

A high order moving interface treatment for fluid-structure interaction in compressible flow

Zepeng Liu¹, Yan Jiang², Chi-Wang Shu³, Mengping Zhang⁴

Abstract: In this study, we present a high order moving interface treatment for fluid-structure interaction (FSI) problems between compressible viscous flows and deformable structure. We consider a loosely partitioned coupling strategy that involves three parts: flow solver, structure solver and fluid-structure interface treatment. In particular, the compressible Navier-Stokes equations are solved by a high order finite difference method on fixed Cartesian Eulerian grids. The structure is considered as a beam under the linearity assumption, which is solved by a standard finite element method in Lagrangian coordinates. In terms of the fluid-structure interaction interface treatment, a new simplified inverse Lax-Wendroff approach is extended and implemented to ensure global high order accuracy. To validate our method, we provide several numerical tests including accuracy test, acoustic wave scattering from cylinder, shock-loaded rigid cylinder and deformation of a panel induced by shock waves. The numerical results demonstrate that our method is capable of handling FSI problems involving shock wave and structure deformations non-oscillatorily and with high order accuracy for smooth solutions.

Key words: Fluid-structure interaction; Viscous compressible flow; Inverse Lax-Wendroff method; High order accuracy; Flow-induced deformation

1 Introduction

In recent years, the fluid-structure interaction (FSI) in a compressible flow has found a wide range of applications in engineering and biological fields [1, 3, 9, 10, 13]. These applications always include problems with complex geometries and large structure deformations. The numerical simulation of FSI modeling is challenging and the key is to properly couple the fluid and structure solvers with enough accuracy.

In general, FSI models can be classified as either monolithic or partitioned approaches, based on the coupling of the flow and structure solvers. The monolithic approaches [6, 32] solve a uniform system which includes equations for the flow and structure together. Thus they are robust without introducing any domain splitting error. However, they may lead to large and poorly conditioned systems due to different scaling of variables. In partitioned

¹School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China. E-mail: lzp0375@mail.ustc.edu.cn.

²School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China. E-mail: jiangy@ustc.edu.cn. Research is supported in part by NSFC grant 12271499.

³Division of Applied Mathematics, Brown University, Providence, RI 02912, USA. E-mail: chi-wang_shu@brown.edu. Research is supported in part by NSF grants DMS-2010107 and DMS-2309249.

⁴School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China. E-mail: mpzhang@ustc.edu.cn. Research is supported in part by NSFC grant 12126604.

approaches [2, 30, 40], fluid and solid are solved separately with existing solvers and they share and exchange data with each other. Varying from monolithic methods, tracking the moving fluid-structure interface is required to distinguish the medium in partitioned methods.

Two different types of treatment for the representation of the time-varying fluid-structure interface are usually used: arbitrary Lagrangian-Eulerian (ALE) and immersed boundary methods (IBM). The ALE method [18, 38, 41] needs to regenerate body-fitted grids at each time level, resulting in large computation. Alternatively, the IBM method, developed initially by Peskin [31], often works in fixed Cartesian meshes and has nice properties for easy implementation and low computational cost. In this method, a force is added into the Navier-Stokes equations to achieve the boundary condition. The force acting on the fluid by the immersed boundaries is distributed onto fluid nodes in the vicinity of the fluid-structure interface. However, this immersed boundary method has usually at most second-order accuracy. Inspired by Peskin, many improved work emerged, such as feedback-forcing [39] and direct-forcing [27, 28, 42] methods. Based on a viscous and subsonic compressible flow solver, Ghias et al. [11] proposed a sharp interface immersed boundary method on body non-conformal Cartesian grids. The ghost-cell technique is used to enforce the boundary conditions at the immersed boundary. Seo and Mittal developed a high order sharp interface immersed boundary method to solve acoustic wave scattering and low-Mach number flow-induced sound problems in complex geometries [33]. Chaudhuri recently employed a sharp interface immersed boundary method for several two-dimensional shock-obstacle interaction simulations. The immersed boundary method has gained popularity for a wide range of applications including fluid-structure interaction [1, 14, 40, 43].

Recently, a new boundary treatment, called the inverse Lax-Wendroff (ILW) method, was proposed in [35] to deal with the inflow boundary conditions for solving hyperbolic conservation laws on a Cartesian mesh. The main idea of the ILW methods is obtaining the normal derivatives from the time derivatives and tangential derivatives by repeatedly using the partial differential equations. Thus with these derivatives, one can obtain the value of ghost points near physical boundaries through Taylor expansions. Considering the ILW procedure might bring heavy algebraic manipulations, a simplified ILW (SILW) method [37] was developed to reduce the complexity, using extrapolations for the high order spatial derivatives. Based on this idea, [5, 36] employed the material derivatives instead of Eulerian time derivatives in the ILW procedure to describe the conditions on the moving boundaries. In [25], Lu et al. proposed a SILW method for convection-diffusion equations on static domain, in which a combination of the boundary treatments for the diffusion-dominated and the convection-dominated cases was considered to obtain a stable and accurate numerical scheme. [22] extended it to the time-varying domain and simulated the interaction between viscous compressible flow and rigid bodies. In addition to high order accuracy, the ILW method also maintains good numerical stability. Stability analysis of ILW method for conservation laws and diffusion equations was given in [19, 20]. Further research has been developed for problems such as Boltzmann type models [8]. Recently, Liu et al. proposed a new type of SILW method to approximate values of

ghost points for conservation laws [24] and convection-diffusion equations [23]. The main difference is that the new proposed method constructs a Hermite extrapolation polynomial based on spatial derivatives at boundary obtained through ILW process and some special point values, which is obtained through interpolation polynomial given the interior points near boundary. With appropriate selection of interpolation points, high-order accuracy and stable results can be achieved. Eigenvalue analysis method was employed to analyze the stability, showing that the new method can improve the computational efficiency on the premise of maintaining accuracy and stability.

In this work, we will follow the framework in [23] and aim to extend the ILW method to the fluid-structure interaction problems. We employ the alternative formulation of high order finite difference weighted essentially non-oscillatory (AWENO) scheme for solving Navier-Stokes equations on fixed Eulerian grids and the standard finite element (FE) method for structure system. The movement of the fluid-structure surface is described in a Lagrangian framework. We attempt to give new treatments of the ILW method for the moving interaction surface where the physical variables satisfy specific boundary conditions. Unlike the previous problems about rigid bodies, the elastic structure may move and deform induced by the flow. Therefore, tracking the interface and exchanging information between solvers are the main parts of this article for fluid-structure interaction problems.

This paper is organized as follows. In Section 2, we introduce the governing equations and the fluid-structure boundary conditions. The numerical methods including spatial and temporal discretizations in flow solver and structure solver, the coupling procedure and the high order ILW boundary treatment will be discussed in detail in Section 3. In Section 4, numerical results are shown to validate our methods. Concluding remarks are given in Section 5.

2 Governing equations

In this section, we describe the governing equations of the FSI modeling we are concerned with. In particular, they consist of the Navier-Stokes equations for compressible viscous flow, linear elastic equations for structure and the fluid-structure interface conditions.

As depicted in Fig. 1, the computational domain can be divided into the fluid domain $\Omega_f(t)$ and the solid domain $\Omega_s(t)$. Let $\Gamma_{\text{FSI}}(t) = \partial\Omega_f(t) \cap \partial\Omega_s(t)$ be the time varying conjoined interface of fluid and structure, with the normal vector \mathbf{n}^Γ pointing from the fluid domain into the solid domain.

2.1 Navier-Stokes equations for compressible viscous flow

The fluid flow is governed by the two-dimensional compressible viscous Navier-Stokes equations in an Eulerian frame, which are given by

$$\mathbf{Q}_t + \mathbf{F}(\mathbf{Q})_x + \mathbf{G}(\mathbf{Q})_y = \frac{1}{Re} [\mathbf{S}_1(\mathbf{Q})_x + \mathbf{S}_2(\mathbf{Q})_y] \quad \text{in } \Omega_f(t), \quad (1)$$

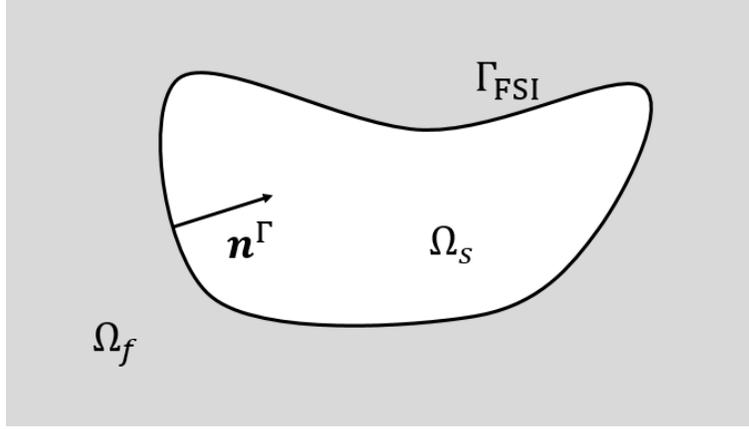


Figure 1. Schematic of the FSI domain.

where

$$\mathbf{Q} = [\rho, \rho u, \rho v, E]^T, \quad (2)$$

$$\mathbf{F} = [\rho u, \rho u^2 + p, \rho uv, u(E + p)]^T, \quad (3)$$

$$\mathbf{G} = [\rho v, \rho uv, \rho v^2 + p, v(E + p)]^T, \quad (4)$$

$$\mathbf{S}_1 = [0, \tau_{xx}, \tau_{xy}, b_x]^T, \quad (5)$$

$$\mathbf{S}_2 = [0, \tau_{xy}, \tau_{yy}, b_y]^T. \quad (6)$$

Here, ρ , u , v , p represent the density, x - and y - velocities, and pressure. E is the total energy per unit of volume, given by

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho(u^2 + v^2). \quad (7)$$

The parameter γ is the adiabatic coefficient, which is equal to 1.4 for an ideal polytropic gas. Re is the Reynolds number. The variables of the viscous terms are given by

$$\tau_{xx} = \frac{2}{3} \left(2 \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \right), \quad (8)$$

$$\tau_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}, \quad (9)$$

$$\tau_{yy} = \frac{2}{3} \left(2 \frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} \right), \quad (10)$$

$$b_x = u\tau_{xx} + v\tau_{xy} + \frac{1}{Pr(\gamma - 1)} \frac{\partial(c^2)}{\partial x}, \quad (11)$$

$$b_y = u\tau_{xy} + v\tau_{yy} + \frac{1}{Pr(\gamma - 1)} \frac{\partial(c^2)}{\partial y}, \quad (12)$$

where Pr is the coefficient of thermal conductivity and $c = \sqrt{\gamma p / \rho}$ is the sound speed.

2.2 Structure dynamics

To describe the elastic structure with geometrically linear and physically linear behaviour, we consider a two-dimensional linear elastic equations in Lagrangian form

$$\rho_s \ddot{\mathbf{d}} = \nabla_0 \cdot \boldsymbol{\sigma}_s + \mathbf{f} \quad \text{in } \Omega_{s_0}. \quad (13)$$

Here, ρ_s is the structure material density. The unknown displacement $\mathbf{d} = (d_1, d_2)^T$, velocity $\dot{\mathbf{d}}$ and acceleration $\ddot{\mathbf{d}}$ describe the motion of deformed solid as a function of Lagrangian coordinates (X, Y) and time t . The dot means the time derivative $\frac{\partial}{\partial t}$. $\nabla_0 = (\frac{\partial}{\partial X}, \frac{\partial}{\partial Y})$ is the material divergence operator, and \mathbf{f} is the body force of structure. Under the geometric and physical linearity assumption, the strain tensor $\boldsymbol{\epsilon}$ and stress tensor $\boldsymbol{\sigma}_s$ satisfy

$$\boldsymbol{\epsilon} = \frac{1}{2} (\nabla_0 \mathbf{d} + \nabla_0 \mathbf{d}^T), \quad \boldsymbol{\sigma}_s = \mathbf{D} : \boldsymbol{\epsilon}, \quad (14)$$

where \mathbf{D} is a tensor of material constants. In fact, the stress tensor $\boldsymbol{\sigma}_s$ can be written as

$$\boldsymbol{\sigma}_s = \begin{pmatrix} (\lambda + 2\mu) \frac{\partial d_1}{\partial X} + \lambda \frac{\partial d_2}{\partial Y} & \mu (\frac{\partial d_1}{\partial Y} + \frac{\partial d_2}{\partial X}) \\ \mu (\frac{\partial d_1}{\partial Y} + \frac{\partial d_2}{\partial X}) & (\lambda + 2\mu) \frac{\partial d_2}{\partial Y} + \lambda \frac{\partial d_1}{\partial X} \end{pmatrix}, \quad (15)$$

where λ and μ are the Lamé constants.

Since the elastic equation (13) is considered in the Lagrangian form, the computational domain Ω_{s_0} is the solid domain at the initial moment $\Omega_s(0)$. The boundary $\partial\Omega_{s_0}$ can be divided into two parts based on the boundary conditions,

$$\partial\Omega_{s_0} = \Gamma_{s;D} \cup \Gamma_{s;N}, \quad (16)$$

where, $\Gamma_{s;D}$ and $\Gamma_{s;N}$ are called the displacement and stress boundaries, respectively, as we have conditions

$$\mathbf{d} = \hat{\mathbf{d}} \quad \text{on } \Gamma_{s;D}, \quad (17)$$

$$\boldsymbol{\sigma}_s \cdot \mathbf{n}_0 = \hat{\mathbf{f}} \quad \text{on } \Gamma_{s;N}, \quad (18)$$

with given boundary conditions $\hat{\mathbf{d}}$ and $\hat{\mathbf{f}}$. Here, \mathbf{n}_0 denotes the normal vector of structure Lagrangian configuration pointing from interior to exterior. The initial conditions are given as follows:

$$\mathbf{d}(\mathbf{X}, t = 0) = \hat{\mathbf{d}}_0, \quad \text{and} \quad \dot{\mathbf{d}}(\mathbf{X}, t = 0) = \hat{\dot{\mathbf{d}}}_0 \quad \text{in } \Omega_{s_0}, \quad (19)$$

where \mathbf{X} defines the initial position.

