European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2006
P. Wesseling, E. Oñate and J. Périaux (Eds)

AN ALE-CHIMERA METHOD FOR LARGE DEFORMATION FLUID STRUCTURE INTERACTION

Peter Gamnitzer, Wolfgang A. Wall

Technische Universität München, Lehrstuhl für Numerische Mechanik,
Boltzmannstraße 15, 85747 Garching (b. München), Germany
e-mail: gamnitzer@lnm.mw.tum.de, wall@lnm.mw.tum.de
web page: http://www.lnm.mw.tum.de/

Key words: Chimera, ALE, deforming grids, domain decomposition

Abstract. *This note discusses a combination of the Chimera fixed grid approach and Arbitrary Lagrangean Eulerian ('ALE') based methods for large deformation fluid structure interaction ('FSI'). The governing equations for incompressible flows in an Eulerian framework, a rotating frame of reference and an ALE-setting are discussed in order to point out the relatedness of these approaches. An algorithm to solve the Navier-Stokes-equations on domains with large deformations in an ALE-Chimera framework is derived from these equations. Some remarks on the implementation of the method are made, and in the end the theory is illustrated by small numerical examples.*

1 INTRODUCTION

A robust handling of the deformation of the computational fluid domain is the basis of every approach to deal with fluid structure interaction problems. These problems may show very large deformations of the fluid domain, including rotations or rigid body movements. Hence some extra effort is necessary if the accuracy of the surface representation should be maintained throughout the calculation.

First we want to draw the readers attention to two classical ways of treating deformations of the fluid domain which are directly related to the algorithm we present. For a more general overview see[1]. The example we use to explain the fundamental idea of these methods is the well known problem of a rigid cylinder immersed in a fluid. Figure 1 illustrates how the different approaches handle the problem when the cylinder is moved around in the computational domain. We summarize:

- *the Chimera approach*

  This is a fixed grid approach. It is capable of describing rigid body movements by coupling of several fixed and moving meshes. See for example the classical approach[2], the multiple body extension[3] or the fully automated version[4].

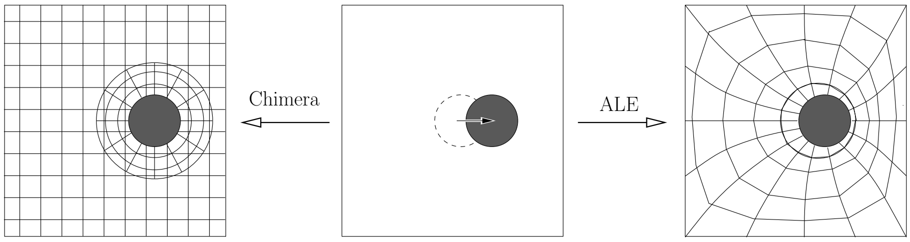

Figure 1: Two ways of moving the cylinder out of the center of the computational domain: The ALE approach fits the mesh to the new geometry, the Chimera-approach uses an overlapping mesh transported with the body.

- *the ALE approach*

  The ALE approach is a deforming grid approach. In contrast to the classical Chimera method it allows to deal with deformable objects. For this purpose, an additional reference configuration is introduced which is related to the actual spatial domain via the deformation of the mesh ('computational mesh dynamics'). This additional reference configuration leads to the well known ALE-formulation of the Navier-Stokes equation. The first papers to introduce this method into the finite element context are Donea et al.[5], Belytschko et al.[6, 7] and Hughes et al.[8]. The method is widely used in the context of fluid structure interaction but shows up some shortcomings when the mesh gets highly distorted due to the large deformations.

We discuss a combination of these two methods in moving a deformable ALE mesh with the solid.

The rest of the paper is structured as follows: The next section contains the governing equations used for the ALE and Chimera approaches. Some information on Chimera methods follows in the next section. The fourth section bridges the gap between ALE and Chimera methods which leads to the ALE-Chimera framework for fluid structure interaction. In the end, the algorithm is illustrated by numerical examples.

2 FORMULATIONS OF THE NAVIER-STOKES EQUATIONS

The two methods mentioned in the last paragraph are based on different formulations of the governing equations. The classical Chimera method uses an Eulerian formulation with a fixed frame of reference for motionless subdomains and an Eulerian formulation with a rotating frame of reference for moving subdomains. For the ALE approach, the momentum equation is restated with the time derivative evaluated with respect to the coordinates of the reference domain.

In this section we want to summarise all these equations and discuss the relationship between the ALE-formulation for special grid movements and the Euler equations on a rotating frame of reference.

2.1 Eulerian formulation for a fixed frame of reference

The Navier-Stokes-equations (1) on a fixed frame of reference are a system of partial differential equations for the velocity $\boldsymbol{u}(\boldsymbol{x}, t)$ and the pressure $p(\boldsymbol{x}, t)$. The physical vector-quantity belongs to a fixed frame of reference, $\boldsymbol{x}$ denotes the spatial coordinate relative to this fixed frame of reference.

$$\left.\frac{\partial \boldsymbol{u}}{\partial t}\right|_{\boldsymbol{x}} + (\boldsymbol{u} \circ \boldsymbol{\nabla}_{\boldsymbol{x}})\, \boldsymbol{u} + \boldsymbol{\nabla}_{\boldsymbol{x}} p - 2\nu \boldsymbol{\nabla}_{\boldsymbol{x}} \circ \boldsymbol{\varepsilon}(\boldsymbol{u}) = \boldsymbol{b} \tag{1}$$

$$\boldsymbol{\nabla}_{\boldsymbol{x}} \circ \boldsymbol{u} = 0$$

The problem is completed by boundary conditions on the boundary of the computational domain.

