Evaluation of a coupling interface for solving fluid structure interaction problems

Luciano Garelli, Marco Schauer, Gustavo Ríos Rodriguez, Sabine C. Langer, Mario A. Storti

PII: S0997-7546(15)30057-1
DOI: http://dx.doi.org/10.1016/j.euromechflu.2016.04.001
Reference: EJMFLU 3003

To appear in: *European Journal of Mechanics B/Fluids*

Received date: 21 April 2015
Revised date: 1 December 2015
Accepted date: 1 April 2016

Please cite this article as: L. Garelli, M. Schauer, G.R. Rodriguez, S.C. Langer, M.A. Storti, Evaluation of a coupling interface for solving fluid structure interaction problems, *European Journal of Mechanics B/Fluids* (2016), http://dx.doi.org/10.1016/j.euromechflu.2016.04.001

Evaluation of a coupling interface for solving fluid structure interaction problems

Luciano Garelli[*,a], Marco Schauer[b], Gustavo Ríos Rodriguez[a], Sabine C. Langer[c], Mario A. Storti[a]

[a] *Centro de Investigaciones de Métodos Computacionales,*
CIMEC (UNL - CONICET), 3000 Santa Fe, Argentina.
Tel.: +54 (0) 342 4511594 Fax: +54 (0) 342 4511595
[b]*Technische Universität Braunschweig - Institut für Statik,*
Beethovenstraße 51, 38106 Braunschweig, Germany.
Tel.: +49 (0) 531 391 7108
[c]*Technische Universität Braunschweig - Institut für Konstruktionstechnik,*
Langer Kamp 8, 38106 Braunschweig, Germany.
Tel.: +49 (0) 531 391 7101

Abstract

This research work evaluates the performance of a Fluid Structure Interaction (FSI) solver, which is created using a generic interface to couple two independent software packages. The basic idea is to combine the advantages of the two independent codes to create a powerful FSI solver for two and three dimensional FSI analysis using the concept of modular programming. A detailed description about the implementation of an interface to couple a three-field system involved in the analysis is given, and this developed interface can be generalized to others codes. Since solving complex FSI problems is very time consuming, the focus of this work is placed on the performance of the coupled solver, for which a FSI benchmark will be solved on a computer cluster in order to measure speed up and efficiency.

Key words: Scalability, Fluid Structure Interaction, Partitioned Coupling.

[*]Centro de Investigaciones de Métodos Computacionales,
CIMEC (UNL - CONICET), 3000 Santa Fe, Argentina.
Tel.: +54 (0) 342 4511594 Fax: +54 (0) 342 4511595

Email addresses: lgarelli@intec.unl.edu.ar (Luciano Garelli), m.schauer@tu-braunschweig.de (Marco Schauer)

Preprint submitted to The European Journal of Mechanics - B/Fluids *December 1, 2015*

1. Introduction

Nowadays, the design of many engineering systems has to consider fluid structure interactions, for instance aircraft, turbines and bridges, but also medical products like artificial heart implants. If effects of oscillatory interactions of fluid and structure is not considered during the design process, it can lead into catastrophic failure of the system. One of the probably most referenced examples of large-scale failure is the Tacoma Narrows Bridge which collapsed in 1940. In aircraft wings and turbine blades when the FSI is strong enough can produce the instability system, which is known as flutter. Other important areas in which FSI plays a fundamental role are, for instance, in biomedical engineering, where the pulsating blood flow can cause the rupture of an abdominal aortic (AAA) or cerebral aneurysm, implying a great risk for the patient. In all these examples it is very difficult to determine a priori the effects of the fluid over the structure. When the interaction is known to be strong enough to produce important deformation on the structure, intensives experimental tests has to be carried out, which are very expensive and time consuming. Numerical solution of different FSI problems can be found in [5, 11, 12, 17, 33].

In this context is where the possibility to perform a numerical simulation of the whole system has a significant importance, allowing to test a wide number of design parameters and having as result a detailed description of the fluid (velocities, pressure, turbulence intensity, etc.) and the structure (stress, strain, displacements, etc.). But the governing equations of the fluid and the structure has to be coupled in some way, one of the possibilities is to combine in a single formulation the fluid and the structural governing equations as it was done in the works of [14] and [19], but this monolithic scheme, in some cases, can be mathematically unmanageable or its implementation can be a laborious task. Furthermore, the monolithic coupled formulation would change significantly if other fluid and/or structure models were considered.

Another possibility is to use specialized codes in order to obtain reliable predictions in FSI problems, considering a staggered fluid-structure coupling algorithm based on a modular programming technique. But the coupling of two different codes is a challenging task because it implies the design of an interface in order to carry out the exchange of information and the synchronization of these codes. Also, when using this approach, the resulting large system of non-linear equations can be solved using (iterative) specifics solvers for each subsystem. Usually, this is done with Block-Jacobi, Block-Gauss-Seidel or related relaxation methods [3, 8]. The solution of this subsystem requires high among of computational power and it is required that both codes are prepared for parallel computation on distributed memory systems, like computer clusters. In this work, in order to evaluate the developed interface and the capabilities of the modular programming, the fluid part and the mesh movement are computed by the solver *PETSc-FEM* which is a software being developed at *Centro de Investigaciones de Métodos Computacionales (CIMEC), Santa Fe, Argentina* [37], hereafter called CFD (Computational Fluid Dynamics) and CMD (Computational Mesh Dynamics) respectively. It is a general purpose, parallel, multi-physics finite element method (FEM) program for CFD applications based on PETSc [42]. It is written in the C++ language with an object oriented programming (OOP) philosophy, keeping in mind the scope of efficiency. CFD and CMD may run in parallel using the MPI standard on a variety of architectures and comprises both a library that allows the user to develop FEM (or FEM-like, i.e. non-structured mesh oriented) programs, and a suite of application programs. These suite of applications allows to deal with compressible [45] and incompressible [39] fluid flows with Streamline Upwind Petrov-Galerkin (SUPG) and for Pressure Stabilizing Petrov-Galerkin (PSPG) stabilization techniques, shallow water [45], free surface [46] and electrokinetic flows [43, 44]. Additionally, these programs can be coupled with mesh refinement [47] and mesh relocation algorithms [40, 41] to improve the quality of the solution or to use when solving FSI problems [11, 15, 33]. The problems can be solved using structured and non-structured mesh with a wide variety of elements types (i.e. hexaedral, tetrahedral, pyramid, wedges, triangle, quadrangle) and boundary conditions (i.e. pressure inlet/outlet, velocity inlet, mass flow inlet, absorbing [45]).

The structural part is computed by the research in-house-code *elementary Parallel Solver* – ELPASO of the *Institut für Konstruktionstechnik, Technische Universität Braunschweig* [34] , hereafter called CSD (Computational Structure Dynamics). It is a FEM based research code written in C++ with an OOP philosophy, which may run in parallel using PETSc and the MPI standard on various architectures. It has been under constant development for many years. It is designed for structural analysis and has been applied to different types of problems in time and frequency domain like structural analysis, eigenfrequency analysis, acoustics, soil-structure interaction and topological optimization [26, 27, 28, 29, 34]. To address these very different types of problems various material models (linear-elastic, visko-elastic, poro-elastic) and 1D (spring and beam), 2D (disc, plate and shell) and 3D (tetrahedron and hexahedron) elements using linear or higher oder shape functions are implemented. Hence for some problems in acoustic analysis and soil-structure integration it is necessary to consider an infinite free-field or infinite half-space, boundary element method (BEM) and scaled boundary finite element method (SBFEM) are implemented as well, since this methods are able to fulfill Sommerfelds radiation condition exaclty.

Due to the high time consuming when solving complex FSI problems, the focus of this work is placed on the performance of the coupled solver, for which a FSI benchmark will be solved on a computer cluster, with an increasing number of degree of freedom (DoF), evaluating in each case the strong scalability and the efficiency.

