An Immersed Boundary Method for Solving the Compressible Navier-Stokes Equations with Fluid-Structure Interaction

Christoph Brehm[*1], Michael F. Barad[†2], and Cetin C. Kiris[‡2]

[1]*Science and Technology Corporation, Moffett Field, CA 94035*

[2]*NASA Ames Research Center, Moffett Field, CA 94035*

An immersed boundary method for the compressible Navier-Stokes equation and the additional infrastructure that is needed to solve moving boundary problems and fully coupled fluid-structure interaction is described. All the methods described in this paper were implemented in NASA's LAVA solver framework. The underlying immersed boundary method is based on the locally stabilized immersed boundary method that was previously introduced by the authors. In the present paper this method is extended to account for all aspects that are involved for fluid structure interaction simulations, such as fast geometry queries and stencil computations, the treatment of freshly cleared cells, and the coupling of the computational fluid dynamics solver with a linear structural finite element method. The current approach is validated for moving boundary problems with prescribed body motion and fully coupled fluid structure interaction problems in 2D and 3D. As part of the validation procedure, results from the second AIAA aeroelastic prediction workshop are also presented. The current paper is regarded as a proof of concept study, while more advanced methods for fluid structure interaction are currently being investigated, such as geometric and material nonlinearities, and advanced coupling approaches.

I.   Introduction

A key motivation of this work is to enable improved aeroelastic predictions for highly complex geometries while reducing the pre-processing time spent during mesh generation. All methods presented in this paper have been developed in the Launch Ascent and Vehicle Aerodynamics (LAVA) framework, within the Computational Aerosciences Branch at NASA Ames Research Center (ARC). LAVA addresses some of the existing shortcomings in computational fluid dynamics (CFD) which are relevant to NASA related applications. Figure 1 displays one of these applications, namely the flow past a parachute using a rigid approximation. Parachutes are being investigated for deceleration of payloads during earth re-entry or Mars entry. These simulations are extremely challenging due to the complex fluid dynamics that are strongly coupled to the large deformations of the parachute. The current paper focuses mainly on the second application, namely wing flutter. Wing flutter analysis may be considered a more conventional application of fluid-structure interaction. In modern airplane design, increasingly more flexible wing designs are being considered. Hence, reliable flutter predictions are crucial for these novel designs.

To assess aeroelastic modeling capabilities, NASA and AIAA organized the Aeroelastic Prediction Workshop (AePW) series. An overview of the workshop and an analysis of the results is provided in Schuster *et al.*[1] and Heeg *et al.*[2] Figure 2 shows the experimental setup for the flutter experiments used in the workshop. Within this workshop, researchers from all over the world are conducting aeroelastic predictions for a given set of test problems where experimental data is available. By comparing the results and analyzing different approaches with each other, the researchers are gaining an improved understanding of current strengths and weaknesses of available FSI approaches. In the present paper, the newly developed capabilities within LAVA

---

[*]Research Professor, University of Arizona and AIAA Senior Member

[†]Research Scientist, Computational Aerosciences Branch

[‡]Branch Chief, Computational Aerosciences Branch and AIAA Senior Member

American Institute of Aeronautics and Astronautics

are applied to some of the test problems from the second AIAA AePW workshop that was held on January $2-3$, 2016, in San Diego, CA. The set of test problems was put together in a way that the complexity of the analysis subsequently increased.

In the first test case, the flow past a stiff wing was simulated. For this test problem, excellent agreement was obtained among all participants. It demonstrates that the simulation capabilities have converged to a level of maturity where the fluid dynamics for an attached flow past a wing geometry can be reliably predicted. The next category of test problems involved the prediction of the aerodynamics of forced oscillation with and without separation. The reliable prediction of unsteady separated flow is still a big challenge in CFD, while reasonable agreement was achieved for forced oscillation without separation. Finally, the structural response of the wing was included. Although, the structural response of the wing could be captured well with a straightforward structural model the prediction of flutter onset, especially in the presence of flow separation, turned out to be challenging. The key outcome of the workshop is that significant research has to go into the understanding and improvement of currently available FSI methods.

Figure 1: Flow visualization of a rigid parachute simulation.

This paper presents and validates a less conventional FSI method, in which a higher-order immersed boundary method for the compressible Navier-Stokes equations is coupled with a structural finite element method (FEM). Immersed Boundary Methods (IBMs) have been developed for many years and have appeared in various forms since they were first introduced by Peskin[3,4] (see for example Goldstein *et al.*,[5] LeVeque and Li,[6] Wiegmann and Bube,[7] Linnick and Fasel,[8] Johansen and Collela,[9] Mittal and Iaccarino,[10] Zhong,[11] Duan *et al.*[12] and many others). These methods were first introduced as a nontraditional approach for numerically solving initial/boundary-value problems for complex geometries on Cartesian meshes and have matured to become increasingly important for a wide range of applications.[13–15]

Figure 2: Experimental setup of flutter experiment.

In the current paper, a higher-order sharp immersed boundary method is utilized. The IBM employed for solving the compressible Navier-Stokes equations was discussed in detail in Brehm *et al.*[16] The approach is similar to that described by Li and Ito.[17] To derive a set of linear equations for the coefficients of the boundary stencil the method aims at determining the finite-difference coefficients by minimizing the order of the local truncation error. In addition to the set of linear equations, this method considers an additional equation based on stability considerations at the irregular grid point. The current method is classified as a sharp immersed boundary method where the boundary conditions are applied at grid line intersection points. In contrast to the ghost cell approaches or forcing approaches, no information inside the geometry is necessary. The use of the current method over ghost cell approaches may be advantageous in the presence of very thin geometries. For example, for the simulation of a contra rotating open rotor system in Brehm *et al.*, the thickness of the rotor blades was close to the Cartesian grid spacings in the vicinity of the geometry. For moving boundary problems the ghost cell method and other volume forcing type approaches have the advantage that "valid" data is available inside the geometry. Thus, when opening up cells that were inside the geometry at a previous timestep no additional treatment of these "freshly cleared cells" becomes necessary.

One of the key advantages of immersed boundary methods is that the computational meshes can be automatically generated starting from a water tight surface triangulation independent of the complexity of the geometry. The grid-generation process for body-fitted structured or unstructured grids for complex geometries is generally very laborious because the process aims at generating well behaved grids that have sufficient local resolution while minimizing the total number of required grid points. Except for simple geometries,
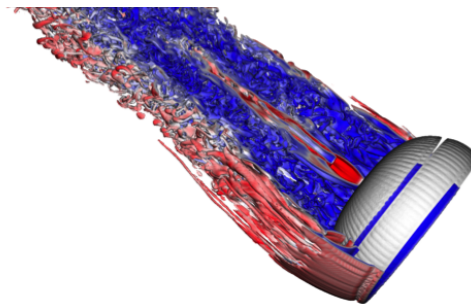
these conflicting requirements can lead to deterioration in grid quality, thereby negatively impacting the accuracy and convergence properties of the flow solver. For highly complex geometries, which are relevant for many fields of science and engineering, the process of generating a high-quality grid is extremely time consuming. For flows involving moving and deforming boundaries, IBMs also provide clear advantages over classical body-fitted grid approaches. Simulating such flows on body-fitted grids requires generating a new grid at each time step and a procedure to project the solution onto this new grid (Tezduyar[18]). These two characteristics (grid deformation and projection of solution) for body-fitted grids associated with simulating flows with moving and deforming boundaries may negatively impact the accuracy, robustness, and computational cost of the numerical solution method. Particularly, in cases where the boundary exhibits large motions, body-conformal grid strategies cause immense difficulties in the solution procedure. This is where an immersed boundary approach with its fully Eulerian approach provides great advantages. IBMs provide a much more convenient way of including the body motion and deformation by using a stationary non-deforming Cartesian grid. While these methods simplify the grid generation process, a detailed mathematical understanding of the IBM is necessary to avoid a negative impact of the boundary treatment on the robustness, the convergence behavior and the accuracy of the numerical scheme. To allow for better control of the grid resolution in the vicinity of the body, Adaptive Mesh Refinement (AMR) can be used.

The present paper proceeds as follows: first, the immersed boundary method by Brehm *et al.*[16] is briefly described and the additional infrastructure to efficiently deal with unsteady body motion is introduced. Next, a Finite Element Method (FEM) used to simulate the structural dynamics is described and its implementation is validated for simple static deformation and vibration test problems. The validation of the standalone and coupled flow solver starts with moving boundary problems of prescribed motion, including test case 1a from the AePW workshop. Finally, fully coupled fluid structure interaction problems are discussed. Experimental data is available for most of the moving boundary and fully-coupled fluid structure interaction validation problems.

II.   Numerical Methods for CFD

A.   LAVA Block-Structured Cartesian Solver

The LAVA computational framework is essentially agnostic to the grid type, such that block-structured Cartesian, curvilinear-overset, and unstructured polyhedral grids can be utilized. For each grid type a different solver module is utilized that are referred to as LAVA-Cartesian, LAVA-Curvilinear, and LAVA-Unstructured. While the long term objective is to provide a common communication interface between all three solver modules, currently the LAVA-Cartesian and LAVA-Unstructured solver modules are fully coupled through an overset approach. LAVA provides additional modules for acoustic far-field propagation and scattering, conjugate heat transfer, and computational structural dynamics. For more details about the LAVA computational framework the reader is referred to Cetin *et al.*[13]

For the work presented in this paper, LAVA-Cartesian is utilized standalone to solve the compressible Navier-Stokes equations in conservative form considering an ideal, Newtonian, non-reactive gas. LAVA-Cartesian is a block-structured, Immersed Boundary (IB) solver module with Adapative Mesh Refinement (AMR). This methodology is capable of automatically generating, refining, and coarsening nested Cartesian volumes. LAVA's AMR-IB method is designed to automatically generate the volume grids from a closed surface triangulation, and dynamically track important flow features as they develop. AMR is a proven methodology for multi-scale problems with an extensive existing mathematical and software knowledge base.[19–23] The code has been extended using data structures and inter-level operators from the high-performance Chombo AMR library[24] to provide a multi-resolution capability that can locally coarsen and refine the grid as a simulation progresses. To account for flow discontinuities, a sixth-order accurate shock-capturing weighted essentially non-oscillatory (WENO) scheme is utilized. The characteristic variable transformation is applied to a Rusanov-type flux vector splitting scheme. The accuracy and computational efficiency of the higher-order shock capturing schemes in the current solver has been previously analyzed in Brehm *et al.*[25] In this work, the viscous terms are discretized with second-order accuracy. Either the fourth-order accurate explicit Runge-Kutta scheme or the implicit second-order backward differencing scheme employing dual-timestepping are used for time-integration depending on the time-scale of the flow physics being simulated.

American Institute of Aeronautics and Astronautics

B. Immersed Boundary Method

In the present paper the IBM introduced in Brehm and Fasel[16, 26] is utilized. The basis of the IBM is that the grid-aligned 1D stencil coefficients are locally optimized in order to improve the stability of the scheme.

One of the drawbacks of the current IBM is that the stencil coefficient have to be recomputed based on the flow conditions. Figure 3 shows the optimal stencil coefficients as a function of the boundary distance ψ and the flow conditions at the irregular grid point. It can be noted that the values of the optimal stencil coefficients is much less sensitive to the flow state than to the flow properties such as the specific heat ratio γ. So far, this has only been analyzed qualitatively and a more thorough analysis will have to be conducted. Nevertheless, for a single gas computation we adopted the strategy of freezing the Jacobians and simply include the dependence on the boundary distance ψ. To avoid solving the minimization problem for the optimal stencil coefficients at each Runge-Kutta substep where the geometry is updated a database containing the stencil coefficients was pre-computed and the coefficients are updated utilizing an interpolation procedure. For all validation cases shown here, this strategy was successfully applied. No negative effect on the stability and robustness of the numerical scheme was noted while large savings in computational expense were obtained.
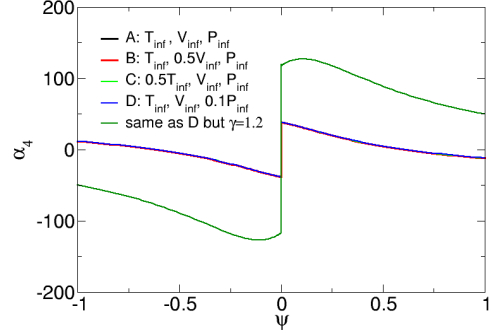

Figure 3: Dependence of optimal coefficient on flow conditions and flow properties. T_{inf} and p_{inf} correspond to sea level conditions and $v_{inf} = 100$m/s.

C. Moving Immersed Boundary Methods

An important aspect for simulating moving boundary and fluid-structure interaction problems is the evaluation of geometry information at each substep the immersed boundary is updated. When dealing with thin geometries a Langrangian type geometry representation is preferred over an Eulerian approach, such as utilizing level set functions. For thin geometries with the Eulerian approach, the volume grid spacings in the vicinity of the geometry need to be equivalent to the thickness of the geometry in order to ensure accurate geometry queries. In the current approach the underlying geometry is represented by the discrete surface triangulation. An x-ray tracing algorithm is used to determine grid line intersection points as visualized in Figure 4. X-rays in x- and y-directions are shown as blue and red arrows, and the intersection points with the surface triangulation are marked as pierce points (green filled circles). In order to reduce the computational cost of generating x-ray maps a boundary volume hierarchy algorithm is utilized. Efficient x-ray/surface-triangulation intersection tests are employed from the Embree Intel library.[27] A more detailed description of the methods employed for moving boundary problems is provided in Barad *et al.*[28]
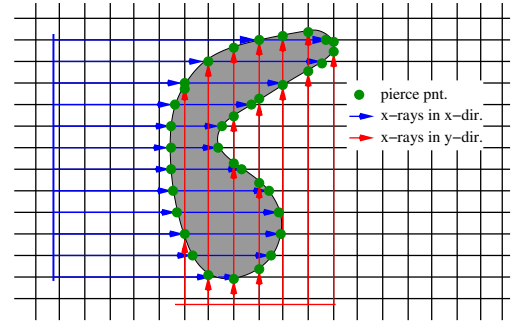

Figure 4: Illustration of x-ray tracing algorithm used to determine grid line intersection points.

