Advances in FSI using body-fitted unstructured grids

R. Löhner, B. Hübner & J. R. Cebral
School of Computational Sciences, George Mason University, Fairfax, USA

Abstract

Several algorithms for the loose coupling of fluid and structural dynamics codes are described. The first class of algorithms treats the displacement of the surface of structure that is in contact with the fluid. It is shown that a straightforward treatment of the displacements for arbitrary choice of timesteps can lead to instabilities. For optimal stability, at each timestep the ending time of the fluid should be just beyond the ending time of the structure. The second class of algorithms treats the movement of the flow mesh in an ALE setting. The use of a projective prediction of mesh velocities, as well as linelet preconditioning for the resulting PCG system, can significantly reduce the effort required. Examples are included that show the effectiveness of the proposed procedures.

1 Introduction

While CFD solvers based on adaptive cartesian or unstructured grids are commonly used for complex Fluid-Structure Interaction (FSI) simulations, for applications that require the proper resolution of boundary layers the moving, body-fitted mesh approach with arbitrary Lagrange–Eulerian (ALE) formulation remains the only viable option to date. The need to extend the range of applicability of this type of approach to long-term FSI runs, such as bridges, chimneys and offshore platforms, has led to a series of algorithmic improvements which are the subject of this paper. In particular, the following subjects will be treated:

- synchronization of times between loosely coupled CFD and CSD solvers in order to minimize grid distortion; and
- projective prediction to accelerate the convergence of the PCG solver of the mesh velocities.
2 Calculation of displacements

A loose coupling between the CFD and CSD codes implies that the ending times of the respective codes can be different. The fluid surface is imposed by the CSD surface, which is moving according to fluid forces. If the time of the CFD code $t_f$ at the end of the CFD timestep lies beyond the time of the CSD code $t_s$, the CSD surface is extrapolated. This can be done in a variety of ways (linear [1], cubic [2], etc.), but in all probability the assumed position of the CSD surface and the one calculated by the CSD code in the next timestep will not coincide. This implies that some form of correction will be required at the beginning of the next CFD timestep. An exaggerated situation where this can happen has been sketched in Figure 1. Note the jump in surface positions at the beginning of the next CFD timestep. Experience with coupled explicit codes indicates that for isotropic (Euler) grids, this jump does not affect the results. However, for highly stretched grids that resolve boundary layers in the fluid (RANS grids), a jump in the surface can easily lead to elements with negative volumes and a breakdown of the simulation.

An alternative is to ignore the new surface velocities at the end of the CSD timestep, and to compute velocities directly from positions. This can be easily done if the surface mesh of the CSD domain remains intact (e.g. no changes in topology).

The surface velocity of the CFD domain is then continued from the previous position in such a way that the CSD surface seen by the CFD code coincides exactly with the position of the CSD surface seen by the CSD code at $t = t_s$. Although intuitively simple, the method is unstable if the ending time of the fluid is larger than the ending time of the structure by half the timestep taken. This behaviour is shown in Figure 2. For optimal stability, the ending time of the fluid should be just beyond the ending time of the structure: $t_f = t_s + \epsilon$.

3 Projective prediction of mesh velocities

The movement of the CFD mesh as the surfaces of bodies immersed or surrounding the flowfield move or deform can have a profound effect on overall code performance. A bad choice of mesh movement leads to frequent remeshings with all associated negative effects (artificial viscosity due to loss of information during reinterpolation, degradation of performance on parallel machines, etc.). Many good mesh movement techniques have been proposed [6, 8, 10]. We have used a non-linear Laplacian-based technique [6, 7] that solves the system:

$$\nabla k \nabla w = 0, \quad w_\Gamma = w_b,$$

where $w$ denotes the velocity of the mesh, $w_b$ the (imposed) velocity at the surface of the bodies immersed or surrounding the flowfield, and $k$ is a diffusivity that depends on the distance from the body. The discretization of Eqn. (1) via finite elements results in a discrete system of the form:
The solution of this system is performed using a preconditioned conjugate gradient solver. Diagonal preconditioning is employed for isotropic grids, while for RANS grids linelet preconditioning [9] is preferred.

