A mesh adaptative method for rotating machines

Didier Chargy, Bastien Sauvage

▶ **To cite this version:**

Didier Chargy, Bastien Sauvage. A mesh adaptative method for rotating machines. [Research Report] RR-9471, Inria. 2022, pp.31. hal-03684715v1

HAL Id: hal-03684715

https://inria.hal.science/hal-03684715v1

Submitted on 1 Jun 2022 (v1), last revised 28 Sep 2022 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A mesh adaptative method for rotating machines

Didier Chargy, Bastien Sauvage

RESEARCH REPORT
N° 9471
2022
Project-Team Ecuador

ISSN 0249-6399 ISRN INRIA/RR--9471--FR+ENG

A mesh adaptative method for rotating machines

Didier Chargy [*], Bastien Sauvage [†]

Équipe-Projet Ecuador

Rapport de recherche n° 9471 — 2022 — 28 pages

Résumé : Les écoulements dans des configurations rotor/stator peuvent être traités avec des discrétisations particulières et une modélisation particulière de la turbulence. Nous introduisons un cadre de référence multiple et un cadre Chimera et nous les combinons à une approche d'adaptation de maillage. Ceci est illustré sur l'écoulement dans une cuve équipée d'un bras mélangeur rotatif en forme de croix. Ce travail a été financé par l'Agence Nationale de la Recherche dans le cadre du projet NORMA, contrat ANR-19-CE40-0020-01.

Mots-clés : Volumes finis, Ordre élevé

[*] Lemma, 2000 Rte des Lucioles, 06410 Biot, didier.chargy@lemma-ing.com
[†] Université Côte d'Azur, INRIA-Ecuador,B.P.93,06902 Sophia-Antipolis, FRANCE, bastien.sauvage@inria.fr

RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

A mesh adaptative convergent method for rotating machines

Abstract: Flows in rotor/stator configurations can be addressed with special discretization and special turbulence modeling. We introduce a multiple Reference Framework and a Chimera framework and combine them to a mesh adatation approach. This is illustrated on the flow in a mixing device with a rotating cross-shape mixer arm. This work was supported by the ANR project NORMA under grant ANR-19-CE40-0020-01.

Key-words: Finite Volume, higher order

Table des matières

1 Introduction

A rotating machine breaks down, in its simplest form, into a rotor and a stator. Taking into account both the fixed part and the rotating part is a delicate numerical issue. For doing this, we identify the following main families :
- Methods which reconstruct a unique and fixed conformal mesh from two parts, with one in rotation. This can be done by applying *edge swapping* between the two components at each time step.
- Methods which apply a common set of finite-volume fluxes between the two components. At each time steps, boundary faces are intersected and split into two interfaces sliding on each others in order to apply finite volume fluxes. The advantage of this approach is that it is a conservative discretization over the global union of both components.
- Methods which reconstruct a unique and fixed conformal mesh, by putting the motion inside the partial differential equation e.g. through a multiple reference frame (MRF). - Chimera methods based on overlapping components, and in which hanging nodes of a component are equipped with informations from the other component.

In this work, we study the two last methods of this list, a MRF method and a Chimera approach, and combine them with a class of mesh adaptation algorithms. As concerns the mesh adaptation, we shall address two different contexts :
- the flow becomes steady,
- the flow becomes periodic in time,
we propose an adaptation of the Transient Fixed Point of [1].

2 Approximation for rotating frames

In this section, we briefly describe the models used and their discretization. First, let us recall the compressible Navier-Stokes equations

$$\begin{cases} \partial_t \rho + \operatorname{div}(\rho \mathbf{u}) = 0, \\ \partial_t (\rho \mathbf{u}) + \operatorname{div}(\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p = \operatorname{div} \mathscr{T}, \\ \partial_t (\rho E) + \operatorname{div}((\rho E + p)\mathbf{u}) = \operatorname{div}(\lambda \nabla T + \mathscr{T} \cdot \mathbf{u}), \end{cases}$$

where ρ denote the density, $\mathbf{u}$ the velocity, E the total energy, p the pressure, T the temperature, μ the laminar dynamic viscosity and λ the laminar conductivity. $\mathscr{T}$ is the laminar stress tensor

$$\mathscr{T} = \mu \left(\nabla \mathbf{u} + {}^t \nabla \mathbf{u} - \frac{2}{3} (\operatorname{div} \mathbf{u}) \mathbb{I}_3 \right).$$

In our study we will rather use the Reynold Average Navier-Stokes system, where if we omit the specific notations for the averaged variables, the system is written in the same way except for the laminar conductivity and the stress tensor which become $\lambda + \lambda_t$ and

$$\mathscr{T} = (\mu + \mu_t) \left(\nabla \mathbf{u} + {}^t \nabla \mathbf{u} - \frac{2}{3} (\operatorname{div} \mathbf{u}) \mathbb{I}_3 \right),$$

where λ_t is the turbulent conductivity and μ_t is the turbulent viscosity. The problem of modeling turbulence is reduced to the determination of μ_t, in order to close the system. We will consider the one equation model of Spalart-Allmaras, which will be described in section 2.6.

2.1 The Navier-Stokes equations in rotating frame

We follow the lines of classical mechanics, see for example [10]. We consider two reference frames, $\mathscr{R}$ and $\mathscr{R}'$, with $\mathscr{R}$ the Galilean reference frame and $\mathscr{R}'$ in rotation with respect to $\mathscr{R}$ around an axis $\mathbf{e}$ ($\mathbf{z}$ in our case), with angular velocity $\boldsymbol{\omega}$ constant. Let $\mathbf{x}$ be a positive vector, the law of velocity compositions gives us

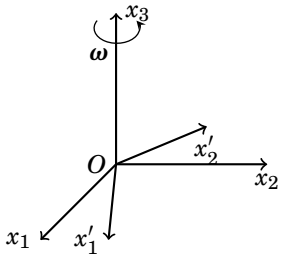

FIGURE 1 – Fixed and mobile references frames

$$\left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\mathscr{R}} = \left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\mathscr{R}'} + \boldsymbol{\omega} \times \mathbf{x}.$$

Let us note the velocity vectors $\left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\mathscr{R}} = \mathbf{u}$ and $\left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\mathscr{R}'} = \mathbf{u}'$. We have

$$\begin{aligned}
\left.\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\right|_{\mathscr{R}} &= \left.\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\right|_{\mathscr{R}'} + \boldsymbol{\omega} \times \mathbf{u} \\
&= \left.\frac{\mathrm{d}(\mathbf{u}' + \boldsymbol{\omega} \times \mathbf{x})}{\mathrm{d}t}\right|_{\mathscr{R}'} + \boldsymbol{\omega} \times (\mathbf{u}' + \boldsymbol{\omega} \times \mathbf{x}) \\
&= \left.\frac{\mathrm{d}\mathbf{u}'}{\mathrm{d}t}\right|_{\mathscr{R}'} + 2\boldsymbol{\omega} \times \mathbf{u}' + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{x}).
\end{aligned}$$

Note that we have

$$\begin{aligned}
\left.\frac{\mathrm{d}\,\mathbf{u}(t,\mathbf{x}(t))}{\mathrm{d}t}\right|_{\mathscr{R}} &= \frac{\partial \mathbf{u}}{\partial t} + \left(\left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\mathscr{R}} \cdot \nabla\right)\mathbf{u} \\
&= \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}.
\end{aligned}$$

where $\mathbf{u}$ is the absolute velocity and $\mathbf{u}'$ is the relative velocity. In the presence of a fluid flow in a rotating frame, the velocity of the fluid can be expressed in the fixed frame and in the moving frame. The relation between the two velocities is $\mathbf{u}' = \mathbf{u} - \mathbf{u}_e$ with $\boldsymbol{u}_e = \boldsymbol{\omega} \times \mathbf{x}$. The compressible Navier-Stokes equations can be written in relative or absolute velocity, we will describe these two formulations.

2.1.1 Relative velocity formulation

In this subsection we will see how the equations of mass conservation, quantity of motion conservation and total energy are formulated in relative velocity.

• Mass conservation equation :

The mass conservation equation is

$$\frac{\partial \rho}{\partial t} + \operatorname{div}\left(\rho \mathbf{u}\right) = \frac{\partial \rho}{\partial t} + \operatorname{div}\left(\rho(\mathbf{u}' + \boldsymbol{\omega} \times \mathbf{x})\right) = \frac{\partial \rho}{\partial t} + \operatorname{div}\left(\rho \mathbf{u}'\right) = 0.$$

• Quantity of motion conservation equation :

Starting from the accelerations composition, the motion equation is

$$\begin{aligned}\rho \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\Big|_{\mathscr{R}} &= \rho \frac{\mathrm{d}\mathbf{u}'}{\mathrm{d}t}\Big|_{\mathscr{R}'} + \rho\left(\boldsymbol{\omega} \times \mathbf{u}' + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{x})\right) = -\nabla p + \operatorname{div} \mathscr{T} \\ &= \rho \frac{\partial \mathbf{u}'}{\partial t} + \rho\left(\mathbf{u}' \cdot \nabla\right)\mathbf{u}' + \rho\left(2\boldsymbol{\omega} \times \mathbf{u}' + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{x})\right) = -\nabla p + \operatorname{div} \mathscr{T},\end{aligned}$$

so considering the mass conservation equation and the motion equation we have

$$\begin{aligned}\frac{\partial(\rho \mathbf{u}')}{\partial t} &= \mathbf{u}' \frac{\partial \rho}{\partial t} + \rho \frac{\partial \mathbf{u}'}{\partial t} \\ &= -\mathbf{u}' \operatorname{div}\left(\rho \mathbf{u}'\right) - \rho\left((\mathbf{u}' \cdot \nabla)\mathbf{u}' + 2\boldsymbol{\omega} \times \mathbf{u}' + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{x})\right) - \nabla p + \operatorname{div} \mathscr{T}\end{aligned}$$

and as

$$\mathbf{u}' \operatorname{div}\left(\rho \mathbf{u}'\right) + \rho(\mathbf{u}' \cdot \nabla)\mathbf{u}' = \sum_{i=1}^{3}\left(\mathbf{u}' \partial_i(\rho \mathbf{u}_i') + \rho \mathbf{u}_i' \partial_i \mathbf{u}'\right) = \sum_{i=1}^{3} \partial_i\left(\rho \mathbf{u}_i' \mathbf{u}'\right) = \operatorname{div}\left(\rho \mathbf{u}' \otimes \mathbf{u}'\right),$$

finally

$$\frac{\partial(\rho \mathbf{u}')}{\partial t} + \operatorname{div}\left(\rho \mathbf{u}' \otimes \mathbf{u}'\right) + \rho\left(2\boldsymbol{\omega} \times \mathbf{u}' + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{x})\right) + \nabla p = \operatorname{div} \mathscr{T}.$$

• Total energy conservation equation :

