We take the fifth order scheme as an example  
288 to describe our boundary algorithm, and other high order schemes can be obtained  
289 similarly. For the fifth order WENO scheme, we need the values of at three ghost  
290 points near the boundary  $x = -1$ , which are  $\mathbf{U}_{-1}, \mathbf{U}_{-2}$  and  $\mathbf{U}_{-3}$ .

291 Similar to the case of scalar equation, in order to obtain the ghost point values,  
292 we will construct the extrapolation polynomials  $\mathbf{q}(x)$  near the boundary. It can be  
293 seen from Table 3 that, for the fifth order scheme, we have to use the point values  
294 and first order spatial derivatives on the boundary through the ILW procedure. To  
295 ensure the order of accuracy, the value  $\mathbf{U}^{*(0)}$  and  $\mathbf{U}^{*(1)}$  should be 5th and 4th order  
296 approximations of  $\mathbf{U}|_{x=-1}$  and  $\mathbf{U}_x|_{x=-1}$ , respectively.

Specifically, we use the left characteristic matrix  $\mathbf{L} = \mathbf{L}(\mathbf{U}_0)$  to do the character-  
istic projection  $\mathbf{V} = \mathbf{L}\mathbf{U}$ . Here,

$$\mathbf{V} = (V_1, V_2, V_3)^T$$

297 is the characteristic variable. In case 2,  $V_1$  is the outflow variable,  $V_2$  and  $V_3$  are the  
298 inflow variables. Combined with the boundary conditions, we can obtain the following  
299 linear system:

$$300 \quad (3.4) \quad \begin{aligned} U_1^{*(0)} &= g_1(t), \\ U_2^{*(0)} &= g_2(t), \\ \mathbf{l}_1 \cdot \mathbf{U}^{*(0)} &= V_1^{*(0)}, \end{aligned}$$

301 where,  $V_1^{*(0)}$  is the 5th order approximation of  $V_1|_{x=-1}$  and can be extrapolated from  
 302 the interior grid points. By solving the above system, we can get the value of  $\mathbf{U}^{*(0)}$ .

303 For  $\mathbf{U}^{*(1)}$ , applying the ILW procedure, we have

$$304 \quad (3.5a) \quad \begin{aligned} \mathbf{a}_1(\mathbf{U}^{*(0)}) \cdot \mathbf{U}^{*(1)} &= -g_1'(t), \\ \mathbf{a}_2(\mathbf{U}^{*(0)}) \cdot \mathbf{U}^{*(1)} &= -g_2'(t), \end{aligned}$$

305 Combine with outflow conditions,

$$306 \quad (3.5b) \quad \mathbf{l}_1 \cdot \mathbf{U}^{*(1)} = V_1^{*(1)},$$

307 where,  $V_1^{*(1)}$  is the 4th order approximation of  $(V_1)_x|_{x=-1}$  and can be extrapolated  
 308 from the interior grid points. By combining and solving the above equations, we can  
 309 get the value of  $\mathbf{U}^{*(1)}$ .

310 At this time, we can construct the point values of the ghost points through the  
 311 new SILW method. The specific construction method is as follows:

Step 1. Construct the vector of interpolation polynomial  $\mathbf{p}(x)$  of degree four with the  
 interior grid points values  $\{\mathbf{U}_0, \dots, \mathbf{U}_4\}$ . Let

$$\mathbf{U}_{k*} = \mathbf{p}(-1 + k\alpha\Delta x), \quad 1 \leq k \leq 3,$$

312 Here,  $\alpha$  can be selected as any number in  $[0.92, 5.11]$  according to the stability  
 313 analysis results in Table 3.

314 Step 2. Construct the extrapolation polynomial  $\mathbf{q}(x)$  of degree four to satisfy:

$$315 \quad \begin{aligned} \mathbf{q}(-1) &= \mathbf{U}^{*(0)} \\ \mathbf{q}'(-1) &= \mathbf{U}^{*(1)} \\ \mathbf{q}(-1 + k\alpha\Delta x) &= \mathbf{U}_{k*}, \quad 1 \leq k \leq 3 \end{aligned}$$

316 Step 3. Let the ghost points values be the values of the extrapolation polynomial  $\mathbf{q}(x)$   
 at the corresponding points:

$$\mathbf{U}_j = \mathbf{q}(x_j) \quad j = -1, -2, -3.$$

317 In particular, if the solution of the equation has discontinuities near the boundary,  
 318 we can apply the WENO extrapolation technique when constructing the extrapolation  
 319 polynomial  $\mathbf{q}(x)$  in step 3. More details about WENO extrapolation can be found  
 320 in [19, 28, 31]. In addition, in many cases, boundary conditions are not directly given  
 321 to the conserved variables. For example, the temperature or pressure is given on the  
 322 boundary. For these cases, when dealing with the boundary conditions, we need to  
 323 convert the conservation equation into equations in terms of primitive variables. This  
 324 process is used in [3, 16, 17, 30], which is very similar to the above process of conserved  
 325 variables. We will not expand it here.

326 **3.2. The new SILW method for two-dimensional Euler equation.** Con-  
 327 sider the two-dimensional Euler equation as follows:

$$328 \quad (3.6) \quad \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{U})}{\partial y} = \mathbf{0}, \quad (x, y)^T \in \Omega,$$

329 where,

$$330 \quad (3.7) \quad \mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E + p) \end{pmatrix}, \quad \mathbf{G}(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E + p) \end{pmatrix}.$$

331 Here,  $\rho$ ,  $\mathbf{u} = (u, v)^T$ ,  $p$  and  $E$  represent the density, velocity, pressure and total energy  
 332 per volume, respectively. In order to close the system, we give the following equation  
 333 of state of ideal gas:

$$334 \quad E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho(u^2 + v^2).$$

335 Here,  $\gamma$  is the adiabatic constant, which equals to 1.4 for an ideal polytropic gas.  
 We use a uniform non body-fitted Cartesian mesh to divide the domain

$$x_{i+1} = x_i + \Delta x, \quad y_{j+1} = y_j + \Delta y,$$

336 with mesh size  $\Delta x$  and  $\Delta y$  in  $x$ - and  $y$ -direction, respectively. Discretize the equation  
 337 into the following conservative semi-discrete scheme:

$$338 \quad \frac{d\mathbf{U}_{i,j}}{dt} + \frac{\hat{\mathbf{F}}_{i+\frac{1}{2},j} - \hat{\mathbf{F}}_{i-\frac{1}{2},j}}{\Delta x} + \frac{\hat{\mathbf{G}}_{i,j+\frac{1}{2}} - \hat{\mathbf{G}}_{i,j-\frac{1}{2}}}{\Delta y} = \mathbf{0},$$

339 where,  $\mathbf{U}_{i,j}(t)$  is approximation to the exact solution  $\mathbf{U}(x_i, y_j, t)$ ,  $\hat{\mathbf{F}}_{i+\frac{1}{2},j}$  and  $\hat{\mathbf{G}}_{i,j+\frac{1}{2}}$   
 340 are numerical fluxes, which can be obtained by the 5th order WENO reconstruction.

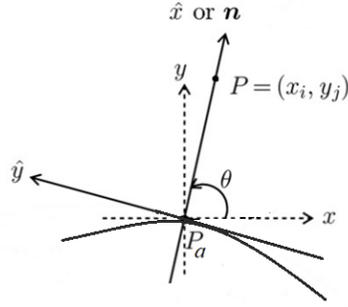


Fig. 3. The local coordinate rotation diagram.