In what follows, the basic assumption is that $K$ does not change significantly in time. For many situations this is indeed the case. The mesh does not move significantly from timestep to timestep, and the distance from the bodies for individual gridpoints also does not change considerably.

If we denote by $w_i, r_i, i = 1, l$ the values of the mesh velocity and right-hand sides at previous timesteps $n - i$, we know that:

$$K \cdot w_i = r_i. \quad (3)$$

Given the new right-hand side $r$, we can perform a least-squares approximation to it in the basis $r_i, i = 1, l$:

$$\left( r - \alpha_i r_i \right)^2 \rightarrow \min. \quad (4)$$
which results in
\[
A\alpha = s, \quad A^{ij} = r^i \cdot r^j, \quad s^i = r^i \cdot r. \tag{5}
\]

Having solved for the approximation coefficients \(\alpha_i\), we can estimate the start value \(w\) from:
\[
w = \alpha_i w^i. \tag{6}
\]

We remark that, in principle, the use of the right-hand sides \(r^i, i = 1, l\) as a basis may be numerically unstable. After all, if any of these vectors are parallel, the matrix \(A\) is singular. One could perform a Gram-Schmidt orthogonalization instead. This option was invoked by Fischer [3] who looked at a number of possible schemes to accelerate the convergence of iterative solvers using successive right-hand sides within the context of incompressible flow solvers based on spectral elements. However, we have not found this to be a problem for any of the cases tried to date. The advantage of using simply the original right-hand sides is that the update of the basis is straightforward. We keep an index for the last entry in the basis, and simply insert the new entries at the end of the timestep in the position of the oldest basis vector. The storage requirements for this projective predictor scheme are rather modest: \(2*\text{ndimn}*\text{npoint}*\text{nvec1}\). We typically use 1-4 basis vectors, i.e. the storage is at most \(24*\text{npoint}\).

The effect of using even 2 search directions is dramatic: for the cylinder and bridge-cases shown below, the number of mesh velocity iterations per timestep drops from \(O(20)\) to \(O(2)\).

4 Examples

4.1 Tacoma narrows bridge

The structure is computed with an implicit eigenmode integrator that uses the second-order Newmark scheme [12]. In the present case, only two eigenmodes are considered: heave and torsion [4, 11, 5]. The eigenmodes are approximated by a parabolic profile along the span, assuming that one end is clamped. Due to symmetry, only half of the bridge (i.e. 50 m) is computed. The incoming flow conditions are set to \(\rho = 1.25 \text{ kg/m}^3, v_\infty = 10.0 \text{ m/sec}, \mu = 0.1 \text{ kg/m/sec}\). The flow mesh had approximately 8.5M elements. The coupled system is advanced in time using the explicit loose coupling strategy, but making sure that for each timestep the ending time of the fluid is just beyond the ending time of the structure \((t_f = t_s + \epsilon)\). Figures 3a, b show a typical result after the structure has started to vibrate. Note the difference in the vortex structure for the planes where the bridge is clamped (left) and free to vibrate (right). The time-histories of the eigenforces and eigenmodes can be seen in Figures 3c, d. Note the onset of the torsional instability after approximately 4 minutes of real time. Even though the mesh was moved through many cycles, and torsion was considerable, no remeshing was required. Figure 3e depicts the exchange of work between the fluid and the structure. Note that with the onset of the instability, the work exchange is no longer balanced, but shifts to a net flow of work (energy) from the fluid to the structure.
Figure 3: (a) Bridge: typical flowfield; (b) velocity of structure; (c) eigenforces; (d) eigenmodes; (e) work exchanged between fluid and solid.
4.2 Heated cylinder