2.2 The extension to rotating frames of reference

We now present equation (2) with an underlying moving frame of reference. For the sake of clarity we restrict ourselves throughout the following paper to rotations (with constant angular velocity $\boldsymbol{\omega}$) around the origin of the global coordinate system. Accelerated rotations and linear motions are a straightforward generalisation. For this formulation, the physical quantities $\hat{\boldsymbol{u}}$ and $\hat{p}$ depend on the spatial coordinate $\boldsymbol{\chi}$ in the rotating frame of reference. The velocity as a vector is given in the same rotating coordinates as the position vector $\boldsymbol{\chi}$.

$$\left.\frac{\partial \hat{\boldsymbol{u}}}{\partial t}\right|_{\boldsymbol{\chi}} + (\hat{\boldsymbol{u}} \circ \boldsymbol{\nabla}_{\boldsymbol{\chi}})\, \hat{\boldsymbol{u}} + 2\boldsymbol{\omega} \times \hat{\boldsymbol{u}} + \boldsymbol{\nabla}_{\boldsymbol{\chi}} \hat{p} - 2\nu \boldsymbol{\nabla}_{\boldsymbol{\chi}} \circ \boldsymbol{\varepsilon}(\hat{\boldsymbol{u}}) = \hat{\boldsymbol{b}} - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{\chi}) \tag{2}$$

$$\boldsymbol{\nabla}_{\boldsymbol{\chi}} \circ \hat{\boldsymbol{u}} = 0$$

2.3 The ALE variant of the Navier-Stokes equation

For the ALE-formulation of the momentum equation we introduce the mappings from figure 2. The displayed mappings give rise to the following expressions for the particle velocity $\boldsymbol{u}(\boldsymbol{x}, t)$ and the grid velocity $\boldsymbol{u}_G(\boldsymbol{x}, t)$:

$$\boldsymbol{u}(\boldsymbol{x}, t) = \left(\left.\frac{\partial \boldsymbol{\varphi}}{\partial t}\right|_{\boldsymbol{X}}\right) \left(\boldsymbol{\varphi}^{-1}(\boldsymbol{x}, t), t\right), \qquad \boldsymbol{u}_G(\boldsymbol{x}, t) = \left(\left.\frac{\partial \boldsymbol{\Phi}}{\partial t}\right|_{\boldsymbol{\chi}}\right) \left(\boldsymbol{\Phi}^{-1}(\boldsymbol{x}, t), t\right)$$

They are used to state the ALE momentum equation in spatial representation for the particle velocity $\boldsymbol{u}$ and the pressure p as follows:

$$\left.\frac{\partial \left(\boldsymbol{u} \circ \boldsymbol{\Phi}\right)}{\partial t}\right|_{\boldsymbol{\chi}} \circ \boldsymbol{\Phi}^{-1} + \left((\boldsymbol{u} - \boldsymbol{u}_G) \circ \boldsymbol{\nabla}_{\boldsymbol{x}}\right) \boldsymbol{u} + \boldsymbol{\nabla}_{\boldsymbol{x}} p - 2\nu \boldsymbol{\nabla}_{\boldsymbol{x}} \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}) = \boldsymbol{b} \tag{3}$$

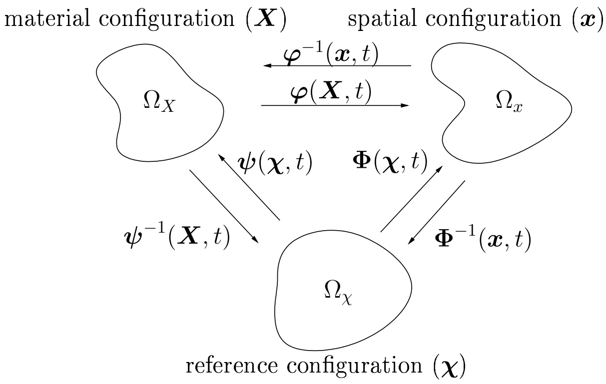

Figure 2: The material (initial) configuration is mapped by the mapping $\boldsymbol{\varphi}$ (particle motion) to the spatial (current) domain at time t. The ALE formulation uses an independently moved reference configuration, which is mapped to the spatial configuration by the mapping $\boldsymbol{\Phi}$.