2.3 Fluid-structure interface conditions

On the fluid-structure interface $\Gamma_{\text{FSI}}(t)$, we have the continuity of velocity and traction, i.e.

$$\mathbf{u} = \dot{\mathbf{d}} \quad \text{and} \quad \boldsymbol{\sigma}_f \cdot \mathbf{n}^\Gamma = \tilde{\boldsymbol{\sigma}}_s \cdot \mathbf{n}^\Gamma, \quad \text{on } \Gamma_{\text{FSI}}(t). \quad (20)$$

Here, $\tilde{\boldsymbol{\sigma}}_s$ is the Cauchy stress tensor of structure, and $\boldsymbol{\sigma}_f$ is the stress tensor of fluid satisfying

$$\boldsymbol{\sigma}_f = -p\mathbf{I} + \frac{1}{Re} \left[\nabla \mathbf{u} + \nabla \mathbf{u}^\text{T} - \frac{2}{3}(\nabla \cdot \mathbf{u})\mathbf{I} \right]. \quad (21)$$

Furthermore, we assume the interface is adiabatic for the fluid, i.e.

$$\frac{\partial T}{\partial \mathbf{n}^\Gamma} = 0, \quad \text{on } \Gamma_{\text{FSI}}(t), \quad (22)$$

where $T = \gamma p / \rho$ is the temperature.

3 Numerical methods

In this section, we will design high order numerical methods for the FSI system. We employ the high order alternative weighted essentially non-oscillatory (AWENO) finite difference method for convective terms and central difference scheme for viscous terms in the Navier-Stokes equations. Standard finite element method is used for spatial discretization of elastic dynamic equations. For the temporal discretization, we employ the Runge-Kutta method and Newmark's method for fluid and structure respectively. Fluid and structure parts need to exchange information at each time step to maintain the fluid-structure interface conditions. In particular, due to a wide stencil needed in high order finite difference schemes, we have to construct the value on the fluid ghost points. Here, a new simplified inverse Lax-Wendroff (SILW) boundary treatment is proposed.

3.1 Fluid solver

When solving the NS equations, we suppose the domain is divided by the fixed Cartesian meshes

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

where Δx and Δy are uniform mesh sizes on x - and y -direction, respectively. We denote $\mathbf{Q}_{i,j}$ as the approximation of point value $\mathbf{Q}(x_i, y_j, t)$.

In the fluid solver, the high order AWENO finite difference scheme [16, 17] is used for the spatial discretization of the convective term,

$$\begin{aligned} \frac{1}{\Delta x} \left(\hat{\mathbf{F}}_{i+1/2,j} - \hat{\mathbf{F}}_{i-1/2,j} \right) &= \mathbf{F}_x|_{(x_i,y_j)} + \mathcal{O}(\Delta x^k), \\ \frac{1}{\Delta y} \left(\hat{\mathbf{G}}_{i,j+1/2} - \hat{\mathbf{G}}_{i,j-1/2} \right) &= \mathbf{G}_y|_{(x_i,y_j)} + \mathcal{O}(\Delta y^k). \end{aligned} \quad (23)$$

The numerical fluxes $\hat{\mathbf{F}}_{i+1/2,j}$ and $\hat{\mathbf{G}}_{i,j+1/2}$ at half point are obtained by the one-dimensional AWENO approximation along x - and y - direction respectively. We briefly introduce the procedure of discretizing the convective term \mathbf{F}_x with fifth order accuracy (k=5). The term \mathbf{G}_y can be treated similarly.

Step 1. Apply the fifth order WENO interpolation [16] on \mathbf{Q} in the x -direction for fixed j , and obtain $\mathbf{Q}_{i+1/2,j}^\pm$ at half point. Here the superscripts \pm refer to one-point left/right biased stencils.

Step 2. Construct the term $h(\mathbf{Q}_{i+1/2,j}^-, \mathbf{Q}_{i+1/2,j}^+)$. The two-argument numerical function h is based on an exact or approximate Riemann solver, such as the Lax-Friedrichs flux or the HLLC flux.

Step 3. Approximate high order derivatives $\partial^{2l}\mathbf{F}/\partial x^{2l}$ at half points with linear schemes

$$\begin{aligned}\Delta x^2 \frac{\partial^2 \mathbf{F}}{\partial x^2} \Big|_{i+\frac{1}{2},j} &\approx \frac{1}{48} (-5\mathbf{F}_{i-2,j} + 39\mathbf{F}_{i-1,j} - 34\mathbf{F}_{i,j} - 34\mathbf{F}_{i+1,j} + 39\mathbf{F}_{i+2,j} - 5\mathbf{F}_{i+3,j}), \\ \Delta x^4 \frac{\partial^4 \mathbf{F}}{\partial x^4} \Big|_{i+\frac{1}{2},j} &\approx \frac{1}{2} (\mathbf{F}_{i-2,j} - 3\mathbf{F}_{i-1,j} + 2\mathbf{F}_{i,j} + 2\mathbf{F}_{i+1,j} - 3\mathbf{F}_{i+2,j} + \mathbf{F}_{i+3,j}),\end{aligned}$$

which give truncation errors of $\mathcal{O}(\Delta x^6)$ and guarantee a fifth-order accuracy of the numerical flux.

Step 4. Obtain the numerical flux

$$\hat{\mathbf{F}}_{i+1/2,j} = h(\mathbf{Q}_{i+1/2,j}^-, \mathbf{Q}_{i+1/2,j}^+) - \frac{1}{24} \Delta x^2 \frac{\partial^2 \mathbf{F}}{\partial x^2} \Big|_{i+1/2,j} + \frac{7}{5760} \Delta x^4 \frac{\partial^4 \mathbf{F}}{\partial x^4} \Big|_{i+\frac{1}{2},j}. \quad (24)$$

Step 5. Finally, we approximate the convection term in the x -direction as

$$\frac{\partial \mathbf{F}}{\partial x} \Big|_{i,j} = \frac{\hat{\mathbf{F}}_{i+1/2,j} - \hat{\mathbf{F}}_{i-1/2,j}}{\Delta x} \quad (25)$$

For the viscous terms on the right hand side, a fourth order central finite difference scheme is used to approximate the derivatives.

$$\begin{aligned}\frac{\partial \phi}{\partial x} \Big|_{i,j} &\approx \frac{1}{12\Delta x} (\phi_{i-2,j} - 8\phi_{i-1,j} + 8\phi_{i+1,j} - \phi_{i+2,j}), \\ \frac{\partial^2 \phi}{\partial x^2} \Big|_{i,j} &\approx \frac{1}{12\Delta x^2} (-\phi_{i-2,j} + 16\phi_{i-1,j} - 30\phi_{i,j} + 16\phi_{i+1,j} - \phi_{i+2,j}), \\ \frac{\partial^2 \phi}{\partial x \partial y} \Big|_{i,j} &\approx \frac{1}{144\Delta x \Delta y} ((\phi_{i-2,j-2} - 8\phi_{i-1,j-2} + 8\phi_{i+1,j-2} - \phi_{i+2,j-2}) \\ &\quad - 8(\phi_{i-2,j-1} - 8\phi_{i-1,j-1} + 8\phi_{i+1,j-1} - \phi_{i+2,j-1}) \\ &\quad + 8(\phi_{i-2,j+1} - 8\phi_{i-1,j+1} + 8\phi_{i+1,j+1} - \phi_{i+2,j+1}) \\ &\quad - (\phi_{i-2,j+2} - 8\phi_{i-1,j+2} + 8\phi_{i+1,j+2} - \phi_{i+2,j+2})).\end{aligned} \quad (26)$$

Remark: To construct the third order scheme, we need to use third order WENO interpolation in Step 1 of the AWENO method and remove the higher order term $\frac{7}{5760}\Delta x^4 \frac{\partial^4 \mathbf{F}}{\partial x^4}|_{i+\frac{1}{2},j}$ in (24). Accordingly, all linear approximation in AWENO and viscous terms can be obtained by lower order central difference schemes.

After spatial discretization, the semi-discrete scheme is equivalent to a first-order ordinary differential equation system $\mathbf{Q}_t = \mathcal{L}(\mathbf{Q})$, where $\mathcal{L}(\mathbf{Q})$ is the spatial operator. Then, the third-order TVD Runge-Kutta time integration [34] is taken for evolving the solution,

$$\begin{aligned} \mathbf{Q}^{(1)} &= \mathbf{Q}^n + \Delta t \mathcal{L}(\mathbf{Q}^n), \\ \mathbf{Q}^{(2)} &= \frac{3}{4}\mathbf{Q}^n + \frac{1}{4}(\mathbf{Q}^{(1)} + \Delta t \mathcal{L}(\mathbf{Q}^{(1)})), \\ \mathbf{Q}^{n+1} &= \frac{1}{3}\mathbf{Q}^n + \frac{2}{3}(\mathbf{Q}^{(2)} + \Delta t \mathcal{L}(\mathbf{Q}^{(2)})). \end{aligned} \quad (27)$$

3.2 Structure solver

The elastic equations (13) are solved by the standard finite element method for spatial discretization. This yields the following system of ordinary differential equations

$$\mathbf{M}\ddot{\mathbf{d}} + \mathbf{K}\mathbf{d} = \bar{\mathbf{F}}, \quad (28)$$

where, \mathbf{d} consists of the displacement vector on each grid point, \mathbf{K} is the stiffness matrix, \mathbf{M} is the mass matrix and $\bar{\mathbf{F}}$ is the traction boundary condition vector.

The temporal integration is achieved with the following Newmark's algorithms [15],

$$\mathbf{d}^{n+1} = \mathbf{d}^n + \Delta t \dot{\mathbf{d}}^n + \frac{\Delta t^2}{2} \left\{ (1 - 2\beta)\ddot{\mathbf{d}}^n + 2\beta\ddot{\mathbf{d}}^{n+1} \right\}, \quad (29)$$

$$\dot{\mathbf{d}}^{n+1} = \dot{\mathbf{d}}^n + \Delta t \left\{ (1 - \gamma)\ddot{\mathbf{d}}^n + \gamma\ddot{\mathbf{d}}^{n+1} \right\}. \quad (30)$$

The parameters β and γ determine the stability and the accuracy characteristics of Newmark's algorithms. In our study, we consider the average acceleration with $\beta = 0.25$ and $\gamma = 0.5$, which give an implicit method with second order accuracy. Substituting equation (28) at time level t_n and t_{n+1} into (29) and (30), we finally have the discretization form

$$(\mathbf{M} + \beta\Delta t^2 \mathbf{K})\mathbf{d}^{n+1} = \mathbf{M}\mathbf{d}^n + \Delta t \mathbf{M}\dot{\mathbf{d}}^n + \frac{\Delta t^2}{2}(1 - 2\beta)(\bar{\mathbf{F}}^n - \mathbf{K}\mathbf{d}^n) + \beta\Delta t^2 \bar{\mathbf{F}}^{n+1}, \quad (31)$$

$$\mathbf{M}\dot{\mathbf{d}}^{n+1} = \mathbf{M}\dot{\mathbf{d}}^n + \Delta t(1 - \gamma)(\bar{\mathbf{F}}^n - \mathbf{K}\mathbf{d}^n) + \gamma\Delta t(\bar{\mathbf{F}}^{n+1} - \mathbf{K}\mathbf{d}^{n+1}). \quad (32)$$

Therefore, given \mathbf{d}^n , $\dot{\mathbf{d}}^n$, $\bar{\mathbf{F}}^n$ and $\bar{\mathbf{F}}^{n+1}$, we can obtain \mathbf{d}^{n+1} and $\dot{\mathbf{d}}^{n+1}$ by solving the above equations.

3.3 FSI coupling

In this work, the compressible fluid model and the deformable solid model are solved separately. We need to couple them through interface boundary conditions by exchanging

data at each time step. Note that the two parts employ different time discretization methods. Hence, we want to exchange data on integer steps only, rather than on the intermediate stages.

Since we have the Lagrangian form of the elastic structure, the solid domain could be represented by finite elements and the FSI boundary is then embedded in a fluid domain discretized using a stationary Cartesian grid. We achieve the coupling by the following procedure.

3.3.1 Boundary treatment for fluid

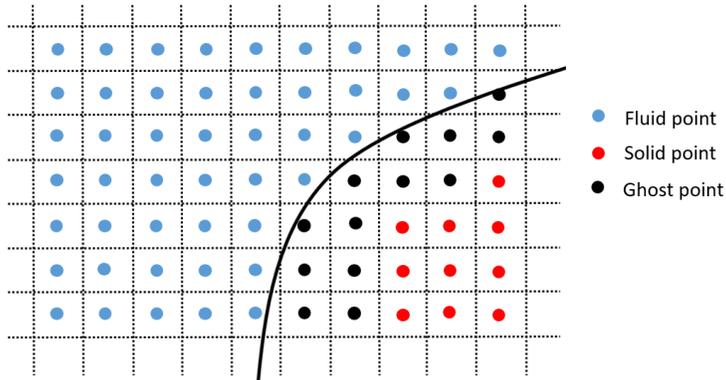


Figure 2. Schematic of the fluid ghost points.

A sketch of a two-dimensional Cartesian grid is shown in Fig. 2. Note that the high order finite difference discretization requires a wide stencil and may include points outside the fluid region, called the ghost points (black nodes in Fig. 2). In the following, we will construct the ghost point values by coupling the fluid-structure boundary conditions, information of interior points and partial differential equations. In particular, we will give a pair of values on each ghost point $P_{ij} = (x_i, y_j)$, denoted as $\mathbf{Q}_{i,j}^c$ and $\mathbf{Q}_{i,j}^d$, which will be used to discrete the convective terms and viscous terms, respectively.