Taking the mass conservation equation and what we did for the momentum conservation equation we have

$$\begin{aligned}\frac{\partial(\rho E_r)}{\partial t} &= \frac{\partial(\rho e)}{\partial t} + \frac{|\mathbf{u}'|^2}{2}\frac{\partial \rho}{\partial t} + \rho \mathbf{u}' \cdot \frac{\partial \mathbf{u}'}{\partial t} \\ &= -\operatorname{div}\left(\rho \mathbf{u}' e\right) - \frac{|\mathbf{u}'|^2}{2}\operatorname{div}\left(\rho \mathbf{u}'\right) - \rho \mathbf{u}' \cdot \left((\mathbf{u}' \cdot \nabla)\mathbf{u}'\right) - \rho \mathbf{u}' \cdot \left(2\boldsymbol{\omega} \times \mathbf{u}' + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{x})\right) \\ &\quad + \operatorname{div}\left(\lambda \nabla T + \mathscr{T} \cdot \mathbf{u}' - p \mathbf{u}'\right),\end{aligned}$$

we can easily verify that $\mathbf{u}' \cdot (\boldsymbol{\omega} \times \mathbf{u}') = 0$ and

$$|\mathbf{u}'|^2 \operatorname{div}\left(\rho \mathbf{u}'\right) + \rho \mathbf{u}' \cdot \left((\mathbf{u}' \cdot \nabla)\mathbf{u}'\right) = \operatorname{div}\left(\rho \mathbf{u}' \frac{|\mathbf{u}'|^2}{2}\right).$$

So finally we have

$$\frac{\partial(\rho E_r)}{\partial t} + \operatorname{div}\left(\left(\rho E_r + p\right)\mathbf{u}'\right) + \rho \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{x}) \cdot \mathbf{u}' = \operatorname{div}\left(\lambda \nabla T + \mathscr{T} \cdot \mathbf{u}'\right).$$

2.1.2 Absolute velocity formulation

This time we just consider our motion equation in the following form

$$\begin{aligned}\rho\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\Big|_{\mathscr{R}} &= \rho\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\Big|_{\mathscr{R}'} + \rho\boldsymbol{\omega}\times\mathbf{u}\\ &= \rho\frac{\partial\mathbf{u}}{\partial t} + \rho\left(\mathbf{u}'\cdot\nabla\right)\mathbf{u} + \rho\boldsymbol{\omega}\times\mathbf{u} = -\nabla p + \operatorname{div}\mathscr{T},\end{aligned}$$

• Mass conservation equation :

The mass conservation equation is unchanged from the relative velocity formulation

$$\frac{\partial\rho}{\partial t} + \operatorname{div}\left(\rho\mathbf{u}'\right) = 0.$$

• Quantity of motion conservation equation :

Considering our mass conservation and motion equation we have

$$\begin{aligned}\frac{\partial(\rho\mathbf{u})}{\partial t} &= \mathbf{u}\frac{\partial\rho}{\partial t} + \rho\frac{\partial\mathbf{u}}{\partial t}\\ &= -\mathbf{u}\operatorname{div}\left(\rho\mathbf{u}'\right) - \rho\left(\mathbf{u}'\cdot\nabla\right)\mathbf{u} - \rho\boldsymbol{\omega}\times\mathbf{u} - \nabla p + \operatorname{div}\mathscr{T},\end{aligned}$$

we check that

$$\mathbf{u}\operatorname{div}\left(\rho\mathbf{u}'\right) + \rho\left(\mathbf{u}'\cdot\nabla\right)\mathbf{u} = \sum_{i=1}^{3}\left(\mathbf{u}\partial_i(\rho\mathbf{u}_i') + \rho\mathbf{u}_i'\partial_i\mathbf{u}\right) = \sum_{i=1}^{3}\partial_i(\rho\mathbf{u}_i'\mathbf{u}) = \operatorname{div}\left(\rho\mathbf{u}'\otimes\mathbf{u}\right),$$

then we have

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \operatorname{div}\left(\rho\mathbf{u}'\otimes\mathbf{u}\right) + \rho\boldsymbol{\omega}\times\mathbf{u} + \nabla p = \operatorname{div}\mathscr{T}.$$

• Energy conservation equation :

$$\begin{aligned}\frac{\partial(\rho E)}{\partial t} &= \frac{\partial(\rho e)}{\partial t} + \frac{|\mathbf{u}|^2}{2}\frac{\partial\rho}{\partial t} + \rho\mathbf{u}\cdot\frac{\partial\mathbf{u}}{\partial t}\\ &= -\operatorname{div}\left(\rho\mathbf{u}'e\right) - \frac{|\mathbf{u}|^2}{2}\operatorname{div}\left(\rho\mathbf{u}'\right) - \rho\mathbf{u}\cdot\left((\mathbf{u}'\cdot\nabla)\mathbf{u}\right) - \rho\mathbf{u}\cdot(\boldsymbol{\omega}\times\mathbf{u}) + \operatorname{div}(\lambda\nabla T + \mathscr{T}\cdot\mathbf{u} - p\mathbf{u})\end{aligned}$$

we can check again that $\mathbf{u}\cdot(\boldsymbol{\omega}\times\mathbf{u}) = 0$ and

$$\frac{|\mathbf{u}|^2}{2}\operatorname{div}\left(\rho\mathbf{u}'\right) + \rho\mathbf{u}\cdot\left((\mathbf{u}'\cdot\nabla)\mathbf{u}\right) = \operatorname{div}\left(\rho\mathbf{u}'\frac{|\mathbf{u}|^2}{2}\right).$$

Finally

$$\frac{\partial(\rho E)}{\partial t} + \operatorname{div}\left(\rho E\mathbf{u}' + p\mathbf{u}\right) = \operatorname{div}(\lambda\nabla T + \mathscr{T}\cdot\mathbf{u}).$$

2.1.3 The equations

To summarize, let us give the formulation of the Navier-Stokes system in relative and in absolute formulation :

- **Relative velocity formulation**

$$\begin{cases} \partial_t \rho + \operatorname{div}(\rho \mathbf{u}') = 0, \\ \partial_t(\rho \mathbf{u}') + \operatorname{div}(\rho \mathbf{u}' \otimes \mathbf{u}') + \rho(\boldsymbol{\omega} \times \boldsymbol{\omega} \times \mathbf{x} + 2\boldsymbol{\omega} \times \mathbf{u}') + \nabla p = \operatorname{div} \mathscr{T}, \\ \partial_t(\rho E_r) + \operatorname{div}((\rho E_r + p)\mathbf{u}') + \rho \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{x}) \cdot \mathbf{u}' = \operatorname{div}(\lambda \nabla T + \mathscr{T} \cdot \mathbf{u}'). \end{cases}$$

here $\mathscr{T} = \mu\left(\nabla \mathbf{u}' + {}^t\nabla \mathbf{u}' - \frac{2}{3}(\operatorname{div} \mathbf{u}')\mathbb{I}_3\right)$ and $E_r = e + \frac{1}{2}\|\mathbf{u}'\|^2$. The relative velocity formulation is appropriate when most of the fluid in the domain is in rotation. In this case it is called SRF for Single Reference Frame.

- **Absolute velocity formulation**

$$\begin{cases} \partial_t \rho + \operatorname{div}(\rho \mathbf{u}') = 0, \\ \partial_t(\rho \mathbf{u}) + \operatorname{div}(\rho \mathbf{u}' \otimes \mathbf{u}) + \rho(\boldsymbol{\omega} \times \mathbf{u}) + \nabla p = \operatorname{div} \mathscr{T}, \\ \partial_t(\rho E) + \operatorname{div}((\rho E + p)\mathbf{u}' + p\,\mathbf{u}_e) = \operatorname{div}(\lambda \nabla T + \mathscr{T} \cdot \mathbf{u}). \end{cases}$$

The absolute velocity formulation is preferred in applications where the fluid flow does not rotate through most of the domain. In this case it is called MRF for Multiple Reference Frame.

In the case where the whole computational domain is referred to a single rotating reference frame, the equations in relative formulation are solved in all fluid cell zones.

Some problems involve multiple moving parts or contain surfaces that are not surfaces of revolution. For these problems, it is appropriate to divide the computational domain into several fluid/solid cell zones, with interface boundaries separating the zones. The zones that contain the moving components can then be solved using the equations of the moving frame while the stationary zones can be solved with the equations of the fixed frame.
When the absolute velocity formulation is used, the equations in each domain are expressed in absolute velocity, so no transformation is required at the interface of the two domains in this approach.

2.2 Discretization (N-S absolute formulation)

We are interested in the spatial discretization of the Navier Stokes system in absolute formulation. The governing equations are discretized in space using a mixed finite-volume/finite-element method. Finite-volumes are used for the convective terms, the diffusion terms are discretized using P1 Galerkin finite-elements on the tetrahedra. Let $\Omega \subset \mathbb{R}^3$ be a bounded domain and $\mathscr{T}_h = \{K\}$ its conformal tetrahedrization. The adopted scheme is vertex-centered and at each node i of the mesh we build a dual cell C_i such that

$$\Omega = \bigcup_{i=1}^{\mathscr{N}} C_i\,, \quad \mathscr{N} \text{ the number of nodes.}$$

We consider the Navier-Stokes system in absolute velocity formulation, written in condensed form,

$$\begin{cases} \partial_t W + \operatorname{div} F^C(W) = \operatorname{div} F^D(W) + F^S(W), & (t,\mathbf{x}) \in [0,+\infty[\times \Omega, \\ W(0,\mathbf{x}) = W_0(\mathbf{x}), & \mathbf{x} \in \Omega, \\ W(t,\mathbf{x}) = \Phi(t,\mathbf{x}), & (t,\mathbf{x}) \in [0,+\infty[\times \partial\Omega. \end{cases} \tag{1}$$

with

$$\partial_t W + \mathrm{div} F^C(W) - \mathrm{div} F^D(W) = \partial_t W + \sum_{j=1}^{3} \partial_{x_j} F_j^C(W) - \sum_{j=1}^{3} \partial_{x_j} F_j^D(W) = F^S, \tag{2}$$

where, with the following notations $\mathbf{u} = (u_1, u_2, u_3)$, $\mathbf{u}' = (u_1', u_2', u_3')$,

$$W = \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho E \end{pmatrix}, \quad F_j^C(W) = \begin{pmatrix} \rho u_j' \\ \rho u_1 u_j' + p\,\delta_{1j} \\ \rho u_2 u_j' + p\,\delta_{2j} \\ \rho u_3 u_j' + p\,\delta_{3j} \\ (\rho E + p) u_j' + p u_{ej} \end{pmatrix}, \quad F_j^D(W) = \begin{pmatrix} 0 \\ \mathcal{T}_{1j} \\ \mathcal{T}_{2j} \\ \mathcal{T}_{3j} \\ \sum_{i=1}^{3} u_i \mathcal{T}_{ij} + \lambda \partial_{x_j} T \end{pmatrix},$$

with

$$\mathcal{T}_{ij} = \mu \left(\partial_{x_j} u_i + \partial_{x_i} u_j - \frac{2}{3} \sum_{k=1}^{3} \partial_{x_k} u_k \delta_{ij} \right),$$

and the source terms fluxes is

$$F^S(W) = \begin{pmatrix} 0 \\ -\rho(\boldsymbol{\omega} \times \mathbf{u}) \\ 0 \end{pmatrix}.$$

The equations are integrated on each cell C_i and using the Stokes formula we have (omitting the boundary conditions)

$$\frac{d}{dt} \int_{C_i} W \,\mathrm{d}\mathbf{x} + \int_{\partial C_i} F^C(W) \cdot \mathbf{n}_i \,\mathrm{d}\sigma = \int_{\partial C_i} F^D(W) \cdot \mathbf{n}_i \,\mathrm{d}\sigma + \int_{C_i} F^S(W) \mathrm{d}\mathbf{x}.$$