1. Thin Geometries

Once the pierce points or grid line intersection points are identified it is straightforward to determine if grid points are located inside or outside of the body by simply accounting for the number of intersections of the x-ray until reaching the particular grid point. The boundary distance of the irregular grid point to the surface triangulation is obtained by a line-triangle intersection algorithm. For all results shown here, the normal vector at the grid line intersection point is taken from the intersected triangle. Hence, the quality of the surface triangulation has a strong effect on the accuracy of geometry queries and low surface quality may cause unwanted speckling in surface quantities, such as pressure and skin friction distributions.

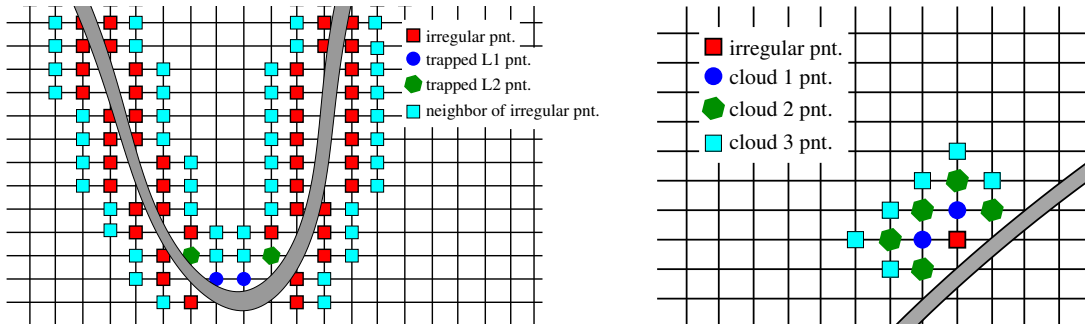American Institute of Aeronautics and Astronautics

Figure 5: (a) Irregular grid layout within a narrow gap and (b) illustration of cloud selection procedure.

Figure 5a displays the layout of irregular grid points in the vicinity of the geometry. The first layer of irregular grid points directly borders the immersed boundary. Since a higher-order immersed boundary scheme is employed additional layers of irregular grid points are marked and at all irregular grid points a special treatment of the finite difference stencils is necessary. For complicated geometries trapped points are frequently encountered. Trapped points occur in gaps that are smaller than the stencil size of the irregular finite difference stencil. In the present paper, trapped points are treated in a way where a lower-order finite difference stencil with a reduced stencil size is utilized. Flux boundary conditions are required at grid line intersection points to apply the irregular finite difference stencil as introduced in Section B. The boundary conditions are obtained by utilizing point clouds in the vicinity of the of the grid line intersection point. For moving boundary problems, the point cloud selection process can become computationally very time-consuming. Especially for thin geometries, where the geometry is represented by a surface triangulation since a minimum distance function approach is not feasible, intersection tests are necessary to determine if a grid point resides on the same side of the geometry. To speed-up the point cloud selection process, base clouds are utilized that can be added to successively grow the point clouds. This graph-walking approach minimizes the number of intersection tests that need to be applied in order to determine if the grid point resides on the same side of the immersed boundary as the irregular grid point, which is the owner of the point cloud. A base cloud contains its direct neighbors $(i \pm 1, j \pm 1, k \pm 1)$, where only a single permutation of $\pm$ in each coordinate direction is permitted. At irregular grid points, direct neighbors which are inside the geometry or lie on the other side of a thin geometry are excluded from the point cloud. The process is visualized in Figure 5b. A point cloud is grown around an irregular grid point by successively adding base clouds in three stages. Additional stages are added depending on the required size of the point cloud. Once the point cloud is obtained a rank based stencil selection procedure is used to down-select grid points which will be included in weighted least-squares interpolation for pressure and temperature boundary conditions at grid line intersection points. In the presence of flow discontinuities, such as shocked flows, a shock sensor is employed to reduce the order of the stencil (both for boundary conditions and the right-hand-side).

2. Freshly Cleared Cells

In the context of solving the governing equations with moving boundaries, the method presented herein can be viewed as an Eulerian-Lagrangian formulation. This means that the governing equations are solved in an Eulerian fashion on a stationary grid, and the moving boundaries are tracked in a Lagrangian manner. The treatment of the governing equation in Eulerian form on stationary, non-deforming grids can provide some advantages for the computation of moving boundary problems over more conventional Lagrangian type methods. In Lagrangian type approaches, one has to deal with moving/deforming grids and discretized equations, which produce additional time derivatives of grid-related quantities such as the changing cell volume. Moreover, for Lagrangian methods, large boundary deformations or movements lead to major complications in the local adjustment of the grid in the vicinity of the boundary.

When considering immersed boundary methods for the moving boundary problem, a further distinction must be made based on how the boundary is tracked as well as how its effect is represented in the underlying Eulerian governing equations. In the pioneering work of Peskin,[3] the boundary was tracked as a distinct and sharp Lagrangian entity, while in the governing equations it was treated as diffuse; thus, spreading its effect over several grid points in the vicinity of the boundary. In so-called Eulerian meth-

American Institute of Aeronautics and Astronautics

ods such as Volume-of-Fluid (Anderson *et al.*[29]), the diffuse nature of tracking the boundary is retained for representing its effect on the flow field. In the present approach, the immersed boundary is treated as a sharp entity for tracking as well as for the representation within the governing equations. The advantage of these methods with respect to the sharp representation of the boundary, implicitly causes an additional complication when dealing with moving boundaries. Figure 6 shows that as the immersed boundary moves across the fixed Cartesian grid, so called "freshly-cleared" cells (FCCs) may appear.

For fluid-solid interfaces, the FCCs are cells that are inside the fluid at t_{n+1} but they were inside the solid domain at the previous time-step t_n. In effect, the spatial discontinuity leads to a temporal discontinuity. It is clear that straightforward temporal discretization of the governing equations for these cells is not possible, because these cells do not have a valid time history. Udaykumar *et al.*[30] handled this issue by merging the FCCs with adjacent fluid cells for the first time step after the FCC appears in the fluid domain. In Udaykumar *et al.*,[31] a slightly different approach was chosen to address this issue. For one time step the velocity in the FCC was determined by applying an interpolation procedure considering the neighboring cells. A similar strategy is pursued in the present work, the state vector of the freshly cleared cell is initialized from a point cloud surrounding the FCC. For the explicit time-integration scheme, other FCCs are excluded from the point clouds so that no FCC depends on another FCC. For the implicit scheme, FCCs can generally be included in the interpolation procedure because they are iteratively updated at each nonlinear iteration. If the number of nonlinear iterations is small, this procedure may, however, not be satisfactory.
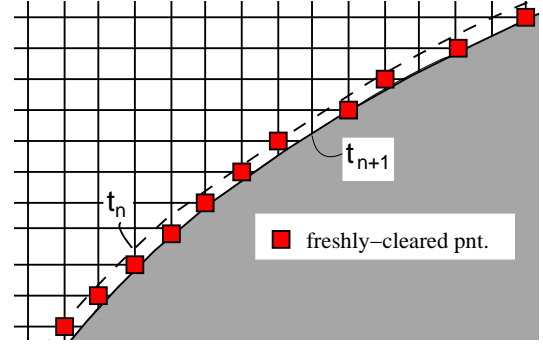

Figure 6: Emergence of freshly cleared cells during geometry update from time t_n to time t_{n+1}.
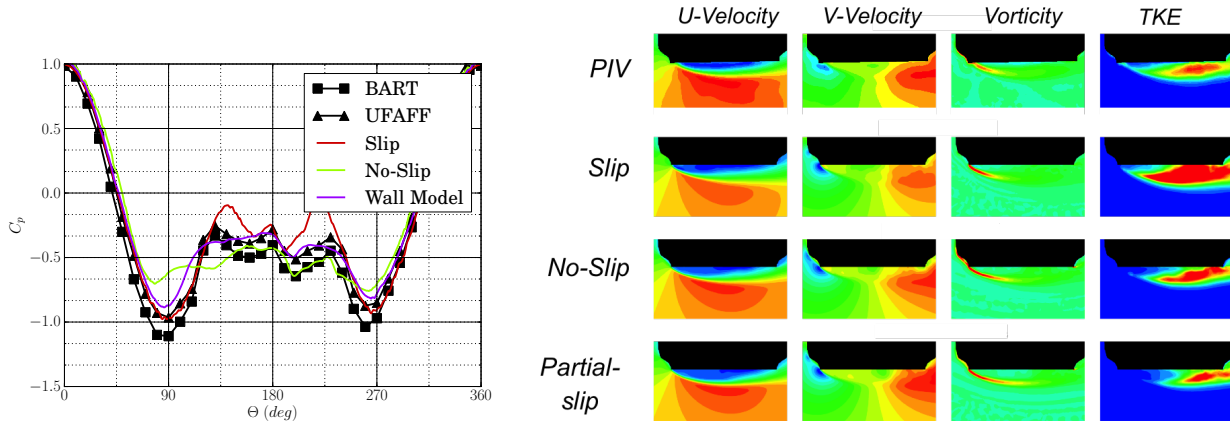
D. Viscous Wall Treatment at High-Reynolds Numbers

Figure 7: Comparison of pressure distribution around star board wheel of the landing gear used in the AIAA BANC-III workshop considering slip wall, no-slip wall, and wall modeling results. The data is compared to two experimental data sets (BART and UFAFF).

One of the key challenges for immersed boundary methods is the treatment of viscous walls at high Reynolds numbers. For high-Reynolds number viscous flow simulations, current immersed boundary approaches are inefficient in resolving the viscous boundary layers since the Cartesian mesh layout generally does not allow the use of (wall-normal) high aspect ratio cells in the vicinity of the wall. This issue has been addressed in several research investigations mainly by employing wall-functions near the immersed boundary.[32] In the present paper, a velocity boundary condition based on a wall model (the wall velocity deficit model) is used to mimic the effect of a viscous boundary layer. This approach was first used to simulate the flow around the partially-dressed cavity-closed nose landing gear for the BANC-III AIAA workshop. One of

American Institute of Aeronautics and Astronautics

the major drawbacks of this approach is that it depends on free parameters which need to be adjusted to the particular flow conditions, which requires experienced users.

Figure 7a shows the effect of the wall velocity boundary condition on the pressure distribution around the star board wheel of the landing gear used in the AIAA BANC-III workshop. The pressure distribution for no-slip wall and slip wall boundary conditions do not match the experimental data (BART and UFAFF) very well due to predicting too early separation for no-slip and failing to predict any separation at all for slip. A fair match between the numerical simulation and the experimental data can be obtained for the wall model. Figure 7b shows a comparison between PIV measurements and CFD data in a cut-plane through the star board wheel. A good match between the experimental and CFD results was obtained. This approach and more general approaches on how to treat viscous walls in high Reynolds number flows with immersed boundary methods are part of ongoing research.
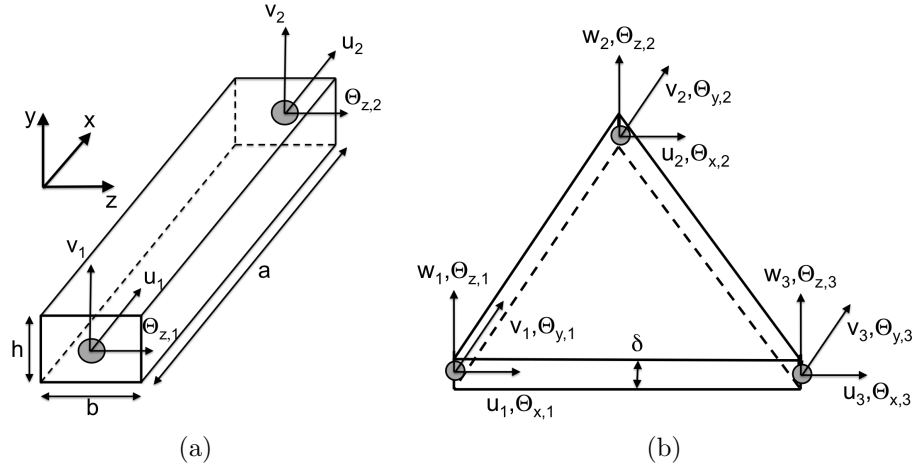
III.   Numerical Methods for CSD

(a)   (b)

Figure 8: (a) Bernoulli-Euler beam element for 2D calculations with a height h, a width b, and a length a. Each node contains three degrees of freedom, i.e., the vertical displacement, horizontal displacement, and the rotation Θ_z. (b) Triangular shell elements for 3D calculations with thickness δ and a cross-sectional area A. Each node contains five degrees of freedom, i.e., the in-plane deformations u and v, the out-of-plane displacement w, rotations Θ_x and Θ_y, and the drilling DOF Θ_z. Additional nodes are inserted at edge midpoints to compute the higher-order stiffness matrix, $\boldsymbol{K}_h$.