The fluid satisfies the adiabatic no-slip boundary condition

$$\mathbf{u} = \mathbf{V}_b, \quad \text{and} \quad \partial T / \partial \mathbf{n}^\Gamma = 0, \quad \text{on } \Gamma_{\text{FSI}}(t). \quad (33)$$

The boundary velocity \mathbf{V}_b , boundary acceleration $\dot{\mathbf{V}}_b$ and its time derivative $\ddot{\mathbf{V}}_b$ can be obtained from the elastic solid model. Assume $P_a = (x_a, y_a)$ is a point on Γ_{FSI} , and the corresponding Lagrangian coordinate of P_a is $\mathbf{X}_a = (X_a, Y_a)$. The displacement, velocity and acceleration of structure can all be represented as polynomials from the structure solver

$$\mathbf{d}(\mathbf{X}, t) = \mathbf{p}_d(\mathbf{X}, t), \quad \dot{\mathbf{d}}(\mathbf{X}, t) = \mathbf{p}_{\dot{d}}(\mathbf{X}, t), \quad \ddot{\mathbf{d}}(\mathbf{X}, t) = \mathbf{p}_{\ddot{d}}(\mathbf{X}, t), \quad (34)$$

where $\ddot{\mathbf{d}}(\mathbf{X}, t)$ is obtained by solving (28). Employing the boundary condition that the fluid velocity is equal to the solid velocity at the fluid-structure interface, i.e., $\mathbf{V}_b = \dot{\mathbf{d}}$ on

Γ_{FSI} , we can derive that

$$\mathbf{V}_b|_{P_a} = \mathbf{p}_{\dot{\mathbf{d}}}(\mathbf{X}_a, t), \quad \dot{\mathbf{V}}_b|_{P_a} = \mathbf{p}_{\ddot{\mathbf{d}}}(\mathbf{X}_a, t). \quad (35)$$

To obtain $\ddot{\mathbf{V}}_b$, we first take the partial time derivative on both sides of the elastic equation (13)

$$\rho_s \ddot{\mathbf{d}}_t = \nabla_0 \cdot (\boldsymbol{\sigma}_s)_t. \quad (36)$$

The term $\nabla_0 \cdot (\boldsymbol{\sigma}_s)_t$ could be represented by a Lagrangian derivative operator $L_{\mathbf{X}}$ on $\dot{\mathbf{d}}$, i.e.

$$L_{\mathbf{X}}(\dot{\mathbf{d}}) = \nabla_0 \cdot (\boldsymbol{\sigma}_s)_t. \quad (37)$$

Therefore, combining (34), (36)-(37) and $\ddot{\mathbf{V}}_b = \ddot{\mathbf{d}}_t$, we have that

$$\ddot{\mathbf{V}}_b = L_{\mathbf{X}}(\mathbf{p}_{\dot{\mathbf{d}}}(\mathbf{X}_a, t))/\rho_s. \quad (38)$$

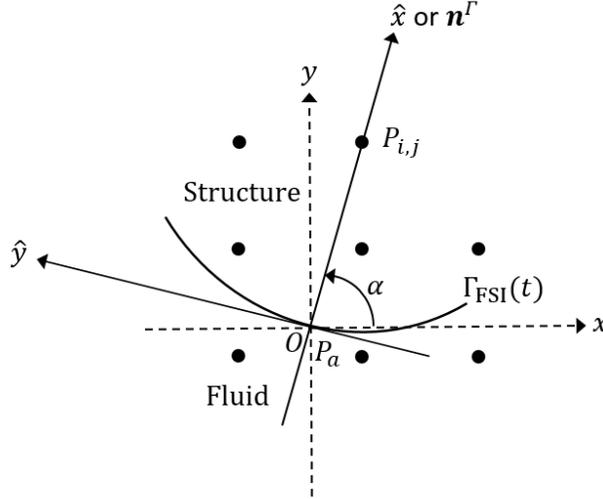


Figure 3. The local coordinate rotation diagram.

Next, we will construct the ghost point values. Similar to [37], we set up a local coordinate system on the boundary for two-dimensional problems to turn the 2D boundary treatment into a 1D algorithm. Specifically, suppose $P_{ij} = (x_i, y_j)$ is a ghost point near the boundary at time level t_n . We can find the pedal $P_a = (x_a, y_a) \in \Gamma_{\text{FSI}}(t)$ such that the outward normal (from fluid to solid) at P_a goes through P_{ij} (shown in Fig. 3). Assume the normal vector $\mathbf{n}^F = (\cos \alpha, \sin \alpha)$. Then, we perform the following local coordinate rotation transformation at P_a by

$$\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \mathbf{T} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (39)$$

Here, \mathbf{T} is the rotation matrix, \hat{x} and \hat{y} represent the normal and tangential direction to $\Gamma_{\text{FSI}}(t)$ at P_a . Consider the new primitive variables

$$\hat{\mathbf{U}} = (\rho, \hat{u}, \hat{v}, T)^T, \quad (40)$$

with

$$\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} = \mathbf{T} \begin{pmatrix} u \\ v \end{pmatrix}.$$

The Navier-Stokes equations (1) can be rewritten under the local coordinate

$$\hat{\mathbf{U}}_t + \mathbf{A}(\hat{\mathbf{U}})\hat{\mathbf{U}}_{\hat{x}} = \mathbf{B}(\hat{\mathbf{U}})\hat{\mathbf{U}}_{\hat{x}\hat{x}} + \mathbf{Res}, \quad (41)$$

where

$$\mathbf{A}(\hat{\mathbf{U}}) = \begin{pmatrix} \hat{u} & \rho & 0 & 0 \\ \frac{T}{\rho} & \hat{u} & 0 & 1 \\ 0 & 0 & \hat{u} & 0 \\ 0 & (\gamma - 1)T & 0 & \hat{u} \end{pmatrix}, \quad \mathbf{B}(\hat{\mathbf{U}}) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{4}{3Re \cdot \rho} & 0 & 0 \\ 0 & 0 & \frac{1}{Re \cdot \rho} & 0 \\ 0 & 0 & 0 & \frac{\gamma}{Pr \cdot Re \cdot \rho} \end{pmatrix},$$

$$\mathbf{Res} = \begin{pmatrix} Res_1 \\ Res_2 \\ Res_3 \\ Res_4 \end{pmatrix} = \begin{pmatrix} -\rho\hat{v}_{\hat{y}} - \hat{v}\rho_{\hat{y}} \\ \frac{1}{Re \cdot \rho}(\hat{u}_{\hat{y}\hat{y}} + \frac{1}{3}\hat{v}_{\hat{x}\hat{y}}) - \hat{v}\hat{u}_{\hat{y}} \\ \frac{1}{Re \cdot \rho}(\frac{1}{3}\hat{u}_{\hat{x}\hat{y}} + \frac{4}{3}\hat{v}_{\hat{y}\hat{y}}) - \hat{v}\hat{v}_{\hat{y}} - T_{\hat{y}} - \rho_{\hat{y}}\frac{T}{\rho} \\ -\hat{v}T_{\hat{y}} - (\gamma - 1)T\hat{v}_{\hat{y}} + \frac{\gamma}{Re \cdot \rho \cdot Pr}T_{\hat{y}\hat{y}} + NLT \end{pmatrix},$$

$$NLT = \frac{\gamma - 1}{Re \cdot \rho} \left[\frac{4}{3}(\hat{u}_{\hat{x}}^2 + \hat{v}_{\hat{y}}^2 - \hat{u}_{\hat{x}}\hat{v}_{\hat{y}}) + (\hat{v}_{\hat{x}} + \hat{u}_{\hat{y}})^2 \right].$$

The matrix $\mathbf{A}(\hat{\mathbf{U}})$ can be diagonalized as

$$\mathbf{A}(\hat{\mathbf{U}}) = \mathbf{L}^{-1}(\hat{\mathbf{U}})\mathbf{\Lambda}(\hat{\mathbf{U}})\mathbf{L}(\hat{\mathbf{U}}), \quad (42)$$

with $\mathbf{\Lambda}(\hat{\mathbf{U}}) = \text{diag}\{\hat{u} - c, \hat{u}, \hat{u}, \hat{u} + c\}$.

According to the Dirichlet velocity boundary condition, we have

$$\hat{u} = \hat{\mathbf{u}} \cdot \hat{\mathbf{n}}^\Gamma = \mathbf{u} \cdot \mathbf{n}^\Gamma = \mathbf{V}_b \cdot \mathbf{n}^\Gamma, \quad (43)$$

where $\hat{\mathbf{n}}^\Gamma = (1, 0)^T$ is the normal vector under the local coordinate. Defining the material operator $\frac{D}{Dt} = \frac{\partial}{\partial t} + \hat{u}\frac{\partial}{\partial \hat{x}} + \hat{v}\frac{\partial}{\partial \hat{y}}$ and applying this operator to the formula above, we can obtain that

$$\frac{D\hat{\mathbf{u}}}{Dt} \cdot \hat{\mathbf{n}}^\Gamma + \hat{\mathbf{u}} \cdot \frac{D\hat{\mathbf{n}}^\Gamma}{Dt} = \frac{D\mathbf{V}_b}{Dt} \cdot \mathbf{n}^\Gamma + \mathbf{V}_b \cdot \frac{D\mathbf{n}^\Gamma}{Dt}, \quad (44)$$

which can be simplified as

$$\frac{D\hat{u}}{Dt} = \frac{D\mathbf{V}_b}{Dt} \cdot \mathbf{n}^\Gamma + \mathbf{V}_b \cdot \frac{D\mathbf{n}^\Gamma}{Dt} - \hat{\mathbf{u}} \cdot \frac{D\hat{\mathbf{n}}^\Gamma}{Dt} = \dot{\mathbf{V}}_b \cdot \mathbf{n}^\Gamma. \quad (45)$$

Here, the second equality results from $\frac{D\hat{\mathbf{n}}^\Gamma}{Dt} = \mathbf{T} \frac{D\mathbf{n}^\Gamma}{Dt}$, $\hat{\mathbf{u}} = \mathbf{T}\mathbf{u}$ and the boundary condition $\mathbf{u} = \mathbf{V}_b$. Moreover, the second derivative has the similar result

$$\frac{D^2\hat{u}}{Dt^2} = \ddot{\mathbf{V}}_b \cdot \mathbf{n}^\Gamma. \quad (46)$$

The value of \hat{u} , $\frac{D\hat{u}}{Dt}$ and $\frac{D^2\hat{u}}{Dt^2}$ would be used in our boundary treatment later.

In the following, we will introduce the ILW process to construct a pair values on $P_{i,j}$, denoted as $\hat{U}_{i,j}^c$ and $\hat{U}_{i,j}^d$, which will be used in the discretization of convective terms and diffusion terms, respectively. Specially, an approximation polynomial $\mathbf{p}(x, y)$ of degree 4 based on the values at internal grid points near P_a is needed. It can be obtained by the least square method with enough points.

Construct the ghost point value $\hat{U}_{i,j}^c$ for convective terms

The main idea is that we construct point value and the first order derivative $\frac{\partial}{\partial \hat{x}} \hat{U}$ at P_a through the ILW procedure, denoted as $\hat{U}_{ilw}^{*(0)}$ and $\hat{U}_{ilw}^{*(1)}$ respectively. Then, we couple them with the point values on some specific inner points to obtain a one-dimensional polynomial \mathbf{q} along the \mathbf{n}^Γ -direction. Finally, we take the value of the polynomial \mathbf{q} at the ghost point P_{ij} . For convenience, we denote the k -th derivative $\frac{\partial^k}{\partial \hat{x}^k} \hat{U}$ at P_a obtained from extrapolation by $\hat{U}_{ext}^{*(k)} = \frac{\partial^k \mathbf{p}}{\partial \hat{x}^k}(P_a)$.

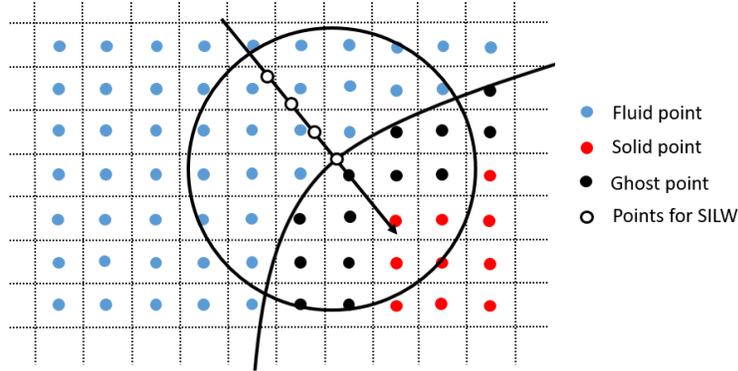


Figure 4. Schematic of the new SILW method used in the fluid solver.

