POLITECNICO DI MILANO

Scuola di Ingegneria Industriale e dell'Informazione

Corso di Laurea Magistrale in Ingegneria Aeronautica

Dipartimento di Scienze e Tecnologie Aerospaziali

POLITECNICO

MILANO 1863

Numerical Characterization of Porous Screens: A Comparison Between Explicit Modeling and Porous Media Approach

Relatore:
Paolo Schito

Candidato:
Gianmarco Cicolini
Matricola:
905732

ANNO ACCADEMICO 2020/2021

Contents

Nomenclature

V_{ref} Reference Volume

α Angle of inclination of the bisector plane

β Porosity

ϵ Turbulent Dissipation

ρ Density

$\mathbf{D}$ DarcyForchheimer Viscous Tensor

$\mathbf{F}$ DarcyForchheimer Inertial Tensor

θ Wind Deflection with respect the normal axis to the screen

A Area of the Screen

d Diameter of the Holes

d_{ij} Viscous Tensor Component

f_{ij} Inertial Tensor Component

F_x Horizontal Force Component Acting on The Screen

F_y Vertical Force Component Acting on The Screen

F_z Normal Force Component Acting on The Screen

I Turbulence Intensity

K Pressure Loss Coefficient

k Turbulent Kinetic Energy

U_x Horizontal Speed Component

U_y Vertical Speed Component

U_z Axial Speed Component

$\mathbf{U}$ Flow Velocity

List of Figures

5

List of Tables

Abstract

Over the years new technologies based on the application of porous screens have been developed. Their capability of changing the properties of the flow in terms of pressure and speed orientation makes them always an interesting object to be investigated. Current applications in civil engineering show their use as an external skin for the buildings aimed to screen solar light and to interact with the incoming wind.

This second effect is studied in this work by resorting to Computational fluid dynamics as principal tool of investigation. Numerical simulations have become a valid and less expensive alternative to experimental tests in wind tunnels: the starting idea is to verify how much numerical results are in agreement with experimental ones and with the existing literature concerning this point for different screen geometries.

Successively a porous media approach has been developed as a way alternative to the explicit simulations: the aim is to reproduce the effects of porous screens without modeling them in the numerical domain. Algebraically these aerodynamic effects are reconstructed with the Darcy Forchheimer model and inserted inside Navier-Stokes momentum equation as a sink term: this term would represent the pressure gradient produced by the presence of a porous media inside the flow. The outcome is a new solver that is able to reproduce the effects of the screen without the physical presence of it.

The results show the capability of the solver to predict the behaviour of the porous screen both for 2D and 3D geometries especially for the conditions in which the maximum loads are registered.

Sommario

Nel corso degli anni sono state sviluppate nuove tecnologie basate sull'applicazione di superfici porose. La loro capacità di cambiare le proprietà dell'aria in termini di pressione e direzione della velocità le rende sempre degli interessanti oggetti di investigazione. Un'attuale applicazione nell'ingegneria civile le vede utilizzate come superfici esterne per gli edifici con lo scopo principale di schermare la luce solare e di interagire con il vento incidente. Questo secondo effetto è studiato in questo lavoro attraverso l'utilizzo della fluidodinamica computazione come principale strumento di calcolo. Le simulazioni numeriche sono diventate una valida e meno costosa alternativa agli studi sperimentali in galleria del vento: l'idea iniziale è quella di verificare quanto i risultati numerici siano in accordo con quelli sperimentali e con la l'esistente letteratura a riguardo. Successivamente è stato sviluppato il porous media approach come alternativa alle simulazione esplicite: l'obiettivo è quello di riprodurre gli effetti degli schermi porosi senza modellarli nel dominio computazionale. Algebricamente questi effetti aerodinamici sono ricostruiti con il modello di Darcy-Forchheimer ad inseriti all'interno dell'equazione di quantità di moto delle equazioni diNavier-Stokes come termine sorgente: questo termine rappresenterebbe il gradiente di pressione prodotto dalla presenza del mezzo poroso dentro al flusso. Il risultato è un nuovo solutore che è capace di riprodurre gli effetti dello schermo senza la sua presenza fisica. I risultati mostrano la capacità del solutore di predire il comportamento degli schermi porosi sia per le geometrie 2D che per quelle 3D specialmente per le condizioni in cui sono registrati i carichi più elevati.

9

1 Introduction

Porous screens are not such a recent technology. Their application can be seen since 50's when they started to be used inside the first wind tunnels to reduce the main differences in speed of the fluid flow before the test chamber. Prandtl[9] seems to be the first to describe this effect and to find an expression for the reduction of these differences in terms of a pressure loss coefficient. This application is still present and has been improved until now. Contemporary over the years porous structures have found more areas of application spacing from civil engineering to environmental one.

A new possible area of interest has been recently considered and still regards the aerodynamic field. It considers the potential reduction of the wind loads acting on buildings by the installation of porous screens on their surfaces. As stated by Richards[13] the wind loads acting on porous screens are almost the same of those acting on the solid structures, however these geometries are able to change the pressure distribution when the flow passes through them and therefore requires consideration.

The aim of this work is the study of the properties of different geometries in terms of the pressure jump provided and of the forces acting on their surfaces through the use of CFD as main instrument of investigation.

However, launching a numerical simulation in a domain in which there is both the building and the porous screen would require a very thin mesh, and so an high and unnecessary computational costs. This because the characteristic length scale of the screen is far more small than that of the building, and so this last one would be modelled with a mesh thicker than the necessary for its dimensions.

By basing on these considerations, follows the definition of a porous media approach, which consists in a new numerical model based on the modification of the Darcy-Forchheimer model that is able to reproduce the effects of porous screens on the fluid flow for different wind flow directions by performing numerical simulations without the presence of the real screen inside the numerical domain. This could avoid the necessity to have a very thin mesh, essential to represent the geometry of the screen, saving then time and computational costs.

About this approach different studies are present in the existing literiture: O.Bistoni, G. Pomaranzi, P.Schito and A.Zasso[15] as well as Buscemi[8] developed this new numerical model for porous media validated with numerical experiments, that is able to represent the aerodynamic characteristics of the screens without representing them in the numerical domain.

A way to represent the porous media was proposed by Xu, Patronu, Lo and De Miranda[12]. They tried to model the presence of the surface inside the numerical domain through appropriate pressure jumps. With this purpose they also proposed a new relation which links the pressure jump with the porosity of the screen.

The actual situation already talks about the adoption of such technology for buildings and skyscrapers surfaces, mainly employed for solar light screening and privacy services.

An example is the O-14 skyscraper in Dubai (fig.38): Kheir Al-Kodmany, in *Sustainability and the 21st Century Vertical City: A Review of Design Approaches of Tall Buildings*[7] refers that O-14 "building's unique exterior serves as solar screens that protect the glass curtain wall from the harsh desert sun, while providing daylight and framing views to the outside" as well as "the facade also serves as a structural exoskeleton that bears the lateral loads of the tower".

Figure 1: O-14, Dubai

As said before investigations on porous screens have already been performed: Taylor was the first one to build a theoretical model that links the resistance produced by a damping screen with its porosity. After him Collar[1], still Taylor and Davies[5] theorized new formulas to analyze the reduction of the differences in mean speed: Taylor himself discovered that through the screen there was not only a change in pressure but also a deflection of the flow if the stream passed obliquely. Following this considerations Vahl Davis[4] tried to predict the aerodynamic effects of porous wire screens normal or inclined to the incident wind, with the purpose of classifying them on the basis of some characteristic parameters. He ended up with the consideration that two fundamental parameters are needed to define the properties of them: the pressure drop produced by the screen and the deflection

of the flow after it.

Then over the years the employment of porous screens has increased rapidly by arriving to appear in various branches of engineering. Different in fact are the articles focused on this technology and the wide panorama of possible applications for these surfaces makes the literature on the object really diversified.

Klementina Gerova[6] studied honeycombs employed as heat exchangers to observe the pressure they could generate for different inclinations to the free stream, finding that with the increase of the porosity and inclination the pressure jump decreases while the permeability shows the opposite trend.

Effects of inclination and porosity variation were studied by Teitel[11] and Xu[14] for different heights of installations of porous screens, basing their attention on the values of the forces registered on the screen as well as the deflection of the fluid flow downstream.

Similarly to Teitel also LetchFord[10] focused his attention on screens placed at variable height. In this case the objects of the paper are hoardings and signboards installed on buildings that are object of a varying wind speed and orientation.