$\mathbf{n}_i$ is the outer normal to the cell surface ∂C_i. If we denote by $\mathcal{V}(i)$ the set of vertices j directly neighboring i, we can write

$$\int_{\partial C_i} F^C(W) \cdot \mathbf{n}_i \,\mathrm{d}\sigma = \sum_{j \in \mathcal{V}(i)} \int_{\partial C_i \cap \partial C_j} F^C(W) \cdot \mathbf{n}_i \,\mathrm{d}\sigma\,,$$
$$\int_{\partial C_i} F^D(W) \cdot \mathbf{n}_i \,\mathrm{d}\sigma = \sum_{j \in \mathcal{V}(i)} \int_{\partial C_i \cap \partial C_j} F^D(W) \cdot \mathbf{n}_i \,\mathrm{d}\sigma\,.$$

For the convective fluxes we consider, with $\mathbf{n}_{ij} = \displaystyle\int_{\partial C_i \cap \partial C_j} \mathbf{n}_i \,\mathrm{d}\sigma$, the following approximation

$$\int_{\partial C_i \cap \partial C_j} F^C(W) \cdot \mathbf{n}_i \,\mathrm{d}\sigma \approx F^C_{|\partial C_i \cap \partial C_j} \cdot \mathbf{n}_{ij}\,.$$

The viscous terms will be discretized by Finite Element, so we will show an equivalence between our Finite Volume formulation and a Finite Element formulation. Let φ_i be the P_1 Finite Element basis function associated with the vertex p_i, we have (see Appendix for a proof)

$$\int_K \nabla \varphi_i \,\mathrm{d}\mathbf{x} = -\int_{\partial C_i \cap K} \mathbf{n} \,\mathrm{d}\sigma\,.$$

We will see that thanks to this formula we will be able to establish an equivalence between the Finite Volumes and the Finite Elements. Consider $T(i)$ the set of elements $K_i = (p_i, p_j, p_k, p_l)$ which have for common vertex p_i, then

$$\begin{aligned}\sum_{j\in\mathcal{V}(i)}\int_{\partial C_i\cap\partial C_j} F^D(W)\cdot\mathbf{n}_i\,\mathrm{d}\sigma &= \sum_{K_i\in T(i)}\int_{\partial C_i\cap K_i} F^D(W)\cdot\mathbf{n}_i\,\mathrm{d}\sigma\\ &= \sum_{K_i\in T(i)} F^D(W)_{|K_i}\cdot\int_{\partial C_i\cap K_i}\mathbf{n}_i\,\mathrm{d}\sigma\\ &= -\sum_{K_i\in T(i)}\int_{K_i} F^D(W)_{|K_i}\cdot\nabla\varphi_i\,\mathrm{d}\mathbf{x}\end{aligned}$$

We will note respectively $\Phi^C_{ij}(W_i, W_j, \mathbf{n}_{ij})$, $\Phi^D_i|_{K_i}(W_i, W_j, W_k, W_l)$ and $\Phi^S_i(W_i)$ the approximations of $\int_{\partial C_i\cap\partial C_j} F^C(W)\cdot\mathbf{n}_i\,\mathrm{d}\sigma$, $\int_{K_i} F^D(W)_{|K_i}\cdot\nabla\varphi_i\,\mathrm{d}\mathbf{x}$ and $\int_{C_i} F^S(W)\,\mathrm{d}\mathbf{x}$. So we have the following semi-discrete scheme,

$$\frac{dW_i}{dt} = \frac{1}{|C_i|}\left(\sum_{j\in\mathcal{V}(i)}\Phi^C_{ij}(W_i, W_j, \boldsymbol{n}_{ij}) + \sum_{K_i\in T(i)}\Phi^D_i|_{K_i}(W_i, W_j, W_k, W_l) + \Phi^S_i(W_i)\right).$$

where W_i stand for the following quantity

$$W_i = \frac{1}{|C_i|}\int_{C_i} W\,\mathrm{d}\mathbf{x}.$$

As described before we have, with φ_i the P_1 Finite Element basis function associated with vertex P_i,

$$\Phi^D_i|_{K_i}(W_i, W_j, W_k, W_l) = -\int_{K_i} F^D(W)_{|K_i}\cdot\nabla\varphi_i\,\mathrm{d}x,$$

and

$$\Phi^S_i(W_i) = |C_i|\begin{pmatrix}0\\ -\omega\times(\rho\mathbf{u})_i\\ 0\end{pmatrix}.$$

For the convective fluxes we consider HLLC approximate Riemann solver, which is described in the following subsection

$$\Phi^C(W_i, W_j, \mathbf{n}_{ij}) = \Phi^{\mathrm{HLLC}}(W_i, W_j, \mathbf{n}_{ij}).$$

2.3 HLLC approximate Riemann solver

We consider for our study the HLLC solver. The idea is to consider locally a simplified Riemann problem with two intermediate states depending on the local left and right states. The simplified solution to the Riemann problem consists of a contact wave with a velocity S_M and two acoustic waves, which may be either shocks or expansion fans. The acoustic waves have the smallest and the largest velocities (S_i and S_j, respectively) of all the waves present in the exact solution. If $S_i > 0$ then the flow is supersonic from left to right and the upwind flux is simply defined from $F(W_i)$ where W_i is the state to the left of the discontinuity. Similarly, if $S_j < 0$ then the flow is

supersonic from right to left and the flux is defined from $F(W_j)$ where W_j is the state to the right of the discontinuity. In the more difficult subsonic case when $S_i < 0 < S_j$ we have to calculate $F(W_i^*)$ or $F(W_j^*)$. Consequently, the HLLC flux is given by

$$\Phi^{\mathrm{HLLC}}(W_i, W_j, \mathbf{n}_{ij}) = \begin{cases} F(W_i)\cdot\mathbf{n}_{ij} & \text{if } S_i > 0, \\ F(W_i^*)\cdot\mathbf{n}_{ij} & \text{if } S_i \leqslant 0 < S_M, \\ F(W_j^*)\cdot\mathbf{n}_{ij} & \text{if } S_M \leqslant 0 \leqslant S_j, \\ F(W_j)\cdot\mathbf{n}_{ij} & \text{if } S_j < 0. \end{cases}$$

where W_i^* and W_j^* are evaluated as follow. Let us denote $\eta = \mathbf{u}\cdot\mathbf{n}$, the following evaluations are proposed

$$W^* = \frac{1}{S - S_M}\begin{pmatrix} \rho(S-\eta) \\ \rho\mathbf{u}(S-\eta) + (p^* - p)\mathbf{n} \\ \rho E(S-\eta) + p^* S_M - p\eta \end{pmatrix},$$

with $p^* = \rho(S-\eta)(S_M - \eta) + p$. For the contact wave we consider :

$$S_M = \frac{\rho_j\eta_j(S_j-\eta_j) - \rho_i\eta_i(S_i-\eta_i) + p_i - p_j}{\rho_j(S_j-\eta_j) - \rho_i(S_i-\eta_i)}$$

and the acoustic wave speeds based on the Roe average ($\bar{\cdot}$) :

$$S_i = \min(\eta_i - c_i, \bar{\eta} - \bar{c}), \quad S_j = \max(\eta_j + c_j, \bar{\eta} + \bar{c}).$$

The formulation that has been described leads to a first order finite volume scheme for the compressible flow equations. A higher order extension may be constructed in the standard form, by using reconstructed interface values in the HLLC solver.

2.4 Discrete Geometric Conservation Law for MRF case

In the code we are using an ALE method is already implemented, we used this method for the implementation of our MRF method.

2.4.1 Notations

We consider only the convective part of our equation in MRF formulation, which read

$$\partial_t W + \mathrm{div} F^C(W) = 0, \tag{3}$$

with

$$F^C(W) = F(W) + F^{MRF}(W) = \begin{pmatrix} \rho\mathbf{u} \\ \rho\,\mathbf{u}\otimes\mathbf{u} + p\mathbb{I}_3 \\ (\rho E + p)\mathbf{u} \end{pmatrix} - \begin{pmatrix} \rho\mathbf{u}_e \\ \rho\,\mathbf{u}_e\otimes\mathbf{u} \\ \rho E\mathbf{u}_e \end{pmatrix}$$

Now integrating (3) on each cells C_i we have

$$|C_i|\frac{dW_i}{dt} + \sum_{j\in V(i)}\int_{\partial C_{ij}} (F(W)\cdot\mathbf{n}_i - W\,\mathbf{u}_e\cdot\mathbf{n}_i)\,d\sigma = 0,$$

and we consider the following approximation

$$\int_{\partial C_{ij}} (F(W)\cdot\mathbf{n}_i - W\,\mathbf{u}_e\cdot\mathbf{n}_i)\,d\sigma \approx |\partial C_{ij}|\Phi(W_i, W_j, \mathbf{n}_{ij}, \sigma_{ij})$$

where

$$\mathbf{n}_{ij} = \frac{1}{|\partial C_{ij}|}\int_{\partial C_{ij}} \mathbf{n}_i \, d\sigma,$$
$$\sigma_{ij} = \frac{1}{|\partial C_{ij}|}\int_{\partial C_{ij}} \mathbf{u}_e \cdot \mathbf{n}_i \, d\sigma,$$

and moreover

$$\Phi(W_*, W^*, \mathbf{n}, \sigma) = -\Phi(W^*, W_*, -\mathbf{n}, \sigma), \quad \text{(conservativity)}$$
$$\Phi(W, W, \mathbf{n}, \sigma) = F(W)\cdot\mathbf{n} - W\sigma. \quad \text{(consistency)}$$

2.4.2 Discretization of moving domains : ALE method

Let us consider a smooth bijective mapping $\pi(t)$ depending on time and equal to identity at time $t = 0$. Defining this mapping is equivalent to defining a velocity field $\mathbf{u}_e$ and moving each point of the space with this velocity. Let for any time $\Omega_{\mathrm{h}}(t)$ the triangulation (tetrahedrisation in 3D) obtained by applying the mapping $\pi(t)$ to any vertex of Ω_{h}^0. It is enough to known the trajectories of each vertex, starting from a vertex of Ω_{h}^0. This is also equivalent to know for any time t the velocity $\mathbf{u}_e(i,t)$ at vertex i of Ω_{h}^t. In order to ensure than any segment or plan of the initial mesh will stay at ant time resp. a serment or a plan, we consider the linear interpolation of vertex values $\mathbf{u}_e(i,t)$ to any element of mesh Ω_{h}^t. For this discrete deformation velocity, we keep the notation $\mathbf{u}_e(t)$. In Ω_{h}^t we build the dual cells :

$$\Omega_{\mathrm{h}}(t) = \bigcup_{i=1}^{n_c} C^i(t).$$

The flux balance writes now :

$$\frac{d\mathcal{V}^i}{dt} + \int_{\partial C^i(t)} \boldsymbol{F}(v)\cdot\mathbf{n}_i - \int_{\partial C^i(t)} v\ \mathbf{u}_e\cdot\mathbf{n}_i = 0, \tag{4}$$

where

$$\frac{d\mathcal{V}^i}{dt} = \frac{d}{dt}\left(\int_{C^i(t)} v\right).$$

For each *interior* cell $C^i(t)$, with its corresponding set $V(i)$ of neighboring cells,

$$\partial C^i(t) = \bigcup_{j\in V(i)} \partial C^{ij}(t), \tag{5}$$

where $\partial C^{ij}(t)$ represents the interface shared by cells $C^i(t)$ and $C^j(t)$. Then, [4] can be written as :

$$\frac{d\mathcal{V}^i}{dt} + \sum_{j\in V(i)}\int_{\partial C^{ij}(t)} \boldsymbol{F}(v)\cdot\mathbf{n}_{ij} - \sum_{j\in V(i)}\int_{\partial C^{ij}(t)} v\ \mathbf{u}_e\cdot\mathbf{n}_{ij} = 0. \tag{6}$$