341 Suppose  $P_{ij} = (x_i, y_j)$  is a ghost point near the boundary. At first, we find its  
 342 foot point  $P_a \in \partial\Omega(t_n)$ , so that the normal  $\mathbf{n}$  at  $P_a$  goes through  $P_{ij}$ , as shown in  
 343 Figure 3. Assume the normal vector from  $P_a$  to  $P_{i,j}$  is  $\mathbf{n} = (\cos \theta, \sin \theta)^T$ . In order to  
 344 simplify the algorithm, we perform a local coordinate rotation transformation at  $P_a$ ,

$$345 \quad \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}.$$

346 In the new coordinate system, the equation (3.6) can be rewritten as

$$347 \quad (3.8) \quad \frac{\partial \hat{\mathbf{U}}}{\partial t} + \frac{\partial \mathbf{F}(\hat{\mathbf{U}})}{\partial \hat{x}} + \frac{\partial \mathbf{G}(\hat{\mathbf{U}})}{\partial \hat{y}} = \mathbf{0},$$

348 where,

$$349 \quad \hat{\mathbf{U}} = \begin{pmatrix} \rho \\ \rho \hat{u} \\ \rho \hat{v} \\ E \end{pmatrix} = \begin{pmatrix} \hat{U}_1 \\ \hat{U}_2 \\ \hat{U}_3 \\ \hat{U}_4 \end{pmatrix}, \quad \begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}.$$

350 Let

$$351 \quad \mathbf{A}(\hat{U}) = \begin{pmatrix} \mathbf{a}_1(\hat{U}) \\ \mathbf{a}_2(\hat{U}) \\ \mathbf{a}_3(\hat{U}) \\ \mathbf{a}_4(\hat{U}) \end{pmatrix} = \mathbf{F}'(\hat{U}), \quad \mathbf{Res} = -\frac{\partial \mathbf{G}(\hat{U})}{\partial \hat{y}} = \begin{pmatrix} \mathit{Res}_1 \\ \mathit{Res}_2 \\ \mathit{Res}_3 \\ \mathit{Res}_4 \end{pmatrix}.$$

352 Then, the equations can be written in the following non conservative form

$$353 \quad (3.9) \quad \hat{U}_t + \mathbf{A}(\hat{U})\hat{U}_x = \mathbf{Res}.$$

The original equation is hyperbolic, so  $\mathbf{A}(\hat{U})$  is diagonalizable:

$$\mathbf{A}(\hat{U}) = \mathbf{R}(\hat{U})\mathbf{\Lambda}(\hat{U})\mathbf{L}(\hat{U}),$$

Here,

$$354 \quad \mathbf{\Lambda}(\hat{U}) = \mathit{diag}(\hat{u} - c, \hat{u}, \hat{u}, \hat{u} + c)$$

$$355 \quad \mathbf{L}(\hat{U}) = \begin{pmatrix} \mathit{l}_1(\hat{U}) \\ \mathit{l}_2(\hat{U}) \\ \mathit{l}_3(\hat{U}) \\ \mathit{l}_4(\hat{U}) \end{pmatrix}.$$

356 The number of boundary conditions that should be given at the boundary point  
 357  $P_a$  is related to the eigenvalues  $\hat{u} - c, \hat{u}, \hat{u}, \hat{u} + c$  at this point. Specifically, it can be  
 358 divided into the following situations:

359 Case 1:  $\hat{u} - c > 0$ , no boundary conditions are required;

360 Case 2:  $\hat{u} - c \leq 0$ ,  $\hat{u} > 0$ , only one boundary condition needs to be given;

361 Case 3:  $\hat{u} \leq 0$ ,  $\hat{u} + c > 0$ , three boundary conditions need to be given;

362 Case 4:  $\hat{u} + c \leq 0$ , four boundary conditions need to be given.

We take case 2 as an example to describe our algorithm. Suppose the boundary  
 condition given at the boundary point  $P_a$  is

$$\hat{U}_2 = g(t).$$

363 As before, we can transform the construction problem of ghost point values into  
 364 the construction problem of extrapolation polynomial  $\mathbf{q}(s)$ . It also can be seen from  
 365 Table 3 that, for the fifth order scheme, we need to use the 0th and 1st order normal  
 366 direction derivatives on the boundary obtained from the ILW procedure when con-  
 367 structing  $\mathbf{q}(s)$ . That is, we need to get the value of  $\hat{U}^{*(0)}$  and  $\hat{U}^{*(1)}$ , which are the  
 368 5th and 4th order approximations of  $\hat{U}|_{P_a}$  and  $\hat{U}_{\hat{x}}|_{P_a}$  respectively, through the ILW  
 369 procedure.

370 Specifically, we use the left characteristic matrix  $\mathbf{L} = \mathbf{L}(\hat{U}_{ext})$  to do the char-  
 371 acteristic projection  $\mathbf{V} = \mathbf{L}\hat{U}$ . Here,  $\hat{U}_{ext}$  is the extrapolation value at  $P_a$ ,  $\mathbf{V} =$   
 372  $(V_1, V_2, V_3, V_4)^T$  is the characteristic variable. For case 2,  $V_2, V_3, V_4$  are the outflow  
 373 variables,  $V_1$  is the inflow variable.

374 Combined with the boundary conditions, we can obtain the following linear sys-  
 375 tem:

$$376 \quad (3.10) \quad \begin{aligned} \hat{U}_2^{*(0)} &= g(t), \\ \mathit{l}_2 \cdot \hat{U}^{*(0)} &= V_2^{*(0)}, \\ \mathit{l}_3 \cdot \hat{U}^{*(0)} &= V_3^{*(0)}, \\ \mathit{l}_4 \cdot \hat{U}^{*(0)} &= V_4^{*(0)}, \end{aligned}$$

377 where,  $V_2^{*(0)}, V_3^{*(0)}$  and  $V_4^{*(0)}$  can be extrapolated from the interior grid points. By  
 378 solving the above system, we can get the value of  $\hat{U}^{*(0)}$ .

379 For  $\hat{U}^{*(1)}$ , apply the ILW procedure and we have

$$380 \quad (3.11a) \quad \mathbf{a}_2(\hat{U}^{*(0)}) \cdot \hat{U}^{*(1)} = -g'(t) + Res_2.$$

381 Combine with the outflow conditions,

$$382 \quad (3.11b) \quad \begin{aligned} \mathbf{l}_2 \cdot \hat{U}^{*(1)} &= V_1^{*(2)}, \\ \mathbf{l}_3 \cdot \hat{U}^{*(1)} &= V_1^{*(3)}, \\ \mathbf{l}_4 \cdot \hat{U}^{*(1)} &= V_1^{*(4)}, \end{aligned}$$

383 where,  $V_2^{*(1)}, V_3^{*(1)}$  and  $V_4^{*(1)}$  can be extrapolated from the interior grid points. By  
 384 solving the above equations, we can get the value of  $\hat{U}^{*(1)}$ .

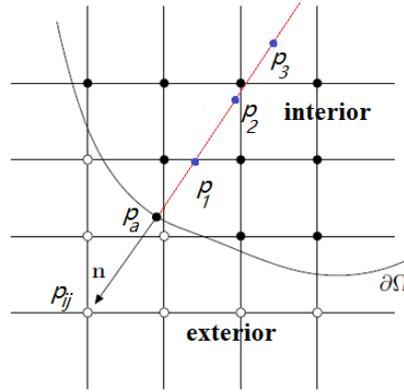


Fig. 4. Two dimensional new SILW method diagram

385 Next, we use the new SILW method to construct the value of ghost point  $P_{i,j}$ . It  
 386 is divided into the following steps:

387 Step 1. Obtain the interpolation polynomial  $\mathbf{p}(x, y)$  of degree 4 with the values inter-  
 388 nal grid points near  $P_a$ .