In this sens P.J.Richards and M.Robinson[13] found a correlation between the pressure loss coefficient and the inclination of the incoming wind. This results are consistent with those visualized in this work taking into account wind orientation and different screens, mainly slabs with rounded holes and woven wire screens.

Also the ocean engineering is a branch of investigation: Patursson[2] investigated the pressure loss through a grid modelled with woven cylinders connected by knots with the aim of understanding the exchanges of water in cages for aquaculture.

For what concerns the maintenance and the preservation of an ambient dedicated to the farming activity, greenhouses are extremely dedicated to the ventilation of the internal environment maintaining at the same time the correct concentration of humidity, the correct level of life conditions and impeding the incoming of undesired insects. For this purpose actual greenhouses are thought as a complete structure composed by porous tiltable fences that let pass the desired quantity of air and solar light. A similar geometry was inspected by Mitchell Cohen[3] with the purpose of avoiding human intrusion and at the same time guaranteeing air ventilation. Its result was the discovery of a correlation between the diameter and the spacing with the pressure drop.

Actual examples of civil applications can be seen in the new Bocconi Campus, which has a porous covering for all its buildings or the Unicredit Tower in Milan, whose spire at the top has an external skin made of this material (Figure 2b).

(a) Bocconi Campus (Milan) (b) Unicredit Tower (Milan)
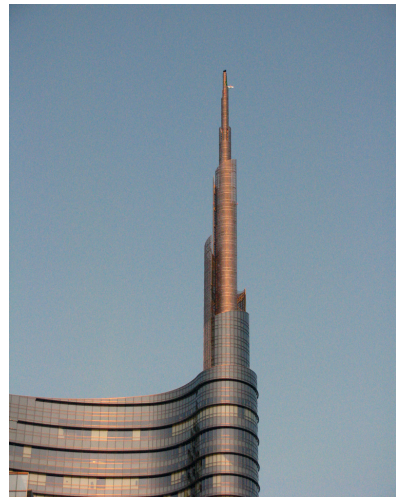
Figure 2: Actual application of porous screen technology

2 Theoretic Model

As stated in the introduction, the aim of this work is the definition and validation of the porous media approach. It consists in the computation of a new Open-Foam solver starting from an existing one which is porous SimpleFoam. Porous SimpleFoam is generally used to treat problems of flow permeability inside a porous media like gauzes and honeycombs. For this reason the solver lets the user the freedom to modify the porosity by specifying in input some characteristics parameters and to choose the porosity model that best fits with the nature of the problem. In accordance with the work [15] the porosity model chosen is the Darcy-Forchheimer one.

2.1 Darcy's Law - 1D Case

Darcy's law is a constitutive equation that describes the motion of a fluid flow in a porous medium. This equation was derived as an expression of conservation of momentum by Henry Darcy in 1856 by basing on the results of his experiments on water through beds of sand. "It is analogous to the Fourier's law in the field of heat conduction, Ohm's law in the field of electrical networks, or Fick's law in diffusion theory". Darcy's empirical flow model represents a simple linear relationship between flow rate and pressure drop in a porous medium and is expressed generally as:

$$q = \frac{kA}{\mu}\frac{dp}{dx} \tag{1}$$

It is necessary to specify that Darcy's Law assumes laminar or viscous flow and it does not involve the inertia term. This implies that the inertia or acceleration forces in the fluid are being neglected when compared to the classical Navier-Stokes equations.
Besides it also considers that in a porous medium a large surface area is exposed to fluid flow, hence the viscous resistance will greatly exceed acceleration forces in the fluid unless turbulence sets in.
Darcy's flow model is then a fluid flow described by the eq.1 and characterized by these properties. Any deviation from these conditions is considered as a non-Darcy flow. An increase in the velocity of the flow can result in a departure from the laminar condition that can change the properties depicted before moving towards a different turbulent kind of flow.

2.2 Darcy-Forchheimer Equation - 1D Case

In 1901, Philippe Forchheimer discovered that the relationship between flow rate and potential gradient is nonlinear at sufficiently high velocity, and that this non-linearity increases with flow rate. He initially attributed this non-linear increase to turbulence in the fluid flow (it is now known that this non-linearity is due to inertial effects in the porous media).
He stated that there was an additional pressure drop due to inertial losses primarily because of the acceleration and deceleration effects of the fluid as it travels through the tortuous flow path of the porous media. For a mono-dimensional case the total pressure drop is thus given by Forchheimer empirical flow model stated traditionally as:

$$\frac{dp}{dx} = \frac{\mu}{k}v + \beta\rho v^2 \tag{2}$$

where μ and k are respectively the dynamic viscosity and the permeability of the flow, β is called inertial factor and ρ is the density of the flow, where v is the speed of the flow.
The Forchheimer equation assumes that Darcy's law is still valid, but that an additional term must be added to account for the increased pressure drop. Hence this equation will be called the Darcy-Forchheimer flow.

2.3 Darcy-Forchheimer Model - 3D Case

Darcy-Forchheimer model consists in the rewriting of Navier-Stokes momentum equation with the addition of a sink term at the second member of the equation:

$$\frac{\partial(\rho\mathbf{U})}{\partial t} + \nabla(\rho\mathbf{U}\mathbf{U}) = \nabla \cdot \boldsymbol{\sigma} + \boldsymbol{S_m} \tag{3}$$

Where σ is the Cauchy Stress Tensor. S_m is the aforementioned sink term which takes the form:

$$\boldsymbol{S_m} = -(\mu\mathbf{D} + \frac{1}{2}\rho tr(\mathbf{U} \cdot \mathbf{I})\mathbf{F})\mathbf{U} \tag{4}$$

$\mathbf{D}$ and $\mathbf{F}$ are respectively the Darcy-Forchheimer viscous and inertial tensors. They need to be estimated for the problem in exam or specified directly from the user in order to obtain the desired porosity effects inside the domain. As the fluid flows moves from a state in which the pressure is higher to a state where the pressure is lower the sink term assumes a negative sign.
The kind of flow here presented is a turbulent flow that works in a range of speed

representative of the wind motion, therefore wind speeds starting from $10\frac{m}{s}$ has been mainly considered. By taking into account the characteristic dimensions of the holes of the different porous screens analyzed, we can compute the Reynolds' Number and see that its value is almost of 4000 units. This means we work in a range in which the inertial forces dominate the physic of the problem with respect to the viscous ones. It is possible then to neglect in first approximation the viscous tensor $\mathbf{D}$ of the Darcy-Forchheimer model because the dominant element inside the sink term is the inertial component represented by the $\mathbf{F}$ tensor. So equation 4 can be rewritten as:

$$\boldsymbol{S_m} = -(\frac{1}{2}\rho tr(\mathbf{U}\cdot\mathbf{I})\mathbf{F})\mathbf{U} \tag{5}$$

2.4 Pressure Loss Coefficient

The existing literature talks about a wide variety of screens. They can differ for their shape, dimension and porosity or still they can have the same shape but different size. Due to this heterogeneity, a possible approach could be the definition of a parameter that would keep into account these differences in shape and geometries, but also the intensity of the forces acting on the screens, the incident speed, (so the dynamic pressure), and if possible verify if there are some conditions for which it is independent from these variables and indicative of the porosity level. This coefficient, called pressure loss coefficient, can be defined in different ways depending on the user requirements (principally on which incident speed is considered, the undisturbed speed or the normal component to the screen). This is only one of the possible parameters which can describe a porous medium: the simplest way to define it is to consider only the normal component of the forces acting on the porous screen. By following the approach depicted in the works of Richards[13] and Xu[14] the pressure loss coefficient is defined as the ratio between the normal force acting on the screen to the area of it divided by the dynamic pressure referred to the undisturbed speed of the fluid flow as presented in equation 6:

$$K = \frac{f_z}{\frac{1}{2}\rho U^2 A} \tag{6}$$

Its definition comes from the expression of the pressure loss through the screen. The pressure drop is in fact defined as the product of a loss coefficient K for the volume flow rate per unit area

$$\Delta p = K\frac{1}{2}\rho V^2 \tag{7}$$

where ρ is the the density of air (1,225 $\frac{Kg}{m^3}$) and V is the volume flow rate of air per unit area expressed in $\frac{m}{s}$.

