Asociación Argentina
de Mecánica Computacional

Mecánica Computacional Vol XXXIV, págs. 3135-3147 (artículo completo)
Sebastián Giusti, Martín Pucheta y Mario Storti (Eds.)
Córdoba, 8-11 Noviembre 2016

CONSERVATIVE COUPLING STRATEGY WITH NON-COINCIDENT MESHES IN PARTITIONED FLUID STRUCTURE INTERACTION PROBLEMS

Gustavo Ríos Rodríguez[a], Luciano Garelli[a], Marco Schauer[b] and Mario Storti[a]

[a] *Centro de Investigación de Métodos Computacionales (CIMEC) Universidad Nacional del Litoral (UNL) - CONICET Predio Conicet-Santa Fe, Colectora Ruta Nac 168 / Paraje El Pozo, Santa Fe (S3000GLN), Argentina e-mail: gusadrr@santafe-conicet.gov.ar*

[b] *Institut für Statik, Technische Universität Braunschweig, Beethovenstrasse 51, 38106 Braunschweig, Germany, e-mail: m.schauer@tu-braunschweig.de, web-page: www.tu-braunschweig.de/statik*

Keywords: conservative projection algorithm, fluid structure interaction, non-coincident meshes, surface tracking.

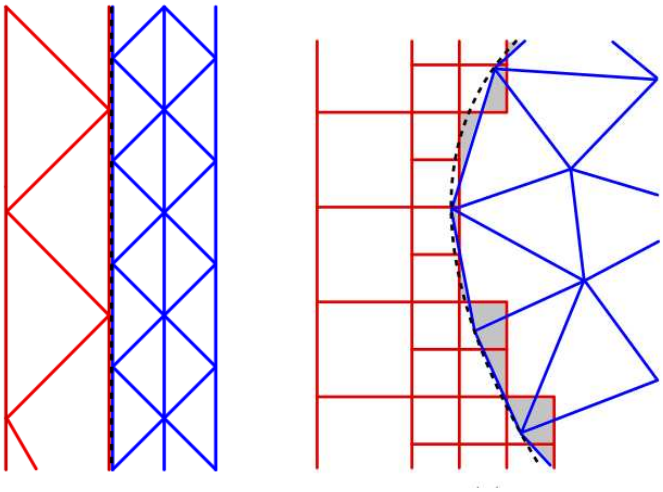
Abstract. Numerical simulations of fluid structure interaction problems have become affordable for a wide range of engineering applications. However, since the complexity of such simulations has increased due to the introduction of other physical phenomena like thermodynamics and acoustics, it is convenient to use a partitioned strategy whenever possible. The coupling of well developed and reliable computer codes can tackle, without introducing major modifications, the numerical solution of such problems. In this scenario, it is also desirable to count with the flexibility of using different discretizations for computing the solution of each physical problem. However, this introduces the requirements of projecting the solution from the interface of one subdomain to that of its neighbour subdomain, to track the displacement of the boundaries and to sychronize the solvers to be able to use differente time step sizes for the computations. In this work, a conservative projection and a tracking algorithm are analyzed by computing the solution of benchmark problems of the fluid structure interaction kind with non-coincident meshes at the interfaces. The accuracy, consistency and conservativeness of the implemented algorithms are evaluated and compared to those of another ones mentioned in the literature. The strengths and weaknesses of both algorithms are finally mentioned.

1 INTRODUCTION

Nowadays, the numerical solution of complex Fluid-Structure Interaction (FSI) and other kind of multiphysics problems like Soil-Structure Interaction (SSI) requires to use parallel computer codes together with high performance computing resources. The development and validation of a parallel FSI computer code is a very complicated task which also involves a lot of time. In some situations, it is possible and desirable to combine the capabilities of well developed and specialized computer codes to tackle the numerical solution of such problems using a partitioned solution scheme. In the FSI case, the Computational Fluid Dynamics (CFD) and the Computational Structure Dynamics (CSD) problems are solved by different codes. However, since the codes are assumed to work independently, there is a need to develop software layers which sychronize and transfer the required information between them.

On the other hand, the usage of a partitioned solution scheme allows to use different spatial domain discretizations for the fluid and the structure, thus providing the flexibility to use different meshes for the CFD and the CSD problems. For example, the fluid domain discretization is usually very refined at the boundaries with a solid object because of the boundary layer existance and the solver may be based on the Finite Volume Method (FVM). On the other hand, the mesh of the structure usually does not need to be that fine at those interfaces and the solver may be based on the Finite Element Method (FEM). In a SSI problem the solution on the structure could be computed with a FEM formulation and the wave propagation in the soil by means of a Scaled Boundary Element Method (SBFEM) formulation (Schauer et al., 2011). In general, different domain discretizations means that both meshes are non-coincident at the interfaces, perhaps with gaps between them. Even if the interface boundaries coincide in the geometrical surface or line, their nodes generally have different locations (Gatzhammer, 2014) (see Fig.1), introducing the difficulty of transferring the information from one of the domain boundaries to the other. In these situations, special care has to be taken in the solution projection step to ensure a conservative transfer of the loads or another solution field from the fluid to the structure boundary. Also, after the structure suffers deformation, the movement of its boundary has to be followed or tracked by that of the fluid domain, which forces to adopt a tracking methodology. The latter should also fulfill the requirements of consistency and conservativeness. Finally, a synchronization strategy must be used between the CFD and the CSD codes in order to consider different time-step sizes for computing the corresponding solutions (Cebral, 1996).