Step 2. Let

$$\mathbf{U}_{k*} = \mathbf{p}(P_k), \quad 1 \leq k \leq 3,$$

where,

$$P_k = P_a - k\alpha h \mathbf{n}, \quad 1 \leq k \leq 3$$

389 As show in Figure 4,  $\{P_k\}_{k=1}^3$  are some non grid points in the interior area  
 390 on the normal line. Here,  $h = \sqrt{\Delta x^2 + \Delta y^2}$ ,  $\alpha$  can be chosen as any number  
 391 in  $[0.92 \frac{\max(\Delta x, \Delta y)}{h}, 5.11 \frac{\min(\Delta x, \Delta y)}{h}]$ .

392 Step 3. Let  $\mathbf{q}(s)$  be the unique polynomial of degree 4 satisfy

$$393 \quad \begin{aligned} \mathbf{q}(0) &= \mathbf{U}^{*(0)}, \\ \mathbf{q}'(0) &= \mathbf{U}^{*(1)}, \\ 394 \quad \mathbf{q}(-k\alpha h) &= \mathbf{U}_{k*}, \quad 1 \leq k \leq 3. \end{aligned}$$

Step 4. Take the function value at  $P_{ij}$

$$U_{ij} = \mathbf{q}(|P_{ij} - P_a|).$$

It should be noted that when  $g(t) = 0$ , we actually get the non penetrating free slip boundary condition:

$$\mathbf{u} \cdot \mathbf{n} = 0.$$

395 As in the one-dimensional case, if the solution is discontinuous near the boundary, we  
396 can use a one-dimensional WENO extrapolation technique in step 3.

397 Notice that, for the problems with changing wind direction, the above inverse Lax-  
398 Wendroff procedure may involve solving an ill-conditioned linear algebraic system,  
399 which may ruin the accuracy or even lead to blowing up. There are two ways to deal  
400 with this problem. One is mentioned in [28], which adds additional extrapolation  
401 equations and solves a least squares problem whenever one of the eigenvalues very  
402 closed to 0. The other method is proposed in [19], which evaluates the solution values  
403 and the flux values at ghost points separately. In the test examples of this paper,  
404 we use the first method. The performance of the second method applied to our new  
405 SILW boundary treatment is unclear.

**4. Numerical tests.** We take some numerical tests to show the efficiency and stability of our new proposed SILW method. We use the third and fifth order FD-WENO scheme for spatial discretization. Correspondingly, the new SILW boundary treatment with third order and fifth order accuracy will be used, respectively. For all the one-dimensional numerical tests we take the parameter  $\alpha = 1.0$ , for all the two-dimensional numerical tests we take  $\alpha = 1.25$ . The third order TVD RK scheme (2.4) is employed for time discretization, with the time step

$$\Delta t = \text{CFL} \frac{\Delta x^{k/3}}{c}$$

for one-dimensional problems, and

$$\Delta t = \frac{\text{CFL}}{c_x/\Delta x^{k/3} + c_y/\Delta y^{k/3}}.$$

406 for two-dimensional problems. Here, the index  $k/3$  help us to guarantee  $k$ -th order  
407 in time.  $c = c_x = \max_{\mathbf{U}} |\lambda(\mathbf{F}'(\mathbf{U}))|$ ,  $c_y = \max_{\mathbf{U}} |\lambda(\mathbf{g}'(\mathbf{u}))|$ , and  $\lambda$  is the eigenvalue of  
408 the Jacobian matrix. Throughout our numerical tests, the CFL number is taken as  
409 0.6.

410 **Example 1.** At first, we consider the accuracy test of the new SILW on the one-  
411 dimensional Euler equation on the computational domain as  $[-\pi, \pi]$ . We choose  
412 suitable boundary conditions such that the exact solution is:

$$413 \quad (4.1) \quad \begin{cases} \rho(x, t) = 1 - 0.2 \sin(2t - x), \\ u(x, t) = 2, \\ p(x, t) = 2. \end{cases}$$

414 In order to verify the applicability of our algorithm to the “cut cell” problem, we  
415 test with difference choices of  $C_a$  and  $C_b$ . The computational errors about density  $\rho$   
416 at final time  $t_{end} = 1$  are shown in Table 4 - 5. We can see that for all cases, the  
417 schemes can achieve the designed order accuracy with mesh refinements.

**Table 4.** Example 1: errors and orders of accuracy of  $\rho$  with third order scheme.

N	$C_a = 0.0001, C_b = 0.7$				$C_a = 0.9999, C_b = 0.7$			
	$L^1$ error	order	$L^\infty$ error	order	$L^1$ error	order	$L^\infty$ error	order
20	1.67E-004	–	5.73E-004	–	1.45E-004	–	4.80E-004	–
40	1.58E-005	3.40	4.78E-005	3.58	1.00E-005	3.85	4.39E-005	3.45
80	2.07E-006	2.92	5.30E-006	3.17	8.52E-007	3.56	4.57E-006	3.26
160	2.69E-007	2.94	7.30E-007	2.86	8.38E-008	3.34	5.24E-007	3.12
320	3.43E-008	2.97	9.50E-008	2.94	9.15E-009	3.19	6.27E-008	3.06
640	4.32E-009	2.98	1.21E-008	2.96	1.06E-009	3.10	7.66E-009	3.03

**Table 5.** Example 1: errors and orders of accuracy of  $\rho$  with fifth order scheme.

N	$C_a = 0.0001, C_b = 0.7$				$C_a = 0.9999, C_b = 0.7$			
	$L^1$ error	order	$L^\infty$ error	order	$L^1$ error	order	$L^\infty$ error	order
20	9.33E-005	–	1.76E-004	–	7.41E-005	–	1.36E-004	–
40	2.99E-006	4.96	6.09E-006	4.84	2.62E-006	4.81	5.47E-006	4.63
80	9.28E-008	5.01	1.99E-007	4.93	8.65E-008	4.92	1.81E-007	4.91
160	2.88E-009	5.01	6.06E-009	5.04	2.77E-009	4.96	5.78E-009	4.97
320	8.89E-011	5.01	1.77E-010	5.09	8.72E-011	4.99	1.73E-010	5.05

418 **Example 2.** Next, we consider the example given in [12] to test the accuracy of  
 419 our method. The governing equations is still the one-dimensional compressible Euler  
 420 equation, with following initial condition:

$$421 \quad (4.2) \quad \begin{cases} \rho(x, 0) = \frac{1 + 0.2 \sin(x)}{2\sqrt{3}}, \\ u(x, 0) = \sqrt{\gamma} \rho(x, 0), \\ p(x, 0) = \rho(x, 0)^\gamma. \end{cases}$$

422 The computational domain is taken as  $[0, 2\pi]$ . We choose the parameter  $\gamma = 3$ .  
 423 Consequently, the exact solution is

$$424 \quad \rho(x, t) = \frac{\mu(x, t)}{2\sqrt{3}}, \quad u(x, t) = \sqrt{\gamma} \rho(x, t), \quad p(x, t) = \rho(x, t)^\gamma,$$

425 where  $\mu(x, t)$  is the solution of the following Burgers' equation:

$$426 \quad (4.3) \quad \begin{cases} \mu_t + \left(\frac{\mu^2}{2}\right)_x = 0, & 0 < x < 2\pi, \quad t > 0, \\ \mu(x, 0) = 1 + 0.2 \sin(x), & 0 \leq x \leq 2\pi. \end{cases}$$

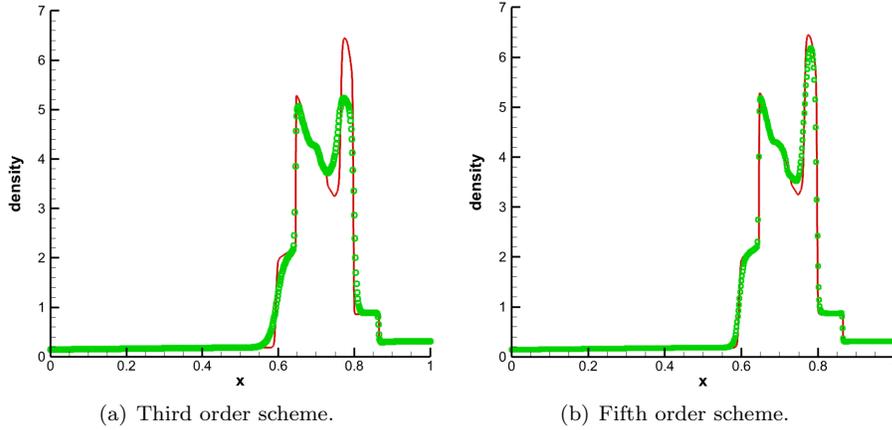
427 We take boundary conditions from the exact solution whenever needed.