2. Coupling strategy

The coupling process between the CFD, CMD and CSD codes has been carried out using a partitioned technique [15, 16, 48]. When such kind of procedure is used, a three-field system is involved in the analysis: the structure, the fluid and the moving mesh solvers. The mesh movement is performed by using a nodal relocation, maintaining the topology unchanged. There are several strategies to perform the nodal relocation of the mesh, for instance, a linear elastic or pseudo-elastic problem is used in order to propagate the boundary motion into the volume mesh, obtaining thereby the mesh deformation. By solving an optimization problem, a mesh with higher quality can be obtained [18]. Another mesh moving technique can be found in [31].

In order to solve FSI problems the fluid dynamic equations must be rewritten in an Arbitrary Lagrangian Eulerian (ALE) framework. Here a geometric conservation law (GCL) compliant scheme based on an Averaged ALE Jacobians Formulation (AJF) [36] is used, which is a new formulation applied to the theta-family of time integration methods and to the three-point Backward Difference Formula (BDF). This scheme was developed in order to satisfy the discrete version of the GCL without needed of change the time integration scheme and with very little modification at elementary level. Several researchers has shown that satisfying the GCL in its discrete version (DGCL) is neither necessary nor sufficient condition for an ALE scheme to preserve on moving grids its time-accuracy established on fixed grids and there is a general consensus to use schemes that satisfy the DGCL when solving FSI problems.

The interaction process is carried out through the exchange of information at the fluid/structure interface in a staggered way. In broad terms, the structural solver establishes the position and velocity of the interface, while the fluid solver provides the pressure and shear stresses on the interface. The principal advantage of the partitioned treatment is that existing optimized solvers can be reused and coupled. Furthermore, the systems to be solved are smaller and better conditioned than in the monolithic case. However, this approach requires a careful implementation to avoid serious degradation of the stability and accuracy. From this approach, either a weak (explicit) or a strong (implicit) time coupling scheme can be developed. Next, these schemes will be described in detail.

During the iterative process three computer codes, CFD, CSD and CMD are run synchronously to complete the following procedure:

i) Transfer the motion of the wet boundary (interface) from solid to fluid problem.

ii) Update the position of the boundary and bulk fluid meshes accordingly.

iii) Advance the fluid flow solution to update the pressures at the interface.

iv) Convert the new fluid pressures (and stress field) into a structural load.

v) Advance the solution of the structural system with updated structural loads.

Taking into account the previous description, two different coupling schemes can be derived depending on how the prediction of the structural displacement for updating the position of the fluid boundary and the computation of the updated pressures are done. To proceed with the explanation of the scheme, the following variables are defined: $\mathbf{w}^n$ is the fluid state vector $(\rho, \mathbf{v}, p)$, $\mathbf{z}^n$ is the vector with the nodal displacement of the structure, $\dot{\mathbf{z}}^n$ is the structure velocity vector and $\mathbf{x}^n$ is a vector with the fluid mesh node positions, at time t_n.

In the weak (explicit) coupling the fluid flow solution is advanced first, using the previously computed structure state $\mathbf{z}^n$ and a current estimated value $\mathbf{z}_p^{n+1}$. In this way, a new solution for the fluid state $\mathbf{w}^{n+1}$ is computed. Next, the structure is updated using the forces of the fluid from states $\mathbf{w}^n$ and $\mathbf{w}^{n+1}$.

The state $\mathbf{z}_p^{n+1}$ is predicted by using a second or higher order approximation given by,

$$\mathbf{z}_p^{(n+1)} = \mathbf{z}^n + \alpha_0 \Delta t \dot{\mathbf{z}}^n + \alpha_1 \Delta t \left(\dot{\mathbf{z}}^n - \dot{\mathbf{z}}^{n-1} \right). \tag{1}$$

were α_0 and α_1 are two real constants. The predictor (1) is trivial if $\alpha_0 = \alpha_1 = 0$, first-order time-accurate if $\alpha_0 = 1$ and second-order time-accurate if $\alpha_0 = 1$ and $\alpha_1 = 1/2$. This coupling scheme has been proposed in [9, 25], with good results in the resolution of aeroelastic problems.

Once the coordinates of the structure are known, those of the fluid mesh nodes are computed with a CMD code, which is symbolized as

$$\mathbf{x}^{n+1} = \mathrm{CMD}\left(\mathbf{z}^{n+1}\right). \tag{2}$$

Alternatively, the strong (implicit) coupling can be adopted [49, 50, 51], which has benefits in terms of stability and is comparable with a monolithic coupling. In this algorithm, the time step loop is equipped with an additional inner loop called "stage", thus if the "stage loop" converges the monolithic solutions is obtained. When using the strong coupling strategy [52, 53] the computational cost increases proportionally to the number of stages needed to achieve the desired error between the states, but large time steps can be used. A more detailed description about the implemented algorithm can be found in [15].

If a slip boundary condition is applied to the interface, at the beginning of each fluid stage a computation of skin normals and velocities are performed. This is necessary because the geometry of the interface Γ is time dependent, whereby

$$\left((\mathbf{v}|_\Gamma - \dot{\mathbf{z}}|_\Gamma)\cdot\hat{\mathbf{n}} = 0\right), \tag{3}$$

If a non-slip boundary condition is used, the interface has the velocity of the moving solid wall

$$\mathbf{v}|_\Gamma = \dot{\mathbf{z}}|_\Gamma. \tag{4}$$

The load vector $\mathbf{p}$ applied to the structure is updated at each time step $\mathbf{n}$. It is composed by the sum of predefined loads applied on the structure $\mathbf{p}_S$ and forces acting on the structure due to the surrounding pressure field of the fluid $\mathbf{p}_F$.

$$\mathbf{p}_n = \mathbf{p}_S + \mathbf{p}_F\,. \tag{5}$$

2.1. Limitations of the coupling strategy

As mentioned above the weak coupling partitioned scheme is more efficient, since it only requires one solution of each subsystem per time step. However the employment of sequentially staggered schemes in FSI problems, where incompressible flows are considered, yields inherent instability [10]. This instability can be overcome by using iteratively staggered partitioned schemes (strong coupling scheme). Because of the high complexity and non-linearity of FSI problems, only few mathematical explanations of the stability or convergence conditions are available (see [6, 10]). In general it has been stated that the convergence depends on the fluid and structure densities and on the geometry of the domain, i.e. when the structure density overtakes a certain threshold or when the domain length is greater than a certain value instability occurs. These unstable computations are mostly due to the so-called *artificial added mass effect*, since the fluids acts as an extra mass on the structural degrees of freedom at the coupling interface. In the weak coupling the fluid forces depend upon the structure displacements and contain thereby incorrect coupling forces. In [23] a threshold for the time step size for staggered analysis on acoustic FSI depending on the ratio of the structural and fluid mass has been obtained. In the incompressible case neither the time step size nor the accuracy influence the instability.

In this work the instability coming from the added mass effect in the weak coupling is avoided because the structure density is greater than the fluid density, whereby a stable scheme is obtained.

2.2. *Interface to manage the bi-directional information transfer*

The coupling of structure and fluid is arranged by an interface. Therefore both computational codes CFD and CSD are executed simultaneously. The interface ensures the bi-directional information transfer between these codes. The structural solver applies nodal forces generated by the surrounding fluid to the structure and returns velocities and displacements to the fluid and mesh movement solvers. Different input-files are to be provided to the software by the user in order to perform the FSI analysis. CFD and CMD are used to solve the fluid dynamic and the mesh movement problem, thus two files, one containing the nodal coordinates and one containing the connectivity of the mesh are needed. Additionally, a file with the setup of the boundaries conditions has to be generated, in the case of the fluid problem velocity inlet, pressure inlet/outlet, wall (slip or non-slip) can be used and for the mesh move problem has to be defined the fixed and one moving boundary in order to be coupled with structure. A more detailed description can be found in [37]. CSD is used to solve the structural dynamic, thus a file containing the nodal coordinates, element connectivity, material parameters and also the boundaries conditions and loads is needed. The other group of files are specific to the coupling interface and has information about the moving boundary. One of these file include a node mapping table (fluid node - structure node) at the moving boundary in order to exchange the information and the other two files include the pressure at the moving boundary to be transferred to the structure and the displacement of the moving boundary to be transferred to the mesh move solver.