2.4 The ALE momentum equation for special mesh movements

For the special case of a rotating mesh with a constant angular velocity (again rotation around $\mathbf{0}$) , the mesh motion $\boldsymbol{\Phi}$ is of the shape

$$\boldsymbol{\Phi}\left(\boldsymbol{\chi}, t\right) = \boldsymbol{A}(t)\boldsymbol{\chi}$$

with a time dependent orthogonal rotation matrix $\boldsymbol{A}(t)$. Introducing this identity in the momentum equation above leads to an equation which could be directly related to the Navier-Stokes-equation on a rotating frame of reference (2). For example the time derivative could be rewritten for a point $\boldsymbol{\chi} = \boldsymbol{\Phi}^{-1}\left(\boldsymbol{x}, t\right)$ as

$$\left.\frac{\partial\left(\boldsymbol{u}\circ\boldsymbol{\Phi}\right)}{\partial t}\right|_{\boldsymbol{\chi}}(\boldsymbol{\chi}) = \ddot{\boldsymbol{A}}(t)\boldsymbol{\chi} + \dot{\boldsymbol{A}}(t)\hat{\boldsymbol{u}} + \boldsymbol{A}(t)\left.\frac{\partial\hat{\boldsymbol{u}}}{\partial t}\right|_{\boldsymbol{\chi}}$$

Here,

$$\hat{\boldsymbol{u}} = \left(\left.\frac{\partial\boldsymbol{\psi}^{-1}}{\partial t}\right|_{\boldsymbol{X}}\right)\left(\boldsymbol{\psi}(\boldsymbol{\chi}, t), t\right)$$

is again the velocity relative to the rotating frame of reference. If this expression for the time derivative is transformed from the spatial domain to the reference domain (i. e. multiplied with the inverse Jacobian $\boldsymbol{A}^T(t)$ of $\boldsymbol{\Phi}$) this leads to

$$\boldsymbol{A}^T(t)\cdot\left.\frac{\partial\left(\boldsymbol{u}\circ\boldsymbol{\Phi}\right)}{\partial t}\right|_{\boldsymbol{\chi}}(\boldsymbol{\chi}) = \boldsymbol{A}^T(t)\ddot{\boldsymbol{A}}(t)\boldsymbol{\chi} + \boldsymbol{A}^T(t)\dot{\boldsymbol{A}}(t)\hat{\boldsymbol{u}} + \left.\frac{\partial\hat{\boldsymbol{u}}}{\partial t}\right|_{\boldsymbol{\chi}}$$

For an orthogonal matrix $\boldsymbol{A}(t)$ with $\boldsymbol{A}^T(t)\boldsymbol{A}(t) = \boldsymbol{E}$ we could easily see by time derivation that the matrix product $\boldsymbol{A}^T(t)\dot{\boldsymbol{A}}(t)$ is skew symmetric. Hence, the expression $\boldsymbol{A}^T(t)\dot{\boldsymbol{A}}(t)\hat{\boldsymbol{u}}$ could be rewritten with the angular velocity $\boldsymbol{\omega}$ as the first half of the Coriolis term:

$$\boldsymbol{A}^T(t)\dot{\boldsymbol{A}}(t)\hat{\boldsymbol{u}} = \boldsymbol{\omega}\times\hat{\boldsymbol{u}}$$

Since

$$
\begin{aligned}
\boldsymbol{A}^T\ddot{\boldsymbol{A}} &= \frac{d}{dt}\left(\boldsymbol{A}^T\dot{\boldsymbol{A}}\right) - \dot{\boldsymbol{A}}^T\dot{\boldsymbol{A}} = \frac{d}{dt}\left(\boldsymbol{A}^T\dot{\boldsymbol{A}}\right) - \dot{\boldsymbol{A}}^T\left(\boldsymbol{A}\boldsymbol{A}^T\right)\dot{\boldsymbol{A}} = \\
&= \frac{d}{dt}\left(\boldsymbol{A}^T\dot{\boldsymbol{A}}\right) - \left(\boldsymbol{A}^T\dot{\boldsymbol{A}}\right)^T\left(\boldsymbol{A}^T\dot{\boldsymbol{A}}\right) = \frac{d}{dt}\left(\boldsymbol{A}^T\dot{\boldsymbol{A}}\right) + \left(\boldsymbol{A}^T\dot{\boldsymbol{A}}\right)\left(\boldsymbol{A}^T\dot{\boldsymbol{A}}\right)
\end{aligned}
$$

we rewrite for a constant angular velocity the expression $\boldsymbol{A}^T(t)\ddot{\boldsymbol{A}}(t)\boldsymbol{\chi}$ as follows:

$$
\boldsymbol{A}^T(t)\ddot{\boldsymbol{A}}(t)\boldsymbol{\chi} = \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{\chi})
$$

All other parts in the ALE-momentum equation could be treated in the same way. The ALE-convective term may be rewritten for our special mesh motion as follows:

$$
\sum_{j=1}^{3} \frac{\partial \boldsymbol{u}}{\partial x_j} \cdot (u_j - u_{G,j}) = \dot{\boldsymbol{A}}\hat{\boldsymbol{u}} + \sum_{i=1}^{3} \boldsymbol{A}\frac{\partial \hat{\boldsymbol{u}}}{\partial \chi_i} \cdot \hat{u}_i
$$

Transformation from the spatial domain to the reference domain yields

$$
\boldsymbol{A}^T \sum_{j=1}^{3} \frac{\partial \boldsymbol{u}}{\partial x_j} \cdot (u_j - u_{G,j}) = \boldsymbol{\omega} \times \hat{\boldsymbol{u}} + \sum_{i=1}^{3} \frac{\partial \hat{\boldsymbol{u}}}{\partial \chi_i} \cdot \hat{u}_i
$$

The same transformation allows to compute the pressure gradient on the reference domain.

$$
\boldsymbol{A}^T \boldsymbol{\nabla}_{\boldsymbol{x}} p = \boldsymbol{\nabla}_{\boldsymbol{\chi}} \hat{p}
$$

Since $\boldsymbol{\Phi}$ is a rigid body motion, the contribution of the mesh velocity $\boldsymbol{u}_G$ to the tensor $\boldsymbol{\varepsilon}(\boldsymbol{u})$ vanishes and we get the identity

$$
\boldsymbol{A}^T \cdot \Big(\boldsymbol{\nabla}_{\boldsymbol{x}} \circ \boldsymbol{\varepsilon}(\boldsymbol{u})\Big) = \boldsymbol{\nabla}_{\boldsymbol{\chi}} \circ (\boldsymbol{\varepsilon}(\hat{\boldsymbol{u}}))
$$

After some calculation, as expected, we have recovered equation (2). Hence, formulations (2) and (3) are equivalent for a rigid body mesh motion.