In this work we describe and analyze two algorithms, one to deal with the load transfer issue and the other to track the structure interface deformation. The former is conservative in the sense that it is devised by enforcing that the load on the fluid boundary must be equal to that on the structure side. The scheme is based on the usage of Gauss numerical quadrature to approach an integral term that arises in the formulation, where the product of mixed finite element shape functions from the structure and the fluid flow appears (Cebral and Löhner, 1997). On the other hand, the methodology known as initial distance vectors is used for carrying out the surface tracking of the structure boundary mesh. This technique is adopted since it is useful when the meshes of the domains do not match and also allows to couple models of different dimensionality (Cebral (1996); Cebral and Löhner (1997)). The analysis of both algorithms are performed by solving benchmark problems presented in de Boer et al. (2006, 2007); de Boer (2008), focused on the accuracy, consistency and conservativeness of the methods. The results obtained are compared to those shown in those works for these strategies and another ones.



(a) Non-conforming meshes due to non-coincident nodes but coinciding interface geometry.

(b) Non-conforming meshes due to non-coincident boundaries.

Figure 1: Different cases of non-conforming meshes. Figure taken from ref.(Gatzhammer, 2014).

2 CONSERVATIVE PROJECTION

Following the formulation presented in Cebral (1996), the conservative strategy states that the integral of the variable of interest (e.g. the pressure $p(x)$, displacemente $u(x)$ or energy $w(x)$) along the fluid domain boundary Γ_f must be equal to the integral along the structure domain boundary Γ_s. Considering for the analysis the pressure variable $p(x)$, assuming that a FEM approximation is used for its computation, and also that a weigthed residual approach is considered on the fluid-structure boundary interface Γ_{fs}, particularly a Galerkin technique, it can be stated that

$$\int_{\Gamma_{fs}} W(x)p_s(x)d\Gamma = \int_{\Gamma_{fs}} W(x)p_f(x)d\Gamma \tag{1}$$

$$\int_{\Gamma_{fs}} N(x)p_s(x)d\Gamma = \int_{\Gamma_{fs}} N(x)p_f(x)d\Gamma \tag{2}$$

where $N(x)$ is the usual FEM shape functions. If $N(x)$ is considered on Γ_s on both sides of the former equation and introducing the FEM approach for the pressures, it can be stated that

$$\int_{\Gamma_{fs}} N_s^i(x)N_s^j(x)\hat{p}_{sj}d\Gamma = \int_{\Gamma_{fs}} N_s^i(x)N_f^j(x)\hat{p}_{fj}d\Gamma \tag{3}$$

where $N_s^j(x)$ is the shape function at vertex-*j* on the structure boundary mesh, $N_f^j(x)$ is the shape function at vertex-*j* on the fluid boundary, $\hat{p}_{fj}$ is the fluid pressure at vertex-*j* of the fluid mesh computed with FEM, $\hat{p}_{sj}$ is the pressure at vertex-*j* on the structure boundary mesh computed by projection (i.e., the unknown of the projection problem). On the left-hand side of eq.(3) there appears the mass matrix of the structure boundary elements and on the right-hand side arises an integral which includes the product of mixed shape functions on both the structure and the fluid discretizations. In this work, such integral is approximated by using a Gauss

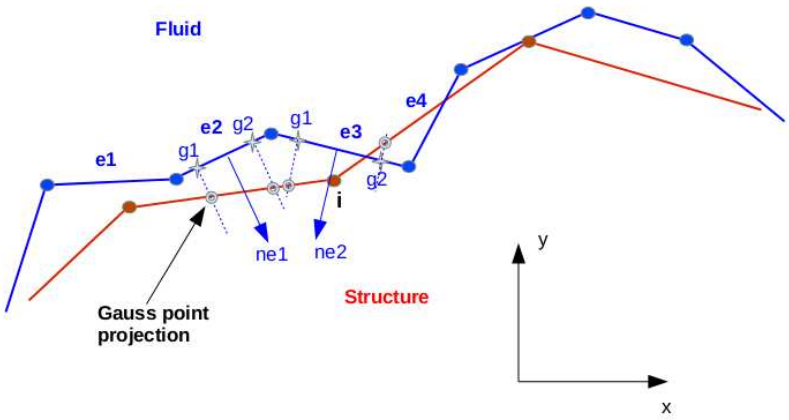


Figure 2: Conservative projection scheme in 2-D

numerical quadrature, introducing the Gauss points on the fluid elements side and projecting them by their corresponding normals onto the structure elements (Cebral, 1996). The shape functions for the weigthed residual approach are evaluated on the structure side in equation (3) because it is easier to compute and more convenient for inverting the mass matrix since the unknown is $\hat{p}_s$. The linear system of equations that is finally solved can be written as,

$$M_{cs}\hat{p}_s = r \tag{4}$$

where M_{cs} is the global consistent mass matrix for the elements on the structure boundary, $\hat{p}_s$ is the solution vector with the projected pressures and r is the right-hand side vector in which the entry associated with vertex-*i* of the structure is evaluated as follows,