This classic case considers a cylinder of unit diameter that is heated uniformly with a source. The surrounding flow is cold, and transports the heat away. The geometry is shown in Figure 4a. A cut through the plane $z = 0$ shows that the grids employed for the flow domain and the solid domain are very different indeed (Figure 4b). The flow domain requires a mesh suited for the boundary layer, with highly stretched elements. The solid domain is discretized with isotropic elements. The material data were set as follows:

- Fluid: $\rho = c_p = 1.0$, $\mathbf{v} = (1.0, 0.0, 0.0)$, $\mu = k = 0.054$.
- Solid: $\rho = c_p = k = 1.0$, $s = 1.0$. 

Figure 4: Heated cylinder: (a) fluid domain; (b) plane $z = 0.0$; (c) velocity; (d) temperature ($z = 0.0$).
The velocity and temperature fields in the same plane are shown in Figures 4c,d for a given time. Note the transport of heat into the fluid. The under-relaxation factor for this case was set to $\alpha = 0.2$ and 5 iterations between the solid and fluid were required per timestep.

4.3 Carotid artery

This example presents results of hemodynamics in a compliant model of a normal carotid bifurcation. The anatomical model was reconstructed from a contrast-enhanced MRA and physiologic flow conditions were measured with PC-MR. Implicit CFD and CSD solvers were used to compute the blood flow and vessel wall motion.

The arterial wall compliance was modelled via a ring approach that assumes that the structural nodes move only in the radial direction. With this approximation, a 1D ODE that has inertial, damping, stiffness and load terms (from the fluid) was obtained for each node of the vessel structure. The flow mesh had approximately 300 K elements. An implicit FSI algorithm with an under-relaxation
factor of $\alpha = 0.4$ was used to solve the coupled problem. On average, 3 iterations between the fluid and structure were required per timestep. Figure 5 shows (top row, from left to right) the contrast-enhanced MRA images, the reconstructed vascular model, the mean wall shear stress magnitude (WSS) for compliant vessel walls, the mean WSS for rigid walls, the oscillatory shear index (OSI) for compliant vessels, and the OSI for rigid vessels. The graph in the bottom left of this figure shows the PC-MR measurements of flow obtained in each branch of the carotid bifurcation (common carotid artery – CCA, internal carotid artery – ICA, and external carotid artery – ECA). This plot also shows the sum of the flow rates in the outflow vessels (ICA + ECA), which is significantly different from the instantaneous flow rate measured at the inflow of the CCA. In the CFD simulation, the flow measured in the ICA and ECA were prescribed at the corresponding outflow boundaries. When the vessel walls were assumed rigid, the inlet flow in the CCA was, as expected, equal to the sum of the ICA and ECA flows, and therefore different to the measured values at this location. On the other hand, when the vessel walls were compliant, the computed flow at the inlet boundary of the CCA was in very good agreement with the corresponding PC-MR measurement, as shown at the bottom right of Figure 5. This example shows that compatible flow conditions can only be prescribed if the vessel wall compliance is incorporated into the models, and that implicit coupling is necessary in order to obtain accurate flow rates.

5 Conclusions and outlook

Several algorithms for fluid-structure interaction have been described. All of them are useful for the loose coupling of fluid and structural dynamics codes.

The first class of algorithms treats the displacement of the surface of structure that is in contact with the fluid. It is the motion of this surface that determines the movement of the fluid domain. Given that CSD codes return positions and velocities, the question is how to treat these. It is shown that a naive treatment of the displacements for arbitrary choice of timesteps can lead to instabilities. For optimal stability, at each timestep the ending time of the fluid should be just beyond the ending time of the structure.

The second class of algorithms treats the movement of the flow mesh in an ALE setting. The use of a projective prediction of mesh velocities, as well as linelet preconditioning for the resulting PCG system can significantly reduce the effort required.

Acknowledgements

This research was partially supported by the Whitaker Foundation, as well as DTRA (technical monitors: Drs. Ali Amini and Seung Lee).
References