428 We consider the extrema situation and set  $C_a = 0.0001, C_b = 0.9999$ . The com-  
 429 putational errors about density  $\rho$  and orders of accuracy at time  $t_{end} = 3.0$  are shown  
 430 in Table 6, indicating that our methods can achieve the designed third order or fifth  
 431 order accuracy.

432 **Example 3.** Now we consider the interaction of two blast waves [28]. In this problem,  
 433 multiple reflections occur between shock and rarefaction off the walls. The initial

**Table 6.** Example 2: errors and orders of accuracy of  $\rho$ .

$N$	third order scheme				fifth order scheme			
	$L^1$ error	order	$L^\infty$ error	order	$L^1$ error	order	$L^\infty$ error	order
40	1.30E-003	—	2.49E-003	—	4.01E-004	—	1.04E-003	—
80	2.04E-004	2.67	6.20E-004	2.00	2.24E-005	4.15	8.03E-005	3.69
160	2.84E-005	2.84	8.50E-005	2.86	7.16E-007	4.97	2.98E-006	4.75
320	3.60E-006	2.97	9.93E-006	3.09	2.38E-008	4.90	1.00E-007	4.89
640	4.52E-007	2.99	1.18E-006	3.06	8.04E-010	4.89	2.90E-009	5.10



**Fig. 5.** Example 3: Density profiles.  $h = 1/640$ . The solid line represents the reference solution and the circle represents the numerical solution.

434 condition is

$$435 \quad (4.4) \quad U(x, 0) = \begin{cases} U_L, & x < 0.1, \\ U_M, & 0.1 < x < 0.9, \\ U_R, & x > 0.9. \end{cases}$$

436 Here,  $\rho_L = \rho_M = \rho_R = 1$ ,  $u_L = u_M = u_R = 0$ ,  $p_L = 10^3$ ,  $p_M = 10^{-2}$ , and  $p_R = 10^2$ .  
 437 We take  $t_{end} = 0.038$  and  $C_a = 0.0001$ ,  $C_b = 0.7$ . At the same time, we use a very  
 438 dense grid with  $\Delta x = 1/2560$  and the original ILW method to obtain the reference  
 439 solution. The numerical results are shown in Figure 5. We can see that the new ILW  
 440 method can distinguish the structure of the solution well, and higher order scheme  
 441 has a better approximation to the complex structure.

442 **Example 4.** We consider two-dimensional linear scalar equations on a disk:

$$443 \quad (4.5) \quad u_t + u_x + u_y = 0, \quad (x, y)^T \in \Omega = \{(x, y) : x^2 + y^2 < 0.5\}.$$

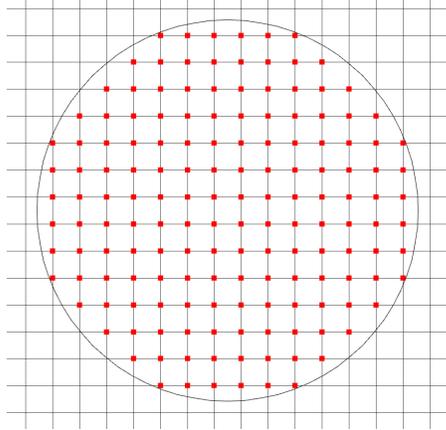
The initial condition is given as

$$u(x, y, 0) = 0.25 + 0.5 \sin[\pi(x + y)],$$

and the boundary is given whenever needed such that the exact solution is

$$u(x, y, t) = 0.25 + 0.5 \sin[\pi(x + y - 2t)].$$

444 The domain is discretized by embedding the domain in a regular Cartesian mesh with  
 445  $x_i = (i - \frac{1}{2})\Delta x, y_j = (j - \frac{1}{2})\Delta y$ , and the non body-fitted Cartesian mesh  $h = \Delta x = \Delta y$ .  
 446 We show Figure 6 as an example. The final time is taken as  $t_{end} = 1.0$ . The numerical  
 447 results are given in Table 7, indicating that our schemes are stable and can achieve  
 448 the design order of accuracy.



**Fig. 6.** Example 4: Non body-fitted Cartesian mesh. The red points are the interior points.

**Table 7.** Example 4: errors and orders of accuracy .

$h$	third order scheme				fifth order scheme			
	$L^1$ error	order	$L^\infty$ error	order	$L^1$ error	order	$L^\infty$ error	order
1/10	1.28E-004	–	4.83E-004	–	4.16E-004	–	1.42E-003	–
1/20	1.33E-005	3.26	4.54E-005	3.41	1.51E-005	4.77	1.38E-004	3.36
1/40	1.43E-006	3.21	5.84E-006	2.95	3.47E-007	5.44	5.54E-006	4.64
1/80	1.46E-007	3.28	6.68E-007	3.12	1.17E-008	4.89	2.71E-007	4.35
1/160	1.19E-008	3.61	8.82E-008	2.92	4.47E-010	4.70	9.72E-009	4.80

449 **Example 5.** We test the vortex evolution problem for the 2D Euler equation. The  
 450 mean flow is  $\rho = u = v = p = 1$  with following isentropic vortex perturbation centered  
 451 at  $(x_0, y_0) = (0, 0)$  (perturbation in  $(u, v)$  and temperature  $T = p/\rho$ , no perturbation  
 452 in the entropy  $S = p/\rho^\gamma$ ):

$$\begin{aligned}
 (\delta u, \delta v) &= \frac{\epsilon}{2\pi} e^{0.5(1-r^2)} (-\bar{y}, \bar{x}), \\
 \delta T &= -\frac{(\gamma-1)\epsilon^2}{8\gamma\pi^2} e^{(1-r^2)}, \\
 \delta S &= 0.
 \end{aligned}
 \tag{4.6}$$

454 where  $(\bar{x}, \bar{y}) = (x - x_0, y - y_0)$ ,  $r^2 = \bar{x}^2 + \bar{y}^2$ , and the vortex strength  $\epsilon = 5$ . It  
 455 is clear that the exact solution is just the passive convection of the vortex with the  
 456 mean velocity. The computational domain is taken as  $(-0.5, 1) \times (-0.5, 1)$  and the  
 457 final time is taken as  $t_{end} = 1.0$ . The boundary conditions are taken from the exact

458 solution whenever needed. We divide the domain with the uniform Cartesian mesh  
 459  $x_i = (i - \frac{1}{2})h$  and  $y_j = (j - \frac{1}{2})h$ , with mesh size  $h = 1.5/N$ . The numerical results in  
 460 Table 8 show that the schemes are stable and can reach the designed high order.