$$r_i = \sum_{e=1}^{N_{efl}} \sum_{g=1}^{N_{gpts}} N_s^i(\tilde{x}_g)\, A_f^{(e)}\, W(g)\, \hat{p}_f(x_g) \tag{5}$$

being N_{efl} the number of elements of the fluid mesh at the interface of both domains for which the summation over the Gauss points is not identically null, N_{gpts} is the number of Gauss points for the current fluid element, $A_f^{(e)}$ is the area (or length) of the current fluid element, $W(g)$ the weight at Gauss point g with coordinates $\mathsf{x_g}$, $\hat{p}_f(x_g)$ the fluid pressure evaluated at Gauss point g and $N_s^i(\tilde{x}_g)$ the value of the shape function associated with vertex-*i* of the structure boundary mesh, evaluated at the normal projection $\tilde{x}_g$ of Gauss point g. The evaluation of the right-hand side term of eq.(4) is done in this way because it guarantees that all the fluid pressure is transmitted to the structure (i.e., conservation of pressure).

Solving the linear system of equations (4) allows to find the pressures on the structure side. Figure (2) depicts the elements previously mentioned.

2.1 Numerical tests for the conservative projection

The following tests were taken from de Boer et al. (2006, 2007); de Boer (2008). Therein, different methods for transferring the pressure from the fluid to the structure side of the interface are evaluated. These include the Nearest Neighbour or Direct interpolation (NN) method, Nearest Neighbour Projected (NN proj), Weigthed Residual plus numerical integration by Gauss

quadrature (WR+GI) (which is the method already implemented in this work), Intersection Method (IS), Multi-quadratic Biharmonic Splines (MQ), Thin Plate Splines (TPS) and Radial Basis functions (RB), considering two approaches: *conservative* and *consistent*. It is worth to mention that these terms are used to denote different methodologies to solve the projection problem. In de Boer (2008) they are defined as

Consistent approach occurs when a constant state or distribution of a variable (e.g. pressure, displacement, load or energy) on one side of the interface is *exactly* recovered, after projection, on the other side of the interface.

Conservative approach it makes reference to conservation of the energy or virtual work at the interface. The projection matrix for the pressures H_{fs} is related to the projection matrix for the displacements H_{sf}, so that conservation of energy is guaranteed. This relation is $H_{fs} = [M_{ff} H_{sf} M_{ss}^{-1}]^T$, where M_{ff} is the consistent mass matrix for the fluid interface and M_{ss} is that for the structure interface.

Both test problems consist in a single transfer of a pressure field defined over an interface with shape given by $y(x) = 0.2\sin(2\pi x)$ and $x \in [-0.5, 0.5]$. The profile of the pressure to be transmited is smooth in the first test and non-smooth in the second one. The pressure, computed with the conservative projection implemented in this work (denoted Weighted Residual + Gauss Integration, WR + GI in the figures) on the structure side of the interface Γ_s, is compared to the analytical values given by the corresponding smooth or non-smooth profiles through the computation of the relative *L2*-transfer error for the pressure,

$$Err_{L2} = \sqrt{\frac{\sum_{i=1}^{n_s} \| \, p_{ex}^{(i)} - p_{app}^{(i)} \, \|^2}{\sum_{i=1}^{n_s} \| \, p_{ex}^{(i)} \, \|^2}} \tag{6}$$

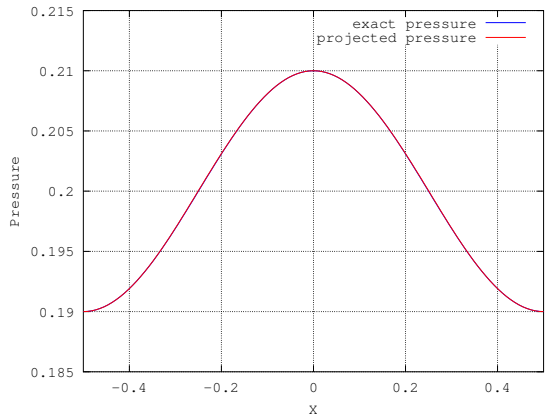
where n_s is the number of structure vertices on Γ_s, $p_{ex}^{(i)}$ is the value of the exact pressure at vertex-*i* of the discretized Γ_s and $p_{app}^{(i)}$ is the corresponding approximate pressure computed by conservative projection.

2.1.1 Smooth pressure profile

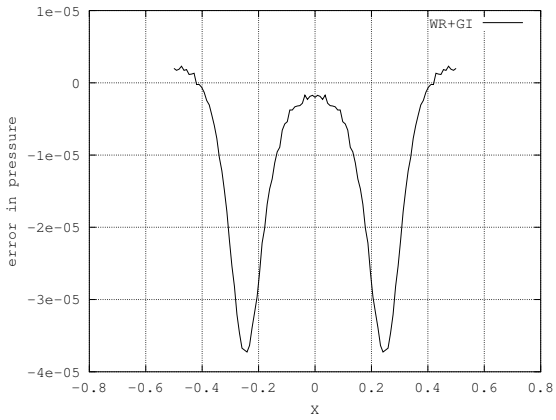
A smooth pressure field is transmitted from the fluid side of the interface to the structure side. The pressure field is given by the following equation, according to de Boer et al. (2006)

$$p(x) = 0.2 + 0.01\cos(2\pi x) \tag{7}$$

The pressure field is discretized on the fluid side and it is projected onto the structure side. The number of cells which discretize the fluid and structure interfaces is $n_f = 21 \cdot 2^k$ and $n_s = 7 \cdot 2^k$, for $k = 0, \ldots, 5$, so that fluid and structure discretizations do not match at the interface. Figure (3.a) shows both the approximate and the exact pressure distributions on the structure side of the interface and figure (3.b) the error in the *L2*-transfer error for the pressure. The results shown in both figures were computed for $n_s = 112$ and $n_f = 336$. It can be seen that there is almost no difference between the exact pressure distribution and that computed with the WR+GI method presented in this work, being the difference of $\mathcal{O}(10^{-5})$ for the whole interval.