Step 1. Do characteristic decomposition at the pedal P_a . Let $\mathbf{V} = \mathbf{L}(\hat{U}_{ext}^{*(0)})\hat{U}$. Taking advantage of the eigenvalues, we know that the components V_2, V_3, V_4 are outflow variables, while V_1 is inflow variable. Notice that velocity on boundary \mathbf{V}_b is already known, so we set

$$\hat{u}_{ilw}^{*(0)} = \mathbf{V}_b \cdot \mathbf{n}^\Gamma. \quad (47)$$

As V_2, V_3, V_4 are outgoing, we can build the following relation

$$\mathbf{l}_m(\hat{\mathbf{U}}_{ext}^{*(0)}) \cdot \hat{\mathbf{U}}_{ilw}^{*(0)} = (V_{ext}^{(0)})_m, \quad m = 2, 3, 4, \quad (48)$$

where $(V_{ext}^{(0)})_m$ are obtained via WENO extrapolation on each component. Then, we can get $\hat{\mathbf{U}}_{ilw}^{*(0)}$ by solving a linear system

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ l_{21} & l_{22} & l_{23} & l_{24} \\ l_{31} & l_{32} & l_{33} & l_{34} \\ l_{41} & l_{42} & l_{43} & l_{44} \end{pmatrix} \begin{pmatrix} (\hat{\mathbf{U}}_{ilw}^{*(0)})_1 \\ (\hat{\mathbf{U}}_{ilw}^{*(0)})_2 \\ (\hat{\mathbf{U}}_{ilw}^{*(0)})_3 \\ (\hat{\mathbf{U}}_{ilw}^{*(0)})_4 \end{pmatrix} = \begin{pmatrix} \mathbf{V}_b \cdot \mathbf{n}^\Gamma \\ (V_{ext}^{(0)})_2 \\ (V_{ext}^{(0)})_3 \\ (V_{ext}^{(0)})_4 \end{pmatrix}. \quad (49)$$

Then we consider $\hat{\mathbf{U}}_{ilw}^{*(1)}$. The second component of system (41) tells us that

$$\frac{T}{\rho} \rho_{\hat{x}} + T_{\hat{x}} = -\frac{D\hat{u}}{Dt} + \frac{1}{Re\rho} \left[\frac{4}{3} \hat{u}_{\hat{x}\hat{x}} + \frac{1}{3} \hat{v}_{\hat{x}\hat{y}} + \hat{u}_{\hat{y}\hat{y}} \right]. \quad (50)$$

Same as before, $\hat{\mathbf{U}}_{ilw}^{*(1)}$ can be solved by

$$\begin{pmatrix} ((\hat{\mathbf{U}}_{ilw}^{*(0)})_4 / (\hat{\mathbf{U}}_{ilw}^{*(0)})_1) & 0 & 0 & 1 \\ l_{21} & l_{22} & l_{23} & l_{24} \\ l_{31} & l_{32} & l_{33} & l_{34} \\ l_{41} & l_{42} & l_{43} & l_{44} \end{pmatrix} \begin{pmatrix} (\hat{\mathbf{U}}_{ilw}^{*(1)})_1 \\ (\hat{\mathbf{U}}_{ilw}^{*(1)})_2 \\ (\hat{\mathbf{U}}_{ilw}^{*(1)})_3 \\ (\hat{\mathbf{U}}_{ilw}^{*(1)})_4 \end{pmatrix} = \begin{pmatrix} -\frac{D\hat{u}}{Dt} + \frac{1}{Re(\hat{\mathbf{U}}_{ilw}^{*(0)})_1} \left[\frac{4}{3} \hat{u}_{\hat{x}\hat{x}} + \frac{1}{3} \hat{v}_{\hat{x}\hat{y}} + \hat{u}_{\hat{y}\hat{y}} \right] \\ (V_{ext}^{(1)})_2 \\ (V_{ext}^{(1)})_3 \\ (V_{ext}^{(1)})_4 \end{pmatrix}. \quad (51)$$

Here, $\hat{u}_{\hat{x}\hat{x}}, \hat{v}_{\hat{x}\hat{y}}$ and $\hat{u}_{\hat{y}\hat{y}}$ can be obtained from WENO extrapolation and $\frac{D\hat{u}}{Dt}$ has already known from boundary conditions.

Step 2. Construct the point value on some special inner points. Define

$$\hat{\mathbf{U}}_{j*}^c = \mathbf{p}(\mathbf{P}_j^c), \quad (52)$$

where

$$\mathbf{P}_j^c = \mathbf{P}_a - j\alpha_c h \mathbf{n}^\Gamma, \quad j = 1, 2, 3, \quad (53)$$

Here, $h = \sqrt{\Delta x^2 + \Delta y^2}$. The parameter α_c is constant and we take $\alpha_c = 1.25$ in our numerical simulations.

Step 3. Let $\mathbf{q}(s)$ be the one-dimensional quartic polynomial along the \mathbf{n}^Γ -direction, satisfying the following conditions

$$\begin{aligned} \mathbf{q}(0) &= \hat{\mathbf{U}}_{ilw}^{*(0)}, & \mathbf{q}'(0) &= \hat{\mathbf{U}}_{ilw}^{*(1)}, \\ \mathbf{q}(-j\alpha_c h) &= \hat{\mathbf{U}}_{j*}^c, & j &= 1, 2, 3. \end{aligned} \quad (54)$$

Step 4. Finally, we obtain the approximation of the ghost value at $P_{i,j}$

$$\hat{U}_{i,j}^c = \mathbf{q}(|\mathbf{P}_{i,j} - \mathbf{P}_a|). \quad (55)$$

Construct the ghost point value $\hat{U}_{i,j}^d$ for viscous terms

For density ρ , we take it as the extrapolation value $p_\rho(\mathbf{P}_{i,j})$ directly.

Note the velocity components $\hat{u}_{i,j}^d$ and $\hat{v}_{i,j}^d$ are given with Dirichlet boundary conditions, while the temperature $T_{i,j}^d$ is given with Neumann boundary condition. Therefore, we design the same method to get $\hat{u}_{i,j}^d$ and $\hat{v}_{i,j}^d$, and a different way for $T_{i,j}^d$.

For the velocity components, we take $\hat{u}_{i,j}^d$ as an example.

Step 1. Define

$$\hat{u}_{k*}^d = p_{\hat{u}}(\mathbf{P}_k^{d,1}), \quad \text{with} \quad \mathbf{P}_k^{d,1} = \mathbf{P}_a - k\alpha_{d,1}h\mathbf{n}^\Gamma, \quad k = 1, 2, 3. \quad (56)$$

Here, $\alpha_{d,1}$ is constant and we take $\alpha_{d,1} = 1.33$ in our present tests.

Step 2. Let $q_{\hat{u}}(s)$ be the one-dimensional cubic interpolation polynomial satisfying

$$q_{\hat{u}}(0) = \hat{u}_b, \quad \text{and} \quad q_{\hat{u}}(-k\alpha_{d,1}h) = \hat{u}_{k*}^d, \quad k = 1, 2, 3. \quad (57)$$

Step 3. Then we obtain the approximation of velocity value at $P_{i,j}$

$$\hat{u}_{i,j}^d = q_{\hat{u}}(|\mathbf{P}_{i,j} - \mathbf{P}_a|). \quad (58)$$

For temperature $T_{i,j}^d$, we consider the following procedure.

Step1. Let

$$\hat{T}_{j*}^d = p_T(\mathbf{P}_j^{d,2}), \quad \text{with} \quad \mathbf{P}_j^{d,2} = \mathbf{P}_a - (j - 0.5)\alpha_{d,2}h\mathbf{n}^\Gamma, \quad j = 1, 2, 3, 4. \quad (59)$$

Here, $\alpha_{d,2}$ is constant and we take $\alpha_{d,2} = 1.16$ in our present tests. In this step, different parameters $\alpha_c, \alpha_{d,1}, \alpha_{d,2}$ are selected for the purpose of stability [23].

Step 2. Construct the one-dimensional quartic Hermite interpolation polynomial $q_T(s)$ satisfying

$$q_T'(0) = 0, \quad \text{and} \quad q_T(-(j - 0.5)\alpha_{d,2}h) = T_{j*}^d, \quad j = 1, 2, 3, 4. \quad (60)$$

Step 3. Then we obtain the approximation of the temperature value at $P_{i,j}$

$$T_{i,j}^d = q_T(|\mathbf{P}_{i,j} - \mathbf{P}_a|). \quad (61)$$

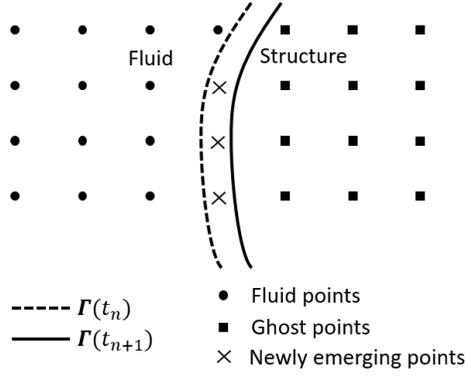


Figure 5. Newly emerging points at time level t_{n+1}

Remark 1. Since the fluid-structure interface moves with time varying, there exists a kind of points which falls in the solid region $\Omega_s(t)$ at time level t_n and enters the fluid region $\Omega_f(t)$ at time level t_{n+1} (see Fig. 5). These points are called “newly emerging” points, and the above procedure is also used to assign values to them. Note that two different values $\hat{U}_{i,j}^c$ and $\hat{U}_{i,j}^d$ have been constructed for the newly emerging points. It is necessary to combine them when doing time evolution. We follow the idea given in [22, 25] and set a convex combination of the two values as that of the “newly emerging” point

$$\hat{U}_{i,j} = \alpha \hat{U}_{i,j}^c + (1 - \alpha) \hat{U}_{i,j}^d. \quad (62)$$

Remark 2. When coupling with the third order RK method, the boundary conditions at the two intermediate stages $\mathbf{Q}^{(1)}$ and $\mathbf{Q}^{(2)}$ need to match the time levels. To achieve third order accuracy, we have to modify the boundary conditions as following [4, 35]. Let \mathbf{x}_b represent the position of pedal P_a at $\Gamma_{\text{FSI}}(t)$, then

$$\hat{u}^{(1)} \sim \hat{u}(\mathbf{x}_b, t_n) + \Delta t \hat{u}_t|_{\mathbf{x}=\mathbf{x}_b, t=t_n}, \quad (63)$$

$$\hat{u}^{(2)} \sim \hat{u}(\mathbf{x}_b, t_n) + \frac{\Delta t}{2} \hat{u}_t|_{\mathbf{x}=\mathbf{x}_b, t=t_n} + \frac{\Delta t^2}{4} \hat{u}_{tt}|_{\mathbf{x}=\mathbf{x}_b, t=t_n}. \quad (64)$$

Here, the time derivatives $\hat{u}_t|_{t=t_n}, \hat{u}_{tt}|_{t=t_n}$ can be obtained by a standard Lax-Wendroff procedure. We have already known

$$\frac{D\hat{u}}{Dt} = \hat{u}_t + \hat{u}_b \hat{u}_{\hat{x}} + \hat{v}_b \hat{u}_{\hat{y}}. \quad (65)$$

The tangential derivative \hat{u}_y above can be obtained by extrapolation. Replacing $\hat{u}_{\hat{x}}$ by $(\hat{U}_{ilw}^{*(1)})_2$, we can obtain the approximation of \hat{u}_t

$$\hat{u}_t = \frac{D\hat{u}}{Dt} - \hat{u}_b (\hat{U}_{ilw}^{*(1)})_2 - \hat{v}_b \hat{u}_{\hat{y}}. \quad (66)$$

with the known values $\frac{D\hat{u}}{Dt}$, \hat{u}_b and \hat{v}_b . Similarly, by applying the material derivative operator, it is easy to get

$$\hat{u}_{tt} = \frac{D^2\hat{u}}{Dt^2} - \left(\frac{\partial\hat{u}_b}{\partial t}\hat{u}_{\hat{x}} + \frac{\partial\hat{v}_b}{\partial t}\hat{u}_{\hat{y}} + \hat{u}_b^2\hat{u}_{\hat{x}\hat{x}} + 2\hat{u}_b\hat{v}_b\hat{u}_{\hat{x}\hat{y}} + \hat{v}_b^2\hat{u}_{\hat{y}\hat{y}} + 2(\hat{u}_b\hat{u}_{t\hat{x}} + \hat{v}_b\hat{u}_{t\hat{y}}) \right), \quad (67)$$

where $(\frac{\partial\hat{u}_b}{\partial t}, \frac{\partial\hat{v}_b}{\partial t})$ is the acceleration of the boundary and satisfies

$$\frac{\partial\hat{\mathbf{u}}_b}{\partial t} = \dot{\mathbf{V}}_b \cdot \mathbf{n}^\Gamma. \quad (68)$$

Since $\hat{u}_{\hat{x}}, \hat{u}_{\hat{y}}, \hat{u}_{\hat{x}\hat{x}}, \hat{u}_{\hat{x}\hat{y}}, \hat{u}_{\hat{y}\hat{y}}$ can be obtained by the ILW procedure or extrapolation, we just need to determine the mixed derivatives $\hat{u}_{t\hat{x}}$ and $\hat{u}_{t\hat{y}}$. Notice that at time $t = t_n$, the time derivative $(\hat{u}_t)_{i,j}$ at every interior point is already known in the first stage of the RK method. Hence, $\hat{u}_{t\hat{x}}$ and $\hat{u}_{t\hat{y}}$ can be obtained by the spatial extrapolation on $\{\hat{u}_t\}_{i,j}$ directly.

3.3.2 Boundary treatment for solid

In order to simulate the elastic structure, we need to compute the integral on the right hand of equation (28),

$$\bar{\mathbf{F}}_i = \int_{\Gamma_{s;N}} \hat{\mathbf{f}}\phi_i d\Gamma. \quad (69)$$

Now we concentrate on how to obtain the traction $\hat{\mathbf{f}}$ on Γ_{FSI} .

Notice that the integration and traction are defined in Lagrangian form, we will transform them into Eulerian form at first. Let $\tilde{\Gamma}_{s;N}$ define the boundary $\Gamma_{s;N}$ in Eulerian framework. Then we have

$$\int_{\Gamma_{s;N}} \hat{\mathbf{f}}\phi_i d\Gamma = \int_{\Gamma_{s;N}} \boldsymbol{\sigma}_s \cdot \mathbf{n}_0 \phi_i d\Gamma = \int_{\tilde{\Gamma}_{s;N}} \tilde{\boldsymbol{\sigma}}_s \cdot \tilde{\mathbf{n}} \tilde{\phi}_i d\tilde{\Gamma}, \quad (70)$$

where $\tilde{\mathbf{n}}$ is the unit outward normal vector in Eulerian form, pointing from solid to fluid. As we defined before, the fluid outward normal \mathbf{n}^Γ points the opposite direction, i.e. $\tilde{\mathbf{n}} = -\mathbf{n}^\Gamma$. Combined with the FSI boundary condition $\boldsymbol{\sigma}_f \cdot \mathbf{n}^\Gamma = \tilde{\boldsymbol{\sigma}}_s \cdot \mathbf{n}^\Gamma$, we can finally obtain that

$$\bar{\mathbf{F}}_i = - \int_{\tilde{\Gamma}_{s;N}} \tilde{\boldsymbol{\sigma}}_s \cdot \mathbf{n}^\Gamma \tilde{\phi}_i d\tilde{\Gamma} = - \int_{\tilde{\Gamma}_{s;N}} \boldsymbol{\sigma}_f \cdot \mathbf{n}^\Gamma \tilde{\phi}_i d\tilde{\Gamma}. \quad (71)$$

Here the pressure and the derivative of velocity in $\boldsymbol{\sigma}_f$ on the structure surface are calculated using WENO interpolation.