In order to evaluate this integral, following [5], let us define

$$\boldsymbol{\nu}_{ij}(t) = \frac{1}{|\partial C_{ij}(t)|}\int_{\partial C_{ij}(t)} \mathbf{n}_{ij}(t)\, ds \tag{7}$$

and

$$\kappa_{ij}(t) \;=\; \frac{1}{|\partial C_{ij}(t)|}\int_{\partial C_{ij}(t)} \mathbf{u}_e(t)\cdot\mathbf{n}_{ij}(t)\,ds\;. \tag{8}$$

$\boldsymbol{\nu}_{ij}(t)$ is the *mean normal* corresponding to $\partial C_{ij}(t)$ and $\kappa_{ij}(t)$, the *mean normal mesh velocity* projection for the same cell interface (the full meaning of this "mean" will be grasped below, when it becomes also a temporal one). Then we get an integral ALE semi-discretization of the conservation law :

$$\frac{d\mathcal{V}^i}{dt} \;+\; \sum_{j\in V(i)} |\partial C_{ij}(t)|\;\Phi(v_i,v_j,\boldsymbol{\nu}_{ij}(t),\kappa_{ij}(t)) \;=\; 0, \tag{9}$$

where Φ is a numerical flux function, typically an approximate Riemann solver, with mesh velocity normal component $\kappa_{ij}(t)$ and with mean value of unknown v over cell i denoted by v_i. In particular it satisfies the following consistency condition :

$$\Phi(v,v,\boldsymbol{\nu},\kappa) \;=\; \boldsymbol{F}(v)\cdot\boldsymbol{\nu} \;-\; \kappa v.$$

Consider now a time discretization of the above formula. Up to first order, v_i can be taken as constant within each cell. Then, if the volume of the partition's cell $C_i(t)$ is $|C_i(t)|$,

$$\mathcal{V}_i(t) = |C_i(t)|\;v_i(t). \tag{10}$$

The θ-parameterized Euler time advancing yields

$$\begin{aligned} |C_i^{n+1}|v_i^{n+1} \;\;=\;\; & |C_i^n|v_i^n \\ & -\Delta t\theta \sum_{j\in V(i)} |\partial\bar{C}_{ij}|\Phi\left(v_i^{n+1},v_j^{n+1},\overline{\boldsymbol{\nu}}_{ij},\overline{\kappa}_{ij}\right) \\ & -\Delta t(1-\theta) \sum_{j\in V(i)} |\partial\bar{C}_{ij}|\Phi\left(v_i^{n},v_j^{n},\overline{\boldsymbol{\nu}}_{ij},\overline{\kappa}_{ij}\right) \end{aligned} \tag{11}$$

where the overlines mean that time averaged values are taken. According to the Geometric Conservation Law principle, a uniform solution is exactly preserved when time-advanced by the numerical scheme. Assume the above system able to reproduce a constant solution $v^n = v^{n+1} = v^*$, it should satisfy :

$$|C_i^{n+1}|v_i^* \;=\; |C_i^n|v_i^* \;-$$

$$\Delta t\theta \sum_{j\in V(i)} |\partial\bar{C}_{ij}|\Phi\left(v_i^*,v_j^*,\overline{\boldsymbol{\nu}}_{ij},\overline{\kappa}_{ij}\right) \;-\; \Delta t(1-\theta) \sum_{j\in V(i)} |\partial\bar{C}_{ij}|\Phi\left(v_i^*,v_j^*,\overline{\boldsymbol{\nu}}_{ij},\overline{\kappa}_{ij}\right).$$

Invoking the consistency condition for Φ and the fact that the cells remain closed during the motion, which writes :

$$\sum_{j\in V(i)} |\partial\bar{C}_{ij}|\overline{\boldsymbol{\nu}}_{ij} \;=\; 0,$$

we see that this gives the usual *Discrete Geometric Conservation Law (DGCL)* :

$$|C_i^{n+1}| \;=\; |C_i^n| \;-\; \Delta t \sum_{j\in V(i)} |\partial\bar{C}_{ij}|\overline{\kappa}_{ij}. \tag{12}$$

As stated in papers like [7], [4] or [3], the DGCL becomes a design condition to impose for the time averaged values $|\partial\bar{C}_{ij}|$, $\overline{\boldsymbol{\nu}}_{ij}$, and $\overline{\kappa}_{ij}$. Evaluated over $\partial C^{ij}(t)$, they should be carefully computed. In [7], both the cell's normals and the grid's velocity mean values determine the geometrical

parameters which enforce the GCL. In [4], this is attained by means of a proper evaluation of the ALE fluxes using suited mesh configurations and grids velocities, showing also an equivalence with the former paper ideas. On the other hand, in [3], a scheme satisfying the GCL is proposed by tuning how the cell volume is evaluated. We also refer to [5] for examples of averagings satisfying the DGCL for the above time advancing scheme.

2.4.3 Adaptation of DGCL to MRF

As concerns the DGCL inside MRF, we observe that the mesh is fictitiously rotating in one part Ω_r of the computational domain and fixed in the rest of the domain. Then the fictitous mesh speed $\mathbf{u}_e$ is discontinuous, typically :

$$\mathbf{u}_e = \boldsymbol{\omega} \times \mathbf{x} \quad \text{in} \quad \Omega_r, \quad 0 \quad \text{elsewhere.}$$

However, the velocity is of zero divergence. Indeed the divergence is zero on both subdomains, and the normal velocity to interface is zero, which implies that the divergence is zero over the complete computational domain. In the discrete case, DGCL is easily checked in both domain. In the neighborhood of the discontinuity, we have checked that in practise, the numerical deviation to DGCL is close to zero machine.

We consider now a second approach for discretizing rotation.

2.5 Numerical Chimera scheme

We describe a second approach for rotor/stator interaction. The Chimera method aims at solving partial differential equations by decomposition into subdomains with overlap in order to avoid having to use a global mesh. These domains can be fixed or mobile, but must have an overlapping area because this method allows the communication between the computational domains thanks to their overlapping. For our rotor/stator applications, we use the Chimera methods to decompose the computational domain $\Omega \subset \mathbb{R}^3$ into two subdomains, one fixed domain Ω_f (the stator part) and one rotating domain Ω_r (the rotor part), see Fig.2. Consider a tetrahedrization $\mathcal{T}_f = \{T_f\}$ of Ω_f and a tetrahedrization $\mathcal{T}_r = \{T_r\}$ of Ω_r. Let $\{p_i^f\}_{i=1,n}$ the vertices of $\mathcal{T}_f$ and $\{p_j^r\}_{j=1,m}$ the vertices of $\mathcal{T}_r$, a crucial assumption is that, in $\Omega_f \cap \Omega_r$, each node p^f is internal to a tetrahedron T_r and reciprocally, each node p^r is internal to a tetrahedron T_f. Note that the boundary nodes, those located in overlap area, are deactivated, the values of our unknows are found at these points by a P_1 interpolation.

We denote by W_f (respectively W_r) the solution computed by our scheme in the domain $\Omega_f \setminus \Omega_f \cap \Omega_r$ (respectively the solution computed by our scheme in the domain $\Omega_r \setminus \Omega_f \cap \Omega_r$). An iteration in time goes as follows

- The domain Ω_r rotates by an angle α with respect to the domain Ω_f.
- We locate the boundary nodes p^f in Ω_r, and reciprocally we locate the boundary p^r in Ω_f.
- The values of W_f and W_r in $\Omega_f \cap \Omega_r$ are determined by a P_1 interpolation.
- Finally W_f and W_r are computed respectively in $\Omega_f \setminus \Omega_f \cap \Omega_r$ and $\Omega_r \setminus \Omega_f \cap \Omega_r$ with the new interpolated values.

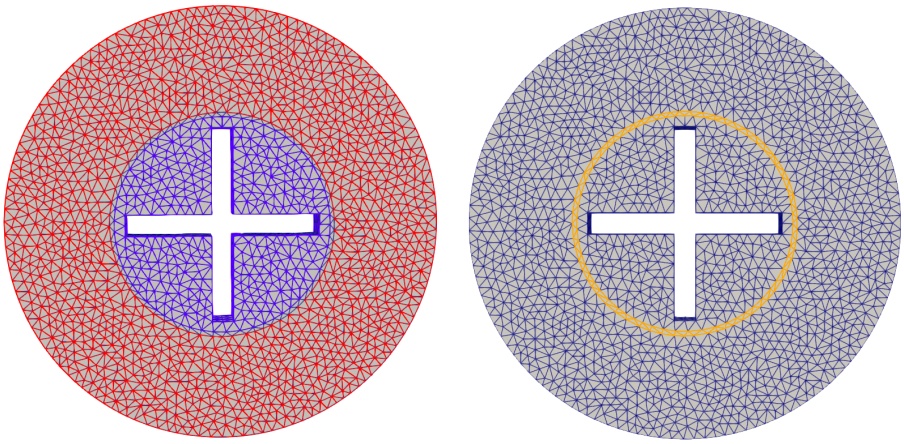

FIGURE 2 – Definition of rotor and stator from an initial mesh : in red, the stator, in blue the rotor, and in yellow, their common part.

At each time step the rotating domain will rotate with respect to the fixed domain according to the rotation speed we have imposed, see example in Fig.3.

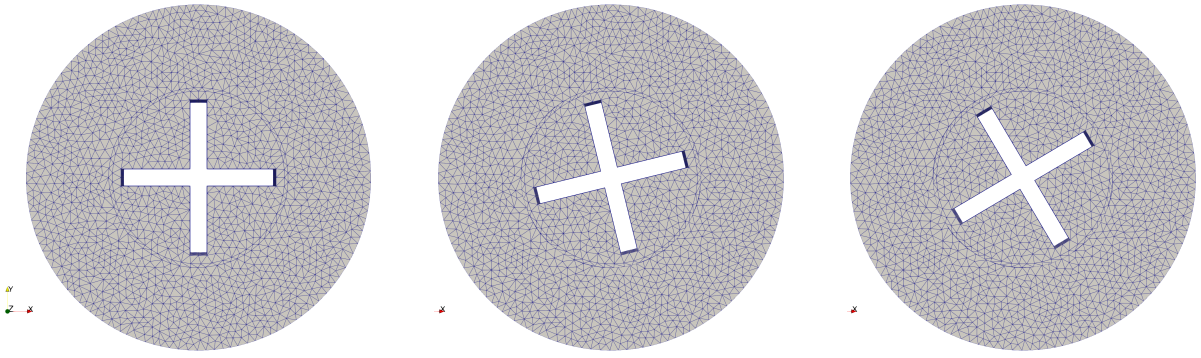

FIGURE 3 – Illustration of the rotation of the cross from the initial position to a physical time $T = 0.005$.