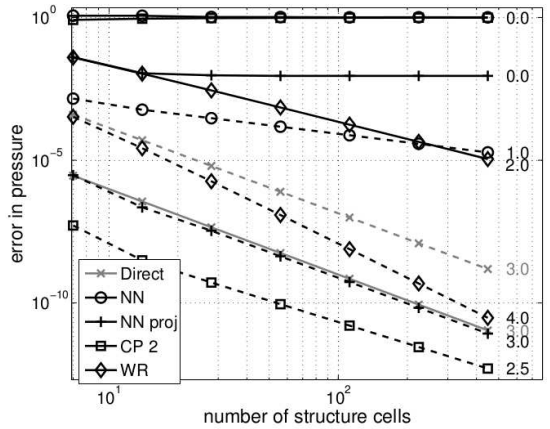
(a) Interpolated and exact pressure distributions computed on the structure side of the interface.



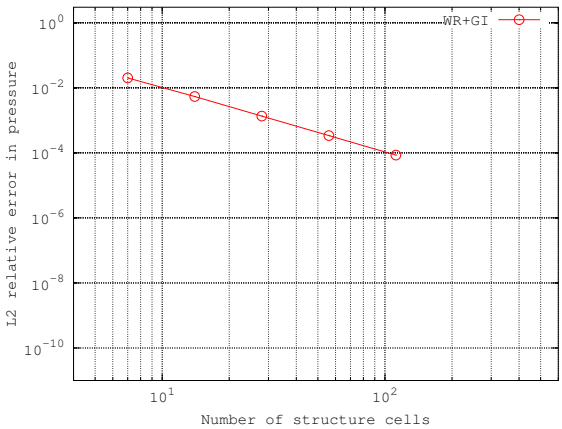
(b) Error in pressure $p_{approx}(x) - p_{exact}(x)$ computed on the structure side of the interface.

Figure 3: Numerial test results for smooth pressure distribution with $n_s = 112$ and $n_f = 336$.

On the other hand, figures (4.a) and (4.b) are used to analyze the accuracy and convergence rate of the conservative projection strategy implemented in this work with regards to other interpolation strategies and to itself as implemented in de Boer (2008). Figure (4.a) is extracted from this reference, and it shows that the implemented method behaves as expected if the line corresponding to the weighted residual method (WR) *for the conservative approach* is compared to that shown in figure (4.b). The convergence rate for this method in this particular test problem is quadratic on the number of cells at the structure interface. Also, the accuracy obtained with this method is the same than that shown by the WR method in de Boer (2008). It can be concluded that the WR+GI method implemented in this work behaves as expected.