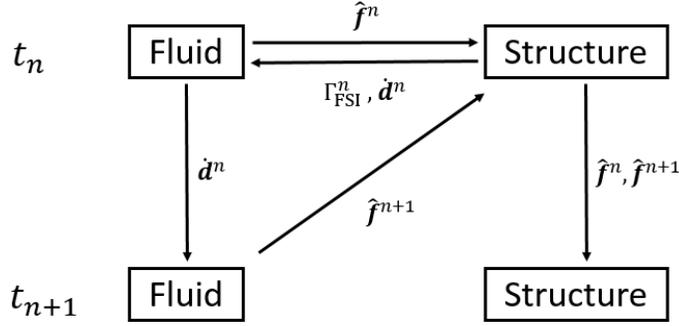


Figure 6. Data exchange between the solvers at the fluid-structure interface.

3.3.3 Coupling procedure

Assume that at time level t_n , the flow variables \mathbf{Q}^n , the fluid-structure interface position Γ_{FSI}^n , the solid displacement \mathbf{d}^n and velocity $\dot{\mathbf{d}}^n$ are known. Then we perform the following loose coupling procedure to update values to time level t_{n+1} , and we give the data exchanging procedure between the solvers at the fluid-structure interface in Fig. 6.

- Step 1. Locate the ghost points of the fluid region around Γ_{FSI}^n and construct the value of ghost points through the ILW procedure. Then solve the fluid equations and update flow variables to \mathbf{Q}^{n+1} .
- Step 2. Interpolate the traction $\hat{\mathbf{f}}^n$ with \mathbf{Q}^n and obtain $\ddot{\mathbf{d}}^n$ by solving (28). Then update the interface position $\tilde{\Gamma}_{\text{FSI}}^{n+1}$ by

$$\tilde{\mathbf{d}}^{n+1} = \mathbf{d}^n + \Delta t \dot{\mathbf{d}}^n + \frac{\Delta t^2}{2} \ddot{\mathbf{d}}^n. \quad (72)$$

- Step 3. Interpolate the traction $\hat{\mathbf{f}}^{n+1}$ with \mathbf{Q}^{n+1} on the updated interface $\tilde{\Gamma}_{\text{FSI}}^{(n+1)}$.
- Step 4. With $\hat{\mathbf{f}}^n$ and $\hat{\mathbf{f}}^{n+1}$, solve the elastic equations (31) - (32) and obtain \mathbf{d}^{n+1} , $\dot{\mathbf{d}}^{n+1}$ and $\Gamma_{\text{FSI}}^{n+1}$.

4 Numerical results

In this section, we will perform several tests to validate the proposed FSI solver. At first, a 1D modified example is simulated to explore the convergence of our method. Then we test the fluid solver with three examples, including 2D accuracy test, acoustic wave scattering from a cylinder and lift-off of a rigid cylinder. Finally, we simulate the deformation of a panel induced by a shock wave in a shock tube.

Without special declaration, the adiabatic constant $\gamma = 1.4$ and the thermal conductivity coefficient $Pr = 0.7$. We employ the third order alternative finite difference WENO

with the HLLC flux, denoted by AWENO-HLLC, as the flow solver. The standard finite element with Q^2 polynomial on triangle meshes is applied for solving the solid. The third-order TVD Runge-Kutta scheme and the second-order Newmark's scheme are used for time integration.

Example 1. Accuracy test for the one-dimensional FSI solver

For simplicity, we consider the 1D Navier-Stokes equations for fluid and wave equation for solid with additional source terms so that we have an explicit exact solution to test accuracy. The modified system is

$$\begin{cases} \rho_t + (\rho u)_x = f_1(x, t), \\ (\rho u)_t + (\rho u^2 + p)_x = \frac{1}{Re} \left(\frac{4}{3} u \right)_{xx} + f_2(x, t), \\ E_t + (u(E + p))_x = \frac{1}{Re} \left(\frac{2}{3} (u^2)_{xx} + \frac{(c^2)_{xx}}{(\gamma-1)Pr} \right) + f_3(x, t), \end{cases} \quad x \in [a, x_{\text{fsi}}(t)] \quad (73)$$

and

$$\rho_s \frac{\partial^2 d}{\partial t^2} = \lambda \frac{\partial^2 d}{\partial X^2} + S(X, t), \quad X \in [x_{\text{fsi}}(0), b]. \quad (74)$$



Figure 7. Example 1: computational domain.

Here, the computational domain is $[a, b]$. As shown in Fig. 7, it is divided into two parts, where $[a, x_{\text{fsi}}(t)]$ represents the fluid part and $[x_{\text{fsi}}(t), b]$ represents the solid part. The fluid-structure interface $x_{\text{fsi}}(t)$ moves with time varying. The exact solution for fluid defined on $x \in [a, x_{\text{fsi}}(t)]$, is

$$\begin{aligned} \rho(x, t) &= 1, \\ u(x, t) &= \sin [(x_{\text{fsi}}(t) - x)^2] e^{-c_2 t} + u_b(t), \\ p(x, t) &= \sin [(x_{\text{fsi}}(t) - x)^2] e^{-c_3 t} + p_b(t). \end{aligned} \quad (75)$$

Here, the velocity $u_b(t)$ and the pressure $p_b(t)$ at $x = x_{\text{fsi}}(t)$ are decided by the boundary conditions. Let x^0 denote the initial position of the interface $x_{\text{fsi}}(0)$. The elastic exact solution is

$$d(X, t) = c_1 (X - b)^3 t^2 - \frac{(X - x^0)(X - b)}{\lambda(x^0 - b)}, \quad X \in [x^0, b]. \quad (76)$$

Thus, the structure is fixed at the right side $X = b$ and the interface $x_{\text{fsi}}(t)$ satisfies

$$x_{\text{fsi}}(t) = x^0 + d(X = x^0, t) = x^0 + c_1 (x^0 - b)^3 t^2. \quad (77)$$

As we discussed before, we have the FSI boundary conditions

$$\begin{cases} u = \dot{d}, \\ \sigma_f = \sigma_s, \\ \partial T / \partial x = 0, \end{cases} \quad (78)$$

at $x = x_{\text{fsi}}(t)$ or $X = x^0$. The stress tensors are $\sigma_f = -p + \frac{1}{Re}(\frac{2}{3}u_x)$ and $\sigma_s = \lambda d_X$. From the conditions above, we can easily obtain

$$u_b(t) = 2c_1(x^0 - b)^3 t, \quad \text{and} \quad p_b(t) = 1 - 3\lambda c_1(x^0 - b)^2 t^2.$$

The source terms $f_1(x, t)$, $f_2(x, t)$, $f_3(x, t)$ and $S(X, t)$ can be simulated by substituting the exact solution. In this example, we take $a = 0, b = 1.2, \rho_s = 1, \lambda = 10, c_1 = -2, c_2 = c_3 = 0.1, Re = 1000$. The final time is $t_{\text{end}} = 0.5$.

When we do numerical simulation, we divide the domain $[a, b]$ into N_f cells with mesh size Δx_f , and divide $[x_{\text{fsi}}(0), b]$ into N_s cells mesh. Then, we set the ghost points of the fluid region as $x_{n_f+j} = x_{n_f} + j\Delta x_f$, $j = 0, 1, 2, \dots$, where the integer n_f satisfies $x_{n_f-1} \leq x_{\text{fsi}}(t) < x_{n_f}$.

At first, we test the flow solver. Assume that the FSI boundary conditions $u = V_b$ and $\frac{\partial T}{\partial x} = 0$ are already known for fluid. Time step is taken as

$$\Delta t = \min \left(\frac{0.6}{\lambda_c / \Delta x + \lambda_d / \Delta x^2}, \frac{\Delta x}{\max(|u_b|)} \right). \quad (79)$$

Here, $\lambda_c = \max_i(|u| + c)$, $\lambda_d = \max_i(\frac{1}{Re\rho}, \frac{4}{3Re\rho}, \frac{\gamma}{PrRe\rho})$. We list the L_∞, L_1 and L_2 errors of u at final time in Table 1. The third order accuracy is achieved for fluid.

Afterwards, we turn to test the solid solver with the known σ_s on $x_{\text{fsi}}(t)$. The time step is taken as $\Delta t = 10^{-7}$ to ensure the spatial errors play a leading role. Errors of \dot{d} are shown in Table 2, indicating the third order accuracy.

In the end, we combine the solvers and validate its efficiency. The fluid and solid exchange information from each other at the fluid-structure interface and they share the same time step as in (79). In Table 3, we show the errors at final time. We can clearly see that the third order accuracy is achieved for the FSI solver.

Table 1. Example 1. Fluid solver: errors and convergence orders of accuracy of u at $t = 0.5$.

N_f	L_∞ errors	order	L_1 errors	order	L_2 errors	order
50	9.726E-05	–	4.270E-05	–	5.227E-05	–
100	5.221E-06	4.219	2.556E-06	4.061	3.055E-06	4.096
200	5.236E-07	3.317	2.453E-07	3.381	3.110E-07	3.295
400	7.233E-08	2.855	3.302E-08	2.893	3.933E-08	2.983
800	8.078E-09	3.162	3.918E-09	3.074	4.651E-09	3.079
1600	9.023E-10	3.162	4.489E-10	3.125	5.465E-10	3.089

Table 2. Example 1. Solid solver: errors and convergence orders of accuracy of \dot{d} at $t = 0.5$.

N_s	L_∞ errors	order	L_1 errors	order	L_2 errors	order
10	7.703E-07	–	1.537E-07	–	3.437E-07	–
20	9.634E-08	2.999	1.924E-08	2.997	4.303E-08	2.997
40	1.203E-08	3.001	2.405E-09	3.000	5.377E-09	3.000
80	1.504E-09	3.000	3.005E-10	3.000	6.720E-10	3.000
160	1.893E-10	2.989	3.758E-11	2.999	8.405E-11	2.999

Table 3. Example 1. FSI solver: errors and convergence orders of accuracy of velocity u and \dot{d} at $t = 0.5$.

(N_f, N_s)	u				\dot{d}			
	L_∞ errors	order	L_2 errors	order	L_∞ errors	order	L_2 errors	order
(300,10)	1.968E-07	–	1.031E-07	–	8.693E-07	–	3.461E-07	–
(600,20)	2.096E-08	3.231	1.225E-08	3.072	9.840E-08	3.143	4.303E-08	3.007
(1200,40)	2.276E-09	3.202	1.390E-09	3.139	1.224E-08	3.006	5.379E-09	2.999
(2400,80)	2.656E-10	3.099	1.629E-10	3.092	1.637E-09	2.903	6.727E-10	2.999
(4800,160)	4.592E-11	2.532	2.662E-11	2.613	2.262E-10	2.855	8.431E-11	2.996

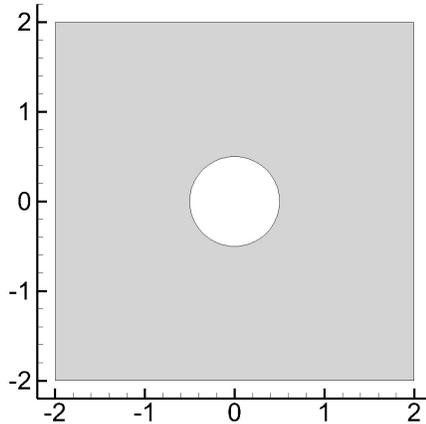


Figure 8. Example 2: computational domain.

Example 2. 2D Accuracy test

Now we consider the 2D Navier-Stokes equations with additional source terms to validate the convergence. The exact solution is

$$\rho(x, y, t) = 1, \quad p(x, y, t) = \exp((\sqrt{x^2 + y^2} - r)^2 t),$$

$$u(x, y, t) = (x^2 + y^2 - r^2) \frac{x}{\sqrt{x^2 + y^2}} \cos(t), \quad v(x, y, t) = (x^2 + y^2 - r^2) \frac{y}{\sqrt{x^2 + y^2}} \cos(t).$$

The computational domain is

$$\Omega = \{(x, y) \mid -2 \leq x \leq 2, -2 \leq y \leq 2, x^2 + y^2 \geq r^2\}, \quad (80)$$

as shown in Fig. 8. Here, we choose $r = 0.5$. The boundaries include the central circle and four sides of the square. Outflow boundary conditions are given for the outer boundary. No-slip and adiabatic boundary conditions are given for the inner circle. We equip the third-order AWENO method and the new SILW boundary treatment for this test. Reynolds number $Re = 500$ in this example and the final time is $t = 0.1$. The domain $[-2, 2] \times [-2, 2]$ is discretized by a uniform mesh with $N_x \times N_y$ points. The time step is taken as

$$\Delta t = \frac{0.6}{\lambda_u/\Delta x + \lambda_v/\Delta y + 6\lambda_d/\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right)}. \quad (81)$$

Here, $\lambda_u = \max_{i,j}(|u| + c)$, $\lambda_v = \max_{i,j}(|v| + c)$, $\lambda_d = \max_{i,j}\left(\frac{1}{Re\rho}, \frac{4}{3Re\rho}, \frac{\gamma}{PrRe\rho}\right)$. In Table 4, we can see third order convergence of the velocity u in L_∞ , L_1 and L_2 norms.