This approach can be sufficient for those kind of flows that are orthogonal to the screen as well as for screens that don't generate important lateral force components. In this conditions pointed out by Richards[13] the pressure loss coefficient shows sensitivity to the variation of some parameters like porosity coefficient β, Reynolds number and geometry configuration such that

$$K = K(\beta, Re, construction) \tag{8}$$

The porosity coefficient is defined as the ratio between the open area of the screen and the total area covered by it:

$$\beta = \frac{Open\,Area\,to\,the\,flow}{Total\,area\,of\,the\,screen} \tag{9}$$

while the Reynolds number is defined as:

$$Re = \frac{\rho U d}{\mu} \tag{10}$$

where U is the speed of the undisturbed flow, d is the diameter of the holes of the screen and μ is the cinematic viscosity. Moreover it is possible to verify that, keeping fixed all the other parameters and varying only the porosity, the loss coefficient shows a decreasing exponential trend by increasing it.

In this phase of the work two different approaches for the simulations have been analyzed:

- Constant speed orientation-varying speed magnitude

- Constant speed magnitude-varying speed orientation

The first trance of simulation is aimed to investigate the pressure drop variations with a 90° incident wind on the porous screen with various intensity. Literature's reviews on the argument show that over a certain Reynolds number the pressure loss coefficient starts being constant with a variation that is not so remarkable. In other words, in the range of wind speed between $0\frac{m}{s}$ and $10\frac{m}{s}$ K coefficient has a decaying trend and then starts flattening over a constant value once overtaken this limit.

2.5 Effects Of The Screen Presence On The Fluid Flow Motion - Vena Contracta

Before treating all the results regarding the numerical simulations it is convenient to explain the physical effects that the presence of the screen provides on the

fluid motion. Being an obstacle, whatever is the geometry, it creates a blockage on the fluid flow, generating consequent normal and shear stresses on the surface of the screen. In this sense, the streamlines of the fluid flow are going to impact and arrest on the part of screen that is full. The remaining part flows inside the holes.

For a simple geometry like rounded holes the streamlines located at the same height of the hole pass through it, those immediately nearby can deflect their direction and flow inside it too, while the most distant finish to impact on the screen. The physics of this phenomenon can be extended for all the holes that compose the screen and can be studied in terms of the pressure loss provided before and after the screen.

What happens instead inside and immediately after the hole is an effect that is generally called *vena contracta* phenomena.

This effect is treated generally for problems in which reservoirs with lateral holes are involved and a liquid jet passes into them. By visualizing the streamlines of the liquid flow it is possible to see that the section of the outgoing jet is less than the section of the hole.
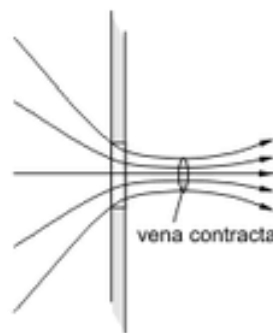
Figure 3: Vena Contracta simplification

There is a gradual contraction of the fluid flow until the smallest section is reached. The same situation can be visualized for the problem in exam. The fluid flow that does not flow inside the hole is interested in a re-circulation located nearby the solid body, the other part contracts inside the hole until a certain distance after it and then turns back to expand (in this case the expansion is intended in the divergence of the streamlines).

For cases in which there is a sudden restriction the loss of energy associated to the flow is to be attributable to a presence of separation inside the hole in proximity of the walls and to the restriction of the flux tube. These effects can be identified as concentrated pressure losses that mathematically can be expressed with a formula

18

similar to 11:

Figure 4: Flux tube inside a hole

These effects can be identified as concentrated pressure losses that mathematically can be expressed with a formula similar to 11:

$$\Delta p = K \frac{1}{2} \rho v^2 \tag{11}$$

where the reference speed is that one corresponding to the minimum section of flow tube.

For what concerns the problem in exam, once the numerical simulation has finished it is possible to visualize the flow field streamlines and their trajectory in proximity of the holes.

(a) R6T8: contraction of the streamlines   (b) R6T8: zoom
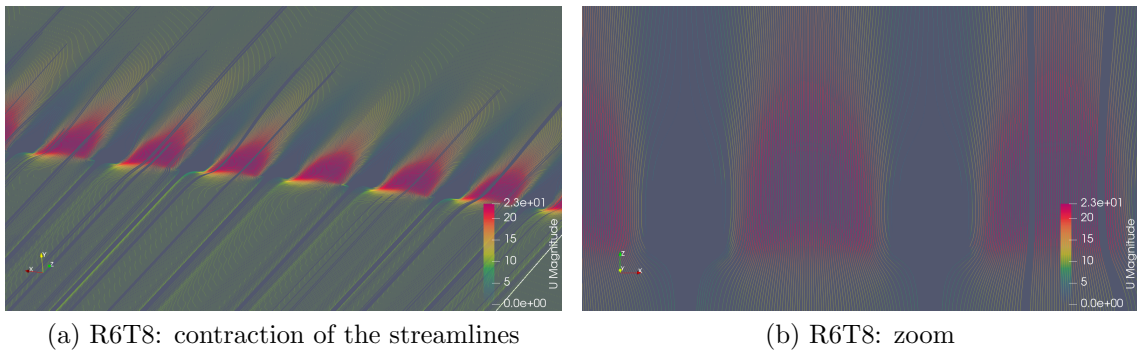
Figure 5: Velocity streamlines

More complex geometries can however create different effects from the ones seen here. The shape of the holes affects the flux orientation before and after

the screen. Some shapes can induce a re-circulation immediately after the screen and the *vena contracta* phenomenon is not seen anymore giving space to a more turbulent behaviour of the flow. The stretched screen for example has a geometry that obstacles in a major way the fluid flow and the remaining part that passes through it is deflected in massive way increasing its rotational speed and then going to re-circulate after the screen.

2.6 Relationship Between Forces and DarcyForchheimer Tensor

To better understand the formulas presented in this chapter it is necessary to depict briefly the steps that this work is going to do. A first run of numerical simulation will be performed by solving stationary Navier-Stokes equations for a channel flow with a porous screen placed inside it. As stated in the introduction the aim is to evaluate the forces acting on the slab and use their values as input for the definition of an analytic method. By starting from these values the analytical method is able to reproduce the effects of the porous screen giving as output the DarcyForchheimer inertial tensor. In this context it is interesting to see which relationship exists between this tensor and the forces acting on the slab. For a generic 2D problem the tensor is a square matrix of only four components. The expression of the forces remembers the form of the equation.11 for the pressure loss. From this expression the vector of forces is defined in a similar way:

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = V \frac{1}{2} \rho |\mathbf{U_i}| \begin{bmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{bmatrix} U_i \tag{12}$$

Where V is the volume of the porous media, ρ is the density and F_{ij} is the inertial tensor. The global reference system is always that referred to the porous body, that is orthogonal to the incoming flow and in this case is the portion of the mesh defined to replace the screen. Even if the inlet flow changes its direction the global reference system still remains the same. The dependence of the forces from the direction of the flow can be expressed by rewriting 12 and by making explicit velocity components:

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = V \frac{1}{2} \rho |\mathbf{U^2}| \begin{bmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{bmatrix} \begin{bmatrix} \frac{U_x}{|\mathbf{U}|} \\ \frac{U_y}{|\mathbf{U}|} \end{bmatrix} \tag{13}$$

where the last vector can be rewritten as $\begin{bmatrix} \frac{U_x}{|\mathbf{U}|} \\ \frac{U_y}{|\mathbf{U}|} \end{bmatrix} = \begin{bmatrix} cos\alpha \\ sin\alpha \end{bmatrix}$ in which α is the wind orientation with respect to z axis. Then the equation 12 can be rewritten as:

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = V \frac{1}{2} \rho |\mathbf{U^2}| \begin{bmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{bmatrix} \begin{bmatrix} cos\alpha \\ sin\alpha \end{bmatrix} \tag{14}$$

2.6.1   Tensor Computation: 2D case

The previous chapter was useful to understand how the forces that act to the screen are related with speed, wind orientation and inertial tensor elements. If these elements are the unique unknown of the problem it is possible to reverse the equation and solve the problem for these variables. Speed, density and wind orientation are all parameters chosen by the user so they are part of the initial data of the problem. The unknowns would be the vector of forces acting in the porous zone and the matrix F_{ij}, however the output of the explicit simulations performed on realistic porous screens seen in chapter 2 gives us exactly the vector of forces we are looking for. This would reduce the unknowns of the problem only to the elements of the inertial tensor.