In order to permit the information transfer in both directions four independent pipes are created. The concept of named pipe or FIFO – first in first out – is used to let the communication and consequently the data transfer between the different codes happen. Named pipes are an extension to the traditional pipe concept on Unix and Unix-like systems, it is one of the methods of inter-process communication. Instead of a conventional, unnamed, shell pipeline, a named pipeline makes use of the file system. It is explicitly created to allow two separate processes to access the pipe by name, thereby one process can open it as a reader, and the other as a writer. To use named pipes for the communication, the communicating parts of the two programs have to be executed on the same physical machine. Here the communication is done by the first MPI process *rank0*. It has to be ensured, that the instances of *rank0* are located in the same physical computer and share the same physical memory. Otherwise the communication via named pipe will fail.

In the present case, the four different pipes which permit the bi-directional coupling are named *adv2str.fifo*, *str2adv.fifo*, *adv2mmv.fifo* and *mmv2adv.fifo*. The data transfers of the nodal forces from CFD to CSD is controlled by the first pipe *adv2str.fifo*, the transfer of nodal velocities and displacements from CSD to CFD is controlled by the the second pipe *str2adv.fifo*. In order to synchronize the CFD code with the CMD code the pipes *adv2mmv.fifo* and *mmv2adv.fifo* are used.

Coupled time domain analysis CSD

```
Data: timestep $\Delta t$, CFD data
Result: timestep $\Delta t$, CSD data
initialize interface to communicate with CFD;
open file adv2str.fifo to read on rank0;
open file str2adv.fifo to write on rank0;
for $n = 1, 2, \ldots, N_{steps}$ do
    send structure state $-1$ massage to str2adv.fifo;
    if adv2str.fifo traces fluid state $\Delta t$ then
        receive fluid forces $\mathbf{p}$;
    end
    perform $n$-th time step for $t = (n + 1)\Delta t$;
    send structure state $\Delta t$ massage to str2adv.fifo;
    if adv2str.fifo traces fluid state $-1$ then
        send nodal velocities $\mathbf{v}$ and displacements $\mathbf{d}$;
    end
end
close pipes and destroy interface
```

Algorithm 1: Linking from CSD to CFD

Algorithm 1 summarizes the coupling procedure of both codes from CSD view to perform the coupled analysis of fluid and structure and link CSD to CFD.

If a coupled analysis is going to be carried out, an interface is initialized, which opens two FIFOs on *rank0* only. When time stepping scheme is performed, the interface is attached to it to organize the communication and synchronization of the two solving processes. The synchronization is organized by sending the current state of the computational processes. After the CSD code receives new nodal forces from CFD the structure state variable is set to -1 and the next time step is performed. As the structural computation is done the structure state variable is set to the current time step Δt and velocities **v** and displacements **d** are send to CFD. This procedure is running until the last time step has been computed. Afterwards CSD closes the pipes, deletes the interface and free the memory to exit the process properly.

Coupled time domain analysis CFD & CMD

```
Data: timestep $\Delta t$, CSD data
Result: timestep $\Delta t$, CFD data
initialize interface to communicate with ELPASO ;
open file str2adv.fifo to read on rank0;
open file adv2str.fifo to write on rank0;
open file adv2mmv.fifo and mmv2adv.fifo on rank0;
for $n = 1, 2, \dots, N_{steps}$ do
    send fluid state $-1$ massage to adv2str.fifo;
    if str2adv.fifo traces structure state $\Delta t$ then
        receive velocities $\mathbf{v}$ and displacements $\mathbf{d}$;
    end
    run CMD and update the moved mesh (using adv2mmv.fifo and mmv2adv.fifo);
    perform $n$-th time step for $t = (n + 1)\Delta t$;
    send fluid state $\Delta t$ massage to adv2str.fifo;
    if str2adv.fifo traces structure state $-1$ then
        send nodal fluid forces $\mathbf{p}$;
    end
end
close pipes and destroy interface
```

Algorithm 2: Linking from CFD to CSD

Algorithm 2 shows the steps to perform on the opposite side by CFD. Here everything acts in the opposite way - velocities and displacements are received and forces are send via the already mentioned pipes. In this particular case the mesh movement has been performed within CFD, hence the CMD is called internally.

3. Validation of fluid and structure solvers

Prior to validate and evaluate the parallel performance of the coupled algorithm, both solvers have to be validated individually.

3.1. Fluid solver validation

To evaluate the overall performance of CFD solver, the fluid flow around a bluff body (square cylinder) with a thin plate aligned to the flow direction, which is attached to its backward face is solved. The problem assumes an incompressible Newtonian flow, hence the following system of equations is solved with the Navier-Stokes module,

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t}+\mathbf{u}\nabla\mathbf{u}\right)-\nabla\boldsymbol{\sigma}=0 \quad \text{in}\,\Omega\times(0,T)\ , \tag{6}$$

and

$$\nabla\cdot\mathbf{u}=0 \quad \text{in}\,\Omega\times(0,T) \tag{7}$$

with ρ and $\mathbf{u}$ as the density and velocity of the fluid respectively, while $\boldsymbol{\sigma}$ is the stress tensor given by

$$\boldsymbol{\sigma}=-p\mathbf{I}+2\mu\varepsilon(\mathbf{u}) \tag{8}$$

$$\varepsilon(\mathbf{u})=\frac{1}{2}\left(\nabla\mathbf{u}+(\nabla\mathbf{u})^T\right), \tag{9}$$

where p and μ are the pressure and dynamic viscosity, respectively, and $\mathbf{I}$ represents the identity tensor. Initial and boundary conditions must be also defined. Details about the discretization of the Navier-Stokes equations using the Finite Element Method (FEM) and its implementation are described in [35, 37]. A sketch of the problem to with geometrical dimensions and boundary condition is shown in Fig. 1. The same problem, but using an elastic cantilever beam, was first proposed by [32] and then by [13] as an FSI benchmark.

The material properties Density and viscosity of the fluid are given by $\rho = 1.18\times10^{-3}$ $[\mathrm{g/cm^3}]$ and $\mu = 1.82\times10^{-4}$ $[\mathrm{g/cm\ s}]$, respectively. The boundary conditions are no-slip on the bluff body and plate surfaces, slip at the top and bottom surfaces of the domain and a reference pressure is set at the outlet. Also, a normal velocity condition is set at the inlet, with a prescribed constant value $V_x = 31.5$ $[\mathrm{cm/s}]$. Hence a Reynolds number of $Re = 204$ is obtained if the length of the square cylinder is used as reference. In this fluid dynamic problem, a

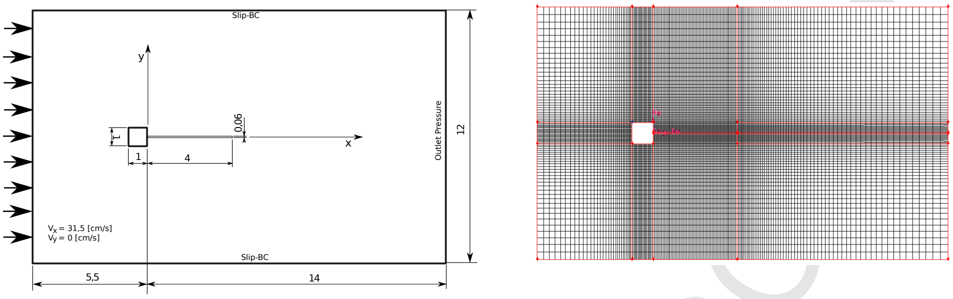

Figure 1: Geometrical description (left) and spatial discretization. Fluid Mesh fl_M1 (right).

vortex shedding is periodically released from the edges of the square. The vortex shedding frequency reported by [13] is $f_F = 3.7$ [Hz], whereby the Strouhal number referenced to the length of the square is $St = 0.117$. Also, different flow regimes and Strouhal numbers as a function of the relation between the splitter plate length and the length of the square edge are reported in [1]. For the particular case under analysis a $St \approx 0.12$ is measured. Another important parameter to verify the accuracy of the results is the mean drag coefficient (Cd_{mean}), which is reported in [1] for different mesh sizes and temporal schemes, ranging between $1.406 \leq Cd_{\text{mean}} \leq 1.507$.