3 CLASSICAL CHIMERA METHODS

In this section we sketch the foundations and implementation of Chimera methods as they are widely used in computational fluid dynamics today. Together with the ideas from the previous section this will allow us to propose a more general algorithm for the solution of the Navier-Stokes equations on a deforming domain in the next section.

3.1 The fundamental idea

Chimera methods are based on a domain decomposition of the fluid domain Ω^f. The solution is therefore calculated by a sequential iteration over the overlapping subdomains Ω_i with $\Omega^f = \bigcup_i \Omega_i$, $\Omega_i \cap \Omega_j \neq \emptyset$. For the sake of clarity we restrict ourselves without loss of generality to two fluid subdomains, Ω_0 ('background') and Ω_1 ('patch'). To understand the Chimera method it is important to know, that the background mesh is automatically generated out of a (favourably) structured mesh covering the whole domain Ω^f by deactivating special nodes under the moving structure (according to its current position). For further explanations we again refer to the simple example of a rigid cylinder moved around in the fluid domain. Figure 3 is intended to introduce the meaning of the artificial inner boundaries Γ_0, Γ_1 on the basis of this example. Later on it will be reused to explain the coupling process between the subdomains.

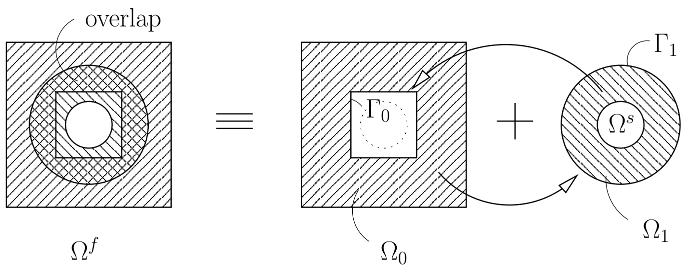

Figure 3: The domain decomposition for the Chimera method. Ω_1 is the moving subdomain connected to the structure Ω_s, Ω_0 the fixed background subdomain. The arrows indicate the transfer of information from the subdomains to the inner boundaries Γ_0, Γ_1 during the alternating solution process.

For the fixed background subdomain Ω_0 the Navier-Stokes-equations are solved for velocity $\boldsymbol{u}_0(\boldsymbol{x}, t)$ and pressure $p_0(\boldsymbol{x}, t)$ on a fixed frame of reference. The index zero indicates the connection of the physical quantities to the subdomain Ω_0. The governing equations are

$$\left.\frac{\partial \boldsymbol{u}_0}{\partial t}\right|_{\boldsymbol{x}} + \left(\boldsymbol{u}_0 \circ \boldsymbol{\nabla}_{\boldsymbol{x}}\right) \boldsymbol{u}_0 + \boldsymbol{\nabla}_{\boldsymbol{x}} p_0 - 2\nu \boldsymbol{\nabla}_{\boldsymbol{x}} \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}_0) = \boldsymbol{b} \quad \text{in } \Omega_0 \times (0, T) \tag{4}$$

$$\boldsymbol{\nabla}_{\boldsymbol{x}} \circ \boldsymbol{u}_0 = 0 \quad \text{in } \Omega_0 \tag{5}$$

The boundary conditions for this problem are the standard boundary conditions on the outer boundary $\partial\Omega^f \cap \Omega_0$ plus the artificial boundary conditions on Γ_0. For the classical Chimera method, Γ_0 is a Dirichlet boundary. The boundary values on Γ_0 are derived by interpolation (Interpolation operator $\boldsymbol{I}$) from the solution of the previous Chimera iteration step on Ω_1:

$$\boldsymbol{u}_0(\boldsymbol{x}) = \boldsymbol{I}\left(\hat{\boldsymbol{u}}_1^{prev}\right) \tag{6}$$

The hat on the solution on Ω_1 indicates like in the previous section, that this velocity $\hat{\boldsymbol{u}}_1(\boldsymbol{\chi}, t)$ is classically calculated on a rotating frame of reference. For this case, the governing equations are:

$$\left.\frac{\partial \hat{\boldsymbol{u}}_1}{\partial t}\right|_{\boldsymbol{\chi}} + \left(\hat{\boldsymbol{u}}_1 \circ \boldsymbol{\nabla}\right)\hat{\boldsymbol{u}}_1 + 2\boldsymbol{\omega} \times \hat{\boldsymbol{u}}_1 - 2\nu \boldsymbol{\nabla} \cdot \boldsymbol{\varepsilon}(\hat{\boldsymbol{u}}_1) + \boldsymbol{\nabla}\hat{p}_1 = \hat{\boldsymbol{b}} - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{\chi}) \tag{7}$$

$$\boldsymbol{\nabla}_{\boldsymbol{\chi}} \circ \hat{\boldsymbol{u}}_1 = 0 \tag{8}$$