If the vector of forces is inspected for different incidence angles α the equations presented in 14 become an over-determined system. To solve these kind of systems different mathematical methods could be used like the LSM (Least-Square Method). The system of equations is linear so it could be represented with the standard formulation:

$$\mathbf{Ax} = \mathbf{b} \tag{15}$$

that with the index formulation can be rewritten as:

$$A_{ij}x_i = b_j \tag{16}$$

Vector $\mathbf{b}$ is the vector containing the system of forces evaluated at different incidence angles:

$$\mathbf{b} = \begin{bmatrix} F_{x\alpha_1} \\ F_{z\alpha_1} \\ F_{x\alpha_2} \\ F_{z\alpha_2} \\ \vdots \\ F_{x\alpha_n} \\ F_{z\alpha_n} \end{bmatrix} \tag{17}$$

$F_{x\alpha_i}$ and $F_{z\alpha_i}$ are respectively the tangential and normal force acting on the screen evaluated for each angle of incidence α_i.

Vector $\mathbf{x}$ is instead the vector of the unknowns, namely the vector containing the

four elements of the inertial tensor:

$$\mathbf{x} = \begin{bmatrix} f_{xx} \\ f_{xz} \\ f_{zx} \\ f_{zz} \end{bmatrix} \tag{18}$$

As a consequence the matrix $\mathbf{A}$ is a rectangular matrix of four columns and of a number of rows equal to the number of elements of vector $\mathbf{b}$ that contains trigonometrical functions as illustrated below:

$$\mathbf{A} = V\frac{1}{2}\rho\mathbf{U^2} \begin{bmatrix} cos(\alpha_1) & sin(\alpha_1) & 0 & 0 \\ 0 & 0 & cos(\alpha_1) & sin(\alpha_1) \\ \vdots & \vdots & \vdots & \ddots \\ cos(\alpha_n) & sin(\alpha_n) & 0 & 0 \\ 0 & 0 & cos(\alpha_n) & sin(\alpha_n) \end{bmatrix} \tag{19}$$

A possible way to solve the problem is that of using the pseudo-inverse method (being $\mathbf{x}$ a rectangular matrix), rewriting the system as:

$$\mathbf{x} = \mathbf{A^+b} \tag{20}$$

where A^+ is the pseudo-inverse matrix:

$$\mathbf{A^+} = \mathbf{(A^T A)^{-1} A^T} \tag{21}$$

2.6.2   Tensor Computation: 3D case

Treating a three-dimensional case means to deal with a more complex problem with an increased number of unknowns. The parameters to be computed are always the elements of the inertial tensor F_{ij} that now are nine instead of four. In order to evaluate these coefficients it is necessary to know the vector of forces acting on the screen first, this time not only on xz plane but also on yz one, so the system of equations becomes:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = V\frac{1}{2}\rho|\mathbf{U^2}| \begin{bmatrix} f_{xx} & f_{xy} & f_{xz} \\ f_{yx} & f_{yy} & f_{yz} \\ f_{zx} & f_{zy} & f_{zz} \end{bmatrix} \begin{bmatrix} \frac{U_x}{|\mathbf{U}|} \\ \frac{U_y}{|\mathbf{U}|} \\ \frac{U_z}{|\mathbf{U}|} \end{bmatrix} \tag{22}$$

The procedure is then the same seen in the previous chapter with an increased number of variables and unknowns naturally. First of all the unknowns are col-

22

lected in a 9x1 vector:

$$\mathbf{x} = \begin{bmatrix} f_{xx} \\ f_{xy} \\ f_{xz} \\ f_{yx} \\ f_{yy} \\ f_{yz} \\ f_{zz} \end{bmatrix} \tag{23}$$

However by basing only to the flow behaviour on xy and xz plane it is not possible to evaluate all the nine values of the unknowns vector because there aren't the conditions to find the f_{zy} term. To solve this problem it is necessary to consider the forces acting on the plane that is the bisector of zy and zx plane. The bisector plane is the plane that divides in equal parts the angle between two reference planes that in this case are zy and zx planes. So the vector of the forces will be constructed by putting in sequence first of all the system of forces acting on xy plane for each incidence angle, then the forces acting on the xz plane and finally the forces acting on the bisector one.

$$\mathbf{b} = \begin{bmatrix} F_{x\alpha_1}^{xy} F_{y\alpha_1}^{xy} F_{z\alpha_1}^{xy} \cdots F_{x\alpha_n}^{xy} F_{y\alpha_n}^{xy} F_{z\alpha_n}^{xy} F_{x\alpha_1}^{xz} F_{y\alpha_1}^{xz} F_{z\alpha_1}^{xz} \cdots F_{x\alpha_n}^{xz} F_{y\alpha_n}^{xz} F_{z\alpha_n}^{xz} F_{x\alpha_1}^{bis} F_{y\alpha_1}^{bis} F_{z\alpha_1}^{bis} \cdots F_{x\alpha_n}^{bis} F_{y\alpha_n}^{bis} F_{z\alpha_n}^{bis} \end{bmatrix} \tag{24}$$

Now the matrix A has 9 columns, the same number of rows of b and is filled as follows:

$$\mathbf{A} = V\frac{1}{2}\rho\mathbf{U^2}
\begin{bmatrix}
\dfrac{U_x^{xy}}{|\mathbf{U}|} & \dfrac{U_y^{xy}}{|\mathbf{U}|} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\[2ex]
& \vdots & & & \vdots & & & \vdots & \\[1ex]
0 & 0 & 0 & \dfrac{U_x^{xy}}{|\mathbf{U}|} & \dfrac{U_y^{xy}}{|\mathbf{U}|} & 0 & 0 & 0 & 0 \\[2ex]
& \vdots & & & \vdots & & & \vdots & \\[1ex]
0 & 0 & 0 & 0 & 0 & 0 & \dfrac{U_x^{xy}}{|\mathbf{U}|} & \dfrac{U_y^{xy}}{|\mathbf{U}|} & 0 \\[2ex]
& \vdots & & & \vdots & & & \vdots & \\[1ex]
\dfrac{U_x^{xz}}{|\mathbf{U}|} & 0 & \dfrac{U_z^{xz}}{|\mathbf{U}|} & 0 & 0 & 0 & 0 & 0 & 0 \\[2ex]
& \vdots & & & \vdots & & & \vdots & \\[1ex]
0 & 0 & 0 & \dfrac{U_x^{xz}}{|\mathbf{U}|} & 0 & \dfrac{U_z^{xz}}{|\mathbf{U}|} & 0 & 0 & 0 \\[2ex]
& \vdots & & & \vdots & & & \vdots & \\[1ex]
0 & 0 & 0 & 0 & 0 & 0 & \dfrac{U_x^{xz}}{|\mathbf{U}|} & 0 & \dfrac{U_z^{xy}}{|\mathbf{U}|} \\[2ex]
& \vdots & & & \vdots & & & \vdots & \\[1ex]
0 & 0 & 0 & \dfrac{U_x^{bis}}{|\mathbf{U}|} & \dfrac{U_y^{bis}}{|\mathbf{U}|} & \dfrac{U_z^{bis}}{|\mathbf{U}|} & 0 & 0 & 0
\end{bmatrix}
\tag{25}$$

For what concerns the bisector plane, it is required only the F_y component of the forces to fully determine the matrix and then to solve the system. From now on the LSM is used to compute the unknowns of the problem. These unknowns then will be used as input parameters in the implicit simulations by filling the F matrix inside the porosityProperties dictionary of porousSimpleFoam solver.

3   Numerical Models

As stated in the introduction OpenFoam has different kind of solvers depending on the properties of the flow analyzed. The porous simulations will be divided into three parts:

- SimpleFoam simulations

- PorousSimpleFoam simulations

- Modified PorosuSimpleFoam simulations

SimpleFoam and PorousSimpleFoam are both steady-state solvers for incompressible and turbulent flow. The main difference is that the latter let the user define in explicit or implicit way a porosity region inside the numerical domain. To identify the properties of this region numerical simulations with SimpleFoam solver will be run for real screens inside the domain.