For more clarity, let's zoom in on an area of the domain with overlap, we can see in Fig.4 the position of the meshes at the initial time and their position after one time step.

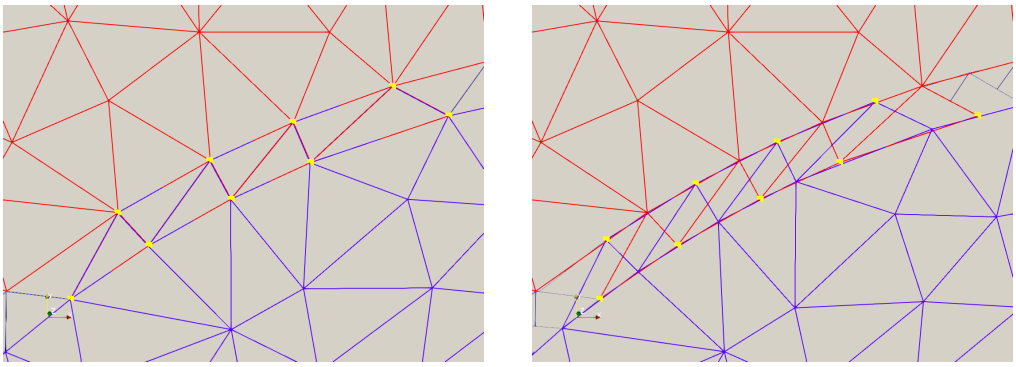

FIGURE 4 – Initial position of meshes, left, and the two meshes after a slight rotation of rotor.

The points in yellow correspond to points that are no longer in their respective mesh, no equation is solved on these points. We calculate the value of these points by interpolation. For example in Fig.5, the value of point 1 is found by interpolation in the blue triangle where it is inscribed.

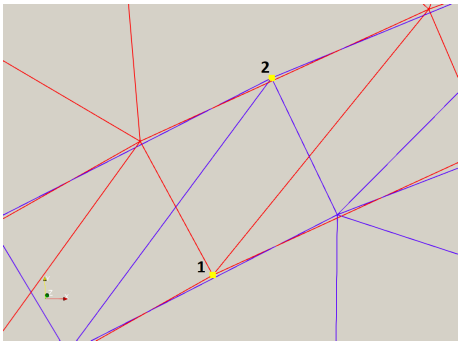

FIGURE 5 – Interpolation of hanging vertices (in yellow).

The Chimera approach proposed here is more complex to define and use than the MRF approach. However, no approximation is introduced to the model, and, with mesh refinement, the solution converges to the actual rotor-stator flow.

We are now equipped with two discrete methods for taking into account the rotation. A last adaptation is the choice of a turbulence model for the rotating context.

2.6 Spalart-Allmaras model

In the case of the Reynold Average Navier-Stokes (RANS) the Navier-Stokes equations are completed by a turbulence model defined by one supplementary equation. The laminar dynamic viscosity μ is replaced in the equations by the sum between the laminar and the turbulent dynamic viscosity $\mu + \mu_t$, and the laminar conductivity λ is replaced by the sum of the laminar and the turbulent conductivity $\lambda + \lambda_t$. The turbulent dynamic viscosity μ_t is given by the turbulence model and the turbulent conductivity is expressed from the Prandtl turbulent number. Here we choose

the following Spalart-Allmaras one equation turbulence model :

$$\partial_t(\rho\tilde{\nu}) + \mathrm{div}(\rho\mathbf{u}\tilde{\nu}) = \rho c_{b1}\tilde{S}\tilde{\nu} - \rho c_{w1} f_w \left(\frac{\tilde{\nu}}{d}\right)^2 + \frac{\rho}{\sigma}\left(\mathrm{div}((\nu+\tilde{\nu})\nabla\tilde{\nu}) + c_{b2}\|\nabla\tilde{\nu}\|^2\right).$$

The turbulent eddy viscosity is computed from :

$$\mu_t = \rho\tilde{\nu} f_{v1}, \quad \text{where} \quad f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3} \quad \text{and} \quad \chi = \frac{\tilde{\nu}}{\nu},\ \nu = \frac{\mu}{\rho}.$$

Additional definitions are given by the following equations :

$$\tilde{S} = \Omega + \frac{\tilde{\nu}}{\kappa^2 d^2} f_{v2} \quad \text{where} \quad \Omega = \|\mathbf{rot}\,\mathbf{u}\|.$$

Symbol d holds for the distance from the field point to the nearest wall and

$$f_{v2} = 1 - \frac{\chi}{1+\chi f_{v1}}.$$

The constants are

$$\sigma = \frac{2}{3},\ \ c_{b1} = 0.1355,\ \ c_{b2} = 0.622,\ \ \kappa = 0.41,\ \ c_{w2} = 0.3,\ \ c_{w3} = 2,\ \ c_{v1} = 7.1,$$
$$c_{w1} = \frac{c_{b1}}{\kappa} + \frac{1+c_{b2}}{\sigma}.$$

Finally, the function f_w is computed as :

$$f_w = g\left(\frac{1+c_{w3}^6}{g^6 + c_{w3}^6}\right)^{1/6} \quad \text{with} \quad g = r + c_{w2}(r^6 - r) \quad \text{and} \quad r = \min\left(\frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}, 10\right).$$

In our case we are interested in a version of the Spalart-Allmaras model with a correction term for the rotation. Spalart and Shur propose a simple modification of the one-equation transport turbulence model of Spalart and Allmaras [9, 8] , to make it more sensitive to rotational and curvature effects. It is a rather simple modification of the original model, the production term $\rho c_{b1}\tilde{S}\tilde{\nu}$ is multiplied by the "rotation function" f_{r1} :

$$f_{r1}(r^*, \tilde{r}) = (1+c_{r1})\frac{2r^*}{1+r^*}\left(1 - c_{r3}\tan^{-1}(c_{r2}\tilde{r})\right) - c_{r1}.$$

The nondimensional quantities r^* and $\tilde{r}$ are given by

$$r^* = \frac{S}{\Omega},$$
$$\tilde{r} = 2\frac{\Omega_{ik}S_{jk}}{D^4}\left(\frac{dS_{ij}}{dt} + (\varepsilon_{imn}S_{jn} + \varepsilon_{jmn}S_{in})\omega_m\right),$$

where the constants are $c_{r1} = 1$, $c_{r2} = 12$, $c_{r3} = 1$ and

$$S_{ij} = \frac{1}{2}\left(\partial_{x_j}v_i + \partial_{x_i}v_j\right), \quad \Omega_{ij} = \frac{1}{2}\left(\left(\partial_{x_j}v_i - \partial_{x_i}v_j\right) + 2\varepsilon_{mji}\,\omega_m\right),$$
$$S^2 = 2S_{ij}S_{ij}, \quad \Omega^2 = 2\Omega_{ij}\Omega_{ij}, \quad D^2 = \frac{1}{2}\left(S^2 + \Omega^2\right).$$

For the discretization of our Spalart-Allmaras model we integrate our equation on each cell C_i, by noting $f^s = \rho c_{b1}\tilde{S}\tilde{\nu} - \rho c_{w1} f_w \left(\frac{\tilde{\nu}}{d}\right)^2$ and using the Stokes formula

$$\frac{d}{dt}\int_{C_i} \rho\tilde{\nu}\,\mathrm{d}\mathbf{x} + \int_{\partial C_i} (\rho\mathbf{u}\tilde{\nu})\cdot\mathbf{n}_i\,\mathrm{d}\sigma = \int_{C_i}\frac{\rho}{\sigma}\left(\mathrm{div}((\nu+\tilde{\nu})\nabla\tilde{\nu}) + c_{b2}\|\nabla\tilde{\nu}\|^2\right)\mathrm{d}\mathbf{x} + \int_{C_i} f^s\,\mathrm{d}\mathbf{x}.$$

Now noticing that $\mathrm{div}(\tilde{\nu}\nabla\tilde{\nu}) = \|\nabla\tilde{\nu}\|^2 + \tilde{\nu}\Delta\tilde{\nu}$ and considering φ_i the P_1 Finite Element basis, we rewrite

$$\begin{aligned}\int_{C_i}\frac{\rho}{\sigma}\left(\mathrm{div}((\nu+\tilde{\nu})\nabla\tilde{\nu}) + c_{b2}\|\nabla\tilde{\nu}\|^2\right)\mathrm{d}\mathbf{x} &\approx \int_{C_i}\mathrm{div}\left(\frac{\rho_i}{\sigma}((\nu+(1+c_{b2})\tilde{\nu})\nabla\tilde{\nu}) + c_{b2}\tilde{\nu}_i\nabla\tilde{\nu}\right)\mathrm{d}\mathbf{x}\\ &= \sum_{j\in\mathcal{V}(i)}\int_{\partial C_i\cap\partial C_j}\left(\frac{\rho_i}{\sigma}((\nu+(1+c_{b2})\tilde{\nu})\nabla\tilde{\nu}) + c_{b2}\tilde{\nu}_i\nabla\tilde{\nu}\right)\cdot\mathbf{n}_i\,\mathrm{d}\sigma\\ &= -\sum_{K_j\in T(i)}\int_{K_j}\left(\frac{\rho_i}{\sigma}((\nu+(1+c_{b2})\tilde{\nu})\nabla\tilde{\nu}) + c_{b2}\tilde{\nu}_i\nabla\tilde{\nu}\right)\Big|_{K_j}\cdot\nabla\varphi_i\,\mathrm{d}\mathbf{x}.\end{aligned}$$

For the advection term we consider the following approximation

$$\int_{\partial C_i\cap\partial C_j}(\rho\mathbf{v}\tilde{\nu})\cdot\mathbf{n}_i\,\mathrm{d}\sigma \approx \Phi^{\rho}_{ij}(W_i,W_j,\mathbf{n}_{ij})\cdot\begin{cases}\tilde{\nu}_i & \text{if } \Phi^{\rho}_{ij}(W_i,W_j,\mathbf{n}_{ij})>0,\\ \tilde{\nu}_j & \text{otherwise,}\end{cases}$$

where Φ^{ρ}_{ij} is the density flux computed with the HLLC solver. Now considering

$$\tilde{\nu}|_K = \frac{1}{4}\sum_{\ell\in K}\tilde{\nu}_\ell, \quad \nabla\tilde{\nu}|_K = \sum_{\ell\in K}\tilde{\nu}_\ell\nabla\varphi_\ell$$

we write the approximation of the diffusion and dissipation terms as follows

$$\begin{aligned}&\int_{K_j}\left(\frac{\rho_i}{\sigma}((\nu+(1+c_{b2})\tilde{\nu})\nabla\tilde{\nu}) + c_{b2}\tilde{\nu}_i\nabla\tilde{\nu}\right)\Big|_{K_j}\cdot\nabla\varphi_i\,\mathrm{d}\mathbf{x} \approx\\ &\quad\approx |K_j|\frac{\rho_i}{\sigma}\left(\left((\nu|_{K_j}+(1+c_{b2})\tilde{\nu}|_{K_j})\nabla\tilde{\nu}|_{K_j}\right) + c_{b2}\tilde{\nu}_i\nabla\tilde{\nu}|_{K_j}\right)\cdot\nabla\varphi_i|_{K_j}.\end{aligned}$$

For source terms we consider the simple following discretization

$$\int_{C_i} f^s\,\mathrm{d}\mathbf{x} \approx |C_i|\rho_i\left(c_{b1}\tilde{S}_i\tilde{\nu}_i - c_{w1}f_w\left(\frac{\tilde{\nu}_i}{d_i}\right)^2\right).$$

3 Adaptation method

We adapt the mesh adaptation method describes in [6]. The metric is built directly with the absolute value of the Hessian of local Mach number.