In order to model the deformation of solid bodies, a linear structural finite-element method (FEM) solver was implemented in the LAVA framework. Two basic element types are considered in this work. For 2D calculations, a Bernoulli-Euler beam element can be utilized that contains three degrees of freedom at each of the two nodes allowing for end displacements (u_1,v_1,u_2,v_2) and end rotations $(\Theta_{z,1},\ \Theta_{z,2})$. The beam element accounts for bending deformations (v and Θ_z) and axial deformation (u). For the stiffness and mass matrices of the 2D beam element the reader is referred to Petyt.[33]

For 3D calculations, triangular flat shell finite elements that incorporate membrane and bending components of displacement are used. Each node is modeled with three displacement and two rotational degrees of freedom. An additional drilling degree of freedom was added to avoid a singular stiffness matrix. With these triangular elements, arbitrary shaped shell structures can be modeled. The membrane and bending strains are treated decoupled from each other, which is a reasonable assumption in the small displacement model of flat shell elements. In this work, a flat shell element is obtained as a combination of a plane stress element and a plate bending element. The plate bending element accounts for two rotation parameters Θ_x and Θ_y and normal displacement w. The strain relations for the bending element are

$$\boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix} = -z_n \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \begin{bmatrix} \Theta_x \\ \Theta_y \end{bmatrix} = -z_n \boldsymbol{L}\boldsymbol{\Theta} \quad \text{and} \quad \boldsymbol{\gamma} = \begin{bmatrix} \gamma_{xz} \\ \gamma_{yy} \end{bmatrix} = \begin{bmatrix} \frac{\partial w}{\partial x} \\ \frac{\partial w}{\partial y} \end{bmatrix} - \begin{bmatrix} \Theta_x \\ \Theta_y \end{bmatrix} = \boldsymbol{\nabla}w - \boldsymbol{\Theta}, \quad (1)$$

American Institute of Aeronautics and Astronautics

where ϵ is the strain tensor, z_n is the coordinate normal to mid-plane of shell element with a thickness δ and cross-section area A. The constitutive relation for an isotropic, linear elastic material is

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix} = \boldsymbol{E}\boldsymbol{\epsilon} \tag{2}$$

where $\boldsymbol{\sigma}$ is the stress tensor, E is Young's modulus, and ν is the Poisson ratio. For thin plates, out-of-plane shear deformations are disregarded, thus, it immediately follows that $\boldsymbol{\gamma} = 0 = \nabla w - \boldsymbol{\Theta}$. Finally, the strong formulation of the equilibrium equation for the thin plate can be derived as $\nabla^T \boldsymbol{L}^T \boldsymbol{D} \boldsymbol{L} \nabla w - q = 0$ with $\boldsymbol{D} = \int_{z_n=-\delta/2}^{\delta/2} z_n^2 \boldsymbol{E} dz_n$. The equilibrium equation for the thin plate was obtained by integrating the equilibrium equation over the thickness of the plate. In order to obtain the FEM equations the governing equations are written in weak form by applying the principle of virtual work where the internal virtual work is given by

$$\delta\Pi_{int} = \int_\Omega (\delta\boldsymbol{\epsilon})^T \boldsymbol{\sigma} d\Omega = \int_\Omega \delta w \left(\boldsymbol{L}\nabla\right)^T \boldsymbol{D} \left(\boldsymbol{L}\nabla\right) w d\Omega \tag{3}$$

and the external virtual work is given by

$$\delta\Pi_{ext} = \underbrace{\int_\Omega \delta w q d\Omega}_{\text{distributed load}} + \underbrace{\sum_i \delta w_i F_i}_{\text{nodal load}} + \underbrace{\int_\Gamma \delta w_s t d\Gamma}_{\text{boundary load}} , \tag{4}$$

where the first term refers to a distributed load, q, the second term represents nodal load, F_i, and last term accounts for line boundary loads, t. The bending stiffness matrix is obtained by evaluating the following integral

$$\boldsymbol{K} = \int_\Omega \boldsymbol{B}^T \boldsymbol{D} \boldsymbol{B} d\Omega \quad \text{with} \quad \boldsymbol{B} = \left(\boldsymbol{L}\nabla\right)\boldsymbol{N}. \tag{5}$$

The triangular elements are formulated in area coordinates to ensure that the normal slope varies linearly along an edge. This formulation guarantees continuity of the lateral displacement and both its first derivatives between the triangular elements.

The equations describing the in-plane components of the shell element are described next. The kinematic relations for in-plane deformations and constitutive relations are

$$\boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \boldsymbol{L}\boldsymbol{u} \quad \text{and} \quad \begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix} = \boldsymbol{E}\boldsymbol{\epsilon}. \tag{6}$$

The equilibrium equation in strong form is $\boldsymbol{L}^T \boldsymbol{E} \boldsymbol{L} = 0$. The internal energy for the in-plane components becomes

$$\delta\Pi_{int} = \int_\Omega (\delta\boldsymbol{\epsilon})^T \boldsymbol{\sigma} d\Omega = \int_\Omega \delta\boldsymbol{u}^T \left(\boldsymbol{L}\right)^T \boldsymbol{E} \left(\boldsymbol{L}\right) \boldsymbol{u} d\Omega. \tag{7}$$

The in-plane deformation stiffness can be written as

$$\boldsymbol{K}_b = \int_\Omega \boldsymbol{B}^T \boldsymbol{E} \boldsymbol{B} d\Omega \quad \text{with} \quad \boldsymbol{B} = \boldsymbol{L}\boldsymbol{N}, \tag{8}$$

where the subscript b refers to the basic lower-order stiffness matrix. A higher-order stiffness matrix, $\boldsymbol{K}_h$, is added in order to introduce the so-called drilling degree of freedom normal to the element plane.

In the combined element exposed to membrane and bending actions, the displacements induced by in-plane forces do not affect the bending deformations. Note the basic formulation of the flat shell element does consider three nodal displacements, u, v, and w, and two rotation parameters, Θ_x and Θ_y but not the normal rotation Θ_z. Hence, as illustrated below zero stiffness is used for the normal rotation,

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{K}_i & 0 & 0 \\ 0 & \boldsymbol{K}_b & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{9}$$

American Institute of Aeronautics and Astronautics

where $\boldsymbol{K}_i$ and $\boldsymbol{K_b}$ refer to previously introduced stiffness matrices associated with in-plane (i) and bending deformations (b). The drilling degree of freedom is obtained by considering the vertex rotation perpendicular to the plane of the plate. Introducing the drilling degree of freedom is necessary to avoid the global stiffness matrix from becoming singular. Several strategies are available to deal with this issue. In the present formulation, higher-order displacement nodes are linked to the corner rotations normal to the plane of the element. The membrane element stiffness matrix is obtained by a superposition of basic and higher-order stiffness, $\boldsymbol{K} = \boldsymbol{K}_b + \boldsymbol{K}_h$. More details about the membrane element with rotational degree of freedom can be found in Bergan and Felippa.[34] The mass matrices are obtained by simply considering the expressions for the kinetic energy.

The global system is dynamically solved for in terms of the displacement vector $\boldsymbol{u} = [u, v, w, \Theta_x, \Theta_y, \Theta_z]$ in the well-known form

$$\boldsymbol{M}\frac{d^2\boldsymbol{u}}{dt^2} + \boldsymbol{C}\frac{d\boldsymbol{u}}{dt} + \boldsymbol{K}\boldsymbol{u} = \boldsymbol{f}, \tag{10}$$

where $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{C}$ is the damping matrix, $\boldsymbol{K}$ is the stiffness matrix, and $\boldsymbol{f}$ is the load vector. External forces and moments can be supplied through the user input file and the pressure and viscous forces/moments are transferred from the CFD solver. For time-integration, full mass matrices are considered and the system of equations are solved in parallel employing Krylov subspace methods. In the present work, the generalized minimal residual method (GMRES) was employed to iteratively solve the linear system at each time step. The well-known Newmark method is used to integrate the system of equation (10) in time. The Newmark method can be written as

$$[a_1\boldsymbol{M} + a_2\boldsymbol{C} + \boldsymbol{K}]\,\boldsymbol{u}_{n+1} = \boldsymbol{f}_{n+1} + [a_1\boldsymbol{M} + a_2\boldsymbol{C}]\,\boldsymbol{u}_n + [a_3\boldsymbol{M} - a_4\boldsymbol{C}]\,\dot{\boldsymbol{u}}_n + [a_5\boldsymbol{M} - a_6\boldsymbol{C}]\,\ddot{\boldsymbol{u}}_n \tag{11}$$

$$\ddot{\boldsymbol{u}}_{n+1} = a_1\,[\boldsymbol{u}_{n+1} - \boldsymbol{u}_n] - a_3\dot{\boldsymbol{u}}_n - a_5\ddot{\boldsymbol{u}}_n \tag{12}$$

$$\dot{\boldsymbol{u}}_{n+1} = a_2\,[\boldsymbol{u}_{n+1} - \boldsymbol{u}_n] + a_4\dot{\boldsymbol{u}}_n + a_6\ddot{\boldsymbol{u}}_n, \tag{13}$$

where $a_1 = 1/\beta\,(\Delta t)^2$, $a_2 = \psi/\beta\Delta t$, $a_4 = (1 - \psi/\beta)$, $a_5 = [(1/2\beta) - 1]$, $a_6 = [1 - (\psi/2\beta)]\,\Delta t$, $\psi \geq 0.5$, and $\beta \geq 0.25\,(\psi + 0.5)^2$. Artificial damping is introduced unless $\psi = 0.5$. For the current simulations, the parameter ψ was chosen so that second-order accuracy in time is achieved.

The computational structural dynamics (CSD) solver was verified for some well-known test problems as shown in Figures 9-13. For the beam bending vibration test the analytical solutions for the first and second bending modes are compared to the numerical solutions. The vibrational modes are determined by solving the eigenvalue problem in the form:

$$\omega_m^2\underline{\boldsymbol{M}}\boldsymbol{v} = \underline{\boldsymbol{K}}\boldsymbol{v}, \tag{14}$$

where ω_m is the mth vibration frequency and $\boldsymbol{v}$ is the corresponding eigenmode. The results of the beam axial and bending vibration tests are shown in Tables 1 and 2. The eigen-frequency of the first and second modes, $\omega_{1,2}$, are non-dimensionalized with the solid density, ρ, cross-sectional area, A, length of the beam, L, Young's modulus, E, and the bending stiffness, $I_z = 1/12bh^3$. The accuracy of the FEM solver for static deformations was validated in Table 3 for a plate clamped on all for edges subjected to a central load. In-plane and out-of-plane vibration modes for the triangular shell element type were verified for a triangular plate supported at a single edge. For these simple test cases the analytical in-plane and bending vibration frequencies are available for the first five eigenmodes. Good agreement for the frequencies of the first five in-plane and out-of-plane vibration modes was obtained.
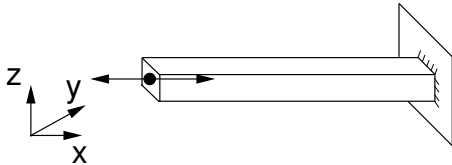

Figure 9: Setup for beam axial vibration test case.

N_e	Error Mode 1	Error Mode 2
2	4.06×10^{-2}	9.17×10^{-1}
5	6.46×10^{-3}	1.75×10^{-1}
10	1.61×10^{-3}	4.36×10^{-2}

Table 1: Error convergence study for the first two axial vibration modes with $\omega_1\sqrt{\rho L^2/E} = 1.571$ and $\omega_1\sqrt{\rho L^2/E} = 4.712$
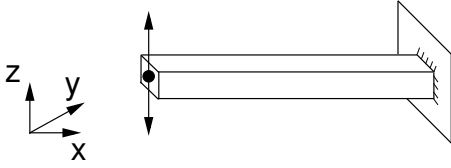
American Institute of Aeronautics and Astronautics

Figure 10: Setup for beam bending vibration test case.

N_e	Error Mode 1	Error Mode 2
2	1.92×10^{-3}	1.87×10^{-1}
5	2.74×10^{-4}	1.14×10^{-2}
10	2.9×10^{-5}	1.13×10^{-3}

Table 2: Error convergence study for the first two bending vibration modes with $\omega_1 \sqrt{\rho A L^4 / E I_z} = 3.516$ and $\omega_2 \sqrt{\rho A L^4 / E I_z} = 22.035$
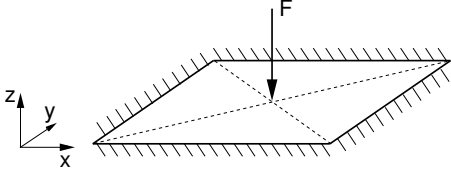

Figure 11: Setup for plate bending test case.

N_e	w_{max} [mm], $\mathcal{O}(\Delta)$	w_{max} [mm], $\mathcal{O}(\Delta^2)$
21×21	6.614	5.700
31×31	6.042	5.593
61×61	5.647	5.517
99×99	5.552	5.500

Table 3: Convergence study for plate bending test case with $w_{max,analytic} = 5.474$mm.
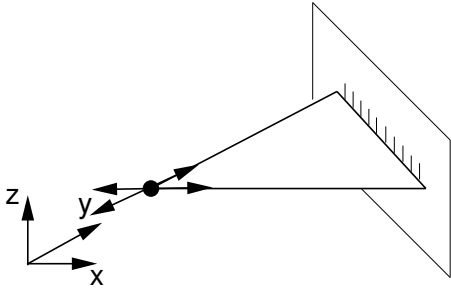
Figure 12: Setup for in-plane vibration test case for triangle.