3.1   SimpleFoam

The first trance of simulations will be treated with SimpleFoam. SimpleFoam is a steady-state solver for incompressible, turbulent flows, that solves stationary Navier-Stokes equations discretized with Simple algorithm. The prefix "Simple" states for Semi-Implicit Method for Pressure Linked Equations. It performs a sequence of steps for each iteration in order to solve momentum and pressure equation as reported below:

- Advance to the next iteration $t = t^{n+1}$

- Initialise u^{n+1} and p^{n+1} using latest available values of u and p

- Construct the momentum equations

- Under-relax the momentum matrix

- Solve the momentum equations to obtain a prediction for u^{n+1}

- Construct the pressure equation

- Solve the pressure equation for p^{n+1}

- Correct the flux for ϕ^{n+1}

- Under-relax p^{n+1}

- Correct the velocity for u^{n+1}

- If not converged, go back to step 2

Momentum equation has been discretized with different numerical schemes depending on the term considered.

$$\frac{\partial[\rho u_i u_j]}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} \tag{26}$$

The advection term is the most critical one to deal with during the simulation. Discretize it with a wrong numerical scheme could compromise the simulations in terms of the quality of the solution and the convergence of the simulation itself. The best strategy to deal with is to start the simulation with a first order scheme, that grants a faster and easier convergence of the solution but loses at the same time all the effects of turbulence, obtaining a very smooth solution. Then restart the simulation from the solution found previously with a second order scheme, slower but more precise.
The advection term has been discretized with a second order upwind scheme, the diffusive term too, while the pressure one with a Gauss linear corrected scheme.
For what concerns the boundary conditions, cyclic boundary conditions have been applied in the two directions tangent to the grid. In this sense we assume that the flow that exits from the domain in a certain face re-enters in the domain in the opposite face. With this condition we can assume an homogeneous and isotropic flow in the lateral and vertical direction of the region of interest.
RANS simulations has been used to solve the fluid flow inside the channel, choosing the k-ϵ turbulence model as closure model. The k-ϵ model is one of the possible choices that OpenFoam guarantees in the turbulence dictionary. It defines the dissipation rate ϵ as proportional to the turbulent kinetic energy k:

$$\epsilon = \frac{C_\mu^{0.75} k^{1.5}}{L} \tag{27}$$

where C_μ is an empirical coefficient usually equal to 0.9 for these kind of simulations and effectively used with this value, L states as a characteristic length while the turbulent kinetic energy is defined as:

$$k = \frac{3}{2}(I|U_{ref}|)^2 \tag{28}$$

I states for the intensity of turbulence, it is usually of 2% while U_{ref} is the reference velocity, that is the undisturbed velocity of the flow.

3.2 Porous SimpleFoam

Porous SimpleFoam is a Steady-state solver for incompressible, turbulent flow with implicit or explicit porosity treatment and support for multiple reference frames (MRF). As SimpleFoam, it solves incompressible Navier-Stokes equations but it considers the addition of a sink term in the momentum balance equation that represents the presence of a porous media inside the problem. The equation in question so figures as follows:

$$\frac{\partial[\rho u_i u_j]}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + S_i \tag{29}$$

Different are the existing models to define the sink term. They differ from each other depending on the type of porosity the user wants to reproduce. The model chosen for this treatment is Darcy-Forchheimer one and it considers the sink term as composed by two terms, a diffusive one and an inertial one. Despite the range of velocity analyzed is not of great entity viscosity starts having remarkable effects at lower speeds and this makes the diffusive term forgettable with respect to the inertial one that will be the term of our interest. Sink term in his totality appears as follows:

$$S_i = -(\mu D_{ij} + \frac{1}{2}\rho[u]F_{ij})u_i \tag{30}$$

D_{ij} represents viscous loss term and is proportional to the velocity's flow, while F_{ij} is the inertial one and is proportional to the square of speed. D_{ij} and F_{ij} are both third order tensors and they are implemented as diagonal matrices. For non complex screens to be reproduced they guarantee a good representation of porous media, with only the diagonal terms non-null, and the maximum contribution that corresponds to the element referred to the normal direction to the grid.
Having non zero terms only on the diagonal elements of the matrix means that there are no interactions between the components of x, y and z axis essentially because the model chosen concerns only about the principal inertial axis.
Numerically porous SimpleFoam follows the same steps seen in SimpleFoam section, dealing with the addition of the sink term. The main limit of this model is however the impossibility to take into account the contribution of the non diagonal components when building the sink term. These elements are important when the geometry of the screen to be represented is complex, like a woven wire screen or a stretched one, or when variable orientations of wind speed are considered. In this scenario tangential forces starts to be non-negligible anymore and their contribution appears in the non-diagonal terms too. In order to consider this effect the first step of this work has been that of modifying the original code on which is based porous SimpleFoam, working on that files that creates the two tensors D_{ij}

and F_{ij} with the aim of constructing also the non-diagonal terms starting from the three components of forces that acts on the screen.

3.3   modified porous SimpleFoam

In order to modify porous SimpleFoam solver we need to recompile it after having modified some files inside the solver's directory. The area of interest of the modifications is that related to the header's and compiler's files of the solver, namely DarcyForchheimer.H and DarcyForchheimer.C.
In the header file there are all the structures necessary to the compiler file to compile the solver, in this sense it is sufficient to rename them for the 3D case. The big changes regards the compiler's file: the basic file considers the system of three forces and three moments acting on the reference system belonging to the porous region defined by the grid (body reference system) to construct the diagonal of the viscous loss tensor and inertial loss tensor.
The purpose is that of defining all the nine components of Darcy's and Forchheimer's matrices, then the new file will contain not only the elements corresponding to the three forces f_{xx}, f_{yy} and f_{zz} acting on their main axes but also their transversal components f_{xy}, f_{xz}, f_{yz}, f_{yx}, f_{zx} and f_{zy}. The discussion that relates the modifications made on the solver are directly presented in appendix.B

4 Porous Screens

Porous screens are employed for different scopes and depending on their functions they exist in a great variety of dimension, form, thickness, etc. The screens considered in this work show round or square openings equally spaced on their surface. For all of them simulations with the OpenFoam solver SimpleFoam have been launched after having meshed them within the channel flow in which numerical simulations have been done. To create these grids two different software have been used: Blender and SolidEdge.

In first instance Blender has been used. It is an open Source software mainly employed for design works like animated models and video makings. Nevertheless it was useful to create screen's geometries at the beginning of the thesis work. Once generated the CAD model, it was exported into an stl file to be used then in the meshing process once in OpenFoam. The problem related to the geometries created with this software was that the STL files had a bad discretization with triangles. The shape of them was different moving from one point to another. The quality of the STL file depended mainly on how good the CAD file was realized. Anyway during the meshing operations in OpenFoam, an unnecessary elevated number of cells was needed to reproduce the model in the computational domain to avoid the generation of a low quality mesh: one recurrent situation in the construction of the mesh is the presence of skew faces, mainly when the surface to be reproduced have sharp edges. This implies that the quality of the results is not the one expected and the simulations struggle in the convergence. For simple geometries like round hole grids no problems are registered in the generation of the mesh, for complex geometries it is necessary to create a more precise cad model to be exported then in an STL file. To do that it is better to use SolidEdge. Being a software purely born to be used for these kind of works it grants the generation of more appropriate files for which the surfaces reproduced have a fine segmentation with triangular or hexahedral elements.

In this sense while in the first instance all the geometries where generated with Blender software, after the problems cited before they have been replicated in SolidEdge.

4.1 types of geometries

Three types of screens have been studied in this work with their behaviour and properties analyzed in terms of a pressure loss coefficient whose definition will be treated in the following chapters. The geometries in question are:

- Stretched Screen

- Rounded hole Screen

- Woven wire Screen

The first one has been recovered from the work did by Andrea Buscemi. It is a geometry with diamond shaped holes deeply three-dimensional. Its three-dimensionality is given by the alternated stretching of the metal sheet in the plane normal to the grid. This shape makes the grid (figure 6) generate forces not only in the normal direction but also in the tangential ones (referring to the body axes).
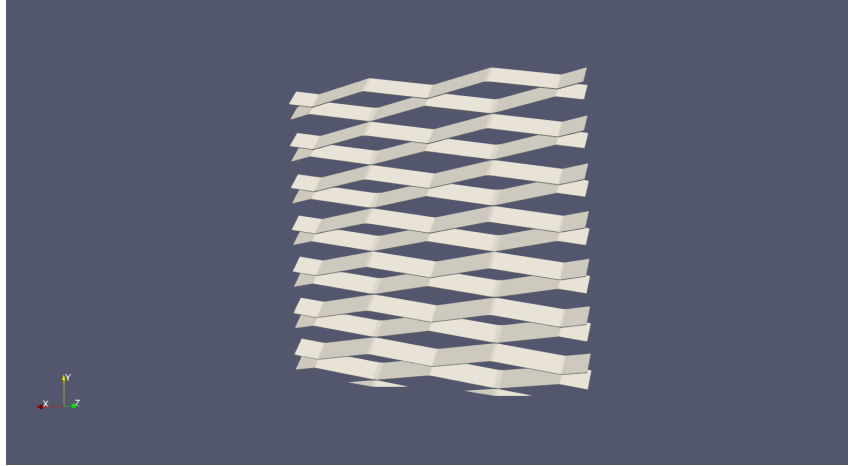