Table 1: Number of nodes, elements, and DoF of the fluid meshes 1 to 4.

Fluid Mesh	fl_M1	fl_M2	fl_M3	fl_M4
# nodes	13672	34262	67045	207410
# elements	13320	33700	66446	206092
# DoF	40027	101217	198719	618669

In order to solve the numerical problem the fluid domain is discretized using quadrilateral elements and four different structured meshes are generated. to evaluate the mesh convergence and later the strong scalability of the software. The number of nodes and elements for the different meshes are detailed in Table 1. The time step adopted was $\Delta t = 0.002$ [s] for a Crank-Nicolson time integration scheme. Two Newton loops were used for the non-linear problem convergence and an iterative Domain Decomposition Method (DDM) with an interface strip preconditioner was used [24].

The results computed with the different meshes for (f_F, St, Cd_{mean}) are summarized in Table 2. To get the vortex shedding frequency a Fourier transforma-

Table 2: Fluid dynamic results.

Fluid Mesh	fl_M1	fl_M2	fl_M3	fl_M4
Vortex shedding freq. (f_F)	3.86	3.79	3.75	3.73
Strouhal Number (St)	0.122	0.120	0.119	0.118
Cd_{mean}	1.508	1.469	1.453	1.422

tion was applied to the lift force. The result of the transformation is plotted in Fig. 2, which shows that they are in very good agreement with the values reported by [1, 13].

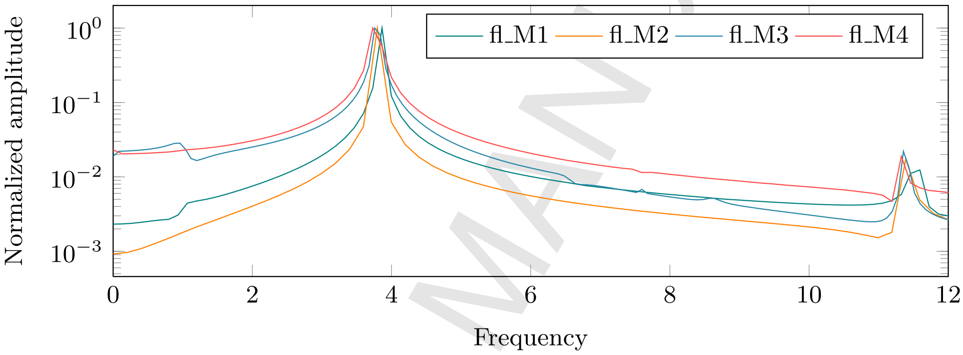

Figure 2: Fourier analysis of the lift force.

3.2. *Structural solver validation*

ELPASO is a FEM-based research code that provides the structural part of the coupled computation. The structure is represented by finite elements for the time domain analysis. At an arbitrary time step the displacement-based finite element method can be written as

$$\mathbf{Ma} + \mathbf{Cv} + \mathbf{Kd} = \mathbf{p} \tag{10}$$

where vector $\mathbf{d}$ represents the nodal displacements, $\mathbf{v}$ the nodal velocities, $\mathbf{a}$ the nodal accelerations and $\mathbf{p}$ denotes the applied nodal forces. Here, $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ is the damping matrix and $\mathbf{K}$ denotes the stiffness matrix. Rayleigh damping is introduced by setting up the damping matrix as a linear combination of the stiffness and mass matrices

$$\mathbf{C} = c_{\mathrm{K}}\mathbf{K} + c_{\mathrm{M}}\mathbf{M}, \tag{11}$$

where c_{K} defines the structure equivalent damping c_{M} the mass equivalent damping.

Consider the time period T divided into n time steps with duration $\Delta t = \frac{T}{n}$ and known initial values for $t = 0$

$$\begin{aligned} \mathbf{d}\left(t=0\right) &= \mathbf{d}_0 \quad \text{and} && (12)\\ \mathbf{v}\left(t=0\right) &= \mathbf{v}_0 && (13) \end{aligned}$$

The equation of motion can be solved using the Generalized-α scheme [7]. The state variables of the next time step are evaluated with

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \Delta t \mathbf{v}_n + \left(\frac{1}{2} - \beta\right) \Delta t^2 \mathbf{a}_n + \beta \Delta t^2 \mathbf{a}_{n+1}\,, \tag{14}$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \left(1-\gamma\right) \Delta t \mathbf{a}_n + \gamma \Delta t \mathbf{a}_{n+1} \tag{15}$$

and

$$\begin{aligned} \left(1-\alpha_{\mathrm{m}}\right) \mathbf{M}\mathbf{a}_{n+1} + \alpha_{\mathrm{m}} \mathbf{M}\mathbf{a}_n &\;+ \\ \left(1-\alpha_{\mathrm{m}}\right) \mathbf{C}\mathbf{v}_{n+1} + \alpha_{\mathrm{m}} \mathbf{C}\mathbf{v}_n &\;+ \\ \left(1-\alpha_{\mathrm{f}}\right) \mathbf{K}\mathbf{d}_{n+1} + \alpha_{\mathrm{f}} \mathbf{K}\mathbf{d}_n &= \mathbf{p}_{n+1} - \alpha_{\mathrm{f}} \Delta t \mathbf{p}_{n+1}\,. \end{aligned} \tag{16}$$

The parameters α_{m}, α_{f}, β, and γ are used to control the method. They should be defined as follows

$$\begin{aligned} \alpha_{\mathrm{m}} &\leq \alpha_{\mathrm{f}} \leq \frac{1}{2} && (17)\\ \beta &\geq \frac{1}{4} + \frac{1}{2}\left(\alpha_{\mathrm{f}} - \alpha_{\mathrm{m}}\right) \quad \text{and} && (18)\\ \gamma &= \frac{1}{2} - \alpha_{\mathrm{m}} + \alpha_{\mathrm{f}}\,. && (19) \end{aligned}$$

If α_{m} and α_{f} are both chosen to be zero, the Generalized-α scheme is equal to Newmarks time integration scheme [22]. Further details about ELPASO can be found in the user manual [4].

The structure solver is validated individually, in a similar way to the fluid flow solver. With this objective in mind, a cantilever beam is discretized with bilinear quadrilateral finite elements. To demonstrate the convergence of the numerical scheme, different meshes with increasing number of nodes are used. Table 3 summarizes relevant data of the meshes: number of nodes, elements and the corresponding number of degrees of freedom.

Table 3: Number of nodes, elements, and DoF of the structure meshes.