The problem on the patch is completed by the boundary conditions on $\Omega_1 \cap \partial\Omega^f$ and on the inner boundary Γ_1. For this inner boundary of the patch various boundary conditions are common. A first possibility is to use a Dirichlet boundary to get a Dirichlet-Dirichlet coupling (a Schwarz method). The problem of that choice is, that the patch subdomain will be purely Dirichlet bounded. Because of the incompressibility constraint this problem will be ill-posed without an additional control of the flux into the patch[9]. A second choice is to make Γ_1 a Neumann boundary and to transmit the force onto the patch subdomain. In the context of finite elements this boundary condition is easily included by adding a surface integral expression to the right-hand side of the weak form. This approach was extended by Houzeaux and Codina[10] to a Dirichlet-Robin coupling. For this coupling, a certain percentage b of the convective term in the weak form of the momentum equation is integrated by parts on Ω_1. With the trial function $\boldsymbol{v}_1$ this leads to an enhanced boundary integral on Γ_1:

$$\int_{\Gamma_1} \left[\boldsymbol{v}_1^T \boldsymbol{\sigma}\left(\boldsymbol{u}_1\right)\boldsymbol{n} - b\left(\boldsymbol{u}_1 \circ \boldsymbol{v}_1\right) \cdot \boldsymbol{u}_1 \circ \boldsymbol{n}\right] da$$

The whole procedure is summarised in table 1.

time-loop			
	determine the inactive nodes on $\Omega^f \backslash \Omega_0$ and the interpolation nodes on Γ_0		
	repeat until STOP CHIMERA		
		loop all subdomains Ω_i	
			update boundary conditions on Γ_i by interpolation from the previous solution of the overlapping domain.
			solve the fluid problem on Ω_i.
		STOP CHIMERA if the boundary values on $\bigcup \Gamma_i$ do not change anymore during one loop over all Ω_i	

Table 1: The solution procedure for the alternating Chimera approach.

3.2 Implementation aspects

Now to some implementation details. The subdomain Ω_0 and its inner boundary Γ_0 is formed in each iteration step as displayed in figure 4 for the well known cylinder example. The node search required for this procedure (and later on required for the interpolation) is done via a quadtree search algorithm.

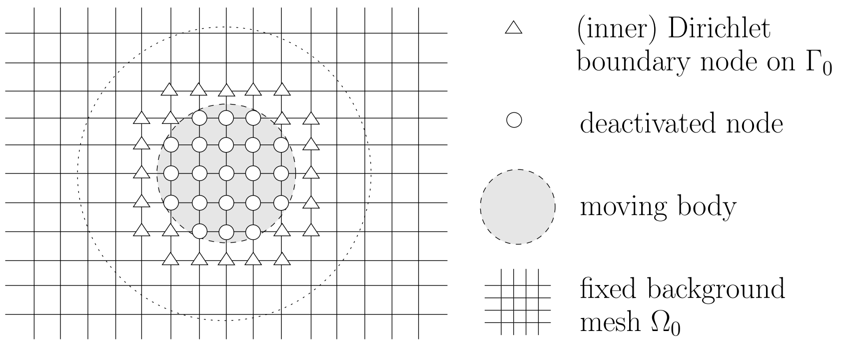

Figure 4: The background mesh is generated automatically within the Chimera method.

The transmission process corresponding to the interpolation operator $\boldsymbol{I}$ interpolates nodal data from one subdomain to the inner boundary of the other subdomain. In the case of Dirichlet boundaries, the information is passed to the nodes, in case of a Neumann boundary, the information is passed to the Gausspoints. For the interpolation of Neumann data it is absolutely mandatory to calculate a least square smoothed superconvergent approximation of the velocity gradients. Useful techniques like the superconvergent patch recovery are described for example in[11, 12].

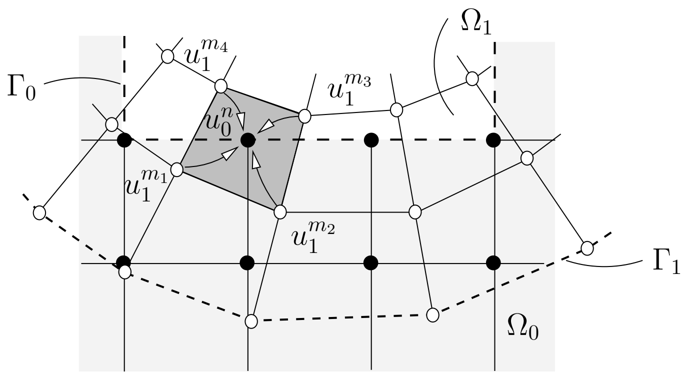

Figure 5: The boundary node on Γ_0 is interpolated from the recent solution on Ω_1.