Figure 6: Stretched Grid

The second one is a flat plate with equidistant rounded holes. Different diameters and thicknesses have been studied with a consequent change of porosity index from one model to another. Two diameters have been tried, respectively 6 mm and 6.35 mm with a variation in porosity from 50% to 55%.
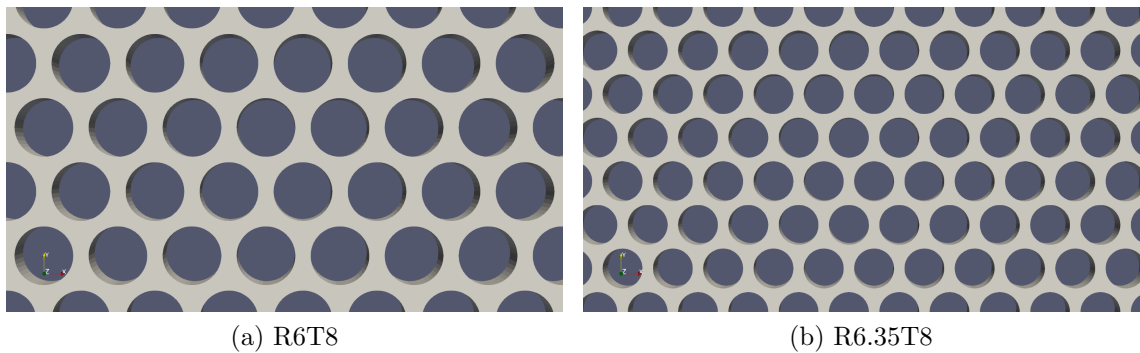

(a) R6T8   (b) R6.35T8

Figure 7: Rounded Hole Grids

The last screen considered is made with woven wires. It consists of round wires

intertwined each other creating squared holes that remembers the geometry of a gauze as can be seen in figure**??**:
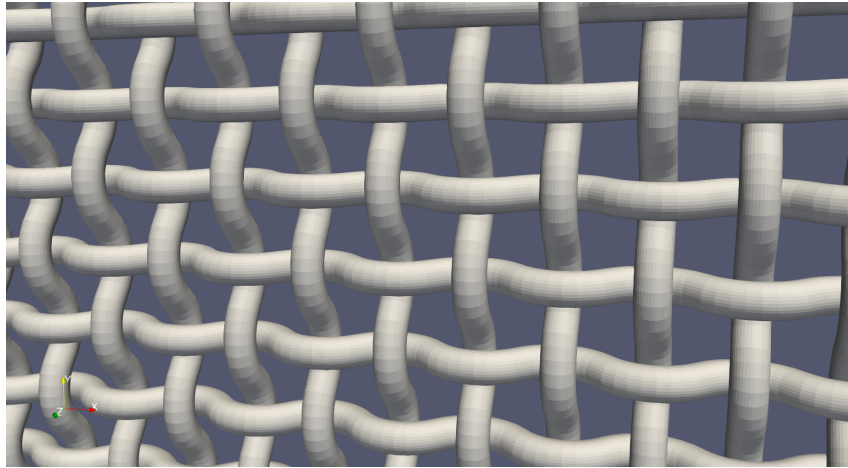

Figure 8: Woven Wire Grid

About the creation of the geometry, it was made by starting from a sketch with two tangent arcs that assume the shape of a sine wave. With a sketch pattern that follows the curve it has been possible to create the piece of wire needed for the construction of all the screen. With a rotational pattern it has been created an alias rotated of 90° and then with a linear pattern each wire has been duplicated in the two directions of the plane of work. This dependency on the sine at the base of the creation of the grid could be an additional reasons that affects the trend of the forces with the variation of the incident speed. It will seen in fact that the the axial force shows a variation with the incoming speed proportional to a trigonometrical function, mainly because to an in increase of the inclination of the wind direction corresponds a decrease to the effective screen area proportional to the cosine.

Also in this case different dimensions of the grid have been studied by varying the wire diameter and changing consequently the free area of the holes. These changes influences the quantity of fluid that passes through the screen, expressed in non dimensional terms with a parameter that is the porosity of the screen itself. Anyway the main case studied for this geometry is that characterized by a wire with a diameter of 4 mm, so with a thickness of the screen of 8mm and squared holes of 6 mm per side.

5 Mesh creation

Once chosen the porous screen to use and created its model on Solidedge the following step is to decide the numerical domain in which simulations have to be performed, its dimensions, how to discretize it, the position of the screen inside it and its boundary conditions. The chosen geometry is that of a channel flow, with the screen positioned far enough from the inlet surface. The principal aim is that of refining the mesh especially in the region in which the screen is located, because there is where the flow is more complex and where we want to catch all its evolution. However refine in same way the channel flow for all its length it is not the best choice. In fact this would mean to refine in an unnecessary way some regions of the domain (like that in proximity of inlet and outlet surfaces) where the flow is ordered and so doesn't need deep investigation.

So the approach followed for the creation of the mesh was that of subdividing the channel flow in different blocks, not necessary equal, each one more refined than the previous one until the screen is encountered. There will be the maximum number of cells with respect to the extension of axial component of the block. Once overtaken the block containing the porous media, the refinement starts decreasing until the last one. In this way the mesh will not have an excessive number of cells, so avoiding large time of calculations and problems of convergence, and at the same time it will be enough precise to catch the behaviour of the flow in proximity of the screen.

5.1 Mesh Discretization

Being x,y,z an orthogonal reference system referred to the body, z axis corresponds to the direction normal to the screen. The refinement of the mesh has been performed in all the three directions following the criterion of doubling up the number of cells in x and y direction until reaching the screen location. The quality of the mesh depends on how well these blocks are connected each other. If the number of the boundary cells at the interface of one block is not equal or a multiple of the cells number that belong to the adjacent block the quality of the simulations can be affected. This because the shifting from one block to another is not regular. There is a patch in the blockMesh dictionary, called *mergePatchPairs*, that allows to pass with more gradual shifting from one block to another, as shown in fig.9. In this case each cell of the low refined block has the half of the cells of the high refined block. This helps the transition from one block to a more refined one.

After the number of cells has been halved until a minimum number to grant enough precision also far away from the porous screen. About the z axis the number of cells in that direction has been chosen in a way in which there was enough amount
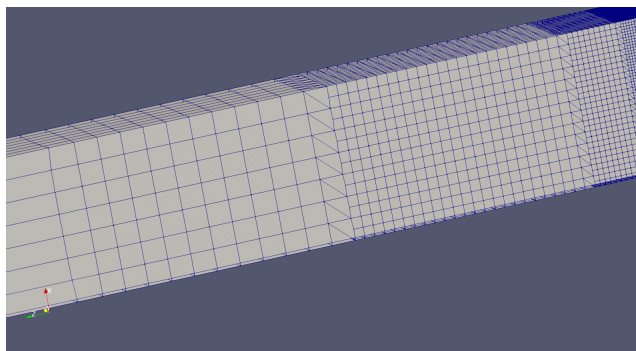
Figure 9: Block Subdivision of the Mesh

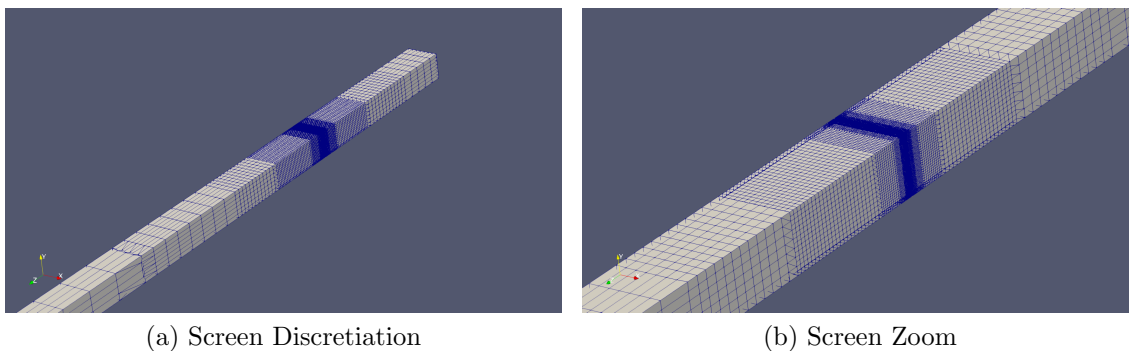of them to deal with the *snappyHexMesh* process to mesh the porous screen inside the channel.