**Table 8.** Example 5: errors and orders of accuracy of  $\rho$ .

$h$	third order scheme				fifth order scheme			
	$L^1$ error	order	$L^\infty$ error	order	$L^1$ error	order	$L^\infty$ error	order
3/40	1.73E-004	–	3.05E-004	–	3.39E-005	–	7.12E-005	–
3/80	2.17E-005	2.99	4.10E-005	2.89	1.09E-006	4.95	2.33E-006	4.93
3/160	2.51E-006	3.10	4.93E-006	3.05	3.46E-008	4.98	1.08E-007	4.43
3/320	2.99E-007	3.07	6.32E-007	2.96	1.12E-009	4.94	4.71E-009	4.51
3/640	3.63E-008	3.04	8.34E-008	2.92	3.77E-011	4.89	1.90E-010	4.63

461 **Example 6.** Next, we consider 2D version of Example 2 [12]. The governing equation  
 462 is the two-dimensional compressible Euler equations with following initial condition:

$$463 \quad (4.7) \quad \begin{cases} \rho(x, y, 0) = \frac{1 + 0.2 \sin(\frac{x+y}{2})}{\sqrt{6}}, \\ u(x, y, 0) = v(x, y, 0) = \sqrt{\frac{\gamma}{2}} \rho(x, y, 0), \\ p(x, y, 0) = \rho(x, y, 0)^\gamma. \end{cases}$$

464 We choose the parameter  $\gamma = 3$ , such that the exact solution is

$$465 \quad \rho(x, y, t) = \frac{\mu(x, y, t)}{\sqrt{6}}, \quad u(x, y, t) = v(x, y, t) = \sqrt{\frac{\gamma}{2}} \rho(x, y, t), \quad p(x, y, t) = \rho(x, y, t)^\gamma,$$

466 where  $\mu(x, y, t)$  is the solution of the following 2D Burgers' equation:

$$467 \quad (4.8) \quad \begin{cases} \mu_t + (\frac{\mu^2}{2})_x + (\frac{\mu^2}{2})_y = 0, & (x, y) \in \Omega \\ \mu(x, y, 0) = 1 + 0.2 \sin(\frac{x+y}{2}). \end{cases}$$

468 We consider following two computational domains:

$$469 \quad (4.9a) \quad \Omega = [0, 4\pi] \times [0, 4\pi],$$

$$470 \quad (4.9b) \quad \Omega = \{(x, y) | x^2 + y^2 < (1.5\pi)^2\},$$

472 and take boundary conditions from the exact solution whenever needed. For the  
 473 square domain (4.9a), we use a grid similar to example 5. And for the circular domain  
 474 (4.9b), we use a non body-fitted grid similar to example 4. The numerical results at  
 475 final time  $t_{end} = 1$  are shown in Table 9 - 10. We can see that our schemes are stable  
 476 and high order accuracy for all cases.

477 **Example 7.** We consider a flow around a cylinder. The center of the bottom surface  
 478 of the cylinder is at the origin with a radius of 1. At the initial moment, a fluid  
 479 with Mach 3 moves towards the cylinder. In consideration of the symmetry, we  
 480 only consider the problem of an upper half plane. For the lower boundary of the  
 481 computation area at  $y = 0$ , we use the reflection technique; for the left boundary

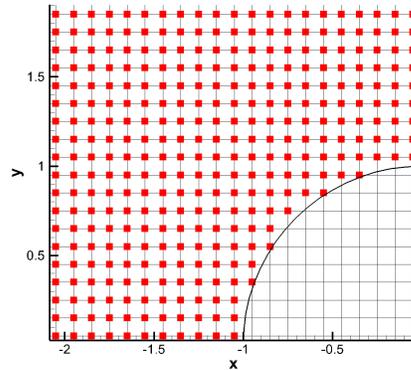
**Table 9.** Example 6: The errors and the orders of accuracy of  $\rho$  on the square domain  $\Omega = [0, 4\pi] \times [0, 4\pi]$ .

$h$	third order scheme				fifth order scheme			
	$L^1$ error	order	$L^\infty$ error	order	$L^1$ error	order	$L^\infty$ error	order
$4\pi/100$	4.09E-004	–	1.13E-005	–	1.42E-005	–	2.08E-006	–
$4\pi/150$	1.22E-004	2.98	3.51E-006	2.89	1.93E-006	4.92	2.89E-007	4.86
$4\pi/200$	5.18E-005	2.98	1.66E-006	2.60	4.59E-007	4.99	7.55E-008	4.67
$4\pi/250$	2.66E-005	2.98	9.14E-007	2.67	1.49E-007	5.01	2.77E-008	4.48
$4\pi/300$	1.54E-005	2.99	5.54E-007	2.74	6.01E-008	5.01	1.20E-008	4.56

**Table 10.** Example 6: The errors and the orders of accuracy of  $\rho$  on the circular domain  $\Omega = \{(x, y) | x^2 + y^2 < (1.5\pi)^2\}$ .

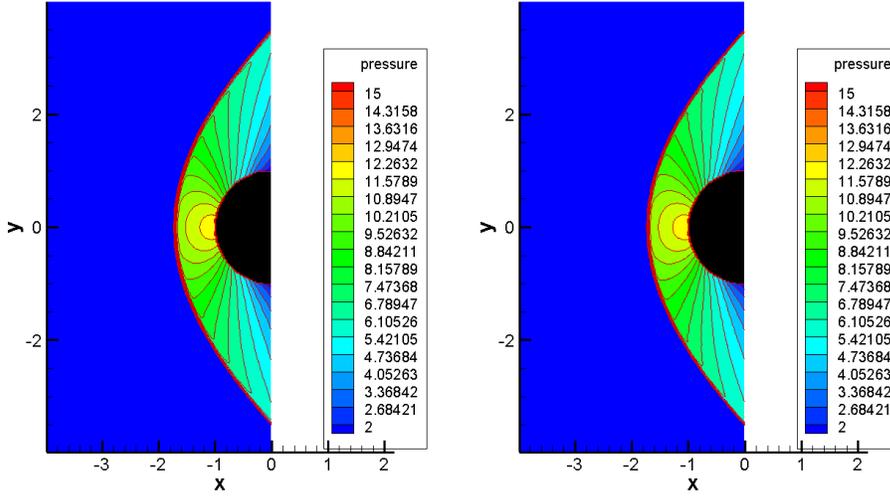
$h$	third order scheme				fifth order scheme			
	$L^1$ error	order	$L^\infty$ error	order	$L^1$ error	order	$L^\infty$ error	order
$4\pi/100$	1.46E-004	–	1.14E-005	–	1.48E-005	–	8.59E-006	–
$4\pi/150$	4.20E-005	3.08	4.28E-006	2.42	2.13E-006	4.78	1.28E-006	4.69
$4\pi/200$	1.81E-005	2.91	1.79E-006	3.03	6.65E-007	4.05	3.51E-007	4.49
$4\pi/250$	9.20E-006	3.05	8.90E-007	3.12	2.21E-007	4.91	1.16E-007	4.95
$4\pi/300$	5.44E-006	2.87	5.23E-007	2.91	8.42E-008	5.31	4.72E-008	4.93

482 of the computation region at  $x = -4$ , we give the inflow boundary condition; for  
 483 the right boundary  $x = 0$  and the upper boundary  $y = 6$  of the computation area,  
 484 the outflow boundary conditions are given. On the surface of a cylinder, our new  
 485 ILW method is used to deal with a no-penetration boundary condition. As before, a  
 486 uniform non body-fitted Cartesian mesh is used, which is shown in Figures 7. Figure  
 487 8 show the numerical results. We can see that the results are comparable with those  
 488 in [5, 19, 28, 31].



**Fig. 7.** Example 7: The non body-fitted Cartesian mesh with near the cylinder boundary. The red points are the interior points.