Mode	Analytic	FEM
1	4.973	4.952
2	26.391	27.799
3	31.944	32.051
4	62.066	71.004
5	95.832	93.452

Table 4: First five in-plane vibration modes for triangle.
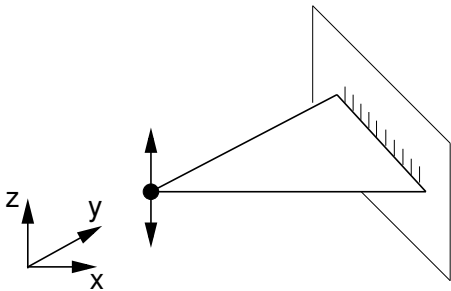
Figure 13: Setup for bending vibration test case for triangle.

Mode	Analytic	5×5-Mesh	10×10-Mesh
1	37.5	39.283	38.047
2	161	173.55	163.91
3	243	223.23	237.48
4	392	435.25	404.12
5	592	522.73	566.54

Table 5: First five bending vibration modes for triangle.

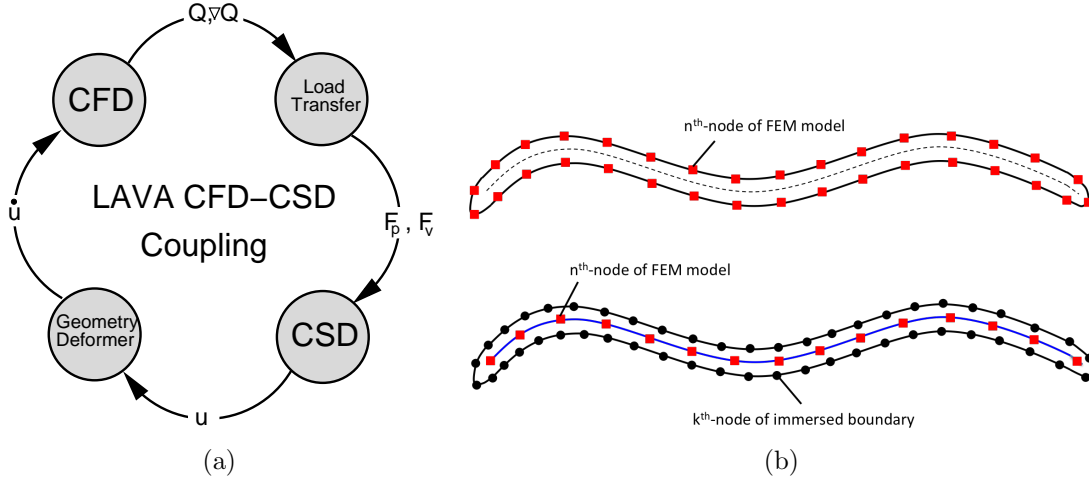American Institute of Aeronautics and Astronautics

Figure 14: (a) Coupling of Computational Fluid Dynamics (CFD) and Computational Structural Dynamics (CSD) solvers within the LAVA framework. (b) top: using the same geometry description for CFD and CSD, (b) bottom: using an auxiliary model for CSD that differs from the CFD geometry.

IV.   Methods for CFD-CSD Coupling

The computational fluid dynamics (CFD) and computational structural dynamics (CSD) solvers are coupled in a loose fashion, while communicating through a common interface within the LAVA framework as illustrated in Figure 14a. Depending on the relevant temporal scales the CFD and CSD solvers can march with different timesteps and data will be exchanged after the CFD/CSD sub-cycling steps are completed. For all results presented in this paper both the CFD and CSD solvers update the geometry and fluid forces at the end of each time-step for the implicit CFD solver and at common sub-steps for the explicit Runge-Kutta time-integrator. For the classical fourth-order accurate Runge-Kutta time-integration scheme the update occurs at two time levels, i.e., $t_{n+1/2}$, and t_{n+1}, where the geometry and fluid forces at t_n are known from the previous update. Hence, a single time-step with a fourth-order accurate explicit Runge-Kutta scheme with four sub-steps of the CFD solver requires two updates with $\Delta t_{CSD} = 0.5\Delta t_{CFD}$ in the structural solver. The CFD solver looses its fourth-order accuracy in time when it is coupled with the second-order accurate structural solver due to insufficient temporal accuracy of the displacement field. In the loosely coupled approach the CSD solution lags behind the CFD solution. Once the flow field has been updated to the next substep the loads are computed and transferred to the FEM solver. For the CFD solver all geometry queries are based on the discrete surface triangulation. As discussed previously, the boundary conditions are enforced at grid line intersection points with the discrete triangles. Hence, the most natural approach is to use the same triangulation in the CSD solver as well. However, for some applications much less complex structural models are sufficient to account for the relevant structural behavior. For many airplane applications for example, the structural response of the entire wing is modeled by simply using connected 3D beam elements. Figure14b displays the two models currently available within LAVA to account for the deformation of the solid body. In the case when an auxiliary geometry is used for the FEM model in the CSD solver, it is ensured that a conservative loads transfer is obtained.

$$\sum_{n=1}^{N} A_n^{(IB)} \boldsymbol{F}_n^{(IB)} = \sum_{k=1}^{K} A_k^{(FEM)} \boldsymbol{F}_k^{(FEM)}, \tag{15}$$

where $A_n^{(ib)}$ and $\boldsymbol{F}_n^{(ib)}$ is the nth element area and force vector of the immersed boundary and $A_k^{(FEM)}$ and $\boldsymbol{F}_k^{(FEM)}$ is the kth element area and force vector of the auxiliary geometry used for the FEM model. A deformer is used in order to map the displacements back to the immersed geometry. In the current paper, the displacements of the reduced-order structural model are simply mapped back by employing an interpolation

American Institute of Aeronautics and Astronautics

approach, such as

$$\boldsymbol{x}_k^{(IB)} = \sum_{n=1}^{N} c_n \boldsymbol{x}_n^{(FEM)}, \qquad (16)$$

where $\boldsymbol{x}_n$ are the nodal displacements of the FEM model of the auxiliary surface and $\boldsymbol{x}_k$ are the displacements of the immersed geometry. The N interpolation coefficients, c_n, are determined by employing a least-squares method. For a large number of nodes in the FEM model, N, an influence region can be specified (through the user input file) only considering the N_i-th closest nodes to the geometry node k. The velocity on the triangulation (or 2D line segments) is transferred to the CFD solver as an added body velocity for the boundary conditions.

V.   Prescribed Body Motion Problems

In the first set of test problems, the present method is applied to moving boundary problems with prescribed body motion. Experimental and other computational results are available for all chosen test problems.

A.   Transversely Oscillating Cylinder

In the first test problem the flow past a transversely oscillating cylinder is simulated. The transversely oscillating cylinder are compared to experimental data obtained by Koopmann[35] and computational results available from Meneghini and Bearman[36] and Udaykumar et al.[30] for a large range of amplitudes and frequencies. The objective of this investigation is to reproduce the phenomena of vortex shedding "lock-on" due to the transversely oscillating cylinder. This classical phenomena is observed in the wake of bluff bodies and refers to situations where the frequency of the vortex shedding in the wake synchronizes with or locks into the frequency of an imposed perturbation. The vortex-shedding lock-on for a transversely oscillating cylinder is studied for a Reynolds number of $Re = 200$ based on the diameter of the cylinder. Grid refinement was applied around and behind the immersed cylinder to better resolve the near wall region and the wake behind the cylinder. All length scales have been normalized by the diameter of the cylinder. For the simulations, a computational domain of $50D \times 50D$ was used. The cylinder was initially placed at $(x_0, y_0) = (10., 25.)$ relative to the lower left corner of the computational domain. The sinusoidal motion of the cylinder can be described by the location of its center, (x_c, y_c) which is given by $x_c(t) = x_0$ and $y_c(t) = y_0 + A\sin(2\pi f t)$. In this equation, the time, t, was non-dimensionalized with diameter of the cylinder and the free stream velocity, where A and f are the non-dimensional amplitude and frequency of the oscillation, respectively. For all calculations, the flow field was initialized with the flow field from a stationary cylinder calculation. The flow past the stationary cylinder at this low Reynolds number displays the classical Karman vortex shedding. For the non-oscillatory case, the shedding frequency in the wake was previously compared for different Reynolds numbers to various experimental and numerical studies (see Brehm et al.[16]). It was found that the frequencies, the unsteady drag coefficients and the unsteady lift coefficient compared very well with the results published in the literature. For a Reynolds number of $Re = 200$ a Strouhal number or non-dimensional vortex shedding frequency of $f_0 = 0.198$ was obtained.

Koopmann[35] experimentally determined the vortex shedding lock-on region for $Re = 200$. The results are shown in Figure 15a as a dashed line connecting the filled squares. The vortex shedding lock-on region is illustrated in Figure 15a with the difference $\Delta f = f_f - f_0$ on the abscissa (between cylinder oscillation frequency, f_f, and the natural shedding frequency, f_0, for $Re = 200$) and the cylinder oscillation amplitude, A, on the ordinate. Only for oscillation amplitudes larger than $5\%D$ could lock-on be observed. The circles in this figure show the simulation results conducted within and outside of the lock-on region. Filled circles indicate simulations where lock-on occurred and open circles indicate results where no lock-on was observed. The observations of vortex shedding lock-on concur with the results by Koopmann.[35] In reference to Udaykumar et al.,[30] the variation of the vortex shedding frequency with cylinder oscillation frequency for an oscillation amplitude of the cylinder of $A = 0.1$ is shown in Figure 15b. Similar to the illustration in Udaykumar et al.[30] the dashed line indicates lock on and the solid line represents the natural vortex shedding frequency of the cylinder flow which is equal to $f_0 = 0.198$ for a Reynolds number of $Re = 200$. For an amplitude of $10\%D$, lock-on can be observed for $\Delta f/f_0 = \pm 5\%$ and $\Delta f/f_0 = \pm 10\%$. Similar to the illustration provided in Koopmann,[35] Lissajous figures (in Figure 16) are shown in order to determine if the velocity signal in the wake lock on to the motion of the oscillating cylinder. When the vortex street locks-on
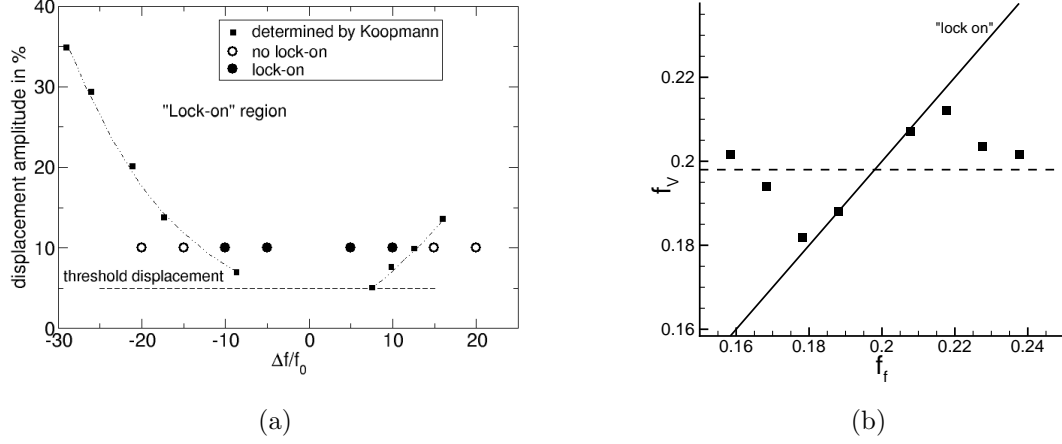
Figure 15: *a)* Illustration of the "Lock-on" region (dashed line) where the cylinder motion controls the vortex shedding behind the cylinder (similar to Koopmann,[35] squares) and *b)* variation of vortex shedding frequency, f_V, with forcing frequency, f_f, for $A = 10\%D$ (similar to Udaykumar *et al.*[30]).

to the motion of the oscillating cylinder, the Lissajous figure appear as a stationary loop. The Lissajous figures nicely display the lock-on, which occurs for $\Delta f/f_0 = \pm 5\%$ and $\Delta f/f_0 = \pm 10\%$. No stationary loops were observed for $\Delta f/f_0 = \pm 15\%$ and $\Delta f/f_0 = \pm 20\%$.