(a) Screen Discretiation

(b) Screen Zoom

Figure 10: Mesh Discretization

5.2 SnappyHexMesh process

The introduction of the porous screen inside the channel is done with the snappyHexMesh process. By modifying the corresponding snappyHexMeshDict directory in the simulation folder it is possible to choose how many levels of refinement will be performed to refine the grid, its position in the channel etc. The snappyHexMesh utility generates 3-dimensional meshes containing hexahedra (hex) and split-hexahedra (split-hex) automatically from triangulated surface geometries, or tri-surfaces, in Stereolithography (STL) or Wavefront Object (OBJ) format.
The mesh approximately conforms to the surface by iteratively refining a starting mesh and morphing the resulting split-hex mesh to the surface.
Cell splitting is performed according to the specification supplied by the user in the

33

castellatedMeshControls sub-dictionary in the snappyHexMeshDict. The entries for castellatedMeshControls are presented below:

- locationInMesh: location vector inside the region to be meshed; vector must not coincide with a cell face either before or during refinement.

- maxLocalCells: max number of cells per processor during refinement.

- maxGlobalCells: overall cell limit during refinement (i.e. before removal).

- minRefinementCells: if minRefinementCells $\geq$ number of cells to be refined, surface refinement stops.

- nCellsBetweenLevels: number of buffer layers of cells between successive levels of refinement (typically set to 3).

- resolveFeatureAngle: applies maximum level of refinement to cells that can see intersections whose angle exceeds resolveFeatureAngle (typically set to 30).

- features: list of features for refinement.

- refinementSurfaces: dictionary of surfaces for refinement.

- refinementRegions: dictionary of regions for refinement.
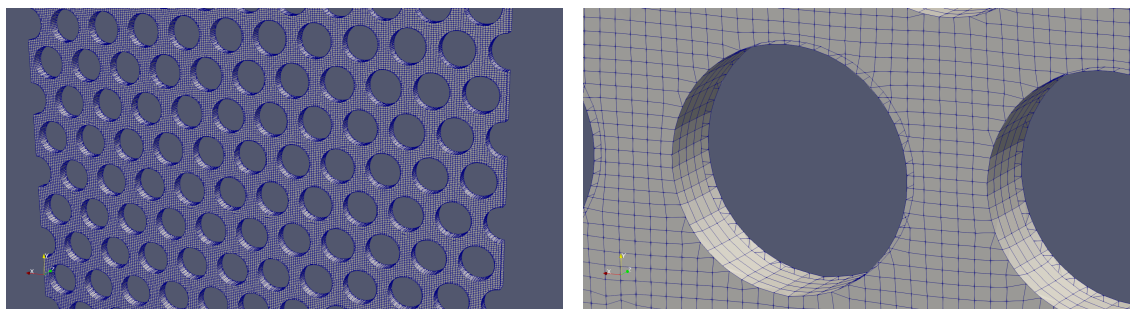
(a) R6T8: meshed surface   (b) R6T8: meshed surface zoom

Figure 11: R6T8: SnappyHex process on the screen

Once terminated the snappyHex process the last step to be performed is the definition of the patches of the numerical domain, in order to make OpenFoam able to understand the properties of each surface. This step is performed by using createPatch directory, in which the user simply defines the inlet and outlet surfaces,

walls and empty regions and so on.

In this section are also defined boundary conditions. To let the numerical results independent from the boundaries of the domain cyclic boundary conditions have been imposed. In this way has been imposed to the domain that the flow that goes of from a certain surface re-enters in the domain from the opposite one.

5.3   Reference Systems

As stated before the reference system is attached to the screen, so we refer to a reference body system because we need the forces developed on its surfaces.
The main quantity of numerical simulations regards a wind with variable orientation. In this sense two approaches are possible: keep the screen fixed and vary the incidence of the velocity vector, or keep the velocity vector fixed and vary the inclination of the screen inside the channel flow. The latter approach is typical of the experimental tests in wind tunnels for which there is the possibility to rotate the model inside the test chamber. The first approach is instead the one used for the numerical simulations in this work. This means to keep fixed the body reference system and move the wind reference one.

Figure 12: Reference System

5.4   Boundary Conditions

Cyclic boundary conditions are applied for all the sides of the channel flow, namely top and bottom surfaces, front and back surfaces, for all the extension of it. These b.c. are applied respectively for the couples top/bottom and front/back so that the fluid flow that goes outside the domain through a certain surface, re-enters inside the channel flow from the opposite surface. In these way the fluid

flow is not affected by the presence of the boundaries feeling free to evolve inside the channel flow and resulting homogeneous.

Initial conditions are chosen in input by the user who specifies the velocity vector, the pressure and the density of the fluid flow as well as the turbulence model.

6 Results

This is the section dedicated to the results of the numerical simulations performed by using the solvers depicted in section.3. The schedule followed in this section starts with a first number of simulations reserved for the convergence analysis both for the rounded hole screen and the woven wire one. Once found the minimum number of cells for which there is an independence of the results from the mesh discretization it is possible to proceed with that mesh to perform numerical simulations in the channel flow.
Two different sub-problems has been considered from now on:

- variable velocity intensity/constant velocity orientation

- variable velocity orientation/constant velocity intensity

The first set of simulations pertains to the variation of the velocity intensity inside the channel flow by considering the range of speed between $5\frac{m}{s}$ and $20\frac{m}{s}$ that is the wind speed range of our interest. Results will be compared with the available experimental data in terms of the pressure loss coefficient. The second set of simulations are characterized by a variable wind orientation. Results will be first considered in terms of the pressure loss coefficient as the previous case. After that the analysis will focus directly on the forces generated on the screens.
Explicit simulations performed with simpleFoam will be compared with the results obtained with the analytic simulations by using the Least Squares Method.
Follows the final comparison between the previous results and those obtained with the implicit simulations through the modified porousSimpleFoam solver.

6.1 Convergence Analysis

After the creation of the mesh and the definition of the boundary conditions, convergence analysis have been performed. The aim is that of refining the quality of the mesh by augmenting the number of cells of the numerical domain to get more precision in the results of the simulation. This procedure is going to be repeated until the results of one simulation from the previous one are the same or their difference falls inside a certain tolerance. This is an usual operation when numerical simulations have to be performed in order to reach the minimum number of cells to have results that are independent from the discretization of the mesh. When this happens, will be chosen the mesh with the minimum number of cells for which the results fall in that tolerance.
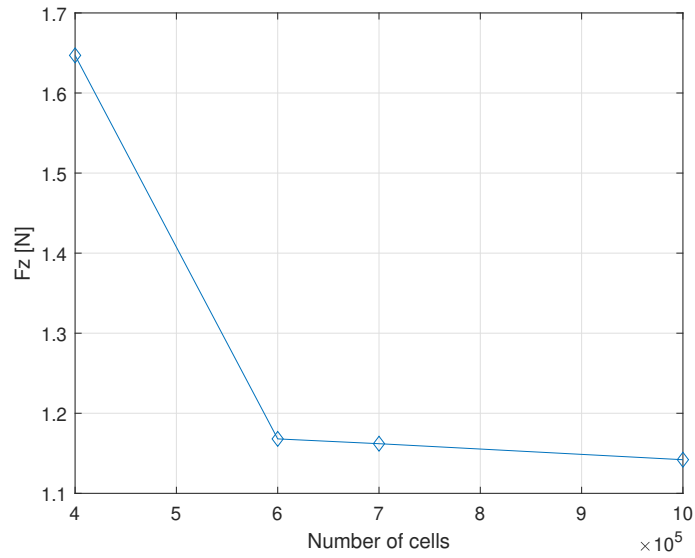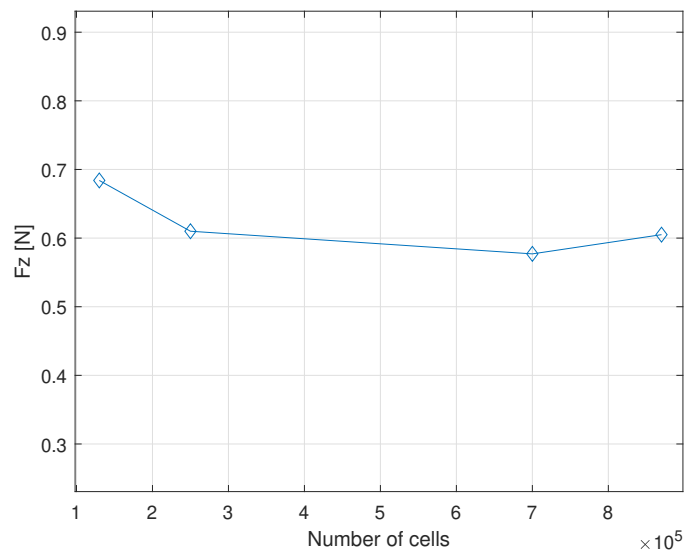
Figure 13: R6T8: Convergence Analysis

Figure 14: WIRE: Convergence Analysis

6.2 Wind Tunnel Experiments

As mentioned before the aim of this work is to find an alternative approach to value the impact of porous screens in civil and environmental engineering. This approach is a compromise between the expensive tests in wind tunnel and the huge

numerical simulations.