Figure 5 exemplifies the interpolation process for the interpolation of the Dirichlet boundary value $\boldsymbol{u}_0^n$ on a node n of Γ_0 (coordinates $\boldsymbol{x}_n$) (The lower index refers to the subdomain, the upper index to the node number). To determine the value of $\boldsymbol{u}_0^n$, a search is performed in domain Ω_1 to determine a 'parent' element containing the point corresponding to the coordinates $\boldsymbol{x}^n$. The value $\boldsymbol{u}_0^n$ is now determined by Lagrange interpolation from the values $\boldsymbol{u}_1^m$ of the nodes m of the parent element, see Figure 5. For the displayed example, the values, which are calculated using the standard finite element shape functions N_{m_i} evaluated at the local coordinates $\boldsymbol{\xi}$ of the point $\boldsymbol{x}^n$ on the parent element are:

$$\boldsymbol{u}_0^n = \sum N_{m_i}\left(\boldsymbol{\xi}\right)\boldsymbol{u}_1^{m_i}$$

A point to be considered here is, that the interpolation requires the real velocities $\boldsymbol{u}_1^m$ on the patch. Thus, a coordinate transformation has to be performed before interpolating the values.

The last aspect of implementation here is the solution process of the subdomains. We applied a stabilized Finite Element approach with a one step theta time integration. The additional coriolis term in the equation (2) requires additional effort for stabilization[13].

4 AN ALE-CHIMERA FRAMEWORK FOR FSI

As we saw in the last but one section, the classic governing equations on the moving frame of reference (2) could be replaced by the ALE formulation (3). For the patch problem, this leads to the equation

$$\left.\frac{\partial\left(\boldsymbol{u}_1\circ\boldsymbol{\Phi}\right)}{\partial t}\right|_{\boldsymbol{\chi}}\circ\boldsymbol{\Phi}^{-1}+\left(\left(\boldsymbol{u}_1-\boldsymbol{u}_{G,1}\right)\circ\boldsymbol{\nabla}_{\boldsymbol{x}}\right)\boldsymbol{u}_1+\boldsymbol{\nabla}_{\boldsymbol{x}}p_1-2\nu\boldsymbol{\nabla}_{\boldsymbol{x}}\cdot\boldsymbol{\varepsilon}(\boldsymbol{u}_1)=\boldsymbol{b} \qquad (9)$$

We now propose to use this ALE formulation on the patch for a mixed ALE-Chimera method.

For real rigid body problems this has already the advantage, that the ALE-equation contains the real particle velocities. Hence, when interpolating between the meshes it is not necessary anymore to transform the velocities between the different frames of reference. In addition to this, the alternative formulation does not require any extra stabilisation or other changes to the element subroutines in a standard ALE code. The further advantage of the new formulation is, that the method is capable of simulating the flow around deformable structures moved over large distances in the fluid.

In addition to the work that has to be done for the classical Chimera approach, the mesh positions and velocities have to be updated in each time step. We propose to use a pseudo-elasticity approach for the computational mesh dynamics. For fluid structure interaction problems this leads to the algorithm described in table 2.

5 NUMERICAL EXAMPLES

5.1 Moving 'breathing' subdomain

The setting of the following academic example is shown in figure 6. It consists of a simple structure which is moved around in a channel. The movement is decomposed into

- time-loop
  - repeat until STOP FSI (iteration over fields)
    - solve for mesh dynamics on deforming sub-domains
    - solve fluid
      - via a Chimera-like procedure
      - ...
    - solve structure
    - if converged, STOP FSI

Table 2: Iterative staggered coupling scheme using a Chimera approach for the fluid solution. Due to the modularity, the changes are ‘local’.

three parts: The center of the structure is moving from left to right and vice versa in the computational domain. The structure rotates around its center and in addition to that, it changes its shape. Throughout the calculation the Reynoldsnumber remains very low.

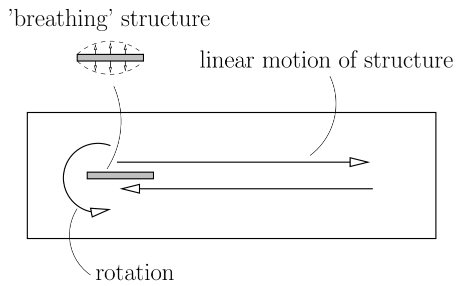

Figure 6: The setting of the numerical example. The total deformation is decomposed in three parts.

The boundary conditions imposed on the upper and lower boundaries and the immersed structure are no-slip. On the outflow we have a do nothing boundary condition an on the inflow we prescribed the traction in such a way that without the obstacle the resulting flow would be a standard channel flow with a parabolic velocity profile and maximum velocity of 1.

The problem was solved for 10 time units, which corresponded to 200 time steps. Each time step required up to 16 iterations over the subdomains. As a result of our choice to control the convergence behaviour of the subdomain iteration by monitoring the changes of the inner boundary values, we had to apply a very small upper bound for this increment to ensure the convergence on the whole domain. During the calculation time the structure oscillates twice from left to right, performes two rotations about its center and changes twice its shape as sketched in figure 6.

The two Dirichlet-Neumann coupled meshes can be seen in figure 7. The same figure contains the results for the horizontal velocities at two time steps. The pictures in the upper row displays a time step when the structure is moving from right to left in the opposite direction of the driving force, the channel flow is decelerated. The other time step corresponds to a movement from left to right. The flux through the channel is increased. As far as we could see from this example, the Dirichlet-Neumann coupled method is capable of producing correct results.