(a) $\Delta f/f_0 = -20\%$ (b) $\Delta f/f_0 = -15\%$ (c) $\Delta f/f_0 = -10\%$ (d) $\Delta f/f_0 = -5\%$

(a) $\Delta f/f_0 = +5\%$ (b) $\Delta f/f_0 = +10\%$ (c) $\Delta f/f_0 = +15\%$ (d) $\Delta f/f_0 = +20\%$
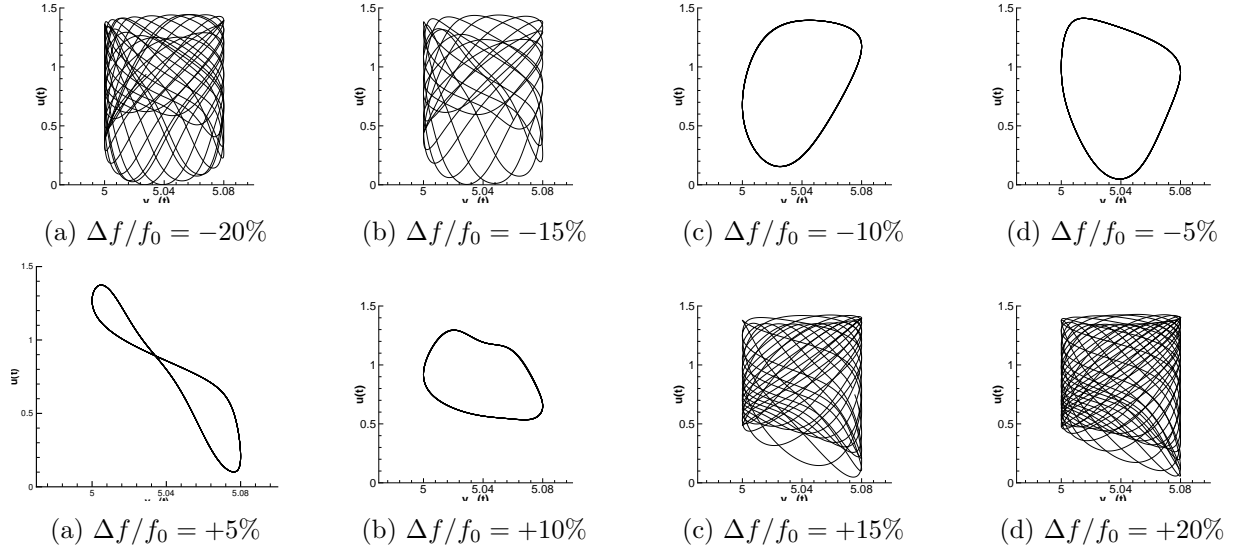
Figure 16: Lissajous figures with the displacement of the cylinder on the abscissa and the velocity signal in the wake of the oscillating cylinder on the ordinate (all for $A = 10\%$)

B. Pitching NACA0015 Airfoil

The wind tunnel experiment by Piziali[37] has been chosen as the second test case. In this paper, the 2D pitching NACA0015 airfoil results are used for comparison. Only the attached flow case was simulated here, due to the limited capability of the immersed boundary method for capturing high Reynolds number boundary layers, in particular the prediction of partial separation during the oscillation cycle. The simulations were conducted for a free-stream Mach number of 0.29 and a chord Reynolds number of 1.95×10^6. The temporal variation of the angle of attack is given by

$$\alpha(t) = \alpha_m + 4.2^\circ \sin\left(2kM_\infty + 1.5\pi\right), \tag{17}$$

American Institute of Aeronautics and Astronautics

where the freestream Mach number M_∞ was 0.29, the reduced frequency was set to $k = 0.1$ and the mean angle of attack is $\alpha_m = 4°$. To enable a smooth start-up of the simulation a phase shift of 1.5π was added leading to $d\alpha/dt = 0$ at $t = 0$.
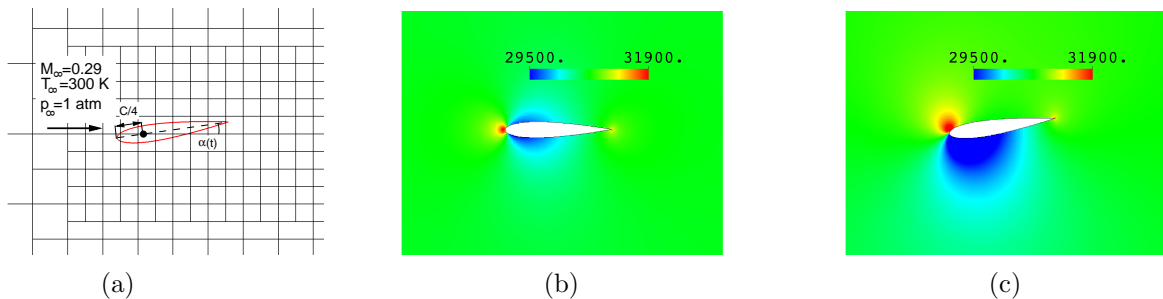

Figure 17: (a) Simulation setup for pitching NACA0015 airfoil calculations. Instantaneous pressure field at (b) upstroke and (c) down-stroke of oscillating motion.

Two grid resolutions were used for this simulation. Figure 17a shows the NACA0015 airfoil immersed into the Cartesian grid, where each Cartesian box contains 16×16 grid points. The pressure field during the up- and down-strokes are displayed in Figures 17b and 17c. A comparison of the lift and momentum coefficients versus angle of attack between experimental and CFD results is presented in Figures 18a and 18b. Note that in the experiment the measured force was determined from pressure data excluding the contributions from the viscous wall shear. The results were time-averaged over 20 cycles. For the CFD simulations, the periodic hysteresis behavior of the force coefficients established after only one period. The discrepancy in the drag coefficient is somewhat similar to what was observed in 2D simulations found in the literature, see for example Ko and McCroskey[38] and Nichols and Heikkinen.[39] Overall, fair agreement between the CFD results and the experiments is obtained.
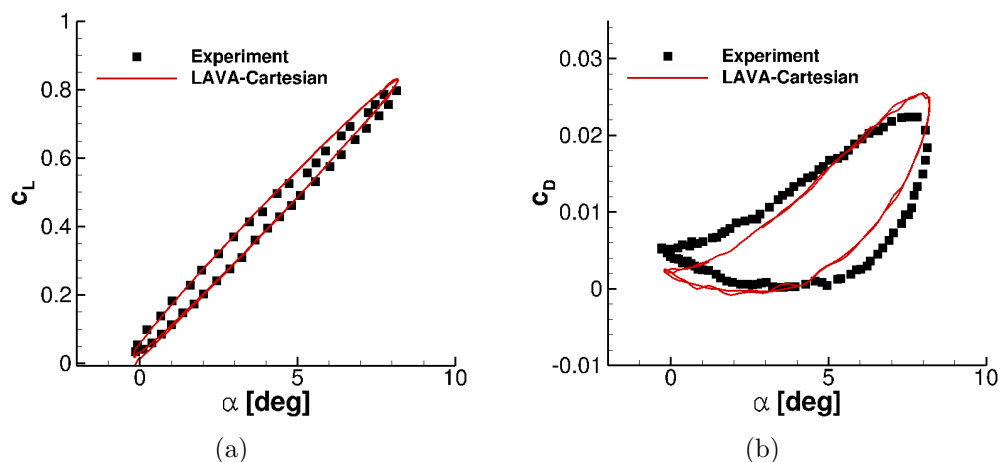

Figure 18: (a) Lift and (b) drag coefficients compared to experimental data from pitching NACA0015 airfoil calculations.

C. AIAA 2nd Aeroelastic Prediction Workshop: Pitching Wing

The pitching wing test case is based on experiments by Piatak and Cleckner[40] that were conducted in NASA Langley's Transonic Dynamics Tunnel (TDT). The experiment was carefully designed to control the motions of the wind tunnel model. An Oscillating Turntable (OTT) was designed for the purpose of oscillating, large semi-span models in pitch at frequencies up to 40 Hz to acquire high-quality unsteady pressure and loads data. This experimental dataset is also used for the AIAA Aeroelastic Prediction Workshop (AePW). The current validation of the numerical methods is confined to a single experimental data set, which in the

workshop is referred to as case 1b. The pitching wing was tested at a Mach number of $M = 0.7$, a mean angle-of-attack of $\alpha = 3°$, a pitching amplitude of $\Delta\alpha = 1°$, a oscillation frequency of $f = 10$Hz and a Reynolds number based on the chord length of 4.56×10^6.
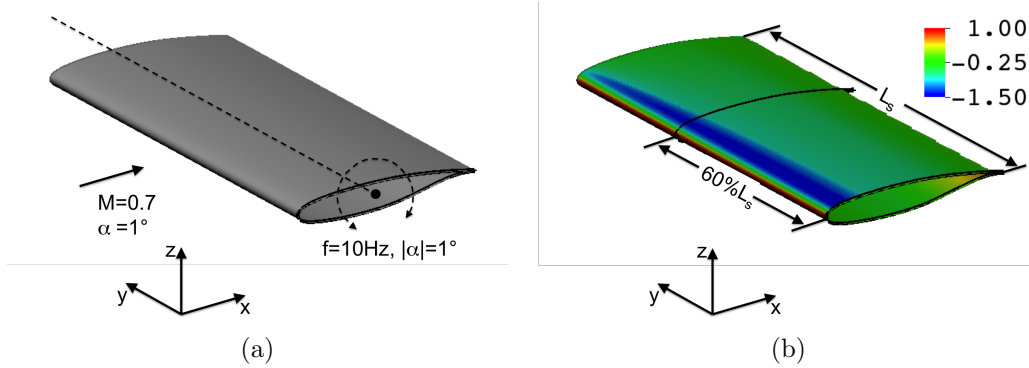

(a)   (b)

Figure 19: (a) Simulation setup for 3D pitching wing. (b) Snapshot of instantaneous pressure distribution on the surface of wing. The black solid line on the surface represents the 60% span location where the CFD and experimental data are compared to each other.

Figure 19a shows the 3D wing geometry used in the experiment and the current CFD simulations as well as a snapshot of the surface pressure distribution during pitch-up. The distribution of mean surface pressure at 60% wing span is compared to experimental data in Figure 18a. The immersed boundary method results are also compared to the body-fitted results utilizing the LAVA-curvilinear solver module. For the experimental data, upper and lower bounds are included to provide a deviation from the mean pressure distribution. The variation of the amplitude and phase along the chord of the wing for the forced oscillation frequency of $f = 10$Hz are shown in Figures 20b and 20c. The low amplitude value in the experimental dataset at $x/c \approx 0.2$ was assumed to be due to a malfunctioning pressure sensor at this location. Interestingly, the LAVA-curvilinear simulations display a similar but much smaller deviation of c_p in this region. While some smaller variations between the experimental and computational results are observed the LAVA results are within the scatter of the other computational results submitted to the workshop.
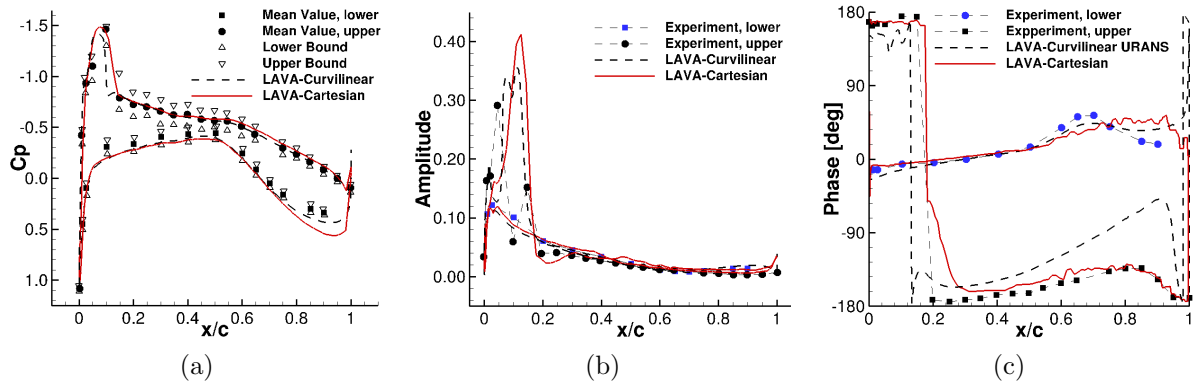

(a)   (b)   (c)

Figure 20: (a) Mean pressure distribution and, (b) amplitude and (c) phase for the forced oscillation frequency $f = 10$Hz. All data was extracted at 60% wing span.

VI.   Fluid-Structure Interaction Problems

Finally, some fluid structure interaction validation cases are presented. All cases were run with both explicit and implicit time-integration schemes of the CFD solver. When utilizing an explicit time-integration scheme a tight CFD-CSD coupling was obtained due to small CFD timestep based on the CFL restriction. For the implicit time-integration scheme, a sufficiently small timestep has to be chosen in order to obtain a good coupling between the CFD and CSD solvers. As customary for dual-time stepping approaches, sufficient sub-iteration convergence is required to eliminate unwanted accumulation of the linearization error.

A.   Deforming Panel
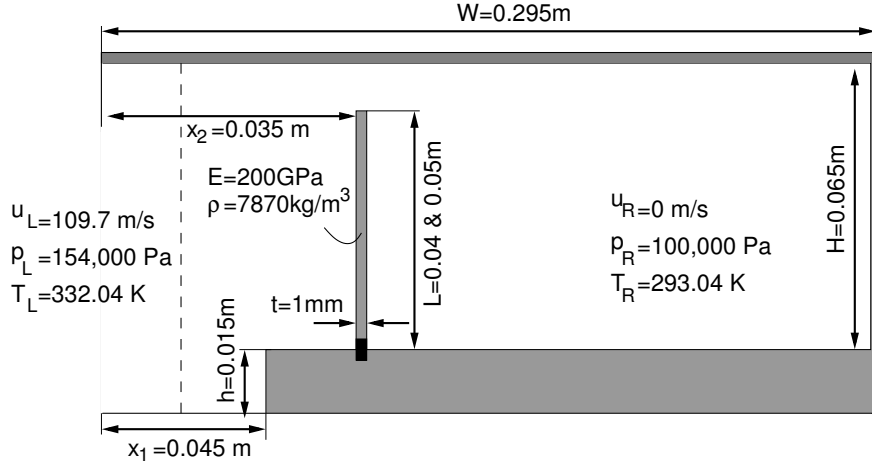
Figure 21: Simulation setup for deforming panel problem.

The first fully coupled fluid-structure-interaction test case involves a shock wave impacting on a deforming panel as first presented in Giordano et al.[41] The objective of this study is to numerically predict the deformation of the cantilever panel motion and the coupled fluid dynamics. The characteristics of the panel were chosen to obtain a strong interaction between the structure and the fluid. Some important static and dynamic parameters of the panel can be estimated considering classical beam theory, such as the maximum displacement $v_{\max} = 3pL^4/2Et^3$ for a uniform load p and the lowest circular eigenfrequency $\omega = \sqrt{Et^2/\rho L^4}$. The simulation setup and the dimensions of the panel are shown in Figure 21. A shock wave moves at Mach 1.2 into ambient air. An ideal gas was assumed in the current simulations and slip-wall boundary conditions were applied on all walls, while viscous effects were neglected. The panel was modeled as a single beam containing 100 elements. To account for the thickness of the panel the beam was surrounded by an outer shell displaced by $t/2$ from the beam similar to the schematic in Figure 14b.