Table 4. Example 2: errors and convergence orders of accuracy of u at $t = 0.1$.

$N_x \times N_y$	L_∞ errors	order	L_1 errors	order	L_2 errors	order
60×60	4.688E-03	–	6.233E-03	–	2.658E-03	–
120×120	8.043E-04	2.543	2.260E-03	1.463	9.103E-04	1.546
240×240	7.494E-05	3.423	1.897E-04	3.574	8.340E-05	3.448
480×480	7.595E-06	3.302	8.739E-06	4.440	4.193E-06	4.313
960×960	5.516E-07	3.783	7.423E-07	3.557	3.419E-07	3.616

Example 3. Acoustic scattering at low Mach numbers

Next, we apply our flow solver to a numerical test case which simulates acoustic wave scattering from a circular cylinder. A cylinder with radius $r = 0.5$ is placed with its center at $(0, 0)$. The initial conditions are localized pressure perturbations of the Gaussian distribution:

$$\delta p(x, y, t = 0) = \epsilon \exp\left(-\frac{\ln 2 \left((x - 4)^2 + y^2\right)}{0.2^2}\right), \quad (82)$$

where $\epsilon = 10^{-3}$. Thus the corresponding initial conditions for the primitive variables are

$$\rho = 1 + \delta p, \quad p = \frac{1}{\gamma} + \delta p, \quad u = 0, \quad v = 0. \quad (83)$$

The Reynolds number for this example is $Re = 5 \times 10^5$. The problem is simulated in a rectangular domain $\Omega = [-6, 12] \times [-12, 12]$ with 900×1200 grid points. On the cylinder

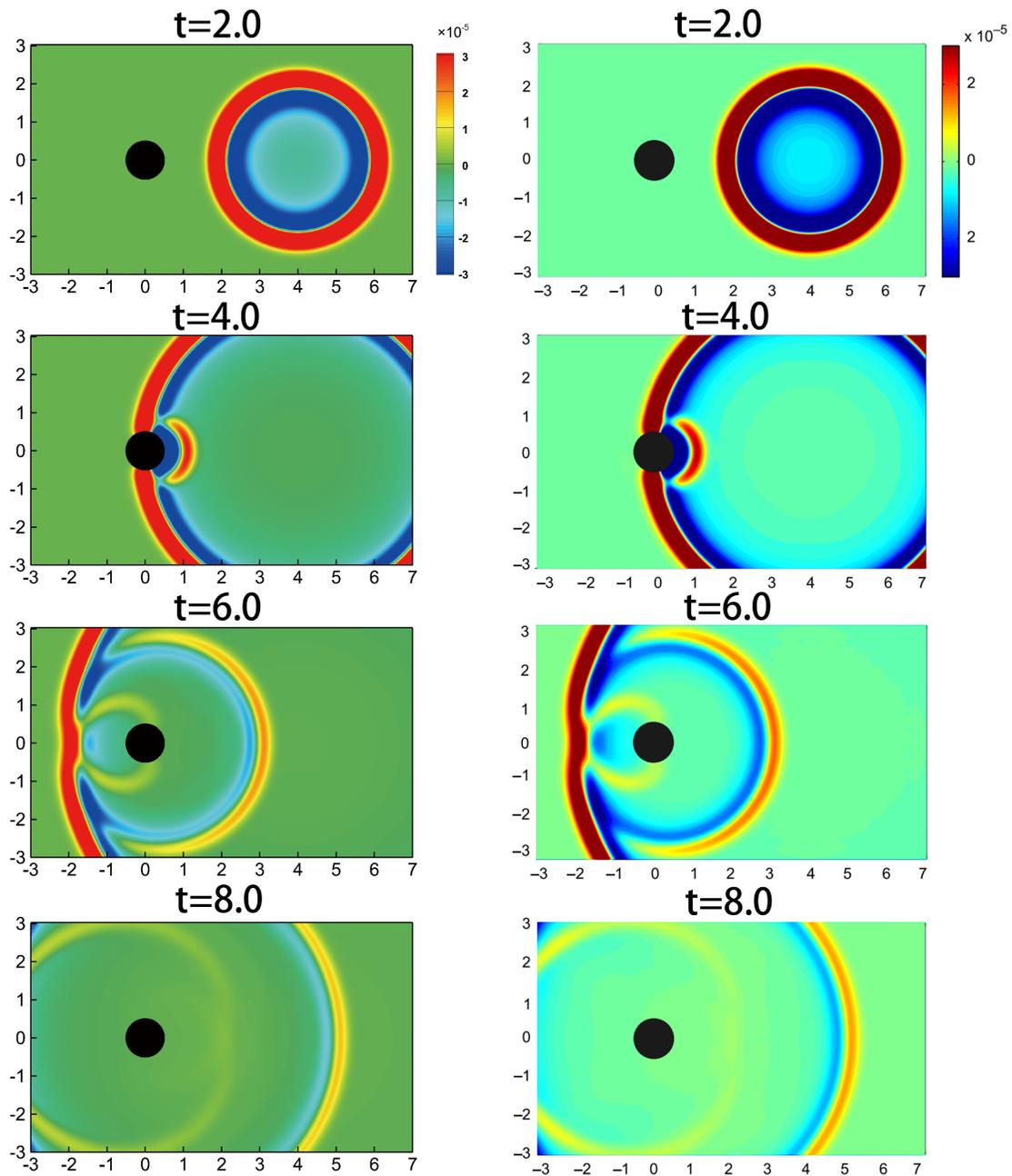


Figure 9. Example 3: Pressure perturbation contours at different times. Left: the results in the present study. Right: the results given in [21].

surface, no-slip wall and adiabatic boundary conditions are applied. Outflow boundary conditions are given for the outer flow boundaries.

The results are shown in Fig. 9. The wave propagates in the flow region and acoustic scatters and reflects from the cylinder. We compare the present results with those of Liu

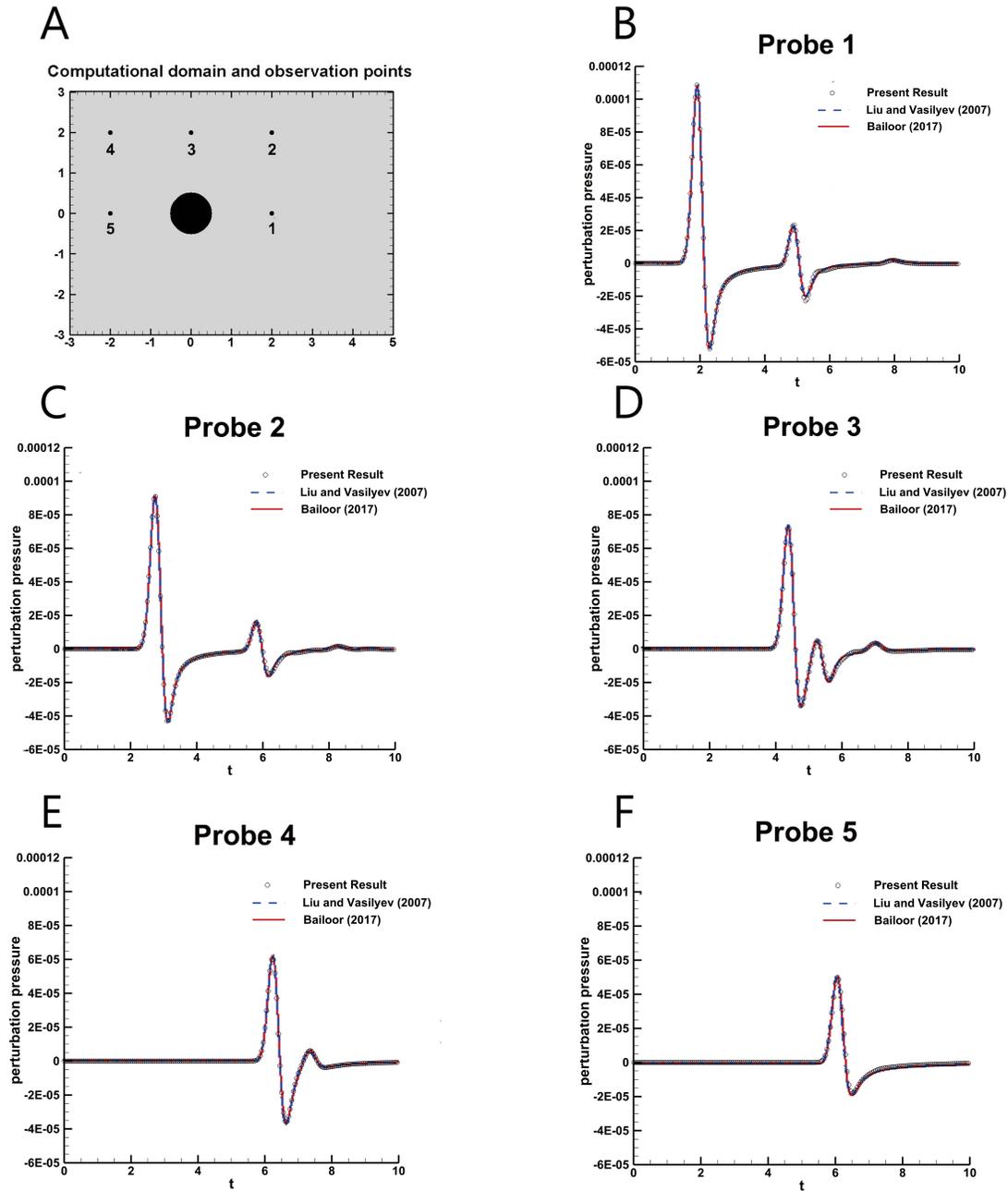


Figure 10. Example 3: (A) Five probes around the cylinder; (B)-(F) Comparison of the pressure at five probes in the present study with the result of Bailoor [1] and Liu and Vasilyev [21].

and Vasilyev [21]. It can be seen that the general structure of the solution is well resolved and our results agree well with the others.

As shown in Fig. 10 (A), we place five numerical probes around the cylinder at $(2, 0)$, $(2, 2)$, $(0, 2)$, $(-2, 2)$ and $(-2, 0)$. The perturbation pressure at each probe is plotted and

compared with result of Bailoor [1] and Liu and Vasilyev [21] in Figs. 10 (B)-(F). We can observe that our results are consistent with those in the literature.

Example 4. Lift-off of a rigid cylinder

In the following, we consider the interaction of a shock wave and a rigid cylinder in a two-dimensional viscous fluid. It was first proposed in [7]. The computational domain is $[0, 1] \times [0, 0.2]$. A rigid cylinder with radius $r = 0.05$ is placed in the domain and its center locates at $(0.15, 0.05)$. The mass density of the cylinder is 7.6. A Mach 3 shock wave moving from left is initially placed at $x = 0.08$. The initial settings of the gas are

$$\rho = 1, \quad u = 0, \quad v = 0, \quad p = 1. \quad (84)$$

We apply inflow and outflow boundary conditions at the left boundary and right boundary, respectively. The upper and lower boundaries are reflective boundaries. No-slip and adiabatic boundary conditions are given for the surface of cylinder. For this example, Reynolds number $Re = 2 \times 10^5$.

The numerical results at different times with the mesh spacing $\Delta x = \Delta y = \frac{1}{1600}$ are shown in Fig. 11. The cylinder is driven and lifted upwards by the shock wave. We also present the density contour at $t = 0.255$ in Fig. 12. In additional, Fig. 13 shows the position of the cylinder and it agrees well with the results in [26], indicating the effectiveness of our scheme.

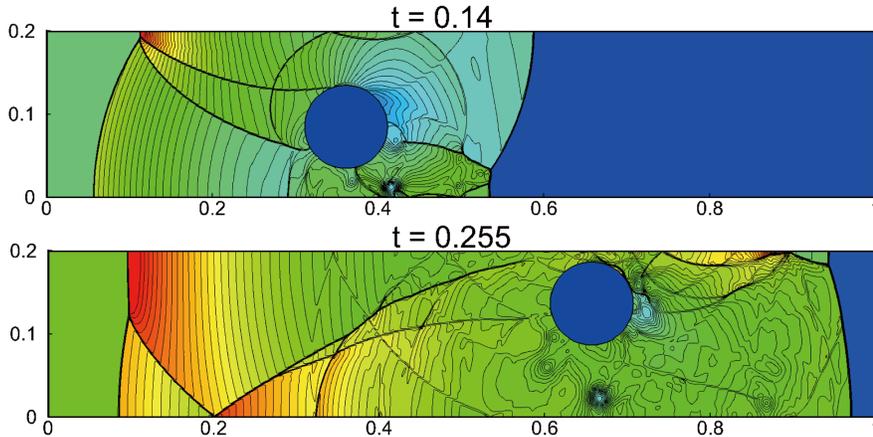


Figure 11. Example 4: Contours of fluid pressure at times $t=0.14$ (60 contours from 0 to 28) and $t=0.255$ (60 contours from 0 to 18).

Example 5. Shock wave impact on a deforming panel

In this example, we calculate the deformation of a panel induced by a shock wave in a shock tube, as shown in Fig. 14. This test case was first investigated both experimentally and numerically in Giordano [12] and used to validate the numerical algorithms later by [1, 29, 30, 40]. The size of the shock tube is shown in Fig. 14. The thickness and the

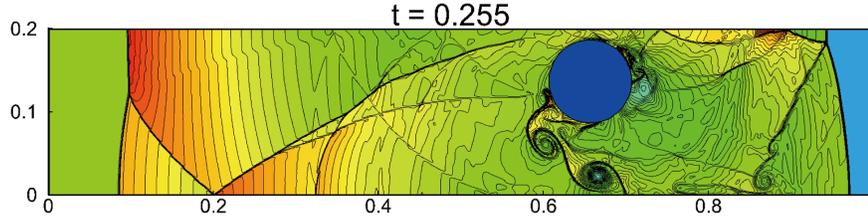


Figure 12. Example 4: Contours of fluid density at time $t=0.255$ (60 contours from 0 to 6).