Mesh	st_M1	st_M2	st_M3	st_M4	st_M5	st_M6	st_M7	st_M8
# nodes	136	405	804	1340	2010	2807	3744	4806
# elements	67	268	600	1068	1670	2400	3269	4264
# DoF	272	810	1608	2680	4020	5614	7488	9612

The material parameters Young's modulus, Poisson's ratio and density of the elastic beam, are given by $E = 2.0 \times 10^5$ [Pa], $\nu = 0.35$ and $\rho_s = 2000$ [kg/m^3], respectively. Considering the Euler-Bernoulli beam theory with the appropriate boundary conditions (clamped on the left side and free on the right side, which is loaded), the natural frequency (f_n) of the cantilever beam can be determined

$$f_n = \frac{k_n^2}{2\pi}\sqrt{\frac{EI}{\rho_s \cdot A \cdot L}} \tag{20}$$

where $I = 1.8 \times 10^{-14}$ [m^4], A $= 6 \times 10^{-7}$ [m^2], L $= 0.04$ [m] are, respectively, the area moment of inertia, the area and the longitude of the beam. Considering the constant $k_n = \{1.875; 4.694; 7.885\}$, the first three approximated natural frequencies $f_1 = 0.606$ [Hz], $f_2 = 3.798$ [Hz] and $f_3 = 10.717$ [Hz] are obtained.

The numerical computation is carried out in time domain for 1×10^6 time steps with a time step interval $\Delta t = 0.0001$ [s]. Newmarks time integration scheme with $\beta = 0.25$ and $\gamma = 0.5$ is used, so that no numerical damping is applied to the analysed structure. To compare the numerical results with the natural frequencies computed with Eq. (20), a point on the neutral axis, at the very right (free) end of the beam, is observed. Applying a Fourier transformation on this time depending recorded data yields the natural frequencies of the numerical model. The number of time steps and the chosen time step lengths lead to a frequency interval of 0.001 [Hz].

Fig. 3 shows the numerical approximation of the three natural frequencies given by Eq. (20), which matches very well. It is clear that the numerically computed frequencies converge to the given natural frequencies when increasing the

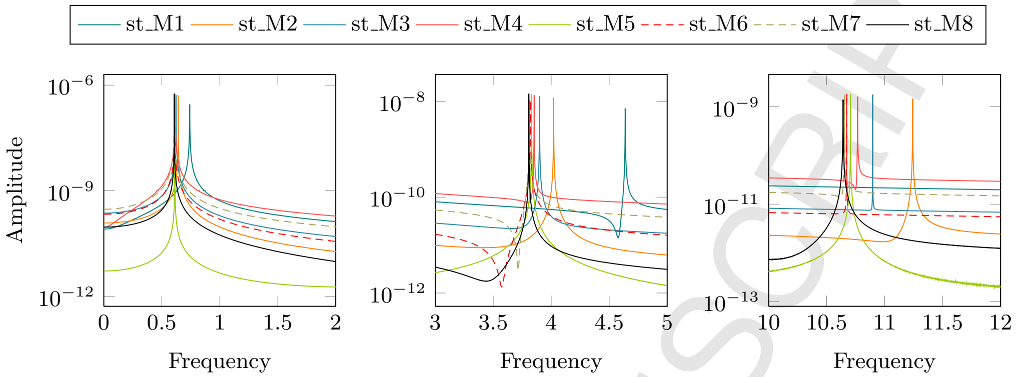

Figure 3: Mesh dependent eigen frequencies $f_1 = 0.606$ [Hz], $f_2 = 3.798$ [Hz] and $f_3 = 10.717$ [Hz] of the cantilever beam.

number of degrees of freedom. For a better survey of the numerical results, the relative errors in the natural frequencies

$$\epsilon_n = \left| \frac{f_n - \tilde{f}_n}{f_n} \right| \tag{21}$$

are summarized in Table 4, where $\tilde{f}_n$ is the nth natural frequency of the numerical model.

Table 4: Mesh dependent computed eigen frequencies and corresponding relative errors ϵ_n.

Mesh	st_M1	st_M2	st_M3	st_M4	st_M5	st_M6	st_M7	st_M8
ω_1 [Hz]	0.741	0.642	0.623	0.615	0.612	0.610	0.609	0.608
ω_2 [Hz]	4.638	4.021	3.899	3.853	3.831	3.819	3.812	3.807
ω_3 [Hz]	12.973	11.242	10.897	10.767	10.706	10.673	10.654	10.642
ϵ_1	22.28%	5.94%	2.81%	1.49%	0.99%	0.66%	0.50%	0.33%
ϵ_2	22.12%	5.87%	2.66%	1.45%	0.87%	0.55%	0.37%	0.24%
ϵ_3	21.05%	4.90%	1.68%	0.47%	0.10%	0.41%	0.59%	0.70%

3.3. Evaluation of the coupled solver

In order to evaluate the numerical solution and the performance of the coupled solver a well-known 2D benchmark with moving interfaces is solved. The benchmark involves the solution of the moving interface, the unsteady fluid flow and structural mechanics. Also, a strategy to deform the mesh is adopted. The test was proposed in [38] in order to evaluate the accuracy and robustness of FSI methods, then several researchers [13, 30] used or adapted this test in different ways. The geometry and boundary conditions are those shown in Fig. 1. The fluid parameters, time integration scheme and time step size for CFD are the same as those described in §3.1. Also, the structure discretization, material parameters and time integration scheme used are the same as those described in §3.2, so that no structural damping is considered.

In this particular case the interface between the fluid and the structure is conformal, so that fluid and structure have the same discretization, wherewith the load from the fluid to the structure and the displacement from the structure to the fluid mesh are directly transferred without any interpolation or tracking. Using a matching interface all the quantities are trivially conserved.

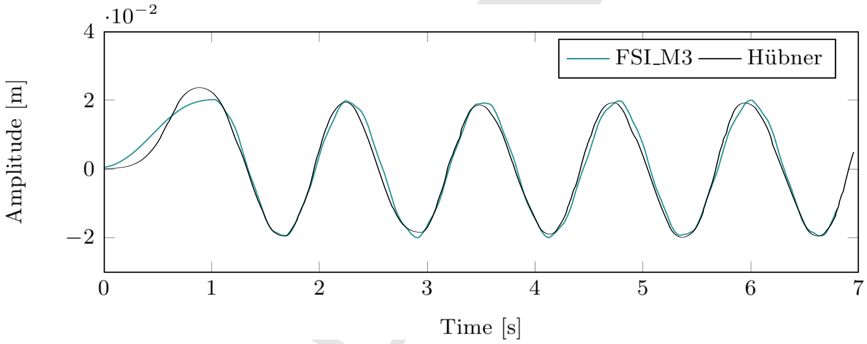

Figure 4: Comparison of tip displacements with [13].

In this simulation the beam starts with an initial deflection $u_y \approx 0.02[m]$, which is produced by a temporary load applied at the tip. Then, the load is removed and the beam starts to oscillate, reaching a periodic stationary solution. The vibration frequency of the coupled system is $f_c = 0.8$ [Hz] with an amplitude at the tip of the beam $u_y \approx 0.02[\mathrm{m}]$ [13].

When the problem is solved using the described coupling algorithm the amplitude at the tip of the beam is $u_y \approx 0.0195$ [m]. The resultant tip displacement is plotted in Fig. 4. Applying a Fourier transformation to the tip displacement yields a frequency of $f_1 = 0.809$ [Hz] $\approx \mathrm{f_c}$ and $f_2 = 5.016$ [Hz], which are in accordance to the results reported in [13], which is solved using a monolithic solver. The results shown were computed with the fluid mesh fl_M3 (see Table 2) and the structure mesh st_M3 (see Table 3).

Also, it is interesting to analyze the pressure around the body to compare it with the results obtained by [13]. In Fig. 5 it is shown a good correlation between the pressure field around the body during the beam deformation, which is very important since this pressure field generates the forces exerted by the fluid onto the structure.