The current CFD-CSD simulations are compared to experimental and numerical results provided by Giordano et al.[41] Figure 24 displays a comparison of Schlieren contours from experiments and numerical Schlieren contours by plotting the magnitude of the density gradient for the numerical simulation. The time of the first snapshot was estimated to closely match the key features of the flow field at $t = 0$s. The current simulation closely matches the main flow features that appear in both the experiment and computation provided by Giordano et al.[41] It must be pointed out that the strong starting vortex dissipates much more quickly in the experiment due to the presence of 3D effects and viscosity. Although a lower grid resolution was used in comparison to Giordano et al.[41] the blended sixth-order accurate WENO scheme[25] is able to sharply capture the flow features.

A more quantitative comparison of the results is provided in Figures 23a and 23b. The horizontal tip deflection is plotted versus time. For the 50mm panel, the horizontal tip deflection agrees well between the other numerical simulation while some discrepancy with the experiment can be observed. Giordano et al.[41] explained the disagreement between their experiment and their numerical simulation by pointing out that structural damping was not taken into account. This argument was quickly disregarded since damping will likely not play an important role during the first period. Hence, an alternate explanation was provided. It was noted that the base of the panel appears to deform slightly for the large panel with $L = 50$mm. In
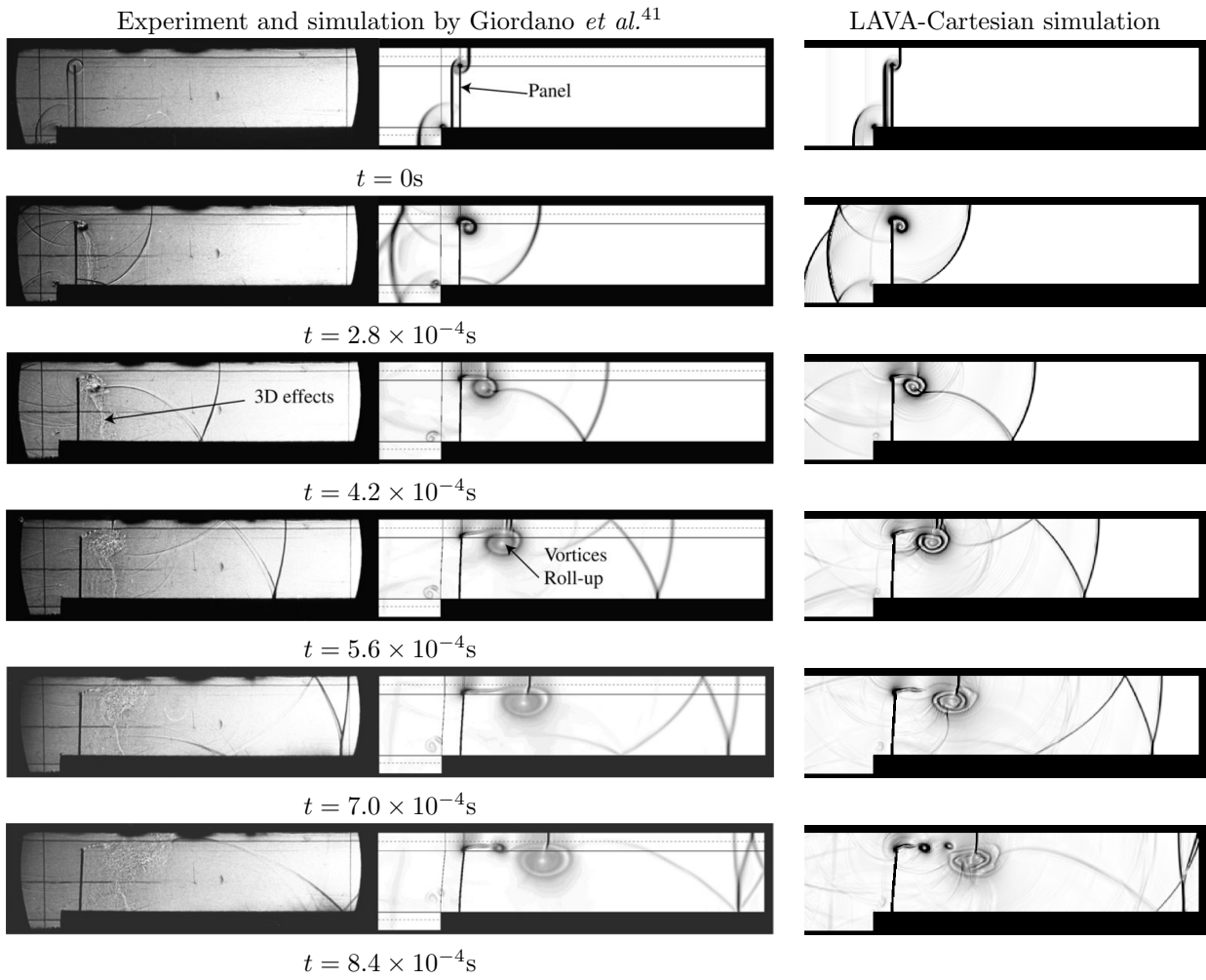
American Institute of Aeronautics and Astronautics

Experiment and simulation by Giordano et al.[41]   LAVA-Cartesian simulation

Panel

$t = 0$s

$t = 2.8 \times 10^{-4}$s

3D effects

$t = 4.2 \times 10^{-4}$s

Vortices
Roll-up

$t = 5.6 \times 10^{-4}$s

$t = 7.0 \times 10^{-4}$s

$t = 8.4 \times 10^{-4}$s

Figure 22: Comparison of instantaneous Schlieren contours for $L = 50$mm and $t = 1$mm.

(a)   (b)
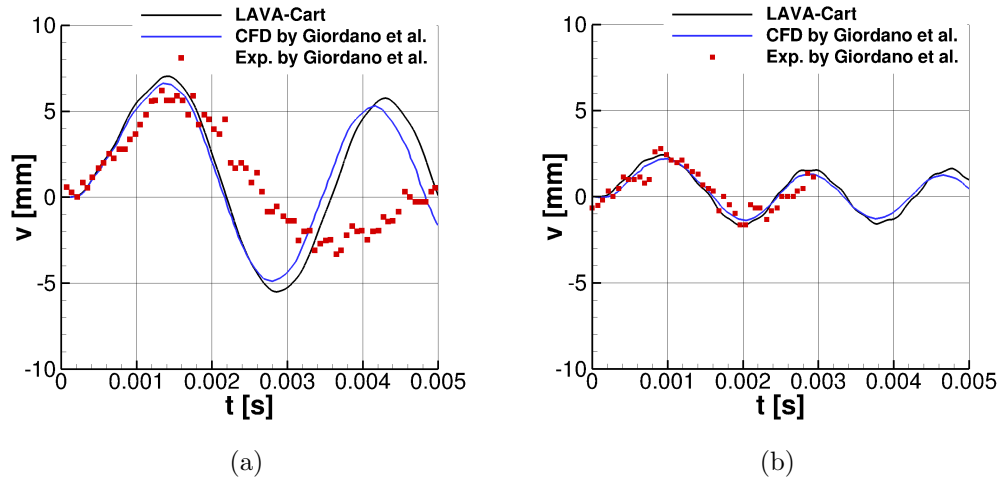
Figure 23: Comparison of horizontal tip deflection for (a) $L = 50$mm and (b) $L = 40$mm.

order to reduce this effect, a shorter panel was tested as well. The results for the shorter panel are presented

American Institute of Aeronautics and Astronautics

in Figure 23b. For the shorter panel, the experimental and computational results are much closer to each other.

B.   Flutter Analysis of Flat Plate

(a)

(b)
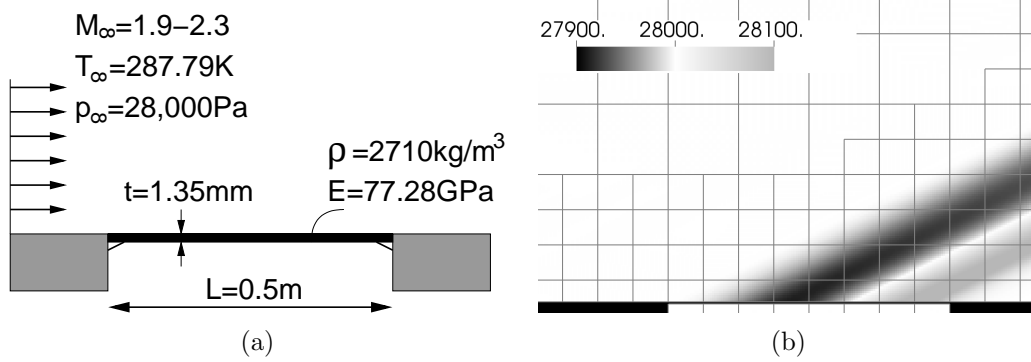
Figure 24: (a) Simulation setup for 2D flat plate flutter analysis. (b) Instantaneous snapshot of pressure for $M = 2.3$.

The next test problem is concerned with predicting the onset of flutter for a flat plate. The interaction of aerodynamic, inertial and elastic forces can cause a phenomenon known as flutter, which is caused by an aeroelastic instability. The simulation setup was chosen in accordance with Dowell[42] and Vito et al.[43] For this setup, the critical Mach number where the instability occurs is $M_{crit} = 2.0$. For larger Mach numbers a growth of the oscillations is expected. The Mach number was chosen in the range $1.9 \geq M \geq 2.3$, to determine if the current FSI approach is able to predict the onset of the aeroelastic instability. The pressure and density of the freestream was set to $p = 28\text{kPa}$ and $\rho_f = 0.339\text{kg/m}^3$. Slipwall boundary conditions were applied at all boundaries. The flat plate was discretized with 300 2D beam elements and all degrees of freedom were set to zero at both ends of the plate $(u_0, v_0, \Theta_0) = (u_N, v_N, \Theta_N) = 0$. Figure 24a provides an overview of the simulation setup. The elastic plate has a length of $L = 0.5\text{m}$, a thickness of $t = 0.00135\text{m}$, a Poisson ratio of $\nu = 0.33$, a density of $\rho_s = 2710\text{kg/m}^3$, and a Young's modulus of 77.28GPa. Initially, a uniform load of $\Delta p = 100\text{Pa}$ is applied over $0 < t \leq 0.01$ to the plate in order to trigger the aeroelatic instability.
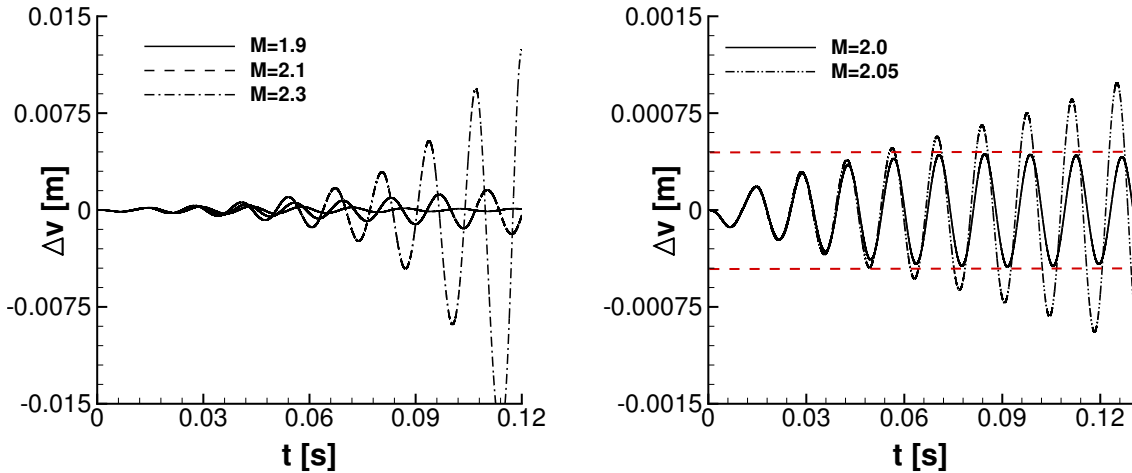

Figure 25: Vertical displacement at $x = L/2$ for different Mach numbers.