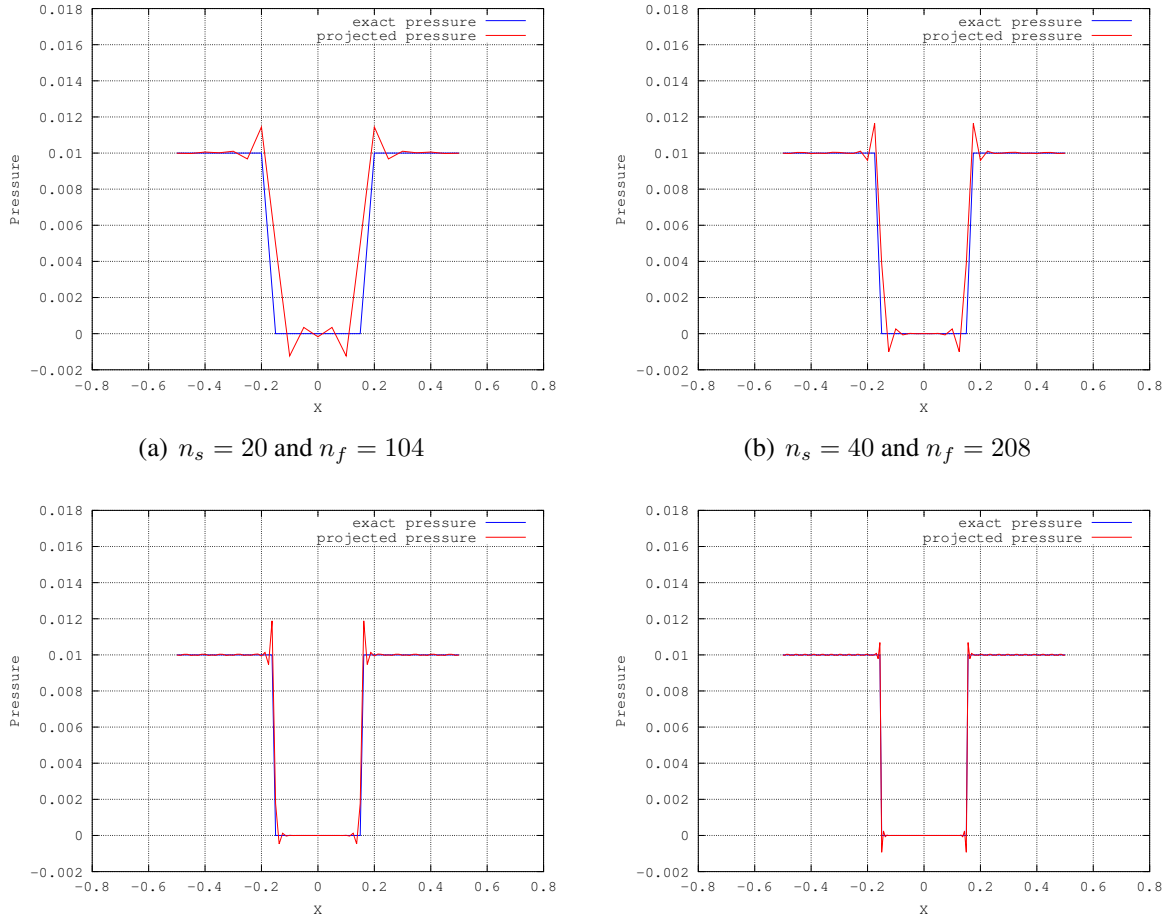


(a) Convergence for the methods presented in de Boer (2008).
(-: conservative approach, - -: consistent approach).



(b) Convergence computed with the implemented conservative method.

Figure 4: Interpolation error and interpolation order of convergence for a smooth pressure distribution.



(a) $n_s = 20$ and $n_f = 104$

(b) $n_s = 40$ and $n_f = 208$

(c) $n_s = 80$ and $n_f = 416$

(d) $n_s = 160$ and $n_f = 832$

Figure 5: Interpolated and exact pressure distributions on structure side of the interface for non-smooth pressure profile.

2.1.2 Non-smooth pressure profile

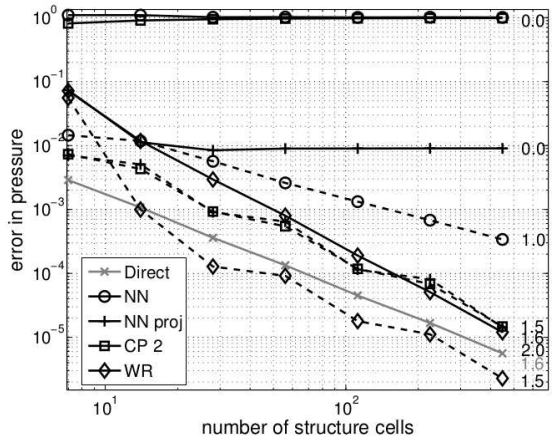
In this case, a non-smooth discretized pressure field is transmitted from the fluid side of the interface to the structure side. Such a discontinuous pressure distribution could be produced, for example, by the presence of shock waves in the fluid. The pressure field is given by the following equation, according to de Boer (2008)

$$p(x) = \begin{cases} 0.01(2 - |x|/a) & |x| < a \\ 0.01 & |x| \geq a \end{cases} \tag{8}$$

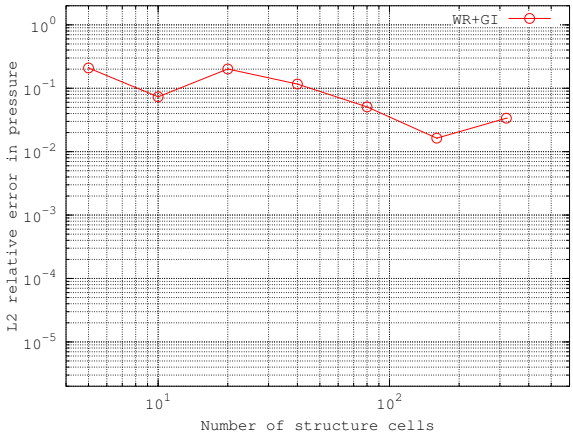
The expressions which define the number of cells that discretize the fluid and structure sides of the interface are $n_f = 26 \cdot 2^k$ and $n_s = 5 \cdot 2^k$, for $k = 0, \ldots, 5$, so that fluid and structure discretizations do not match at the interface. Figures (5.a) through (f) show both the approximate and the exact pressure profiles on the structure side of the interface for increasing number of elements for both the fluid and structure sides.

From these figures it can be concluded that the conservative interpolation introduces oscillations in the projected solution. The magnitude of these oscillations tends to decrease when both the fluid and the structure interfaces are refined, although they never dissapear. Figures 6.(a)

and 6.(b) are used to analyze the accuracy and convergence rate of the conservative projection strategy implemented in this work for this particular test, with regards to other interpolation strategies and to itself as implemented in de Boer (2008). Figure 6.(a) is extracted from this reference, and it shows that the implemented method does not behave as expected if the weighted residual (WR) line in this figure is compared to that shown in figure 6.(b). The convergence rate for the WR method in this particular test problem is between 1.5 and 2.0 on the number of cells at the structure interface but this convergence rate is not achieved by the convervative method implemented in this work.



(a) Convergence for the methods presented in de Boer (2008).
(-: conservative approach, - -: consistent approach)



(b) Convergence computed with the implemented conservative method.