3.1 The transient fixed Point

The simulation time frame $[0,T]$ is split into several subintervals :

$$[0,T] \;=\; [0=t_0,t_1]\cup\ldots\cup[t_i,t_{i+1}]\cup\ldots\cup[t_{n-1},t_n].$$

The whole algorithm is presented in Algorithm 1. *The external loop* goes from the first subinterval to the last one. On any subinterval $[t_i,t_{i+1}]$ is applied a fixed-point iteration, combining alternatively solution computation and mesh regeneration, which corresponds to the internal loop.

Algorithm 1 Transient $\mathbf{L}^{\infty}(0,T;\mathbf{L}^{p}(\Omega))$ fixed-point mesh adaptation algorithm

```
//- Loop over time subintervals i = 1, n_adap
```
For i=1,n_{adap}

```
    //- Solve adaptively on time subinterval S_i = [t_{i-1}, t_i]
    //- Fixed point adaptation loop
```
For j=1,n_{nptfx}

- $\mathcal{W}_{0,i}^{j}$ = `ConservativeSolutionTransfer`($\mathcal{H}_{i-1}^{j}, \mathcal{W}_{i-1}^{j}, \mathcal{H}_{i}^{j}$)
- $\mathcal{W}_{i}^{j}$ = `SolveStateForward`($\mathcal{W}_{0,i}^{j}, \mathcal{H}_{i}^{j}$)
- $\mathcal{M}_{i}^{j}$ = `ComputeFeatureOrientedMetric`($\varepsilon, \mathcal{W}_{i}^{j}, \mathcal{H}_{i}^{j}$)
- $\mathcal{H}_{i}^{j+1}$ = `GenerateAdaptedMeshe`($\mathcal{H}_{i}^{j}, \mathcal{M}_{i}^{j}$)

End for j

End for i

In the internal loop, knowing the spatial mesh, the time subinterval $[t_i, t_{i+1}]$ is divided into m time-integration intervals $[t_i^k, t_i^{k+1}]$, with $k = 0, m$ and $t_i^0 = t_i$, , $t_i^m = t_{i+1}$, see Figure 6. For any $k = 0, m-1$, the flow variables are initialised by an interpolation of the solution obtained at the end of previous subinterval and advanced by means of the numerical scheme from time level t_i^k to time level t_i^{k+1}. Then, a single metric for the whole subinterval $[t_i, t_{i+1}]$ can be defined by intersecting for all $k = 0, m$ the metrics associated with solutions at time level t_i^k. In practice, not all metrics are intersected, but a few tens of them. Then, a new adapted mesh is generated according to this metric and to the error threshold ε. The computation starts again from an interpolation of the solution obtained at the end of previous subinterval. This internal loop is stopped, as for steady case, when the deviation between two successive solutions at t_{i+1} is sufficiently small. The next action is performed by the *external loop* which starts the next internal fixed point adaptation loop for $[t_{i+1}, t_{i+2}]$. For each subinterval n_{adap}, each maximal metric $|\mathbf{H}_{\max}|_i^j\}$ is computed from the different variables and different time levels of the subinterval by applying a *metric intersection*, [2].

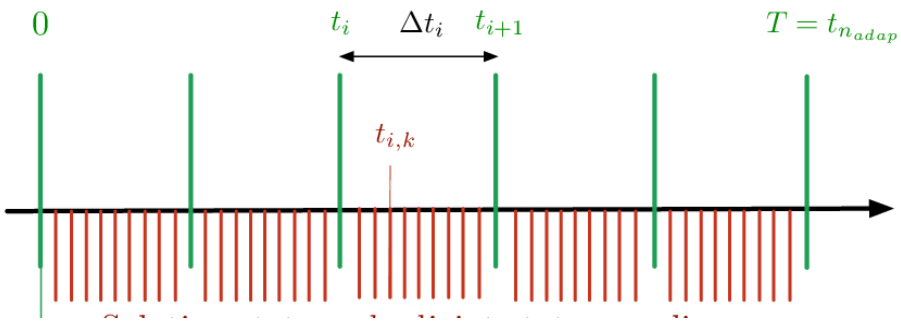

FIGURE 6 – Transient Fixed Point : the adaptive mesh is frozen on each time subinterval $[t_i, t_{i+1}]$. Time integration works on time intervals $[t_i^k, t_i^{k+1}]$.

3.2 Adaptation to steady

When the MRF method is choosen it is possible to get finally a steady flow. Only the inner fixed point adaptation loop of Algorithm 1 is applied in order to get the steady coupling of mesh and flow.

3.3 Adaptation to periodic

In the case of a *periodic flow* as arises frequently to rotor/stator flows, we propose to apply Algorithm 2 on a time interval equal to the period. After some convergence, Algorithm 2 can be followed by a 1-2 computation over a period without recomputing the sequence of meshes in order to allow a better stabilisation of the periodic solution.

Algorithm 2 Restarted Transient $\mathbf{L}^{\infty}(0,T;\mathbf{L}^{p}(\Omega))$ fixed-point mesh adaptation algorithm

`//- Loop over global adaptations` $m = 1, m_{adap}$
For m*=1,*m_{adap}
`//- Loop over time subintervals` $i = 1, n_{adap}$
For i*=1,*n_{adap}
`//- Solve adaptively on time subinterval` $S_i = [t_{i-1}, t_i]$
`//- Inner fixed point adaptation loop`
For j*=1,*n_{nptfx}
- $\mathcal{W}_{0,i}^{j} =$ `ConservativeSolutionTransfer`$(\mathcal{H}_{i-1}^{j}, \mathcal{W}_{i-1}^{j}, \mathcal{H}_{i}^{j})$
- $\mathcal{W}_{i}^{j} =$ `SolveStateForward`$(\mathcal{W}_{0,i}^{j}, \mathcal{H}_{i}^{j})$
- $\mathcal{M}_{i}^{j} =$ `ComputeFeatureOrientedMetric`$(\varepsilon, \mathcal{W}_{i}^{j}, \mathcal{H}_{i}^{j})$
- $\mathcal{H}_{i}^{j+1} =$ `GenerateAdaptedMeshe`$(\mathcal{H}_{i}^{j}, \mathcal{M}_{i}^{j})$

End for j

End for i

Update initial mesh and conditions
End for m

4 Numerical results

We now present our simulation results. We will compare the two methods presented in this report, namely the MRF method and the Chimere method, for a rotating cross.

4.1 Application to a rotating cross in a cylinder

For both methods the dimensions of the domains are the same, the only difference is the superposition of the rotor/stator part which is not present in MRF. This superposition is replaced by a fictitious boundary.

The domain is a disk of radius 2 meters and thickness 0.3 meter. A blade of the cross is 0.9 meter long and 0.2 meter wide.

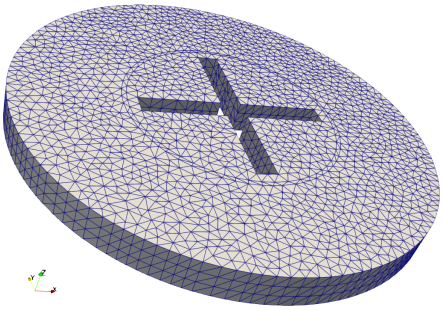

FIGURE 7 – 3D view of our domain.

The initial conditions of the following results are $\rho_0 = 1,2 kg.m^{-3}$ in the stator part, $\rho_0 = 1,4 kg.m^{-3}$ in the rotor part (see Fig.2), $\mathbf{v}_0 = 0$, $p_0 = 101325 Pa$, $T_0 = 288,15 K$ and the rotation speed is 1000 rpm. For the turbulence model we also initialize the Spalart variable at $10^{-6} m^2.s^{-1}$.

We will present and compare three simulation results, one simulation performed with the chimera method and two others with the MRF method. For the MRF, the two test cases correspond to a test with the interface close to the cross (MRF-C) and another with the interface far from the cross (MRF-F) (see Fig.8). For the second MRF test case the distance of the interface from the centre of the domain has increased from a radius of 1.05 metres to 1.55 metres, note that the domaine near the cross (the red domain) has been preserved for the initial condition on density.

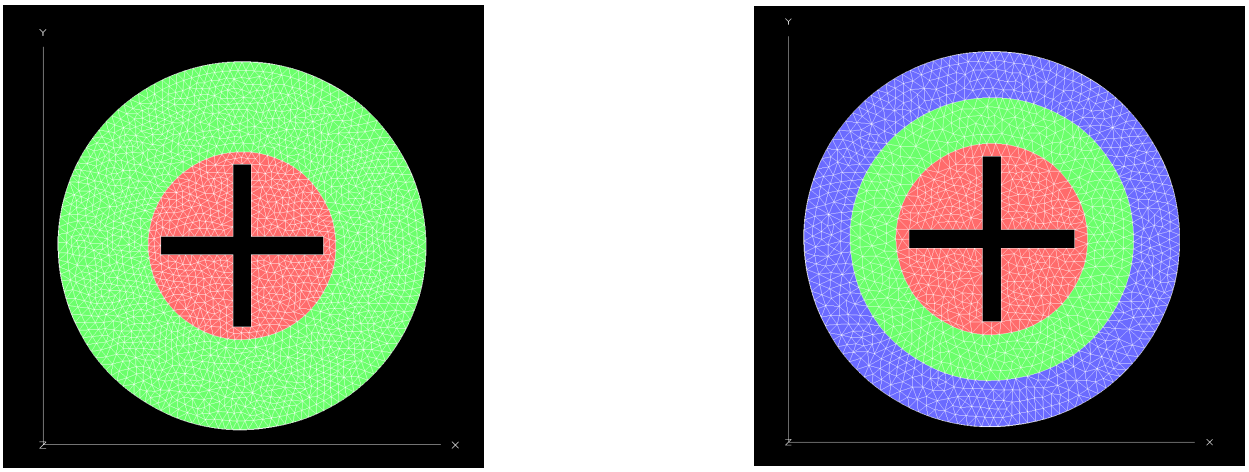

FIGURE 8 – On the left view of the MRF domaine with interface close to the cross and on the right view of the MRF domaine with interface far from the cross.

The physical time of the simulations presented in the following is 0.165s, this time corresponds exactly to 2.75 turns of the cross. The number of points of the initial meshes is around 7.500 for the three simulations and at the physical time 0.165s, the number of points of the meshes for MRF-C, MRF-F and Chimera are, respectively, 15.126, 14.271 and 17.360 .