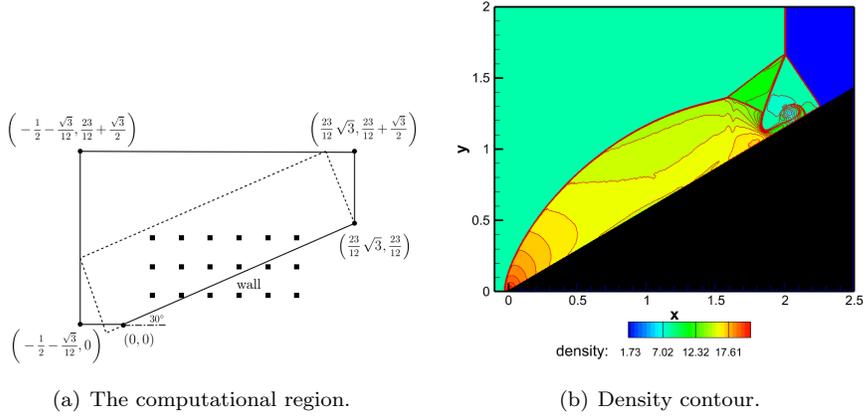
489 **Example 8.** We consider the double Mach reflection problem. At the initial moment,



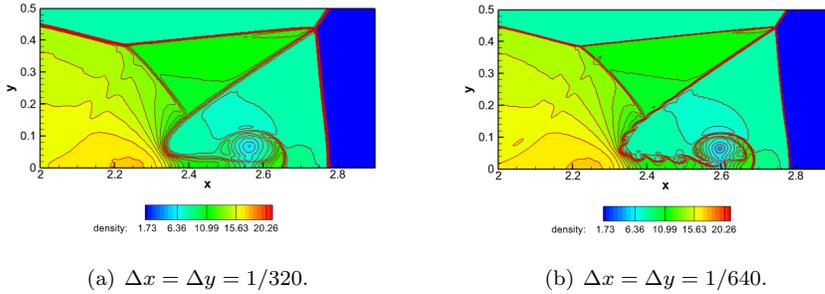
**Fig. 8.** Example 7: Pressure contour of flow past a cylinder, 20 contours from 2 to 15.  $\Delta x = \Delta y = 1/40$ . left: third order scheme. right: fifth order scheme.

490 a horizontally moving Mach 10 shock wave passes through a wedge with an inclination  
 491 angle of  $30^\circ$ . In common practice, the wedge is placed horizontally to apply reflective  
 492 boundary conditions. At this time, the shock wave forms an angle of  $60^\circ$  with the  
 493 wall. In [8, 25], the original double Mach number reflection problem is computed  
 494 respectively. With the ILW method, people can also do numerical simulation on the  
 495 original region [28, 31]. Here, we use the new ILW method to simulate this problem.  
 496 In detail, at the top of the calculation area, we give the exact flow value according  
 497 to the shock Mach number. at the left and right boundary, we give the supersonic  
 498 inlet and outlet boundary conditions respectively. On the lower right boundary, a  
 499 new ILW method is adopted. The discretization of space and time is consistent with  
 500 the previous example. Figure 9 show the computational region and density contour  
 501 respectively. The zoomed in region near the double Mach stem is presented in Figure  
 502 10. We rotate and translate the region for ease of comparison. It is observed that  
 503 the new ILW method captures the shock wave well, and it is comparable with the  
 504 previous results.

505 **5. Concluding remarks.** In this paper, we propose a new SILW method for  
 506 conservation laws, which decomposes the procedure of construction ghost points into  
 507 two steps: interpolation and extrapolation. At first, we approximate some special  
 508 point values through an interpolation polynomial based on interior points near the  
 509 boundary. Then, we construct a Hermite extrapolation polynomial based on the  
 510 special point values and spatial derivatives at the boundary obtained through the  
 511 ILW process. After that, we can get the approximation of the the ghost point values.  
 512 Through the linear stability analysis with the eigenvalue method, we can conclude that  
 513 our new SILW method is more efficient than the original SILW method while ensuring  
 514 the stability. Then we extend our new SILW method to systems and high-dimensional



**Fig. 9.** Example 8: Left: The computational region of the double mach reflection problem. The dashed line indicates the computational domain used in [8, 25]. Right: The density contour. 30 contours from 1.731 to 20.92.  $\Delta x = \Delta y = 1/320$ .



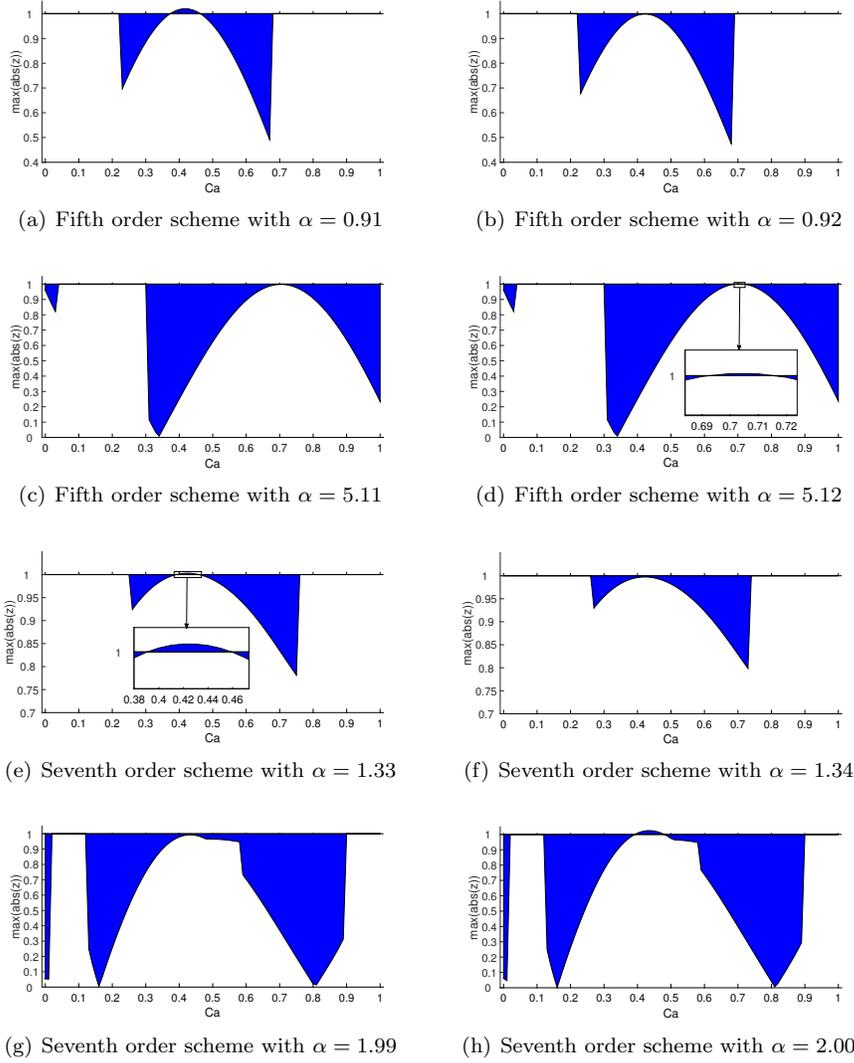
**Fig. 10.** Example 8: Density contour on the local area. 30 contours from 1.731 to 20.92.

515 cases, and carry out a series of numerical experiments. The numerical results show  
 516 that our new SILW method is stable and can achieve the expected accuracy. In the  
 517 future, we are going to extend this new SILW method to deal with the initial-boundary  
 518 value problems of diffusion equations and convection-diffusion equations.

#### 519 **Appendix A. More results about linear stability analysis.**

520 The linear stability analysis results of the new SILW method with different in-  
 521 ternal schemes and different  $k_d$  are shown in Figure 11-13. The parameter  $\alpha$  is taken  
 522 as the critical value between stable and unstable. These figures verify the correctness  
 523 and the optimality of the  $\alpha$  range given in Table 3.

524 Next, we want to verify the results of the above stability analysis numerically.



**Fig. 11.** The result of linear stability analysis with  $k_d = 2$ . The horizontal axis represents  $C_a$  and the vertical axis represents the largest  $|z(\mu)|$ .