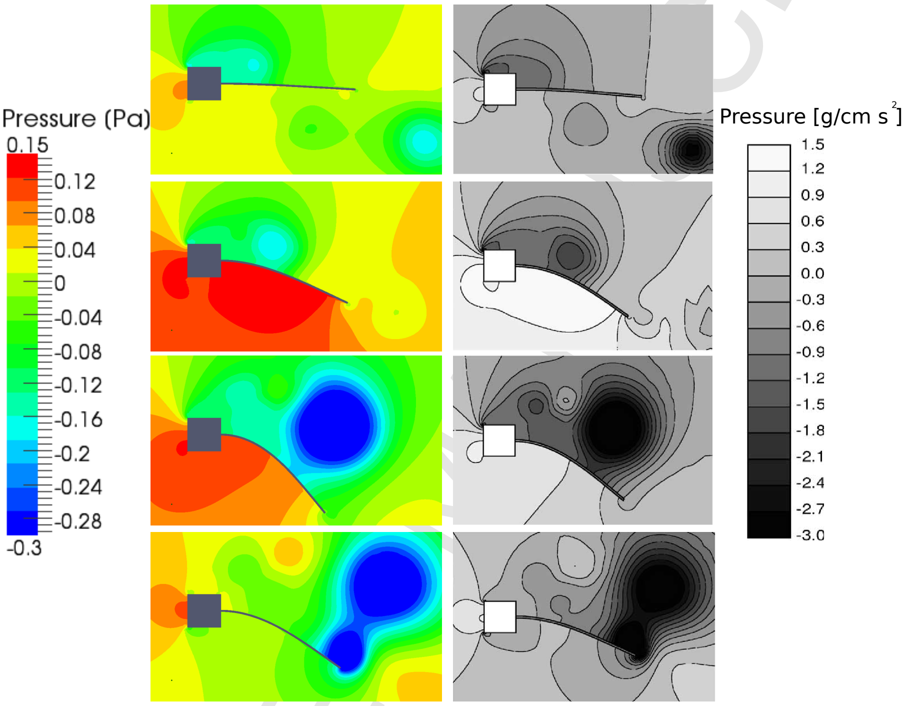

Figure 5: Pressure field around the body. Reprinted figure from [13] with permissions.

After the fluid, structural and coupled solvers are validated, the next step is to measure the performance and scalability of the coupled solver. To carry out these performance measurements, it is chosen to evaluate first the scalability of the fluid solver. This evaluation will be used as reference, since the fluid part dominates the coupled process in the sense that it is the most time consuming.

4. Parallel performance

In a previous work [35] it has been described in detail how CFD solver works and the overall parallel performance of the code, but due to changes in the cluster software and hardware, a new general analysis is accomplished by using a flat

MPI parallelization, i.e. not using hybrid parallelism with OpenMP. With regard to CSD solver a recent work [34] evaluates the performance of the code, whereby it is not needed a new evaluation. Also, the small number of degrees of freedom of the numerical problems used as benchmarks are not suited to do performance measurements and will not lead to reasonable results.

First the parallel performance of the CFD solver is analyzed by carrying out time measurements and then the coupled fluid structure interaction process is evaluated employing the same procedure.

All problems are solved using the cluster "Coyote" from CIMEC, which is of the Beowulf kind, with a server and 8 compute nodes Intel®Xeon®E5-1660 (DDR3-1600Mhz) of 3.3 GHz CPU, with 6 cores and 16 Gb RAM per node, interconnected with a Gigabit Ethernet network.

The performance of each problem is analyzed based on the execution time, being T_s the execution time to solve the problem on 1 node using 1 core and T_p, the parallel execution time with n process. Then the speedup

$$Sp = \frac{T_s}{T_p} \tag{22}$$

and efficiency

$$E = \frac{Sp}{n} = \frac{T_s}{n \cdot T_p} \tag{23}$$

can be computed [2]. First, the speedup and efficiency of the CFD solver are analyzed by – Node Consecutive Filling (NCF) – from 1 to 12 cores. Therefore the programs are executed in one compute node using multiple MPI tasks. For each time measurement the number of MPI tasks is increased by one. After the first node is fully loaded with 6 MPI tasks, a second compute node is added to increase the number of MPI tasks up to 12. The second evaluation – Kernel Consecutive Filling (KCF) – uses 2 compute nodes right from the beginning, and 2 additional MPI tasks are added for each timing, hence each node initializes the same number of tasks.

4.1. Fluid solver performance evaluation

The parallel performance of the fluid solver is evaluated using the meshes described in Table (1). Figure 6 depicts the measured parallel speedup, which reaches at its maximum a factor of 6.4, and its corresponding efficiency for the analyzed meshes. The biggest speedup is archived when 12 cores are addressed.

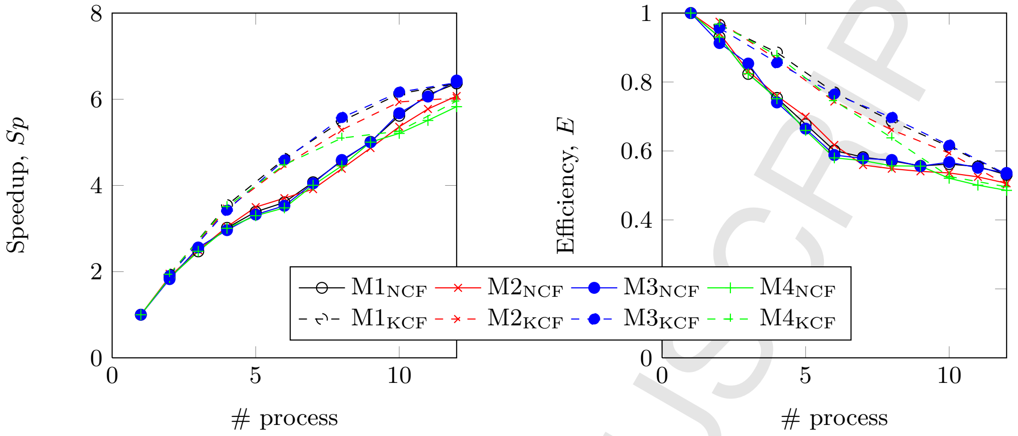

Figure 6: Parallel speedup and efficiency depending on mesh, and initialization of the parallel computation.

With increasing number of initialized MPI tasks the speedup increases as well. Which implies that the computational time decreases at the same time. Due to the parallel overhead introduced by the communication of the processes the efficiency of a single core is decreasing when more cores are added. Hence more communication between the processes is needed, and this additional communication time can not be used to solve the original problem. From the shown graphs it is obvious that the KCF distribution performs better than the NCF initialization. This behaviour arises from the leak of memories bandwidth of the compute nodes. The memory is not able to supply all cores with sufficient data at the same time [20, 21]. Thus the gap between the two initialization procedures – KCF and NCF – appears.

4.2. *Fluid Structure solver performance evaluation*

The evaluation of the parallel performance is carried out using the problem described in §3.3. During the numerical simulation three codes, i.e. CFD, CSD and CMD are executed, whereby the procedure to measure the performance slightly differs to that used in the performance measurement of the fluid solver. The speedup and efficiency of the FSI solver are analysed by NCF from 1 to 12 cores for the fluid and mesh move solvers, while the structure is run in one core only due to the small size of the numerical problem. In each node, a maximum of three MPI for the fluid solver plus three MPI process for the mesh move solver are launched, reaching the maximum number of physical cores of the node. Then, additional nodes are added in order to reach the total of 12 core, using a total of four nodes. In figure 7 the speedup and efficiency for increasing mesh size problems are plotted. When the first two meshes are used the exchange of information and synchronization among the three codes plays an important role and the efficiency starts to decay after using more than 6 core. This problem is overcome when bigger meshes are used, which can be observed when solving the problem with fl_M3 and fl_M4, whereby a weak scalability is achieved.