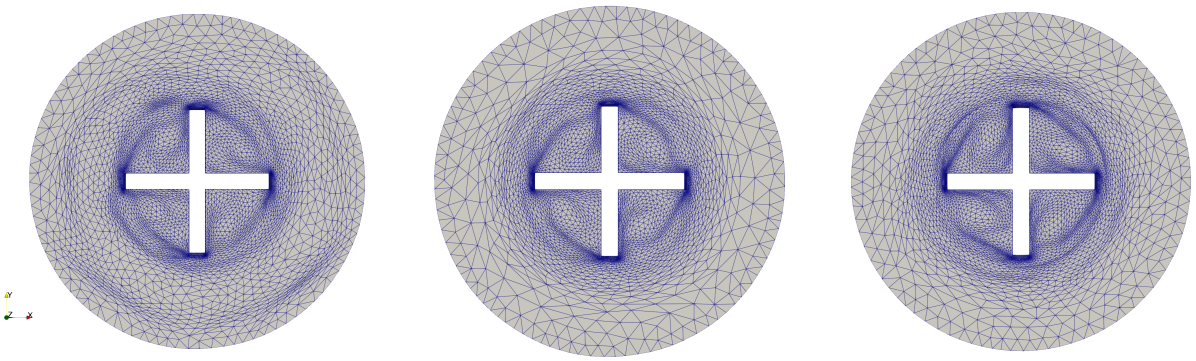

FIGURE 9 – Mesh representation for MRF-C (left), MRF-F (center) and Chimera (right).

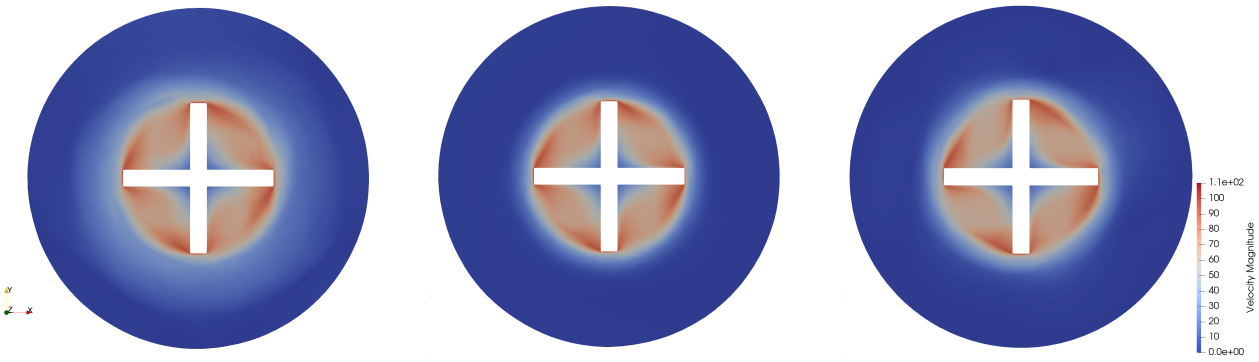

FIGURE 10 – Velocity magnitude representation for MRF-C (left), MRF-F (center) and Chimera (right).

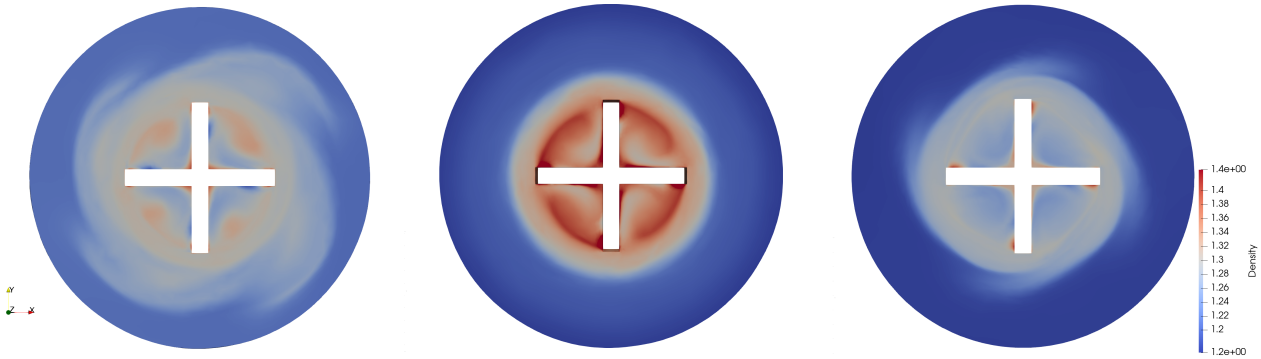

FIGURE 11 – Density representation for MRF-C (left), MRF-F (center) and Chimera (right).

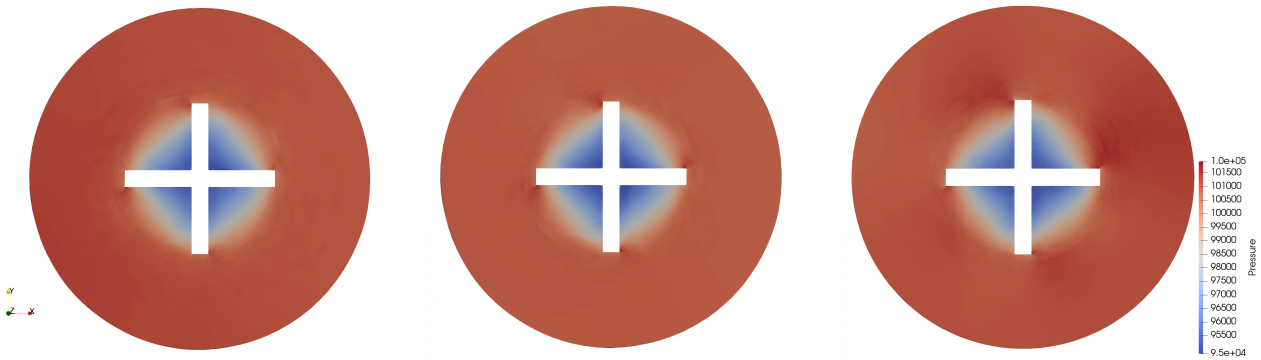

FIGURE 12 – Pressure representation for MRF-C (left), MRF-F (center) and Chimera (right).

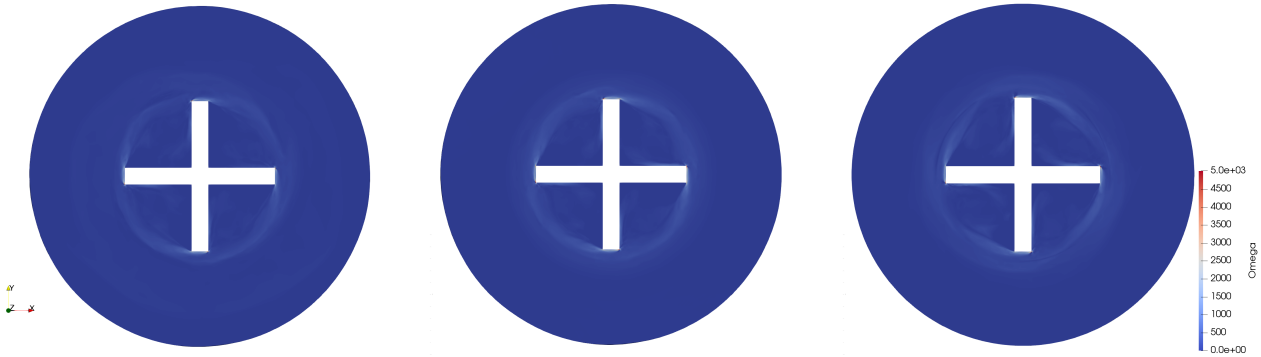

FIGURE 13 – Vorticity representation for MRF-C (left), MRF-F (center) and Chimera (right).

Except for density, the similarity of the results seems encouraging. For the density one can discuss the influence of the distance of the interface from the cross, which has an important influence on the start of the computation (see Fig.14, Fig.15, Fig.16 and Fig.17). For the MRF method, closer the interface is to the cross, longer the flow takes to stabilise.

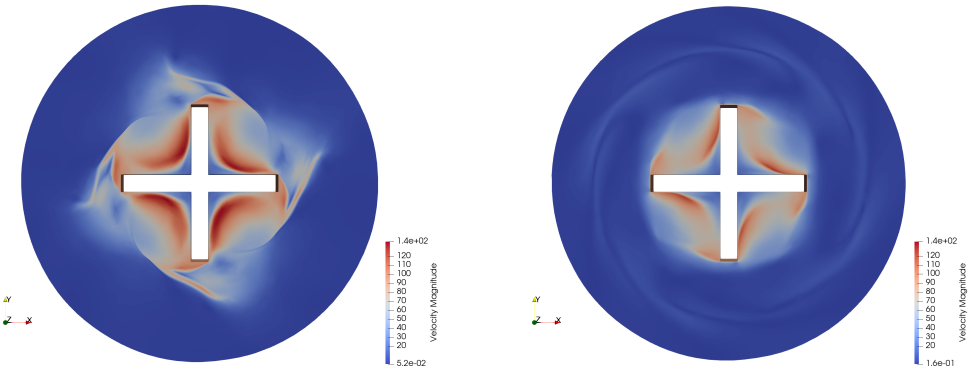

FIGURE 14 – Velocity magnitude comparisons of the MRF-C (left) and the MRF-F (right) at physical time 0.03s

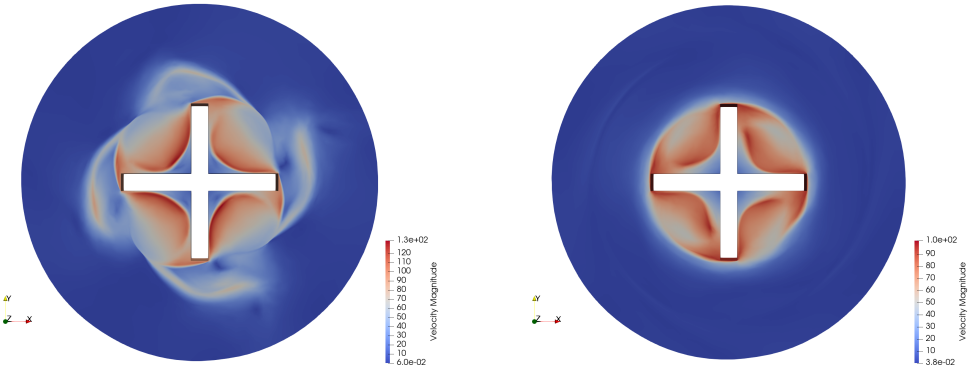

FIGURE 15 – Velocity magnitude comparisons of the MRF-C (left) and the MRF-F (right) at physical time 0.05s

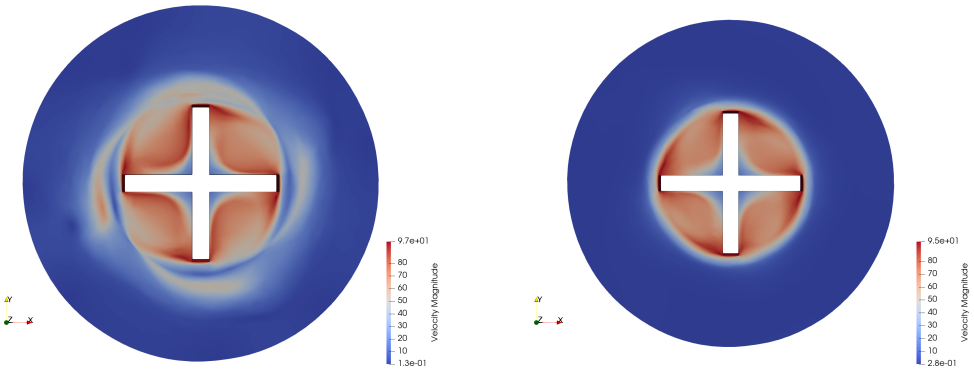

FIGURE 16 – Velocity magnitude comparisons of the MRF-C (left) and the MRF-F (right) at physical time 0.1s