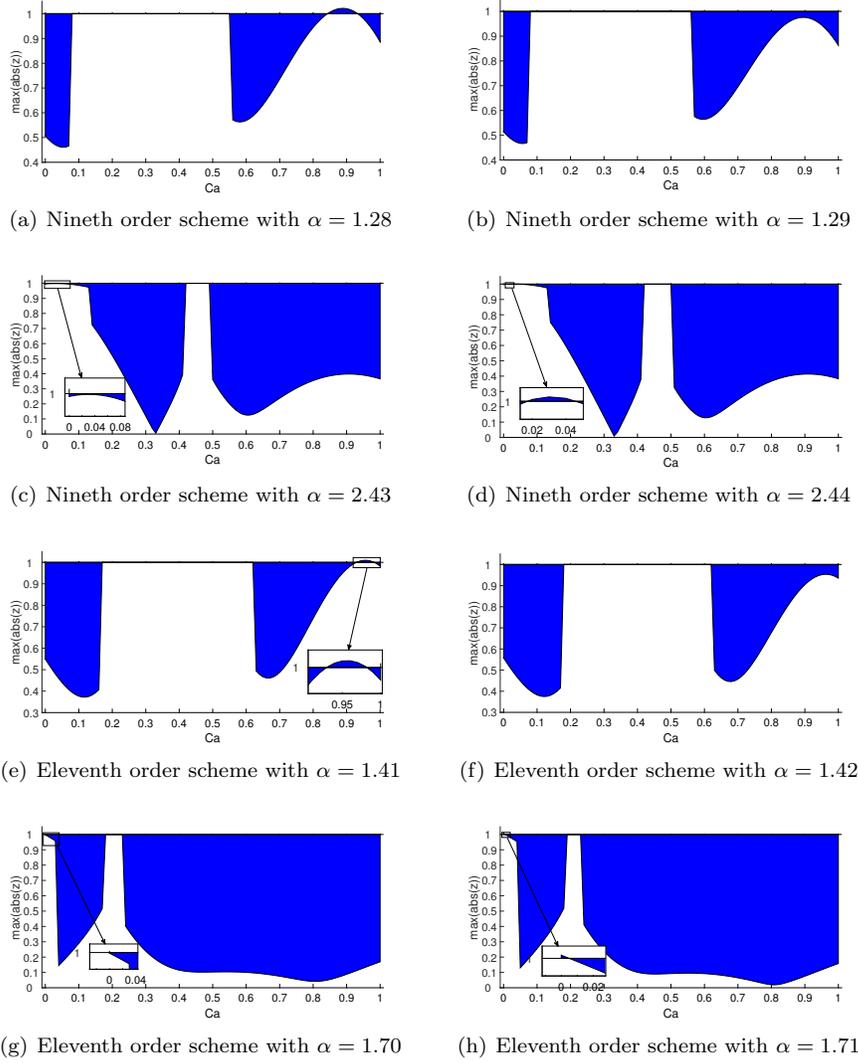
525 Consider the following problem:

$$526 \quad (A.1) \quad \begin{cases} u_t + u_x = 0, & -1 < x < 1, t > 0, \\ u(x, 0) = 0.25 + 0.5 \sin(\pi x), & -1 \leq x \leq 1, \\ u(-1, t) = 0.25 + 0.5 \sin(\pi t), & t > 0. \end{cases}$$

527 The exact solution is

$$528 \quad u(x, t) = 0.25 + 0.5 \sin(\pi(x - t)).$$

529 We use the  $d$ -th order upwind scheme for spatial discretization and the third-order



**Fig. 12.** The result of linear stability analysis with  $k_d = 3$ . The horizontal axis represents  $C_a$  and the vertical axis represents the largest  $|z(\mu)|$ .

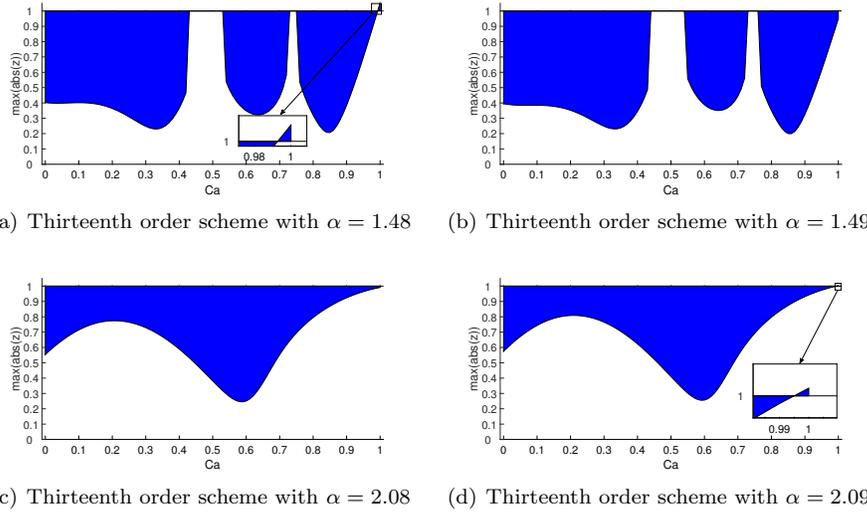
530 TVD RK scheme for time discretization. Let  $t_{end} = 30$ ,  $N = 200$ . Take time step

531 
$$\Delta t = (\lambda_{cfl})_{max} \Delta x.$$

532 We test the problem with  $\alpha$  in or out the range given in Table 3. The numerical  
 533 results are shown in Table 11. It can be observed that when  $\alpha$  falls in the range, the  
 534 scheme will be stable for all tested  $C_a$ . Otherwise, if  $\alpha$  is out of the range, we can  
 535 always find one  $C_a$  such that the scheme is unstable.

536

REFERENCES



**Fig. 13.** The result of linear stability analysis with  $k_d = 4$ . The horizontal axis represents  $C_a$  and the vertical axis represents the largest  $|z(\mu)|$ .

**Table 11.** Numerical verification results of linear stability analysis

$d$	Stable $\alpha$ in Table 3	$\alpha$	Result
3	[0.61,10]	0.60	Unstable for $C_a = 10^{-6}$
		1.00	Stable for all tested $C_a$
5	[0.92,5.11]	0.91	Unstable for $C_a = 0.38$
		1.00	Stable for all tested $C_a$
		5.12	Unstable for $C_a = 0.70$
7	[1.34,1.99]	1.33	Unstable for $C_a = 0.40$
		1.50	Stable for all tested $C_a$
		2.00	Unstable for $C_a = 0.40$
9	[1.29,2.43]	1.28	Unstable for $C_a = 0.85$
		1.50	Stable for all tested $C_a$
		2.44	Unstable for $C_a = 0.03$
11	[1.42,1.70]	1.41	Unstable for $C_a = 0.93$
		1.50	Stable for all tested $C_a$
		1.71	Unstable for $C_a = 0.01$
13	[1.49,2.08]	1.48	Unstable for $C_a = 1 - 10^{-6}$
		1.75	Stable for all tested $C_a$
		2.09	Unstable for $C_a = 1 - 10^{-6}$

- 537 [1] A. BAEZA, P. MULET, AND D. ZORÍO, *High order boundary extrapolation technique for finite*  
538 *difference methods on complex domains with cartesian meshes*, Journal of Scientific Com-  
539 *puting*, 66 (2016), pp. 761–791.
- 540 [2] M. H. CARPENTER, D. GOTTLIEB, S. ABARBANEL, AND W.-S. DON, *The theoretical accuracy*  
541 *of Runge–Kutta time discretizations for the initial boundary value problem: a study of the*  
542 *boundary error*, SIAM Journal on Scientific Computing, 16 (1995), pp. 1241–1252.