Figure 6: Interpolation error and interpolation order of convergence for a non-smooth pressure distribution.

3 A BRIEF DESCRIPTION OF THE INTERFACE TRACKING STRATEGY AND ITS IMPLEMENTATION

In this work, the strategy called "initial distance / difference vectors" described in (Cebral, 1996) is followed. This algorithm allows the Computational Fluid Dynamics (CFD) mesh to follow the Computational Structure Dynamics (CSD) mesh when both meshes are non-coincident at the interface. The strategy can be described by the following steps,

1. First, the initial distance vectors between the fluid and the structure mesh are computed for the initial setup. A distance vector is defined as a vector which measures the distance between a vertex-*i* of the CFD mesh with respect to the boundary of the CSD mesh. To this end, the host element (i.e., the segment of line or the surface element of the CSD mesh where the projection by the normal at the CFD vertex-*i* falls within) has to be determined (see figure 7). This is done by computing the normal vector n_i at the CFD vertex-*i* as the average of the normals to the elements which share vertex-*i* and finding its intersection on the CSD boundary. Previously, an Approximate Nearest Neighbour (ANN) search problem is solved by using the algorithms and data structures of the ANN library (Arya and Mount, 2010). The problem consists in building a list of probable or potential host elements to search on the CSD boundary for vertex-i on the fluid boundary. This list is constructed by using a kd-tree search structure built on the centroids of the CSD boundary

elements. Then, a list is assembled with the n-nearest structure elements to vertex-i and iterations are performed on this list to determine the structure host element by solving a line-segment intersection problem between the line that passes through both vertices of the potential host element and the line normal to the fluid boundary mesh at vertex-i (or equivalently, a line-plane intersection problem in 3-D). If intersection exists, it has to be evaluated if it effectively falls within the potential host element by applying the condition $\min(N^k(p), 1 - N^k(p)) \geq 0, \quad \forall k$, where $N^k(p)$ is the finite element shape function associated to local vertex-k of the potential host element evaluated at the intersection point p. Also, at this instance, it is found more valuable to store the values of the finite element shape functions at the intersection point than the intersection point itself, since they will be used in the next stages of the strategy.

2. Compute the normals to the CSD boundary vertices.

3. Use the shape function values stored in step 1 to compute a direction vector at the intersection point by interpolation of the normals to the CSD boundary vertices computed in item 2.

4. Compute the distance between vertex-*i* and its intersection and scale the length of the direction vector computed in item 3 to this length. This is the initial distance or difference vector associated to vertex-*i* of the CFD boundary mesh which will be used to track the CSD mesh boundary in subsequent time steps.

In a weakly-coupled FSI problem simulation, after the fluid flow solution is computed, the loads or pressures are transferred to the structure, which suffers a deformation so that its boundaries move. Now, the CFD boundary mesh has to follow the CSD boundary movement. This is done by updating the difference vectors by proper translation and rotation in the following manner,

1. Rotation of the distance vectors follows those of the normals to the vertices of the corresponding CSD element. Since the structure host element wherein projection of the CFD mesh vertex-i falls within is known at this stage, the new direction of the distance vector associated with this vertex can be computed by averaging the normals to the vertices of the CSD host element using the values of the finite element shape functions already computed in the initial position (see figure 8). As it is mentioned in Cebral (1996), this way of updating the distance vectors allows to get smoother CFD boundary surfaces if severe deformation of the CSD boundary occurs.

2. Translation of the difference vector is done by computing the new coordinates of its origin (point **p** in figures 7, 8 and 9) by averaging the coordinates of the vertices of the corresponding CSD element.

3. Length of the distance vector is kept constant throughout the time steps. Consequently, the new coordinates for vertex-*i* of the CFD mesh can be computed by adding new coordinates of vertex **p** to the new distance vector, so that the boundary of the CFD mesh can be moved (see figure 9).

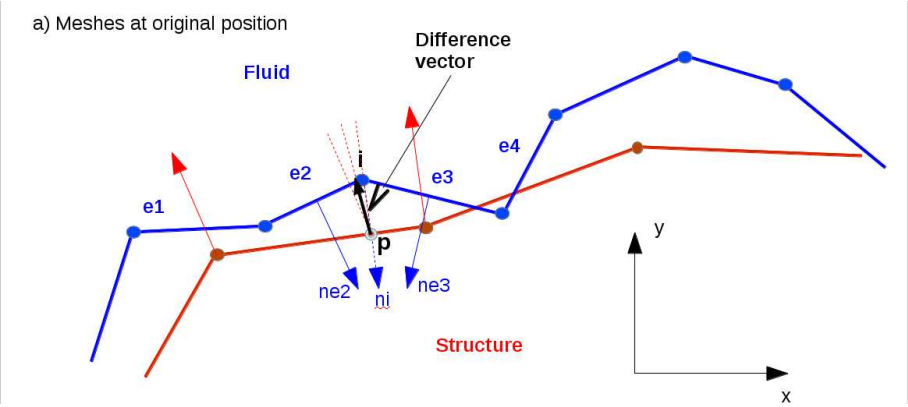


Figure 7: Schematic representation of the initial distance vector computation for both meshes at their starting positions.