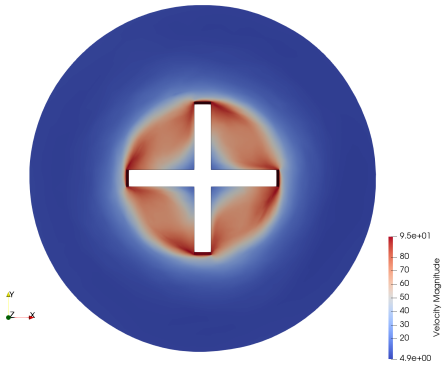

FIGURE 17 – Velocity magnitude for the MRF-C at physical time 0.6s

5 Concluding remarks

Two numerical methods for a rotor-stator flow have been derived and compared, in combination with a mesh adaptation method. The MRF method is a robust and efficient approach, with an approximation keeping from predicting the fluctuation of flow on the stator domain, due to the rotation of the geometry in the rotor domain. The Chimera method presented is less robust and therefore less efficient, but it approximates accurately the initial problem. Both methods couple with mesh adaptation in good conditions. The MRF allows reaching a steady flow and then an easier mesh adaptation with an unique mesh. The coupling of Chimera and adaptation is mode delicate but the proposed algorithm works satisfactorily.

6 Acknowledgements

This work is done in the ANR project NORMA which is supported by the french ministery of Research under contract ANR-19-CE40-0020-01.

7 Appendix

We want to show the following relationship

$$\int_K \nabla\varphi_i \, \mathrm{d}x = -\int_{\partial C_i \cap K} \mathbf{n} \, \mathrm{d}\sigma \, .$$

Let $\Omega \subset \mathbb{R}^d$ be a bounded domain with $d = 2,3$. Let $\mathcal{T}_h$ be a conforming triangulation or tetrahedralization of Ω, and $K \in \mathcal{T}_h$ a genetic element. For each vertex i we build a dual cell C_i according to the median rule.

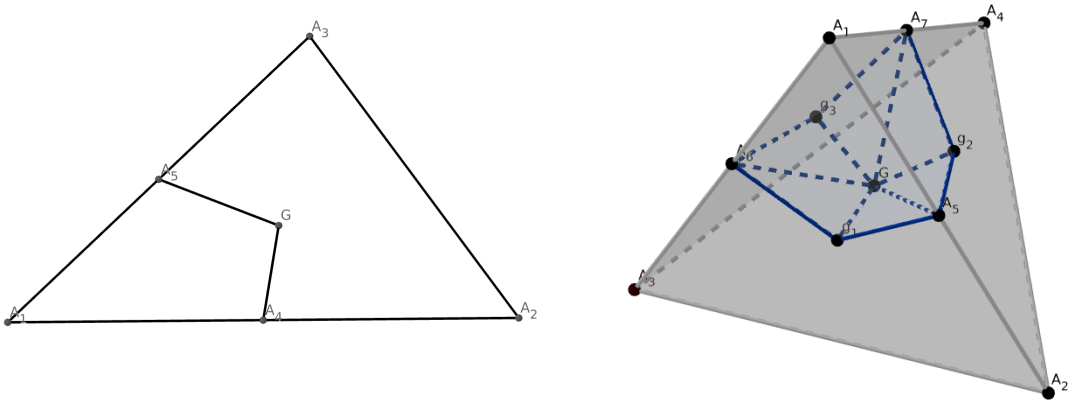

FIGURE 18 – 2D and 3D element with dual cell.

Notations :

- In $2D$:
  - $K = A_1A_2A_3$
  - $\partial K = A_1A_2 \cup A_1A_3 \cup A_2A_3$
  - $\partial C_i \cap K = A_4G \cup A_5G$
  - $C_i \cap \partial K = A_1A_4 \cup A_1A_5$

- In $3D$:
  - $K = A_1A_2A_3A_4$
  - $\partial K = A_1A_2A_3 \cup A_1A_2A_4 \cup A_1A_3A_4 \cup A_2A_3A_4$

- $\partial C_i \cap K = A_6 g_1 G g_3 \cup A_5 g_1 G g_2 \cup A_7 g_2 G g_3$
- $C_i \cap \partial K = A_1 A_6 g_1 A_5 \cup A_1 A_5 g_2 A_7 \cup A_1 A_7 g_3 A_6$

First, let's consider a matrix $\mathbf{\Psi}$ of size 3×3. Using Stokes' formula we have

$$\int_{C_i \cap K} \operatorname{div} \mathbf{\Psi} \, \mathrm{d}\mathbf{x} = \int_{\partial C_i \cap K} \mathbf{\Psi}\, \mathbf{n} \, \mathrm{d}\sigma + \int_{C_i \cap \partial K} \mathbf{\Psi}\, \mathbf{n} \, \mathrm{d}\sigma,$$

so, taking $\Psi = \mathbb{I}_3$

$$-\int_{\partial C_i \cap K} \mathbf{n} \, \mathrm{d}\sigma = \int_{C_i \cap \partial K} \mathbf{n} \, \mathrm{d}\sigma = \mathbf{n}_{|C_i \cap \partial K} \, |C_i \cap \partial K|.$$

Furthermore, we have

$$\int_K \nabla \varphi_i \, \mathrm{d}\mathbf{x} = \int_{\partial K} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x}.$$

• In dimension 2 :

$$\begin{aligned}\int_{\partial K} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} &= \int_{A_1A_2} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} + \int_{A_1A_3} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} + \int_{A_2A_3} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x}, \\ &= \mathbf{n}_{|A_1A_2} \int_{A_1A_2} \varphi_i \, \mathrm{d}\mathbf{x} + \mathbf{n}_{|A_1A_3} \int_{A_1A_3} \varphi_i \, \mathrm{d}\mathbf{x} + \mathbf{n}_{|A_2A_3} \int_{A_2A_3} \varphi_i \, \mathrm{d}\mathbf{x},\end{aligned}$$

in $2D$ the basis functions $\mathbb{P}_1$ are the hat functions which are equal to 1 on their respective vertex and 0 on adjacent vertices. Thus, on an edge, the calculation of the integral can be reduced to the calculation of the area of a right-angled triangle of height 1, with a base of $|A_1A_2|$ or $|A_1A_3|$. Note that φ_i is zero on A_2A_3. We have

$$\begin{aligned}\int_{\partial K} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} &= \frac{1}{2} \mathbf{n}_{|A_1A_2} \, |A_1A_2| + \frac{1}{2} \mathbf{n}_{|A_1A_3} \, |A_1A_3|, \\ &= \mathbf{n}_{|C_i \cap \partial K} \, |C_i \cap \partial K|,\end{aligned}$$

car $\frac{1}{2}|A_1A_2| = |A_1A_4|$ et $\frac{1}{2}|A_1A_3| = |A_1A_5|$.

• In dimension 3 :

$$\begin{aligned}\int_{\partial K} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} &= \int_{A_1A_2A_3} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} + \int_{A_1A_3A_4} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} + \int_{A_1A_2A_4} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} + \int_{A_2A_3A_4} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x}, \\ &= \mathbf{n}_{|A_1A_2A_3} \int_{A_1A_2A_3} \varphi_i \, \mathrm{d}\mathbf{x} + \mathbf{n}_{|A_1A_3A_4} \int_{A_1A_3A_4} \varphi_i \, \mathrm{d}\mathbf{x} + \mathbf{n}_{|A_1A_2A_4} \int_{A_1A_2A_4} \varphi_i \, \mathrm{d}\mathbf{x} \\ &+ \mathbf{n}_{|A_2A_3A_4} \int_{A_2A_3A_4} \varphi_i \, \mathrm{d}\mathbf{x},\end{aligned}$$

we have the same arguments as in $2D$, this time the integral over a triangle is reduced to the calculation of the volume of a tetrahedron of height 1 and base equal to $|A_1A_2A_3|$, $|A_1A_3A_4|$ or $|A_1A_2A_4|$, φ_i is zero $A_2A_3A_4$. So

$$\begin{aligned}\int_{\partial K} \mathbf{n}\, \varphi_i \, \mathrm{d}\mathbf{x} &= \frac{1}{3} \mathbf{n}_{|A_1A_2A_3} \, |A_1A_2A_3| + \frac{1}{3} \mathbf{n}_{|A_1A_3A_4} \, |A_1A_3A_4| + \frac{1}{3} \mathbf{n}_{|A_1A_2A_4} \, |A_1A_2A_4|, \\ &= \mathbf{n}_{|C_i \cap \partial K} \, |C_i \cap \partial K|,\end{aligned}$$

because $|A_1A_5g_1A_6| = \frac{1}{3}|A_1A_2A_3|$, $|A_1A_5g_2A_7| = \frac{1}{3}|A_1A_2A_4|$ and $|A_1A_6g_3A_7| = \frac{1}{3}|A_1A_3A_4|$.

□

Références

[1] F. Alauzet. *Adaptation de maillage anisotrope en trois dimensions. Application aux simulations instationnaires en Mécanique des Fluides*. PhD thesis, Université Montpellier II, Montpellier, France, 2003. (in French).

[2] F. Alauzet, P.J. Frey, P.L. George, and B. Mohammadi. 3D transient fixed point mesh adaptation for time-dependent problems : Application to CFD simulations. *J. of Computational Physics*, 222 :592–623, 2007.

[3] L. Dubuc, F. Cantariti, M. Woodgate, B. Gribben, K.J. Badcock, and B.E. Richards. A grid deformation technique for unsteady flow computations. *Int. J. Num. Meth. Fluids*, 32 :285–311, 2000.

[4] C. Farhat. High performance simulation of coupled nonlinear transient aeroelastic problems. Special course on parallel computing in CFD R-807, NATO AGARD Report, October 1995.

[5] C. Farhat, P. Geuzaine, and C. Grandmont. The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. *J. of Computational Physics*, 174 :669–694, 2001.

[6] D. Guégan, O. Allain, A. Dervieux, and F. Alauzet. An l^∞-l^p mesh adaptive method for computing unsteady bi-fluid flows. *IJNME*, 84(11) :1376–1406, 2010.

[7] B. Nkonga and H. Guillard. Godunov type method on non-structured meshes for three-dimensional moving boundary problems. *Comput. Methods Appl. Mech. Eng.*, 113(1-2) :183–204, 1994.

[8] M.L. Shur, M.K. Strelets, A.K. Travin, and P.R. Spalart. Turbulence modeling in rotating and curved channels : Assessing the Spalart-Shur correction. *AIAA Journal*, 38(5) :784–792, 2000.

[9] P.R. Spalart and M.L. Shur. On the sensitization of turbulence models to rotation and curvature. *Aerospace Science and Technology*, 5 :297–302, 1997.

[10] J.R. Taylor. *Classical Mechanics*. University Science Books, 2005.

RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399