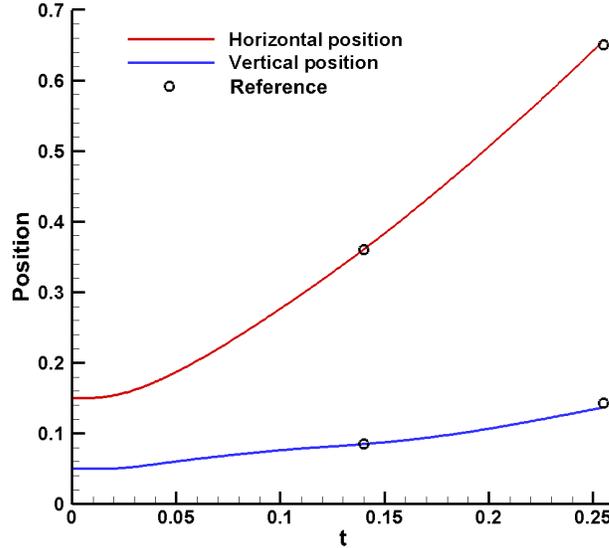


Figure 13. Example 4: Comparison of the horizontal and vertical position of the center of the cylinder with Monasse et al.[26].

length of the panel are respectively 1mm and 50mm. The panel is fixed on a forward-facing non-deforming step at its lower end. A Mach 1.21 shock wave moves from left and interacts with the panel. The parameters and properties of the air and panel are listed in Table 5, following the set up in [29].

Table 5. Example 5: Initial material properties of the impacted panel model.

Material	ρ (kg/m ³)	p (kPa)	E (GPa)	ν	γ	u_x (m/s)
Stationary air	1.20	101.0	-	-	1.4	0
Inflow air	1.63	155.7	-	-	1.4	109.6
Steel (panel)	7600	0	220	0.3	-	0

The computational domain is $[0, 0.4] \times [0, 0.08]$. We employ non-uniform Cartesian grid with 600×400 points for fluid which is denser ($\Delta x = 0.08\text{mm}$, $\Delta y = 0.2\text{mm}$) within the

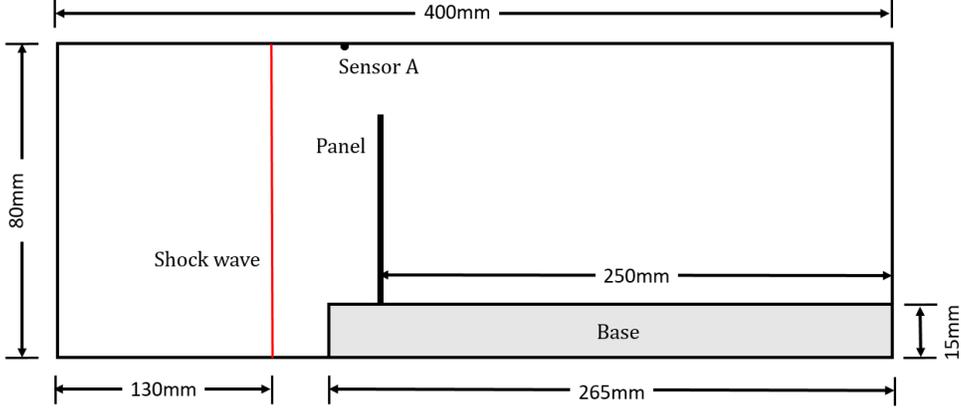


Figure 14. Example 5: Geometric setup for the fluid-structure interaction.

range of motion of the elastic plate. Then, we solve the equations after mapping the grid to a uniform Cartesian one. The detail of AWENO method on curvilinear grid can be found in [17]. 4×200 rectangular elements are equipped for panel solver. The time step is taken as

$$\Delta t = \min \left\{ 0.9 \frac{\Delta x}{\max|u|}, 0.9 \frac{\Delta y}{\max|v|}, 0.6 / \left(\max|\lambda_{\bar{u}}| \frac{1}{\Delta x} + \max|\lambda_{\bar{v}}| \frac{1}{\Delta y} + 6 \max|\lambda_d| \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right) \right) \right\}. \quad (85)$$

In the previous tests, we used the third order AWENO for fluid spatial discretization, since we can overall maintain third order accuracy with the high order boundary treatment and third order solid solver. However, we find that fifth order AWENO shows higher resolution in fluid region for this test. Thus the results in this example are all obtained by fifth order AWENO.

The shock wave arrives around the panel at about $100\mu s$. We take this moment as $t = 0$. Fig. 15 shows the simulation results, namely, the mass density gradient distribution during $t = [0, 630\mu s]$. The left column is experimental shadowgraphs and the middle column is the numerical schlieren by Giordano [12]. Our result (right column) catch the details of the flow when interacting with the elastic panel. When the shock wave interacts with the structure, reflected and transmitted shock waves appear around the panel. We can clearly see that there are vortices dropping from the panel ending. Our results show good performance on the position of waves and vortices, compared with those in [12].

In Fig. 16, we plot the displacement and the von Mises stress distribution during $t = [0, 4.9ms]$, where the von Mises stress is defined as

$$\sigma_\nu = \sqrt{\sigma_{s,XX}^2 - \sigma_{s,XX} \sigma_{s,YY} + \sigma_{s,YY}^2 + 3\sigma_{s,XY}^2}. \quad (86)$$

The panel deforms periodically under the loaded shock wave and reaches the largest horizontal displacement at the tip. We can find that when the panel swing severely

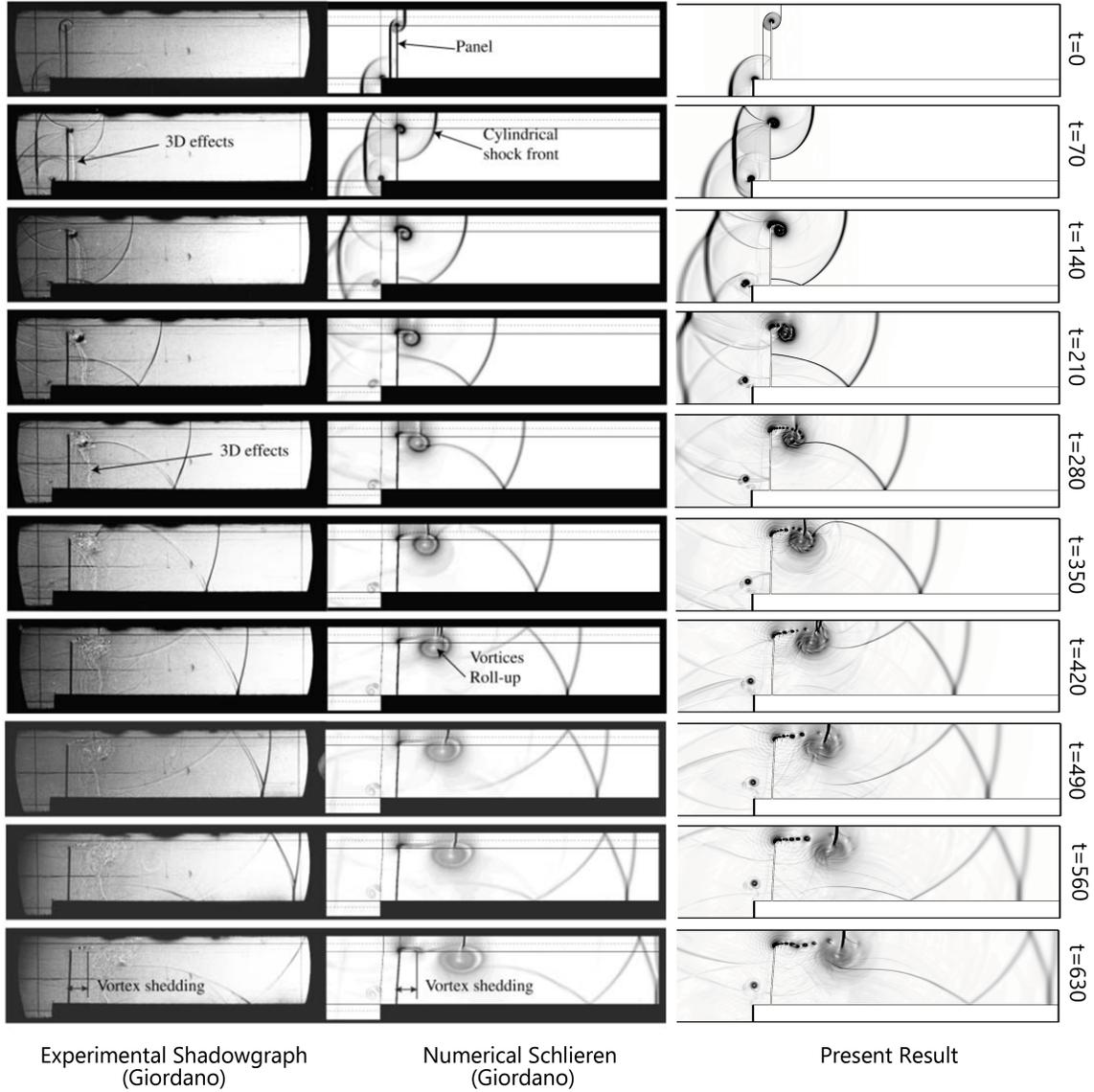


Figure 15. Example 5: Qualitative comparison of shock propagation during $t = [0, 630\mu s]$. The plots, from left to right, are obtained from Giordano’s experiments, numerical simulation [12] and the present result with a locally denser grid about $\Delta x = 0.08\text{mm}$ around the panel.

($t = 1.4\text{ms}$), the stress σ_ν is greater at the bottom boundary. We test the both 50mm and 40mm panel. The displacement of the tip of the panel is compared with available data from the literature [12] in Fig. 17 (A) and (C). Besides, the time-varying pressure at sensor A (at the top, 10 mm to the left of the panel) are plotted in Fig. 17 (B) and (D). We observe that the results of the present numerical simulation and Giordano’s numerical

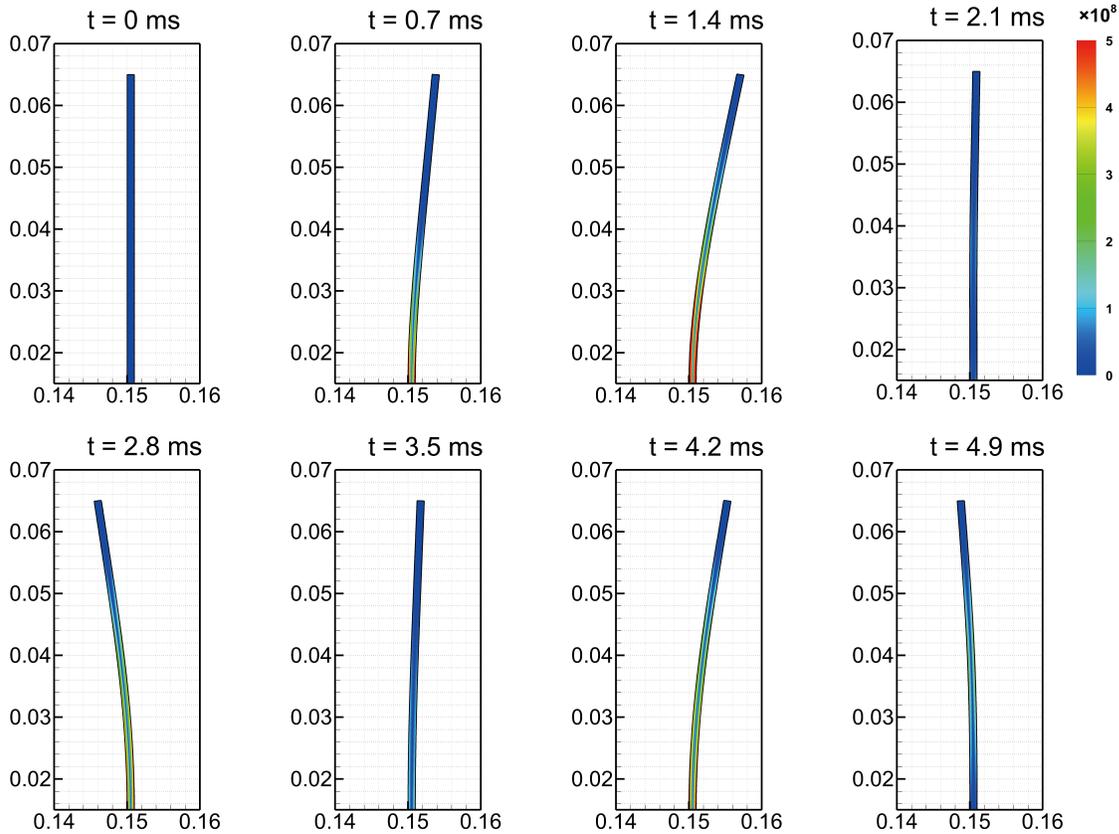


Figure 16. Example 5: The von Mises stress distribution σ_v of the 50 mm panel during $t = [0, 4.9\text{ms}]$.

results show good agreement.

5 Conclusion

In this paper, we present an FSI solver based on the inverse Lax-Wendroff boundary treatment for fluid-structure interaction problems between compressible viscous flows and deformable structure. The method consists of three parts: a flow solver based on the high order finite difference AWENO method, a structure solver using the finite element method, and loosely partitioned coupling strategy based on the ILW procedure. With the help of the ILW method, the solver can maintain high order accuracy globally.