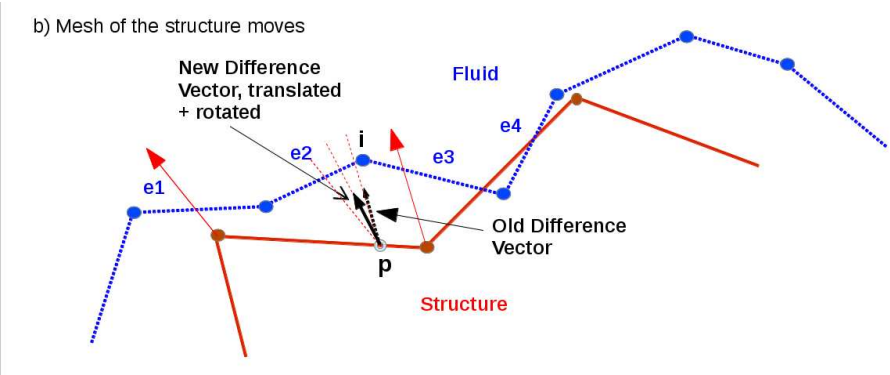


Figure 8: Schematic representation of the difference vector updating after the CSD mesh boundary moves.

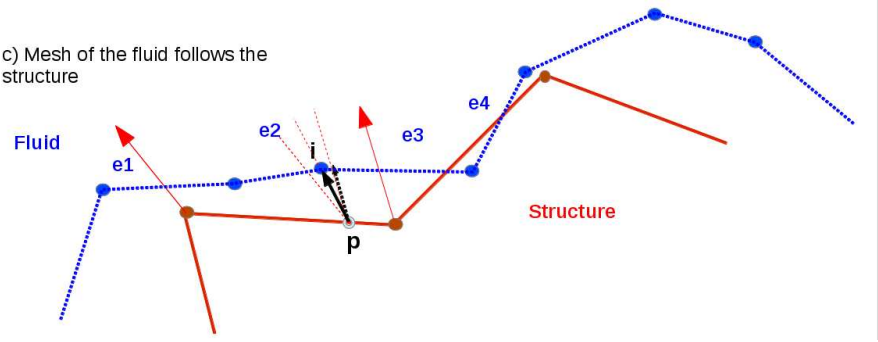


Figure 9: Schematic representation of the CFD position updating with the new difference vector.

3.1 Numerical tests for the interface tracking strategy

The following test is proposed in de Boer et al. (2007) to evaluate the accuracy and convergence rate of different surface tracking algorithms. This latter is measured by computing the L2-norm of the displacement error, considering the exact coordinates that the fluid vertices should have at the interface against the coordinates which are obtained with the surface tracking method, that is

$$e_{L2} = \sqrt{\sum_{i=1}^{nf} [y(x_i) - y_i]^2} \tag{9}$$

where $y(x_i)$ is the y-axis component of the exact position and y_i is the corresponding approximate value. The initial position of the interface for both the structure and the fluid boundaries is given by the following sinusoidal law

$$y_0(x) = 0.5 \sin(2\pi x) \tag{10}$$

where $x \in [-0.5, 0.5]$. The fluid side of the interface is subdivided into 2559 intervals of equal length. On the other hand, the structure side of the interface is also subdivided into intervals of equal length, but the number of points increases as 20, 40, 80, 160 and 320 in order to explore the convergence of the method. Then, the structure side of the interface is displaced according to the following law

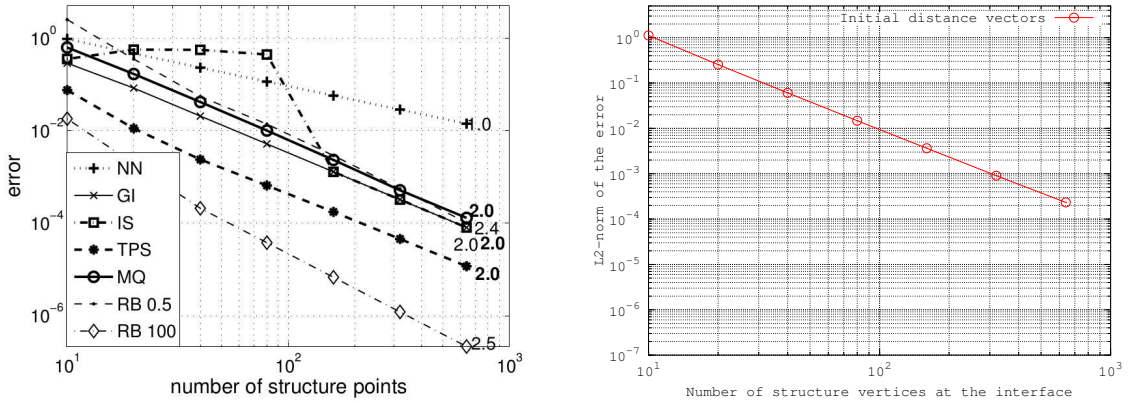
$$dy(x) = 0.05 \cos(2\pi x) \tag{11}$$

so the final coordinates of the vertices on the structure side are given by $y(x) = y_0(x) + dy(x)$. Figure 10.(a) is extracted from de Boer et al. (2007) and it is taken as reference to analyze the implemented method. On the other hand, figure (10.b) shows the results computed with the tracking method based on the initial distance vectors algorithm implemented in this work. It can be seen that the method has second order convergence rate and it is as accurate as those based on the Gauss Integration technique and Multi-quadratic Bihamornic Splines (denoted as GI and MQ in figure (10.a), respectively).