- 543 [3] Z. CHENG, S. LIU, Y. JIANG, J. LU, M. ZHANG, AND S. ZHANG, *A high order boundary scheme*  
544 *to simulate complex moving rigid body under impingement of shock wave*, Applied Math-  
545 ematics and Mechanics, 42 (2021), pp. 841–854.
- 546 [4] F. DE VANNA, F. PICANO, AND E. BENINI, *A sharp-interface immersed boundary method for*  
547 *moving objects in compressible viscous flows*, Computers & Fluids, 201 (2020), p. 104415.
- 548 [5] S. DING, C.-W. SHU, AND M. ZHANG, *On the conservation of finite difference WENO schemes*  
549 *in non-rectangular domains using the inverse Lax-Wendroff boundary treatments*, Journal  
550 of Computational Physics, 415 (2020), p. 109516.
- 551 [6] F. FILBET AND C. YANG, *An inverse Lax-Wendroff method for boundary conditions applied to*  
552 *Boltzmann type models*, Journal of Computational Physics, 245 (2013), pp. 43–61.
- 553 [7] L. HUANG, C.-W. SHU, AND M. ZHANG, *Numerical boundary conditions for the fast sweeping*  
554 *high order WENO methods for solving the Eikonal equation*, Journal of Computational  
555 Mathematics, 26 (2008), pp. 336–346.
- 556 [8] G.-S. JIANG AND C.-W. SHU, *Efficient implementation of weighted ENO schemes*, Journal of  
557 computational physics, 126 (1996), pp. 202–228.
- 558 [9] H.-O. KREISS AND N. A. PETERSSON, *A second order accurate embedded boundary method for*  
559 *the wave equation with Dirichlet data*, SIAM Journal on Scientific Computing, 27 (2006),  
560 pp. 1141–1167.
- 561 [10] H.-O. KREISS, N. A. PETERSSON, AND J. YSTRÖM, *Difference approximations for the second*  
562 *order wave equation*, SIAM Journal on Numerical Analysis, 40 (2002), pp. 1940–1967.
- 563 [11] H.-O. KREISS, N. A. PETERSSON, AND J. YSTRÖM, *Difference approximations of the Neumann*  
564 *problem for the second order wave equation*, SIAM Journal on Numerical Analysis, 42  
565 (2004), pp. 1292–1323.
- 566 [12] J. LI, C.-W. SHU, AND J. QIU, *Multi-resolution HWENO schemes for hyperbolic conservation*  
567 *laws*, Journal of Computational Physics, 446 (2021), p. 110653.
- 568 [13] T. LI, J. LU, AND C.-W. SHU, *Stability analysis of inverse Lax-Wendroff boundary treatment of*  
569 *high order compact difference schemes for parabolic equations*, Journal of Computational  
570 and Applied Mathematics, 400 (2022), p. 113711.
- 571 [14] T. LI, C.-W. SHU, AND M. ZHANG, *Stability analysis of the inverse Lax-Wendroff boundary*  
572 *treatment for high order upwind-biased finite difference schemes*, Journal of Computational  
573 and Applied Mathematics, 299 (2016), pp. 140–158.
- 574 [15] T. LI, C.-W. SHU, AND M. ZHANG, *Stability analysis of the inverse Lax-Wendroff boundary*  
575 *treatment for high order central difference schemes for diffusion equations*, Journal of  
576 Scientific Computing, 70 (2017), pp. 576–607.
- 577 [16] S. LIU, Z. CHENG, Y. JIANG, J. LU, M. ZHANG, AND S. ZHANG, *Numerical simulation of a*  
578 *complex moving rigid body under the impingement of a shock wave in 3D*, Advances in  
579 Aerodynamics, 4 (2022), pp. 1–29.
- 580 [17] S. LIU, Y. JIANG, C.-W. SHU, M. ZHANG, AND S. ZHANG, *A high order moving boundary*  
581 *treatment for convection-diffusion equations*, preparation.
- 582 [18] J. LU, J. FANG, S. TAN, C.-W. SHU, AND M. ZHANG, *Inverse Lax-Wendroff procedure for nu-*  
583 *merical boundary conditions of convection-diffusion equations*, Journal of Computational  
584 Physics, 317 (2016), pp. 276–300.
- 585 [19] J. LU, C.-W. SHU, S. TAN, AND M. ZHANG, *An inverse Lax-Wendroff procedure for hyperbolic*  
586 *conservation laws with changing wind direction on the boundary*, Journal of Computational  
587 Physics, 426 (2021), p. 109940.
- 588 [20] R. MITTAL AND G. IACCARINO, *Immersed boundary methods*, Annu. Rev. Fluid Mech., 37  
589 (2005), pp. 239–261.
- 590 [21] S. NILSSON, N. A. PETERSSON, B. SJÖGREEN, AND H.-O. KREISS, *Stable difference approx-*  
591 *imations for the elastic wave equation in second order formulation*, SIAM Journal on  
592 Numerical Analysis, 45 (2007), pp. 1902–1936.
- 593 [22] C. S. PESKIN, *Numerical analysis of blood flow in the heart*, Journal of computational physics,  
594 25 (1977), pp. 220–252.
- 595 [23] C. S. PESKIN, *The immersed boundary method*, Acta numerica, 11 (2002), pp. 479–517.
- 596 [24] Y. QU, R. SHI, AND R. C. BATRA, *An immersed boundary formulation for simulating high-*  
597 *speed compressible viscous flows with moving solids*, Journal of Computational Physics,  
598 354 (2018), pp. 672–691.
- 599 [25] J. SHI, Y.-T. ZHANG, AND C.-W. SHU, *Resolution of high order WENO schemes for complicated*  
600 *flow structures*, Journal of Computational Physics, 186 (2003), pp. 690–696.
- 601 [26] C.-W. SHU AND S. OSHER, *Efficient implementation of essentially non-oscillatory shock-*  
602 *capturing schemes*, Journal of computational physics, 77 (1988), pp. 439–471.
- 603 [27] B. SJÖGREEN AND N. A. PETERSSON, *A Cartesian embedded boundary method for hyperbolic*  
604 *conservation laws*, Communications in Computational Physics, 2 (2006), pp. 1199–1219.

- 605 [28] S. TAN AND C.-W. SHU, *Inverse Lax-Wendroff procedure for numerical boundary conditions of*  
606 *conservation laws*, Journal of Computational Physics, 229 (2010), pp. 8144–8166.
- 607 [29] S. TAN AND C.-W. SHU, *A high order moving boundary treatment for compressible inviscid*  
608 *flows*, Journal of Computational Physics, 230 (2011), pp. 6023–6036.
- 609 [30] S. TAN AND C.-W. SHU, *Inverse Lax-Wendroff procedure for numerical boundary conditions of*  
610 *hyperbolic equations: survey and new developments*, in Advances in applied mathematics,  
611 modeling, and computational science, Springer, 2013, pp. 41–63.
- 612 [31] S. TAN, C. WANG, C.-W. SHU, AND J. NING, *Efficient implementation of high order in-*  
613 *verse Lax-Wendroff boundary treatment for conservation laws*, Journal of Computational  
614 Physics, 231 (2012), pp. 2510–2527.
- 615 [32] F. VILAR AND C.-W. SHU, *Development and stability analysis of the inverse Lax- Wendroff*  
616 *boundary treatment for central compact schemes*, ESAIM: Mathematical Modelling and  
617 Numerical Analysis, 49 (2015), pp. 39–67.
- 618 [33] J. WANG, X. GU, AND J. WU, *A sharp-interface immersed boundary method for simulating*  
619 *high-speed compressible inviscid flows*, Advances in Aerodynamics, 2 (2020), pp. 1–19.
- 620 [34] T. XIONG, M. ZHANG, Y.-T. ZHANG, AND C.-W. SHU, *Fifth order fast sweeping WENO scheme*  
621 *for static Hamilton-Jacobi equations with accurate boundary treatment*, Journal of Scien-  
622 tific Computing, 45 (2010), pp. 514–536.