To validate our method, we conduct several numerical tests, including 1D and 2D accuracy tests and some benchmark cases. We also compare the results with the experimental data and other simulations in the literature. Results show excellent agreement and we conclude that our method could handle two-dimensional FSI problems involving shock wave and structure deformations with high order accuracy and non-oscillatory

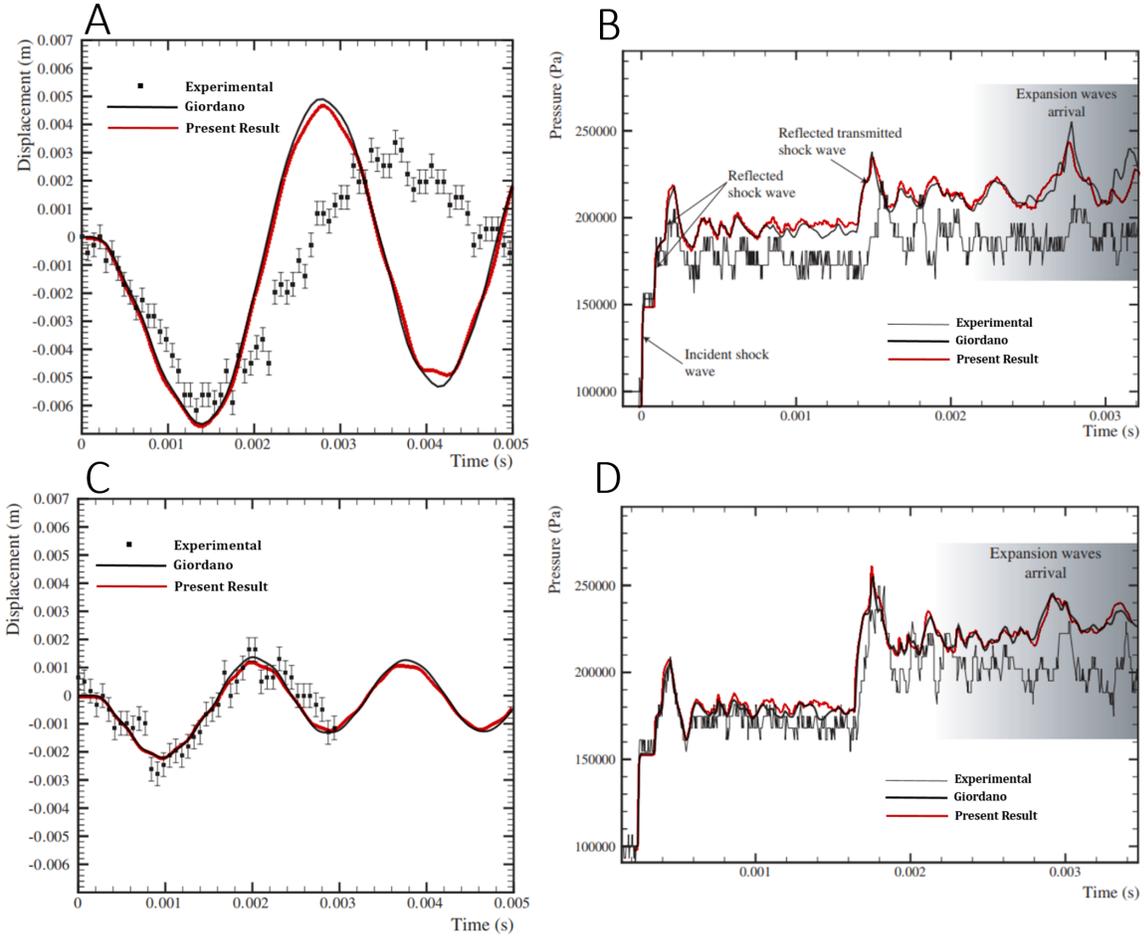


Figure 17. Example 5: Horizontal tip displacement and time-varying pressure at sensor, compared with experimental and numerical results in [12]. (A) Horizontal tip displacement of the 50mm panel end. (B) Time-varying pressure at sensor for 50mm panel. (C) Horizontal tip displacement of the 40mm panel end. (D) Time-varying pressure at sensor for 40mm panel.

performance.

It should be pointed out that the present partitioned solvers are loosely coupled, and we do not consider and analyze any possible instability problems here. The detailed design on the interface boundary treatment could also be time consuming, especially when considering extending it to three dimensions or problems including large deformations. In future work, we hope to improve the existing procedure to be more efficient and robust and extend it to FSI problems involves incompressible flow or more complicated structure.

Conflict of interest

The authors declare that they have no conflict of interest.

References

- [1] Shantanu Bailoor, Aditya Annangi, Jung Hee Seo, and Rajneesh Bhardwaj. Fluid–structure interaction solver for compressible flows with applications to blast loading on thin elastic structures. *Applied Mathematical Modelling*, 52:470–492, 2017.
- [2] Rajneesh Bhardwaj and Rajat Mittal. Benchmarking a coupled immersed-boundary-finite-element solver for large-scale flow-induced deformation. *AIAA Journal*, 50(7):1638–1642, 2012.
- [3] Rajneesh Bhardwaj, Kimberly Ziegler, Jung Hee Seo, KT Ramesh, and Thao D Nguyen. A computational model of blast loading on the human eye. *Biomechanics and Modeling in Mechanobiology*, 13:123–140, 2014.
- [4] Mark H Carpenter, David Gottlieb, Saul Abarbanel, and Wai-Sun Don. The theoretical accuracy of Runge–Kutta time discretizations for the initial boundary value problem: a study of the boundary error. *SIAM Journal on Scientific Computing*, 16(6):1241–1252, 1995.
- [5] Ziqiang Cheng, Shihao Liu, Yan Jiang, Jianfang Lu, Mengping Zhang, and Shuhai Zhang. A high order boundary scheme to simulate complex moving rigid body under impingement of shock wave. *Applied Mathematics and Mechanics*, 42(6):841–854, 2021.
- [6] Thomas Dunne and Rolf Rannacher. Adaptive finite element approximation of fluid–structure interaction based on an Eulerian variational formulation. In *Fluid-structure Interaction: Modelling, Simulation, Optimisation*, pages 110–145. Springer, 2006.
- [7] J Falcovitz, G Alfandary, and G Hanoch. A two-dimensional conservation laws scheme for compressible flows with moving boundaries. *Journal of Computational Physics*, 138(1):83–102, 1997.
- [8] Francis Filbet and Chang Yang. An inverse Lax–Wendroff method for boundary conditions applied to Boltzmann type models. *Journal of Computational Physics*, 245:43–61, 2013.
- [9] Shailesh Ganpule, Aaron Alai, Erwan Plougonven, and Namas Chandra. Mechanics of blast loading on the head models in the study of traumatic brain injury using experimental and computational approaches. *Biomechanics and Modeling in Mechanobiology*, 12:511–531, 2013.

- [10] Luciano Garelli, Rodrigo R Paz, and Mario A Storti. Fluid–structure interaction study of the start-up of a rocket engine nozzle. *Computers & Fluids*, 39(7):1208–1218, 2010.
- [11] Reza Ghias, Rajat Mittal, and Haibo Dong. A sharp interface immersed boundary method for compressible viscous flows. *Journal of Computational Physics*, 225(1):528–553, 2007.
- [12] Jerome Giordano, Georges Jourdan, Y Burtschell, M Medale, DE Zeitoun, and L Houas. Shock wave impacts on deforming panel, an application of fluid-structure interaction. *Shock Waves*, 14(1):103–110, 2005.
- [13] Jaroslava Hasnedlová, Miloslav Feistauer, Jaromír Horáček, Adam Kosík, and Václav Kučera. Numerical simulation of fluid–structure interaction of compressible flow and elastic structure. *Computing*, 95:343–361, 2013.
- [14] Wei-Xi Huang and Hyung Jin Sung. An immersed boundary method for fluid–flexible structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 198(33-36):2650–2661, 2009.
- [15] Thomas JR Hughes. *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation, 2012.
- [16] Yan Jiang, Chi-Wang Shu, and Mengping Zhang. An alternative formulation of finite difference weighted ENO schemes with Lax–Wendroff time discretization for conservation laws. *SIAM Journal on Scientific Computing*, 35(2):A1137–A1160, 2013.
- [17] Yan Jiang, Chi-Wang Shu, and Mengping Zhang. Free-stream preserving finite difference schemes on curvilinear meshes. *Methods and Applications of Analysis*, 21(1):1–30, 2014.
- [18] Antoine Legay, Jack Chessa, and Ted Belytschko. An Eulerian–Lagrangian method for fluid–structure interaction based on level sets. *Computer Methods in Applied Mechanics and Engineering*, 195(17-18):2070–2087, 2006.
- [19] Tingting Li, Chi-Wang Shu, and Mengping Zhang. Stability analysis of the inverse Lax–Wendroff boundary treatment for high order upwind-biased finite difference schemes. *Journal of Computational and Applied Mathematics*, 299:140–158, 2016.
- [20] Tingting Li, Chi-Wang Shu, and Mengping Zhang. Stability analysis of the inverse Lax–Wendroff boundary treatment for high order central difference schemes for diffusion equations. *Journal of Scientific Computing*, 70:576–607, 2017.
- [21] Qianlong Liu and Oleg V Vasilyev. A brinkman penalization method for compressible flows in complex geometries. *Journal of Computational Physics*, 227(2):946–966, 2007.

- [22] Shihao Liu, Yan Jiang, Chi-Wang Shu, Mengping Zhang, and Shuhai Zhang. A high order moving boundary treatment for convection-diffusion equations. *Journal of Computational Physics*, 473:111752, 2023.
- [23] Shihao Liu, Tingting Li, Ziqiang Cheng, Yan Jiang, Chi-Wang Shu, and Mengping Zhang. A new type of simplified inverse Lax-Wendroff boundary treatment for convection-diffusion equations. *in preparation*.
- [24] Shihao Liu, Tingting Li, Ziqiang Cheng, Yan Jiang, Chi-Wang Shu, and Mengping Zhang. A new type of simplified inverse Lax-Wendroff boundary treatment for hyperbolic conservation laws. *in preparation*.
- [25] Jianfang Lu, Jinwei Fang, Sirui Tan, Chi-Wang Shu, and Mengping Zhang. Inverse Lax–Wendroff procedure for numerical boundary conditions of convection–diffusion equations. *Journal of Computational Physics*, 317:276–300, 2016.
- [26] Laurent Monasse, Virginie Daru, Christian Mariotti, Serge Piperno, and Christian Tenaud. A conservative coupling algorithm between a compressible flow and a rigid body using an embedded boundary method. *Journal of Computational Physics*, 231(7):2977–2994, 2012.
- [27] Nishant Nangia, Neelesh A Patankar, and Amneet Pal Singh Bhalla. A DLM immersed boundary method based wave-structure interaction solver for high density ratio multiphase flows. *Journal of Computational Physics*, 398:108804, 2019.
- [28] Maria Giuseppina Chiara Nestola, Barna Becsek, Hadi Zolfaghari, Patrick Zulian, Dario De Marinis, Rolf Krause, and Dominik Obrist. An immersed boundary method for fluid-structure interaction based on variational transfer. *Journal of Computational Physics*, 398:108884, 2019.
- [29] Jianguo Ning, Hetao Zhang, Xiangzhao Xu, and Tianbao Ma. A novel fluid-structure interaction algorithm for compressible flows and deformable structures. *Journal of Computational Physics*, 426:109921, 2021.
- [30] Vito Pasquariello, Georg Hammerl, Felix Örley, Stefan Hickel, Caroline Danowski, Alexander Popp, Wolfgang A Wall, and Nikolaus A Adams. A cut-cell finite volume–finite element coupling approach for fluid–structure interaction in compressible flow. *Journal of Computational Physics*, 307:670–695, 2016.
- [31] Charles S Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25(3):220–252, 1977.
- [32] PB Ryzhakov, Riccardo Rossi, SR Idelsohn, and Eugenio Onate. A monolithic Lagrangian approach for fluid–structure interaction problems. *Computational Mechanics*, 46:883–899, 2010.

- [33] Jung Hee Seo and Rajat Mittal. A high-order immersed boundary method for acoustic wave scattering and low-Mach number flow-induced sound in complex geometries. *Journal of Computational Physics*, 230(4):1000–1019, 2011.
- [34] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77(2):439–471, 1988.
- [35] Sirui Tan and Chi-Wang Shu. Inverse Lax-Wendroff procedure for numerical boundary conditions of conservation laws. *Journal of Computational Physics*, 229(21):8144–8166, 2010.
- [36] Sirui Tan and Chi-Wang Shu. A high order moving boundary treatment for compressible inviscid flows. *Journal of Computational Physics*, 230(15):6023–6036, 2011.
- [37] Sirui Tan, Cheng Wang, Chi-Wang Shu, and Jianguo Ning. Efficient implementation of high order inverse Lax–Wendroff boundary treatment for conservation laws. *Journal of Computational Physics*, 231(6):2510–2527, 2012.
- [38] Paulo Roberto de Freitas Teixeira and Armando Miguel Awruch. Numerical simulation of fluid–structure interaction using the finite element method. *Computers & Fluids*, 34(2):249–273, 2005.
- [39] Fang-Bao Tian, Haoxiang Luo, Luoding Zhu, James C Liao, and Xi-Yun Lu. An efficient immersed boundary-lattice Boltzmann method for the hydrodynamic interaction of elastic filaments. *Journal of Computational Physics*, 230(19):7266–7283, 2011.
- [40] Li Wang, Gaetano MD Currao, Feng Han, Andrew J Neely, John Young, and Fang-Bao Tian. An immersed boundary method for fluid–structure interaction with compressible multiphase flows. *Journal of Computational Physics*, 346:131–151, 2017.
- [41] Thomas Wick. Coupling of fully Eulerian and arbitrary Lagrangian–Eulerian methods for fluid-structure interaction computations. *Computational Mechanics*, 52(5):1113–1124, 2013.
- [42] Lincheng Xu, Fang-Bao Tian, John Young, and Joseph CS Lai. A novel geometry-adaptive Cartesian grid based immersed boundary–lattice Boltzmann method for fluid–structure interactions at moderate and high Reynolds numbers. *Journal of Computational Physics*, 375:22–56, 2018.
- [43] Jianming Yang and Frederick Stern. A simple and efficient direct forcing immersed boundary framework for fluid–structure interactions. *Journal of Computational Physics*, 231(15):5029–5061, 2012.