An instantaneous snapshot of grey-scale pressure contours are shown in Figure 24b. The computational mesh is outlined in the background while each box contains 16×16 grid points. Adaptive mesh refinement

was used to track expansion and compression waves through the computational domain. The vertical displacement of the panel was recorded at $x = L/2$ to determine if the configuration is stable or unstable. Figure 25 displays the vertical displacement for different Mach numbers. As predicted by linear stability theory in Dowel,[42] the system becomes unstable between $M = 2.0$ and 2.05. The current results are in good agreement with Dowel[42] and Vito *et al.*[43]

C.   AIAA 2nd Aeroelastic Prediction Workshop: Flutter Analysis of 3D Wing

(a)   (b)
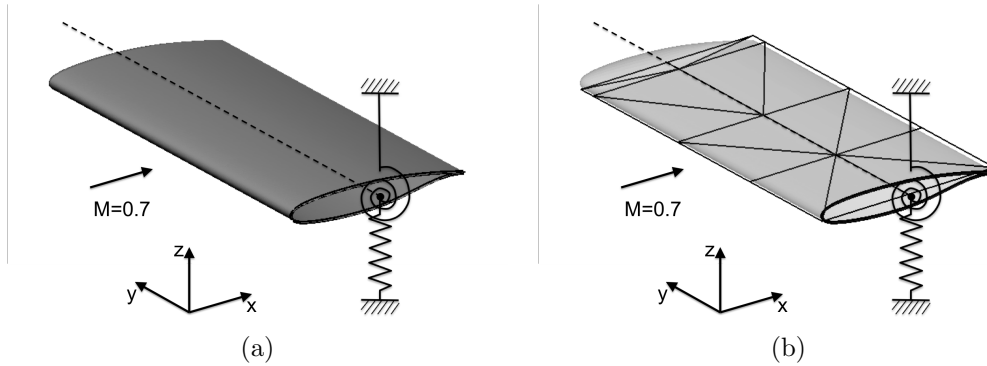
Figure 26: Simulation setups for flutter calculation: (a) two-degrees of freedom structural model and (b) triangulated centerplane representing structural characteristics of wing geometry.

The next FSI test problem was taken from the AePW workshop. For this test problem, both structural modeling approaches as shown in Figure 14b were tested. Figures 26a and 26b display the two structural models which were used for the wing. Since the two results are almost indistinguishable from each other only the results from the two-degree of freedom structural model of the wing are shown here. In the two-degrees of freedom structural model, the structural behavior of the wing is modeled by fixing the stiff wing to a displacement spring and a torsional spring. Following the guidelines of the workshop, in this model it is assumed that the wing can pitch about the spanwise axis centered at 50% chord at the root of the wing. Furthermore, a plunging motion in the vertical direction (here, z-coordinate axis) is also considered while the surface of the wing was modeled as a solid body. The deformation of the wing can then be modeled as two simple transformations of the original surface triangulation. First, the surface triangulation is rotated around the spanwise axis to account for the pitching motion and then the surface triangulation is translated in vertical direction to account for the plunging motion of the wing. Note the stiffness and inertia of the spring models were chosen to match the first plunging and pitching modes of the wing with frequencies of 3.33Hz and 5.20Hz, respectively. For the results shown here, an implicit time-integration scheme was utilized because the expected frequency of the wing motion is relatively low in comparison to the explicit timestep, Δt, based on the CFL restriction. In order to eliminate significant temporal errors the timestep was chosen to $\Delta t = 2 \times 10^{-4}$s, which roughly amounts to 1200 timesteps per period of the flutter mode. The coupled simulation was initialized by applying a distributed load of 20N/m^2 over the first 0.01s of the simulation. Two different grid resolutions with roughly 15 and 30 million grid points were used for the present FSI study. On the coarse and medium mesh, the flutter frequency of the wing motion was estimated to 4.14Hz and 4.16Hz, which is in close agreement with other participants of the workshop and the experimental results where a flutter frequency of 4.3Hz was recorded.

In Figures 27a-c the computational results are compared to the experimental dataset extracted at 60% span. On the medium mesh, the computational results appear to approach the experimental result. In the current simulation, the wall-modeling approach briefly outline in Section D was utilized. The mismatch near the trailing edge in the pressure distribution on the upper surface in Figure 27a may be associated to partial separation near the trailing edge that is not captured correctly in current simulation. Overall, a fair agreement between the computational and experimental results is obtained. Furthermore, the computational results are within the scatter of the other computational results submitted to the AePW workshop. In Figures 28a-c the computational results are also compared to the experimental dataset extracted at 95% span. Similar observations as in Figures 27a-c are made.

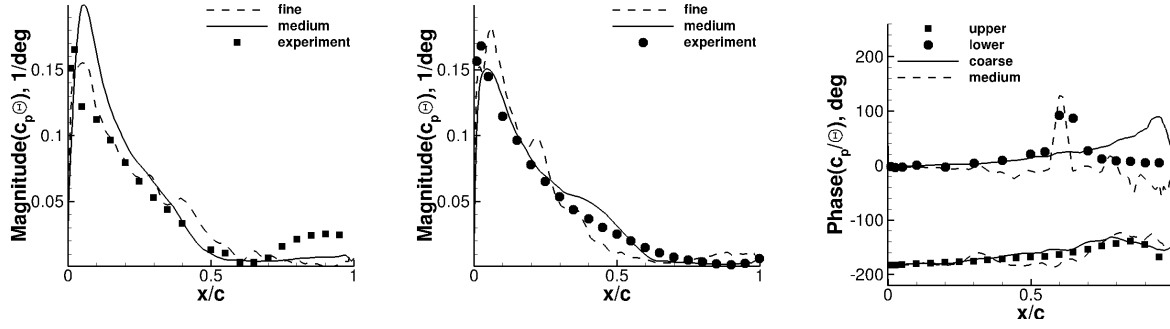American Institute of Aeronautics and Astronautics

Figure 27: Surface pressure magnitude on (a) upper and (b) lower surfaces. (c) Phase on upper on lower surfaces. All data was extracted at 60% span.
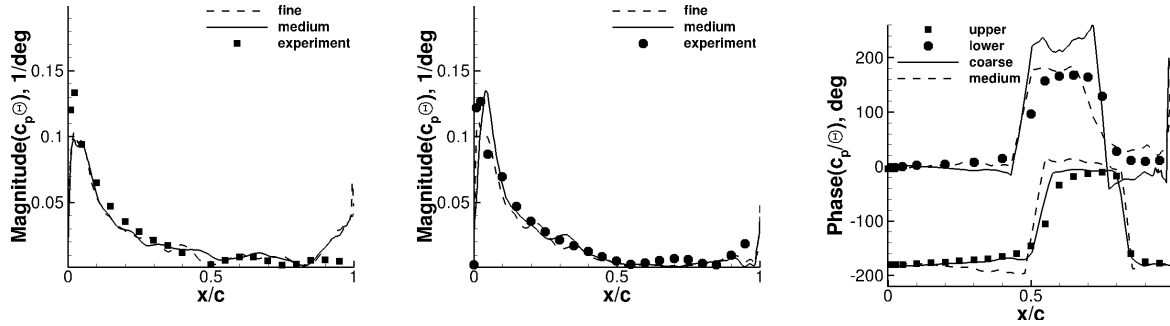

Figure 28: Surface pressure magnitude on (a) upper and (b) lower surfaces. (c) Phase on upper on lower surfaces. All data was extracted at 95% span.

D.   3D Bending Tower

In the last FSI test problem, the interaction of a super elastic tower considering a viscous fluid was simulated to demonstrate the capability of modeling fully 3D FSI problems. Figure 29a shows the simulation setup where a highly flexible tower with a square base of 0.2m×0.2m and a height of $h = 2.0$m is placed into a $L = 6$m long, 2.4m×2.4m channel. The flow at the inlet is assumed to be subsonic with a Mach number of $M = 0.08$ and a Reynolds number of $Re = 500$. The bending tower is modeled with a super elastic material with a Young's modulus of $E = 2 \times 10^7$kg/ms^2, a Poisson's ratio of $\nu = 0.3$ and a shell thickness of $\delta = 5$cm. The surface triangulation used in the structural solver and a cut through the volume mesh which is used in the computational fluid dynamics solver is shown in Figure 29b. The instantaneous snapshot of an iso-contour surface with vorticity magnitude $|\omega| = 200s^{-1}$ displays the unsteady vortex shedding occurring behind the tower. The structural parameters were chosen to obtain sub-critical conditions where a continuous vortex induced vibrations of the tower occurs

The motion of the tower consists of a superposition of several deformation modes of the tower. The deformation of the tower is illustrated in Figures 30a-c. The tower displacements are enhanced for illustration purposes. The first two predominant motions of the tower are visualized by snapshots of the cross-sectional area of the tower in the (x, y)-plane. It consists of bending of the tower in the (x, z)-plane. Note that each snapshot is represented by a different color. In addition to the bending motion in the (x, z)-plane, the unsteady vortex shedding also induces an oscillatory motion of the flexible tower in the (x, y)-plane. The motion is visualized in Figure 30c.

The motion of the tower and the flow field synchronize after an initial start-up transient is passed. In order to illustrate this process, the horizontal displacement u of the tower in x-direction is related to the velocity fluctuations in the wake of the tower. Figure 31a shows the setup of the sampling points which were used to record the structural deformation of the tower and the unsteady data in the wake of the tower. A very coherent pattern can be identified for the static tower in Figure 31b. Figure 31c clearly illustrates a

American Institute of Aeronautics and Astronautics
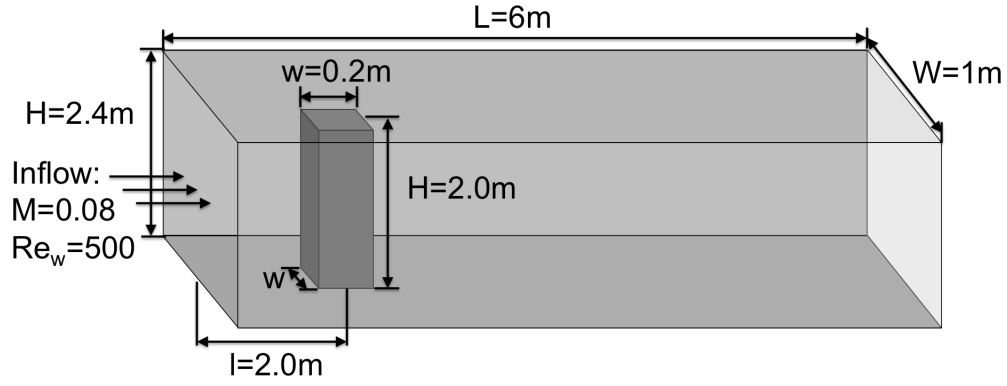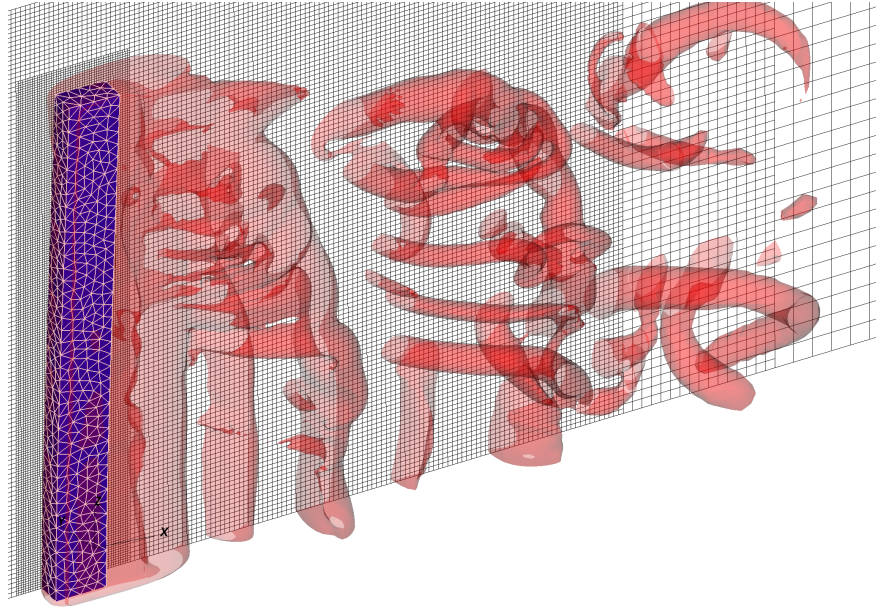
(a) Simulation setup.

(b) CFD and CSD meshes with flow visualization.

Figure 29: (a) Simulation setup for tower bending test case. (b) Surface triangulation for structural FEM solver and cut through volume mesh used in CFD solver. Instantaneous snapshot of an iso-contour surface with vorticity magnitude $|\boldsymbol{\omega}| = 200 s^{-1}$.

lock-on between the unsteady motion of the tower (visualized by a red solid line) and the unsteadiness in the wake (visualized by grey scale contours). It should be noted that the fluctuation amplitudes appear to be larger for the flexible tower in comparison to the static tower. The motion of the tower seems to enhance coherence in the wake. At $t \approx 0.35s$, the flow field and the motion of the tower seem to get out of sync. The fluctuations in u-displacement are significantly reduced during this event. The flow field and the tower motion resynchronize after this brief intermittent event. The occurrence of this event is repeated over time. While the main purpose of the last FSI test case is to showcase the FSI capabilities within LAVA, additional analysis of this fluid-structure interaction case (and higher Re flows) will be provided in follow-up works.
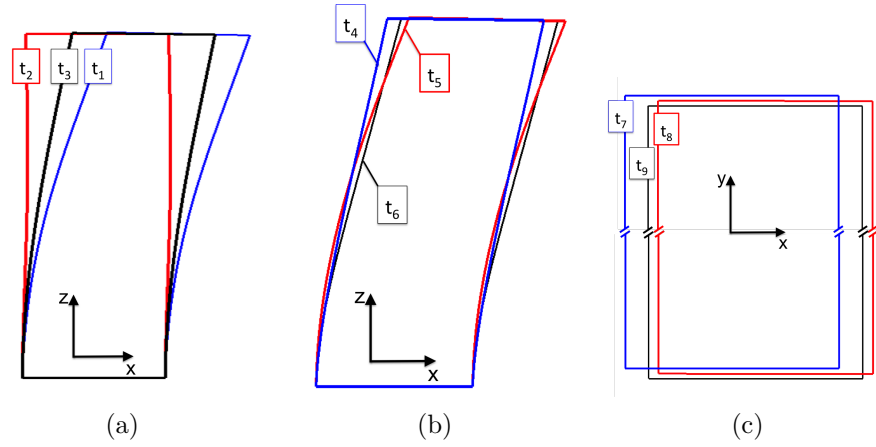
American Institute of Aeronautics and Astronautics

(a) (b) (c)

Figure 30: Visualization of tower motion for cross-sections in (x, z)-plane with $y = 0$ for (a) and (b), and in (x, y)-plane with $z = 1.95$ for (c). Note that the deformations in x-direction for (a) and (b), and in y-direction for (c) are enhanced for illustration purposes. The red, blue, and black colors refer to nine different instances in time.