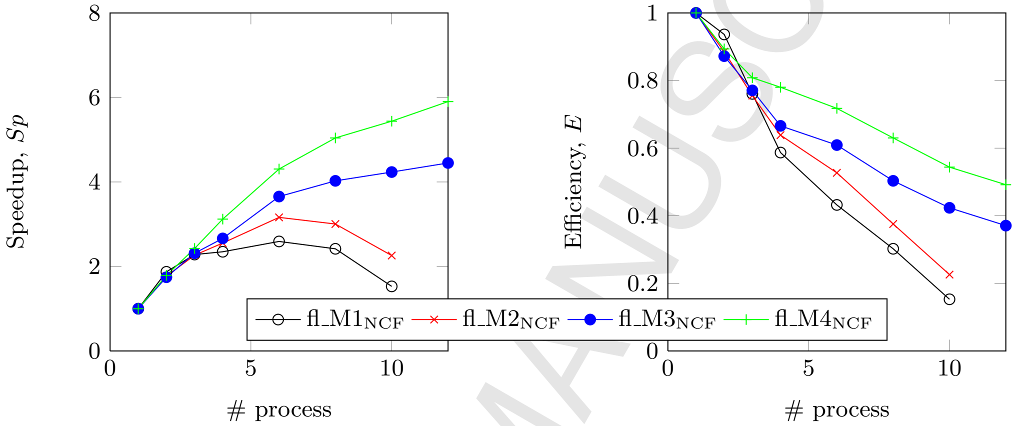

Figure 7: Parallel speedup and efficiency depending on mesh mesh size.

When partitioned schemes are used to couple multiphysics systems, an exchange of information between the codes, as described in the algorithms (1) and (2), is required. This exchange of information and synchronization has an important influence over the possibility to reach a strong scalability.

5. Conclusion

In this research work were presented the algorithm and interfaces used to coupled two independent software packages in order to obtain a FSI solver. The coupling algorithms were presented in §(2.2) and the used strategy was described in detail in §(2). Also, the limitations of this strategy were stated in §(2.1).

Both computer codes (fluid and structural) were individually validated by solving specific benchmark problems. In addition, a mesh convergence analysis was carried out showing good results. In the fluid flow problem the Strouhal Number and mean drag are in very good agreement with the numerical and experimental values reported by several references. On the other hand, the solution obtained with the structural solver was validated against the analytic solution of the problem, also showing good agreement and convergence.

The parallel performance of the solvers were measured separately in order to know the possible speedup and efficiency of the coupled solver, although it is known that the required synchronization among the three solvers (fluid-structure-mesh mover) and the exchange of information will lead to an efficiency decrement, which makes the achievement of strong scalability difficult.

As future work in this collaborative project, it can be mentioned the use of non matching meshes, which will allow to use different spatial discretizations in the fluid and the structure. Also a surface tracking strategy is been developed and implemented in order to manage the movement of the non matching meshes. This new capabilities of the solver will be used to solve a 3D benchmark, in order to validate the results and the performance obtained.

Acknowlegments

This common project work is supported by International Research Staff Exchange Scheme (IRSES), project NumSim PIRSES-GA2009-246977 Marie Curie Actions funded under the 7th Framework Programme of the European Commission. Also this work has received financial support from *Argentinean National Agency for Technological and Scientific Promotion* (ANPCyT, Argentina, grant PICT-1506/2006, PICT-1141/2007, PICT-0270/2008,PICT 2492/2010); Argentinean Council for Scientific Research (CONICET project PIP 112-20111-978) and Universidad Nacional del Litoral, Argentina (grant CAI+D 2009-65/334). Extensive use of freely distributed software such as *GNU/Linux* OS, MPICH, PETSc, Metis, Octave, ParaView, and many others is done in this work.

References

[1] Ali, M.S.M., Doolan, C.J., Wheatley, V.: The sound generated by a square cylinder with a splitter plate at low reynolds number. Journal of Sound and Vibration **330**, 3620–3635 (2011)

[2] Amdahl, G.M.: Validity of the single processor approach to achieving large-scale computing capabilities. AFIPS Conference Proceedings **30**, 483–485 (1967). DOI 10.1145/1465482.1465560

[3] Artlich, S., Mackens, W.: Newton-coupling of fixed point iterations. Numerical Treatment of Coupled Systems (1995)

[4] Beck, S., Clasen, D., Lehmann, L., Rurkowska, K., Schauer, M., Wulkau, M.: ELPASO – manual. Tech. Rep. Revision: 369, Institut für Angewandte Mechanik, Technische Universität Braunschweig (2013)

[5] Bodard, N., Deville, M.: Fluid-structure interaction by the spectral element method. Journal of Scientific Computing **27**(1-3), 123–136 (2006). DOI 10.1007/s10915-005-9031-2.

[6] Causin, P., Gerbeau, J., Nobile, F.: Added-mass effect in the design of partitioned algorithms for fluid-structure problems. Computer Methods in Applied Mechanics and Engineering **194**(42-44), 4506–4527 (2005)

[7] Chung, J., Hulbert, G.M.: A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. Journal of Applied Mechanics **60**, 371–375 (1993). DOI 10.1115/1.2900803

[8] Codina, R., Cervera, M.: Block-iterative algorithms for nonlinear coupled problems. Advanced Computational Methods in Structural mechanics, CIMNE, Barcelona. pp. 115–134 (1995)

[9] Farhat, C., Lesoinne, M., Maman, N.: Mixed explicit/implicit time integration of coupled aeroelastic problems: Three-field formulation, geometric conservation and distributed solution. International Journal for Numerical Methods in Fluids **21:(10)**, 807–835 (1995)

[10] Föster, C., Wall, W.A., Ramm, E.: Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows. Computer Methods in Applied Mechanics and Engineering **196**, (7):1278–1293 (2007)

[11] Garelli, L., Paz, R., Storti, M.: Fluid-structure interaction study of the start-up of a rocket engine nozzle. Computers & Fluids **39:7**, 1208–1218 (2010)

[12] Hron, J., Turek, S.: A monolithic FEM/Multigrid solver for an ALE formulation of fluid-structure interaction with applications in biomechanics. Lecture Notes in Computational Science and Engineering **53**, 146–170 (2006)

[13] Hübner, B., Walhorn, E., Dinkler, D.: A monolithic approach to fluid-structure interaction using space-time fiberboardnite elements. Comput. Methods Appl. Mech. Engrg. **193**, 2087–2104 (2004)

[14] Idelsohn, S.R., Oñate, E., del Pin, F., Calvo, N.: Fluid-structure interaction using the particle finite element method. Computer Methods in Applied Mechanics and Engineering **195:17-18**, 2100–2123 (2006)

[15] Storti, M. and Nigro, N. and Paz, R.R. and Dalcin, L. D.:Strong coupling strategy for fluid structure interaction problems in supersonic regime via fixed point iteration. Journal of Sound and Vibration **30**, 859–877 (2009)

[16] M. Breuer and G. De Nayer and M. Münsch and T. Gallinger and R. Wüchner.:Fluid-structure interaction using a partitioned semi-implicit predictor-corrector coupling scheme for the application of large-eddy simulation. Journal of Fluids and Structures **29**, 107–130 (2012)

[17] Löhner, R., Camelli, F., Baum, J., Soto, O.: Simulation of multiphase blast-structure interaction via coupled cfd and csd codes. AIAA-10-0096 (2010) **10**, 0096 (2010)

[18] Lopez, E., Nigro, N., Storti, M., Toth, J.: A minimal element distortion strategy for computational mesh dynamics. International Journal for Numerical Methods in Engineering. **69:9**, 1898–1929 (2007)

[19] Michler, C., Hulshoff, S.J., van Brummelen, E.H., De Borst, R.: A monolithic approach to fluid-structure interaction. Computers & Fluids **33**, 839–848 (2004)