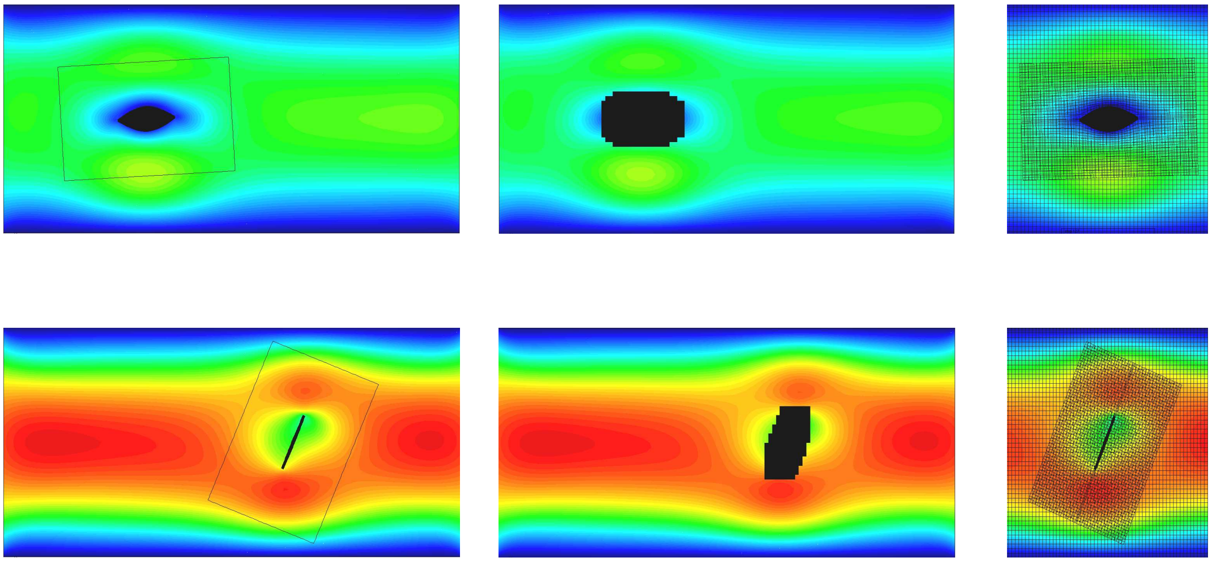

Figure 7: Results (horizontal velocities) for two time steps. Each line contains the result of the overlapping domain decomposition (left), the result on the background subdomain (center) and the two overlapping meshes around the moving body (right).

Closer numerical investigation of the velocities along the inner Neumann boundary shows up some deficiencies of the method, especially when the background mesh was chosen too coarse. For these meshes the quality of the gradient approximation was not high enough, so the Neumann coupling of the patch failed.

5.2 Moving 'snake-subdomain'

The geometry for this example is basically a 4.0×2.0 channel with a small cylinder-beam object immersed in a viscous fluid (kinematic viscosity $\nu = 1$) which is slowly moved around in the channel. The motion of the object is completely prescribed. It may be decomposed in a translatoric part (cos-shaped velocity) and a 'snake like' deformation of the beam. The boundary conditions on the left side of the channel is 'do nothing', on the right border we have a parabolic velocity profile with maximum horizontal velocity $u_{max,in} = 1.0$ in negative x-direction.

The problem was solved for 40 time units which corresponded to 800 time steps. During this time the structure moved 8 times from left to right and back and kept continuously

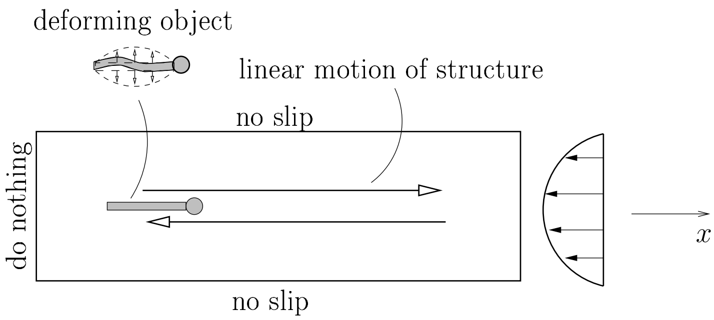

Figure 8: The setting of the numerical example. The prescribed total deformation is decomposed into two parts. We place emphasis on the fact that this problem does not have any biological background but was simply designed to illustrate the capabilities of our algorithm.

'waving' the beam. In every time step of the fluid solution, the fluid problem was solved iteratively on the two Dirichlet-Robin-coupled subdomains. The results for the horizontal velocity of different time steps is displayed in figure 9. The velocity values on the inner Robin boundary showed a better quality than the values from a pure Neumann-coupling, even for coarser grids.