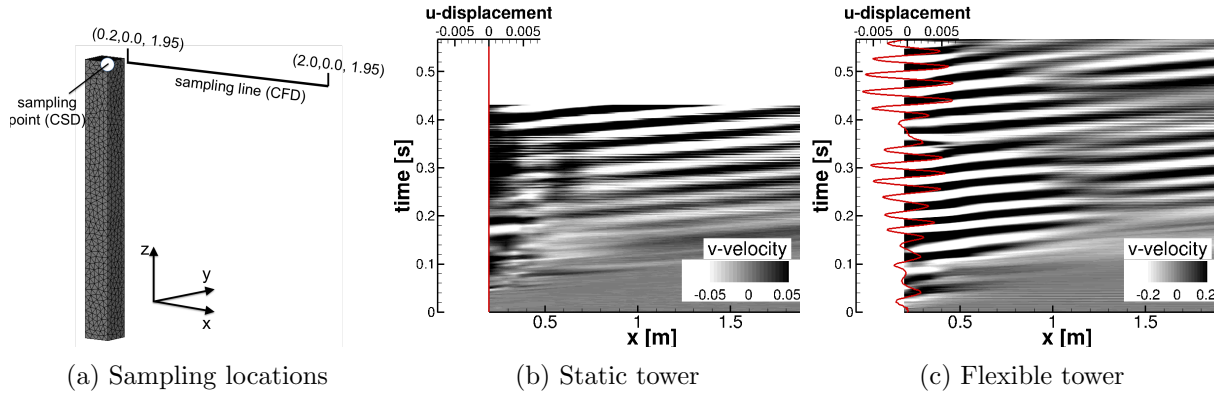(a) Sampling locations   (b) Static tower   (c) Flexible tower

Figure 31: (a) Sampling location on the tower to record structural displacement and sampling line to extract flow information. Contours of v-velocity fluctuations along x-direction and time, and structural displacement, u, in streamwise direction for (b) static and (c) flexible tower.

VII. Conclusion and Outlook

An immersed boundary method for moving boundary problems with prescribed body motion and fluid-structure interaction was presented and validated. By utilizing a sharp immersed boundary method and motion tracking based on a discrete surface triangulation, the method is able to handle highly complex and very thin geometries. Parallel geometry kernels were implemented to allow fast inside-outside testing and exact computation of the distance from the irregular grid points to the surface triangulation (including point to plane and point to edge cases). A strong focus has been put on developing the geometry query algorithms in an efficient manner to increase computational efficiency. For highly complex geometries involving moving boundaries the time spent inside these algorithms can become significant and comparable to the time spent for the solution of the governing equations in the volume.

The linear structural FEM solver was validated in stand-alone mode for canonical test problems. LAVA's newly developed capabilities for moving boundaries and fluid-structure interaction problems were validated for 2D and 3D test problems. For all test cases good agreement with available experimental data was obtained. The next step in the development of LAVA-Cartesian involves the improvement of the structural FEM solver to account for non-linear effects such that the solver can be applied to problems involving large deformations, for example in parachute simulations.

Acknowledgments

References

[1] Schuster, D. M., Heeg, J., Wieseman, C., and Chwalowski, P., "Analysis of test case computations and experiments for the Aeroelastic Prediction Workshop," *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2013, AIAA-2013-788.

[2] Heeg, J., Chwalowski, P., Raveh, D., and Dalenbring, M., "Analysis of test case computations and experiments for the Aeroelastic Prediction Workshop," *34th AIAA Applied Aerodynamics Conference, Washington, D.C.*, 2016, AIAA Paper 2016.

[3] Peskin, C., "Numerical Analysis of Blood Flow in the Heart," *J. Comput. Phys.*, Vol. 25, 1977, pp. 220–252.

[4] Peskin, C. S., "The Immersed Boundary Method," *Acta Numerica, Cambridge University Press*, 2002, pp. 1–39.

[5] Goldstein, D., Handler, R., and Sirovich, L., "Modeling a Non–Slip Flow Boundary with an External Force Field," *J. Comp. Phys.*, Vol. 105, 1993, pp. 354–366.

[6] LeVeque, R. J., *Time-Split Methods for Partial Differential Equations*, Ph.D. thesis, Stanford University, 1982.

[7] Wiegmann, A. and Bube, K., "The Explicit–Jump Immersed Interface Method: Finite Difference Methods for PDEs with Piecewise Smooth Solutions," *SIAM J. Numer. Anal.*, Vol. 31(4), 1992, pp. 1019–1044.

[8] Linnick, M. and Fasel, H., "A High–Order Immersed Interface Method for Simulating Unsteady Incompressible Flows on Irregular Domains," *J. Comput. Phys*, Vol. 204, 2004, pp. 157–192.

[9] Johansen, H. and Collela, P., "A Cartesian Grid Embedded Boundary Method for Poisson's Equation on Irregular Domains," *Journal of Computational Physics*, Vol. 147, 1998, pp. 60–85.

[10] Mittal, R. and Iaccarino, G., "Immersed Boundary Methods," *Annu. Rev. Fluid Mech.*, Vol. 37, 2005, pp. 239–261.

[11] Zhong, X., "A New High–Order Immersed Interface Method for Solving Elliptic Equations with Embedded Interface of Discontinuity," *Journal of Computational Physics*, Vol. 225, 2007, pp. 1066–1099.

[12] Duan, L., Wang, X., and Zhong, X., "A High–Order Cut–Cell Method for Numerical Simulation of Hypersonic Boundary–Layer Instability with Surface Roughness," *Journal of Computational Physics*, Vol. 229, 2010, pp. 7207–7237.

[13] Kiris, C., Barad, M., Housman, J., Sozer, E., Brehm, C., and Moini-Yekta, S., "The LAVA Computational Fluid Dynamics Solver," *52st AIAA Aerospace Sciences Meeting, National Harbor, Maryland*, January 13-17 2014, AIAA-2014-0070.

[14] Nemec, M. and Aftosmis, M., "Aerodynamic Shape Optimization Using a Cartesian Adjoint Method and CAD Geometry," *24th AIAA Applied Aerodynamics Conference, AIAA 2006-3456*, 2006.

[15] Mittal, R., Dong, H., Bozkurttas, M., Najjar, F., Vargas, A., and von Loebbecke, A., "A versatile sharp inerface immersed boundary method for incompressible flows with complex boundaries," *Journal of Computational Physics*, Vol. 227, 2008, pp. 4825–4852.

[16] Brehm, C., Hader, C., and Fasel, H., "A Locally Stabilized Immersed Boundary Method for the Compressible Navier-Stokes Equations," *J. Comput. Phys.*, Vol. 295, 2015, pp. 475 – 504.

[17] Li, Z. and Ito, K., *The Immersed Interface Method; Numerical Solutions of PDEs Involving Interfaces and Irregular Domains*, SIAM book, 2006.

[18] Tezduyar, T., "Finite Element Methods for Flow Problems with Moving Boundaries and Interfaces," *Arch. Comput. Methods Eng.*, Vol. 8, 2001, pp. 83–130.

[19] Berger, M. J. and Colella, P., "Local Adaptive Mesh Refinement for Shock Hydrodynamics," *J. Comput. Phys.*, Vol. 82, No. 1, May 1989, pp. 64–84.

[20] Almgren, A. S., Bell, J. B., Colella, P., Howell, L. H., and Welcome, M. L., "A Conservative Adaptive Projection Method for the Variable Density Incompressible Navier-Stokes Equations," *J. Comp. Phys.*, Vol. 142, 1998, pp. 1–46.

[21] Barad, M. F. and Colella, P., "A Fourth-Order Accurate Local Refinement Method for Poisson's Equation," *J. Comp. Phys.*, Vol. 209, No. 1, October 2005, pp. 1–18.

[22] Barad, M. F., Colella, P., and Schladow, S. G., "An Adaptive Cut-Cell Method for Environmental Fluid Mechanics," *Int. J. Numer. Meth. Fluids*, Vol. 60, No. 5, 2009, pp. 473–514.

[23] Zhang, Q., Johansen, H., and Colella, P., "A Fourth-Order Accurate Finite-Volume Method with Structured Adaptive Mesh Refinement for Solving the Advection-Diffusion Equation," *SIAM Journal on Scientific Computing*, Vol. 34, No. 2, 2012, pp. 179–201.

[24] Colella, P., Graves, D. T., Ligocki, T. J., Martin, D. F., Modiano, D., Serafini, D. B., and Straalen, B. V., "Chombo Software Package for AMR Applications - Design Document," unpublished.

[25] Brehm, C., Barad, M., Housman, J., and Kiris, C., "A comparison of higher-order finite-difference shock capturing schemes," *Computers & Fluids*, Vol. 122, 2015, pp. 184 – 208.

[26] Brehm, C. and Fasel, H., "A novel concept for the design of immersed interface methods," *Journal of Computational Physics*, Vol. 242, No. 0, 2013, pp. 234 – 267.

[27] Wald, I., Woop, S., Benthin, C., Johnson, G. S., and Ernst, M., "Embree: A kernel framework for efficient cpu ray tracing," *ACM Transactions on Graphics (TOG)*, Vol. 33, No. 4, 2014, pp. 143.

[28] Barad, M., Brehm, C., and Kiris, C., "Solving Moving Boundary Problems with an Immersed Boundary Method," *accepted abstract, ICCFD9-2016, July 11-15, Istanbul, Turkey*, 2016.

[29]Anderson, D., Mc Fadden, G., and Wheeler, A., "Diffuse-Interface Methods in Fluid Mechanics," *Annu. Rev. Fluid MEch.*, Vol. 30, 1998, pp. 139–165.

[30]Udaykumar, H. S., Mittal, R., Rampunggoon, P., and Khanna, A., "A Sharp Interface Cartesian Grid Method for Simulating Flows with Complex Moving Boundaries," *J. Comput. Phys.*, Vol. 174, 2001, pp. 345–380.

[31]Udaykumar, H. S., Mittal, R., and Shyy, W., "Computation of Solid–Liquid Phase Fronts in the Sharp Interface Limit on Fixed Grids," *J. Comput. Phys.*, Vol. 153, 1999, pp. 534–574.

[32]Ruffin, S. M. and Lee, J., "Adaptation of a k-epsilon Model to a Cartesian Grid Based Methodology," *International Journal of Mathematical Models and Methods in Applied Sciences*, Vol. 3, 2009, pp. 238–245.

[33]Petyt, M., *Introduction to finite element vibration analysis*, Cambridge university press, 2010.

[34]Bergan, P. and Felippa, C., "A triangular membrane element with rotational degrees of freedom," *Computer Methods in Applied Mechanics and Engineering*, Vol. 50, No. 1, 1985, pp. 25–69.

[35]Koopmann, G. H., "The Vortex Wakes of Vibrating Cylinders at Low Reynolds Numbers," *J. Fluid Mech.*, Vol. 28, 1967, pp. 501–512.

[36]Meneghini, J. R. and Bearman, P. W., "Numerical Simulation of High Amplitude Oscillatory Flow About a Circular Cylinder," *Journal of Fluids and Structures*, Vol. 9, 1995, pp. 435–455.

[37]Piziali, R., "2-D and 3-D oscillating wing aerodynamics for a range of angles of attack including stall," NASA Technical Memorandum TM-1994-4632, Sep. 1994.

[38]Ko, S. and McCroskey, W., "Computations of Unsteady Separating Flows Over An Oscillating Airfoil," *33rd Aerospace Sciences Meeting, Reno, Nevada*, January 1995, AIAA-1995-0312.

[39]Nichols, R. and Heikkinen, "Validation of Implicit Algorithms for Unsteady Flows Including Moving and Deforming Grids," *43rd AIAA Aerospace Sciences Meeting, Reno, Nevada*, January 2005, AIAA-2005-683.

[40]Piatak, D. J. and Cleckner, C. S., "Oscillating Turntable for the measurement of unsteady aerodynamic phenomena," *Journal of aircraft*, Vol. 40, No. 1, 2003, pp. 181–188.

[41]Giordano, J., Jourdan, G., Burtschell, Y., Medale, M., Zeitoun, D., and Houas, L., "Shock wave impacts on deforming panel, an application of fluid-structure interaction," *Shock Waves*, Vol. 14, No. 1-2, 2005, pp. 103–110.

[42]Dowell, E. H., *Aeroelasticity of plates and shells*, Vol. 1, Springer Science & Business Media, 1974.

[43]Pasquariello, V., Hammerl, G., Örley, F., Hickel, S., Danowski, C., Popp, A., Wall, W. A., and Adams, N. A., "A cut-cell finite volume–finite element coupling approach for fluid–structure interaction in compressible flow," *Journal of Computational Physics*, Vol. 307, 2016, pp. 670–695.