[20] Moor, S.: Multicore is bad news for supercomputers. Spectrum, IEEE **45**(11), 15 (2007). DOI 10.1109/MSPEC.2008.4659375

[21] Murphy, R.: On the effects of memory latency and bandwidth on supercomputer application performance. IEEE 10th International Symposium on Workload Characterization, 2007. IISWC 2007. pp. 35–43 (2007). DOI 10.1109/IISWC.2007.4362179

[22] Newmark, N.M.: A method of computation for structural dynamics. Journal of Engineering Mechanics, ASCE. **85(EM3)**, 67–94 (1959)

[23] Park, K., Felippa, C., DeRuntz, J.: Stabilization of staggered solution procedures for fluid- structure interaction analysis. AMD **26**, 95–124 (1977)

[24] Paz, R.R., Nigro, N., Storti, M.: On the efficiency and quality of numerical solutions in CFD problems using the interface strip preconditioner for domain decomposition methods. International Journal for Numerical Methods in Fluids **52**, 89–118 (2006). DOI 10.1002/fld.1176

[25] Piperno, S.: Explicit/implicit fluid/structure staggered procedures with a structural predictor and fluid subcycling for 2d inviscid aeroelastic simulations. International Journal for Numerical Methods in Fluids **25(10)**, 1207–1226 (1997)

[26] Beck, S. C.; Langer, S.: Modellierung von strömungsinduzierter Schalleinleitung in poroelastische Materialien Proceedings of DAGA 2012 Deutsche Gesellschaft für Akustike. V. (DEGA), 927–928, (2012).

[27] Beck, Silja C.: Strömungsinduzierter Körperschalleintrag in Strukturen mit porösen Oberflächen Dissertation, Technische Universität Braunschweig, Braunschweiger Schriften zur Mechanik, (2012).

[28] Beck, Silja C.; Langer, Sabine: Flow-induced Sound Radiation from Air-ducting Structures. Proceedings of the International Conference on Acoustics NAG/DAGA Nederlands Akoestisch Genootschap (NAG) und Deutsche Gesellschaft für Akustik e. V. (DEGA), 1496–1498 (2009).

[29] Schauer, M; Langer, S; Roman, J, E.; Quintana-Ortí, E, S.: Large scale simulation of wave propagation in soils interacting with structures using FEM and SBFEM. Journal of Computational Acoustics. **19** (1), 75–93, (2011).

[30] Rossi, R., Oñate, E.O.: Analysis of some partitioned algorithms for fluid-structure interaction. International Journal for Computer-Aided Engineering and Software **27**(1), 20–56 (2010)

[31] S. Yigit and M. Schäfer and M. Heck.: Grid movement techniques and their influence on laminar fluid-structure interaction computations. Journal of Fluids and Structures **24**(6), 819–832 (2008)

[32] S. Idelsohn E. Oñate, E.D. (ed.): Fluid-structure interaction based upon a stabilized (ALE) finite element method, Computational Mechanics–New Trends and Applications (Proceedings of WCCM IV). CIMNE, Barcelona. (1998)

[33] Sassetti, F., Guarnieri, F.A., Garelli, L., Storti, M.: Characterization and simulation of an active microvalve for glaucoma. Computer Methods in Biomechanics and Biomedical Engineering. **15(12)**, 1273–1280 (2012). DOI 10.1080/10255842.2011.585978

[34] Schauer, M., Roman, J., Quintana-Ortí, E., Langer, S.: Parallel computation of 3-d soil-structure interaction in time domain with a coupled fem/sbfem approach. Journal of Scientific Computing **52**(2), 446–467 (2012). DOI 10.1007/s10915-011-9551-x.

[35] Sonzogni, E., Yommi, A., Nigro, N., Storti, M.: A parallel finite element program on a Beowulf cluster. Adv. Eng. Softw. **33**(7–10), 427–443 (2002)

[36] Storti, M., Garelli, L., Paz, R.R.: A finite element formulation satisfying the discrete geometric conservation law based on averaged jacobians. International Journal for Numerical Methods in Fluids **69**(12), 1872–1890 (2011)

[37] Storti, M., Nigro, N., Paz, R., Dalcin, L., Lopez, E.: Petsc-fem: A general purpose, parallel, multi-physics FEM program (1999-2010). http://www.cimec.org.ar/petscfem (2010).

[38] Wall., W.A.: Fluid-struktur-interaktion mit stabilisierten finiten elementen. Ph.D. thesis, Universitat Stuttgart, Institut für Baustatik (1999)

[39] Franck, G., Nigro, N., Storti, M., D'Elía, J.: Numerical simulation of the flow around the ahmed vehicle model. Latin American applied research. **39**, 295–306 (2009) ISSN(0327-0793).

[40] López, E., Nigro, N., Storti, M., Toth, J.: A Minimal Element Distortion Strategy for Computational Mesh Dynamics. International Journal for Numerical Methods in Engineering. **69**, 1898–1929 (2007)

[41] López, E J., Nigro, N M., Storti, M A.: Simultaneous untangling and smoothing of moving grids International Journal for Numerical Methods in Engineering **76** (7), 994–1019 (2008)

[42] Satish Balay, Kris Buschelman, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Barry F. Smith, Hong Zhang PETSc Web pag: http://www.mcs.anl.gov/petsc

[43] Kler, P A., Berli, C., Guarnieri, F A.: Modeling and high performance simulation of electrophoretic techniques in microfluidic chips Microfluidics and Nanofluidics **10**(1), 187–198 (2010)

[44] Kler, P A., López, E J., Dalcín, L D., Guarnieri, F A., Storti, M A.: High performance simulations of electrokinetic flow and transport in microfluidic chips. Computer Methods in Applied Mechanics and Engineering **198**(30-32), 2360–2367 (2009)

[45] Storti, M., Nigro, N., Paz, R.R.:Dynamic boundary conditions in Computational Fluid Dynamics. Computer Methods in Applied Mechanics and Engineering **197** 13-16), 1219–1232 (2008)

[46] Battaglia, L., D'Elía, J., Storti, M.A., Nigro, N.M.: Numerical simulation of transient free surface flows ASME-Journal of Applied Mechanics **73**, 1017–1025 (2006).

[47] L. Garelli, G. Ríos Rodríguez, R. Paz, M. A. Storti.:Adaptive Simulation of the internal flow in a Rocket Nozzle. Latin American Applied Research **44** 267–276 (2014).

[48] Degroote, J., Bruggeman, P., Haelterman, R., Vierendeels, J: Stability of a coupling technique for partitioned solvers in FSI applications Computers & Structures **86**(23-24) 2224–2234 (2008)

[49] Banks, J.W., Henshaw W.D., Schwendeman D.W.:An analysis of a new stable partitioned algorithm for {FSI} problems. Part I: Incompressible flow and elastic solids. Journal of Computational Physics **269** 108–137 (2014).

[50] Suliman, R., Oxtoby, O.F., Malan, A.G., Kok, S.:A matrix free, partitioned solution of fluid-structure interaction problems using finite volume and finite element methods European Journal of Mechanics - B/Fluids. **49** 272–286 (2015)

[51] Bogaers, A.E.J., Kok, K., Reddy, B.D., Franz, T.: Extending the robustness and efficiency of artificial compressibility for partitioned fluid-structure interactions Computer Methods in Applied Mechanics and Engineering. **283** 1278–1295 (2015).

[52] Matthies, Hermann G., Steindorf Jan.:Partitioned strong coupling algorithms for fluid-structure interaction Computers & Structures **81** (8-11) 805–812 (2003).

[53] Matthies, H.G., Niekamp, R., Steindorf, J.: Algorithms for strong coupling procedures Computer Methods in Applied Mechanics and Engineering **195**(17-18) 2028–2049 (2006).