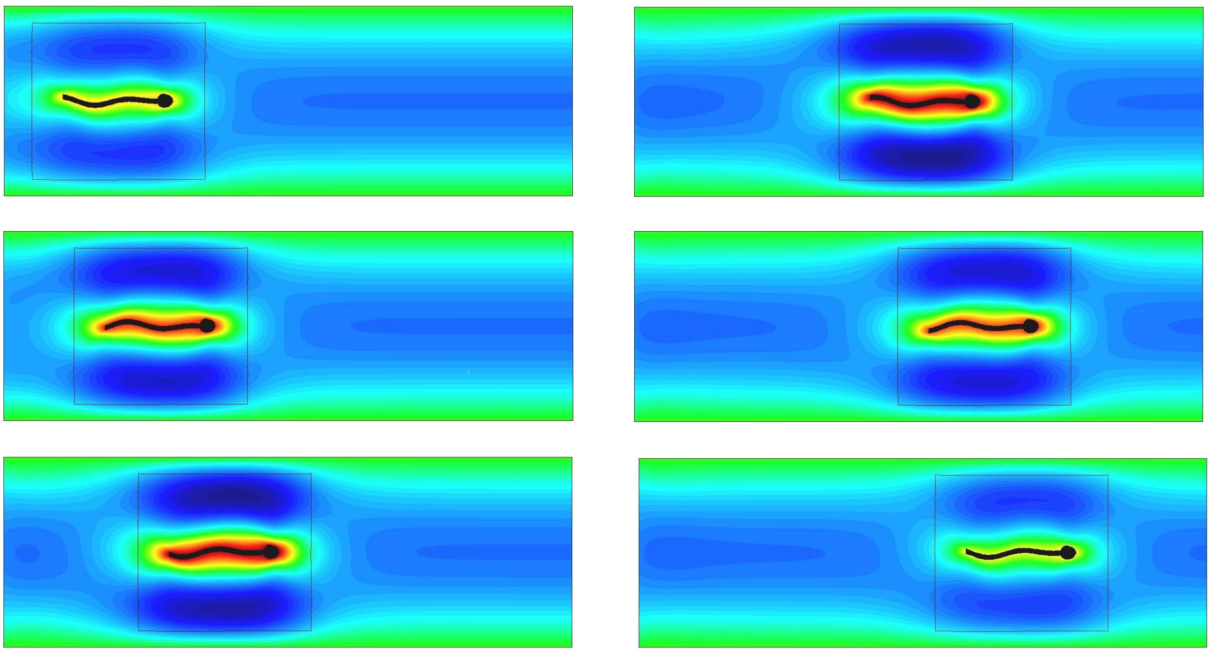

Figure 9: Results (horizontal velocities) for several time steps between 400 and 440.

6 CONCLUSIONS

We presented an extended Chimera method with an alternative governing equation on the patch subdomain using an ALE formulation. The new formulation was the basis for a finite element implementation which is capable of treating moving deformable structures immersed in fluids. This enables us to focus on the treatment of fluid structure interaction problems with very large deformations within such a framework. By the time of the conference we will present the basics of our implementation and we are looking forward to show first results of fluid structure interaction problems solved in combination with an ALE-Chimera method.

REFERENCES

[1] W. A. Wall, A. Gerstenberger, P. Gamnitzer, C. Förster and E. Ramm: Large deformation fluid-structure interaction – advances in ALE methods and new fixed grid approaches, *Fluid-Structure Interaction: Modelling, Simulation, Optimisation, LNCSE (H.-J. Bungartz and M. Schäfer eds.),* Springer Verlag (2006)

[2] J.L. Steger, F.C. Dougherty, J.A. Benek: A Chimera grid scheme, *Advances in Grid Generation (Ghia, Ghia eds.).* Volume ASME FED-5 (1983) 59-69

[3] R.L. Meakin, N.E. Suhs: Unsteady aerodynamic simulation of multiple bodies in relative motion, *AIAA Paper* 89-1996-CP (1989)

[4] Z.J. Wang, V. Parthasarathy: A fully automated Chimera methodology for multiple moving body problems, *International Journal for Numerical Methods in Fluids* **33**, 7 (2000) 919–938

[5] J. Donéa, P. Fasoli-Stella, S. Giuliani: Lagrangian and Eulerian Techniques for Transient Fluid-Structure Interaction Problems, *Transactions of the 4th int. Conference on SMIRT*, San Francisco. B1/2

[6] T. Belytschko, J.M. Kennedy: Computer Modells for Subassembly Simulation, *Nuclear Engineering and Design*, **49**, 17–38

[7] T. Belytschko, J.M. Kennedy, D.F. Schoeberle: Quasi-Eulerian Finite-Element Formulation for Fluid Structure Interaction, *Journal of Pressure Vessel Technology*, **102**, 62–69

[8] T.J.R. Hughes, W.K. Liu, T.K. Zimmermann: Lagrangian-Eulerian Finite Element Formulation for Viscous Flows, *Computer Methods in Applied Mechanics an Engineering*, **29**, 329–349

[9] G. Houzeaux, R. Codina: Transmission conditions with constraints in finite element domain decomposition methods for flow problems, *Communications in Numerical Methods in Engineering*, **17**, 179–190 (2001)

[10] G. Houzeaux, R. Codina: A Chimera method based on a Dirichlet/Neumann(Robin) coupling for the Navier-Stokes equations, *Computer Methods in Applied Mechanics and Engineering* **192**, 31-32 (2003), 3343–3377

[11] O.C. Zienkiewicz, J.Z. Zhu: Superconvergent patch recovery and a posteriori error estimation in the finite element method, Part I: A general superconvergent recovery technique, *Internat. J. Num. Meth. Eng.*, **33**, 1331–64, (1992)

[12] O.C. Zienkiewicz, J.Z. Zhu: The superconvergent patch recovery (SPR) and a posteriori error estimates. Part 2: Error estimates and adaptivity, *Internat. J. Num. Meth. Eng.*, **33**, 1365–82, (1992)

[13] R. Codina, O. Sato: Finite element solution of the Stokes problem with dominating Coriolis force, *Comput. Methods. Appl. Mech. Engrg.* **142** (1997) 215–234