Finally, for the surface tracking strategy to satisfy the *consistency* property, it should be considered that a constant displacement applied to the structure interface is exactly recovered at the fluid interface after the surface tracking is applied (de Boer et al., 2006). To this end, it is considered the same initial geometry for the structure interface that was used in the previous test, given by eq.(10). Then, $\forall x \in \Gamma_S$ a constant displacement $dy(x) = 0.1$ is applied, where Γ_S is the structure interface. Then, the surface tracking is applied and the fluid interface is displaced accordingly. Thereafter, the following values are computed, $|0.1 - \max_{\forall x \in \Gamma_F}(dy(x))| = 3.572e-5$ and $|0.1 - \min_{\forall x \in \Gamma_F}(dy(x))| = 1.045e-6$, where Γ_F is the fluid interface. Based on these values it can be concluded that the constant displacements of the structure interface have been exactly transferred to the fluid interface and therefore the surface tracking strategy of initial distance vectors satifies the consistency property.

4 CONCLUSIONS

On regards to the conservative state projection method introduced in section 2 in this work, based on the results obtained in the numerical tests it can be concluded that this scheme is much more accurate than a direct interpolation scheme (NN) or a projected interpolation scheme (NN



(a) Results for the methods considered in de Boer et al. (2007) - Figure taken from the same reference.

(b) Results computed with the initial distance vectors tracking.

Figure 10: Interpolation error and interpolation order of convergence for surface tracking test 4.

proj) if what matters is to preserve the total load transfer from the fluid boundary mesh to the structure boundary mesh. This behaviour is even more notorious when the solution on the fluid side exhibits jump discontinuities or steep gradients. The meshes used in the tests are very simple but all of them are non-matching at the interface so that the method presented in this work provides a great flexibility for computing the solution of FSI problems if compared to the solution schemes that require matching meshes. However, care should be taken since it is shown in the results corresponding to the projection of the non-smooth solution that the conservative method introduces spurious oscillations due to the lack of monotonicity of the numerical formulation. Monotonicity preserving methods based on the Flux-Corrected Transport (FCT) technique (Kuzmin et al., 2012) are the following step to be analyzed and implemented.

On the other hand, based on the results obtained for the interface tracking method, it can be concluded that the algorithm works as expected, considering that the structure and the fluid meshes do not have a common interface. Also, the accuracy of the method, based on the benchmark proposed in de Boer et al. (2007) is similar to that based on Multi-quadratic Bihamornic Splines and has second order convergence rate. Finally, the tracking strategy satisfies the consistency property.

ACKNOWLEDGMENTS

This work has received financial support from Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET, Argentina, grant PIP 11220150100588CO), Universidad Nacional del Litoral (UNL, Argentina, grant CAI+D 2011-01-00012-LI, CAI+D-501-201101-00233-LI), Agencia Nacional de Promoción Científica y Tecnológica (ANPCyT, Argentina, grants PICT 2660–14, PICT-E 0191–14, PICT 0938–13, PICT-2015-2904), Secretaría de Ciencia, Tecnología y Producción para la Defensa (grant PIDDEF-4/14), Agencia Santafesina de Ciencia, Tecnología e Innovación (ASACTEI, Argentina, grant 00010-18-2014).

This work was performed with *Free Software Foundation/GNU-Project* resources like GNU–Linux OS, GNU–GFortran, GNU–Octave, GNU–Git and GNU–GIMP, as well as other Open Source resources such as NETGEN, Para-View, Xfig and LaTeX.

REFERENCES

Arya S. and Mount D. *ANN: Approximate Nearest Neighbors Library*. Department of Computer Science, University of Maryland, 2010.

Cebral J. *Loose Coupling Algorithms for Fluid-Structure Interaction*. PhD Thesis, Institute for Computational Sciences and Informatics, George Mason University, 1996.

Cebral J. and Löhner R. Conservative load projection and tracking for fluid-structure problems. *AIAA*, 35(4):687–692, 1997.

de Boer A. *Computational fluid-structure interaction. Spatial coupling, coupling shell and mesh deformation*. Ph.D. thesis, Technische Universiteit Delft, 2008.

de Boer A., van Zuijlen A., and Bijl H. Comparison of the conservative and a consistent approach for the coupling of non-matching meshes. In *ECCOMAS CFD*. TU Delft, 2006.

de Boer A., van Zuijlen A., and Bijl H. Review of coupling methods for non-matching meshes. *Comput. Methods Appl. Mech. Engrg.*, pages 1515–1525, 2007. doi:10.1016/j.cma.2006.03.017.

Gatzhammer B. *Efficient and Flexible Partitioned Simulation of Fluid-Structure Interactions*. PhD Thesis, Technische Universität München, 2014.

Kuzmin D., Turek S., and Löhner R., editors. *Flux-Corrected Transport: Principles, Algorithms and Applications*. Springer Dordrecht Heidelberg, second edition, 2012. ISBN 978-94-007-4037-2.

Schauer M., Langer S., Roman J., and Quintana-Ortí E. Large scale simulation of wave propagation in soils interacting with structures using FEM and SBFEM. *Journal of Computational Acoustics*, 19(01):75–93, 2011. doi:10.1142/S0218396X11004316.