At the beginning of the simulations, experimental data in wind tunnel tests have been available. The simulations have been performed on two types of perforated sheets with round holes whose dimensions were 6 mm of diameter and 8 millimeter of spacing between two holes for the first one, and 6,35 mm of diameter and 8 mm of spacing for the second one. Both panels were 2 mm thick and were installed in a wind tunnel with circular section as in the scheme reported below:
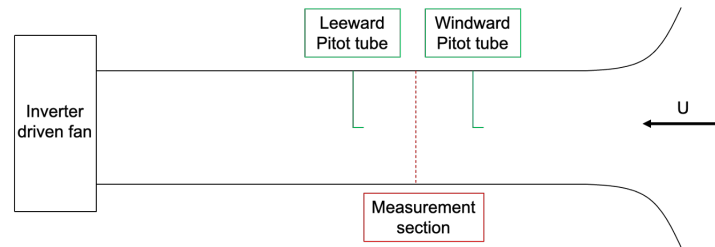

Figure 15: experimental setup

Before and after the porous screen two static Pitot tubes were installed to detect the dynamic pressure loss through the sheet. The dimensions of the sensors were negligible with respect to the characteristic length scale of the panel, then the perturbations of the flow produced by their presence can be assumed negligible too.

The wind speed at the inlet section is orthogonal to the sheet and for each test it was kept constant until the force signal was clear. A wind interval from 5 m/s to 30 m/s was investigated for both the screens and then the pressure loss coefficient trend was plot with the variation of the speed.
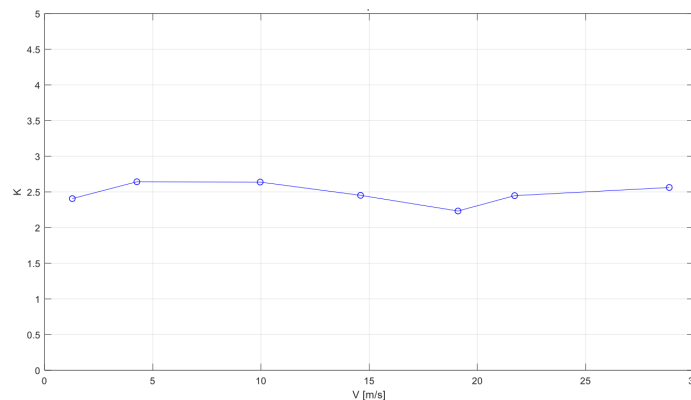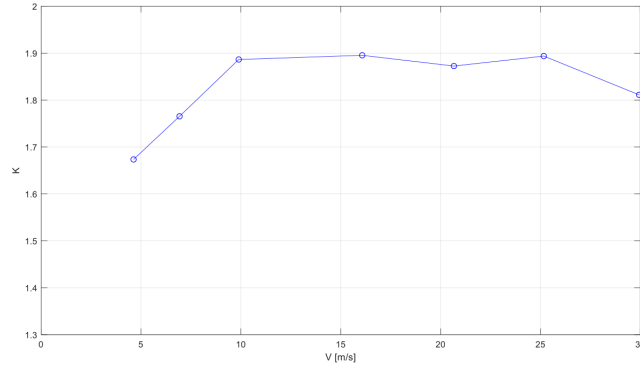

Figure 16: R6T8 wind tunnel

Figure 17: R6.35T8 wind tunnel

The porosity of the first screen, called R6T8, is around 50% while that of the second is around 55%, then we expect a greater value of the pressure loss coefficient for the first one due to the fact that the fluid flow volume rate through the screen is lower. In fact we can see that for the R6T8 the loss coefficient stabilizes around 2.5/2.6 values, while for the R6.35T8 it is around 1.9/2.0 units.

then it is clear that after a certain speed ($\approx 10\frac{m}{s}$) the pressure loss coefficient starts to be independent from the wind speed orthogonal to the sheet. In the next chapters it will be possible to see that this behaviour is analogous also for different incident angles, in the sense that, by keeping fixed the angle at a certain orientation and changing the speed magnitude, the pressure loss coefficient normalized with respect the upstream velocity shows the same constant trend.

Now the aim of this chapter is to verify if the numerical simulations can detect the same trend seen for the "experimental" pressure loss coefficient and possibly results closer to the experimental ones.

6.3 Constant Speed Orientation Simulations

A first number of numerical simulations was dedicated to convergence analysis. By thickening the mesh, it was found that the one with 600.000 cells was the first to be independent from the further discretization of the mesh, so it was used to perform the numerical simulations.

A sequence of numerical simulations was performed changing every time the inlet speed of the fluid flow. The chosen turbulence model is k-epsilon. For sake of completeness also some simulations were performed by using k-omega model in order to verify if there was a difference in the results but no great changes have been seen.

For what concerns the results, the simulations revealed the same trend of the loss

coefficient seen in wind tunnel for R6T8 and R6.35T8 grids. For both the configurations the results seems overestimated, with values that exceeds of 0.5/0.6 the experimental ones. Several can be the reasons behind these differences. Anyway it is possible to confirm that after $10\frac{m}{s}$ the pressure loss coefficient flattens on a constant value for both cases as already seen in [6]. This means it is independent from the speed variation and then from the Reynolds number keeping fixed the geometry of the screen. A sharp decaying can be seen instead in the interval of speeds before $10\frac{m}{s}$. At this range of speeds we work with lower Reynolds number that are indicative of the fact we deal with a flow field in which viscosity effects are still not negligible. So the diffusion term in Navier-Stokes equation plays a determinant role in the definition of the forces that act on the screen. The more the speed is increased the bigger will be the Reynolds number and the inertial effects that will affect the problem and start dominating the physics of it.
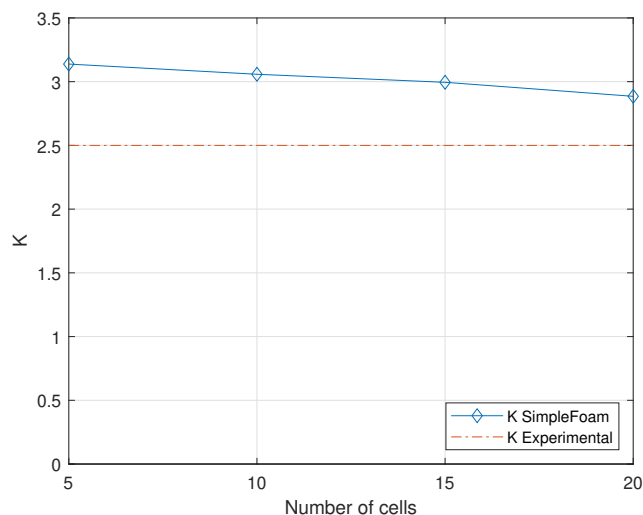

Figure 18: R6T8: Pressure Loss Coefficient/Speed Variation

6.3.1 Woven Wire Screen Results

The same approach has been followed for the woven wire grid screen. In this case no experimental data were available from wind tunnel testings so the information about this kind of screen has been searched in the literature about it.
The choice of analyzing the woven wire screen was made in order to work with a geometry more three-dimensional than the R6T8 and R6.35T8 that were totally planar geometries. The woven wire screen shows a porosity of 65%, but in this

case the evaluation of porosity could be ambiguous. While for the flat porous screens seen before the calculation follows that of (7) dealing with areas because the screen is planar, here the geometry changes also in the orthogonal direction and working only with areas could let unconsidered some open spaces created by the sine wave trend of the wire. Then it would be better to work with volumes instead of areas. Also in this case the variation of the pressure loss coefficient with the speed magnitude stabilizes on a constant value around 2.5 as could be seen in figure below.

It is possible to conclude that the behaviour of porous screens has one common denominator: when the inertial effects start prevailing on the viscous ones (around $10\frac{m}{s}$) the pressure loss coefficient is independent from the speed. This means that for this kind of problem each grid of that treated until now can be characterized by a constant parameter that defines the grid's capability of blocking the fluid flow. So given the density of the flow, the upstream speed and the geometry of the screen it is possible to know with good approximation the pressure jump through the screen.
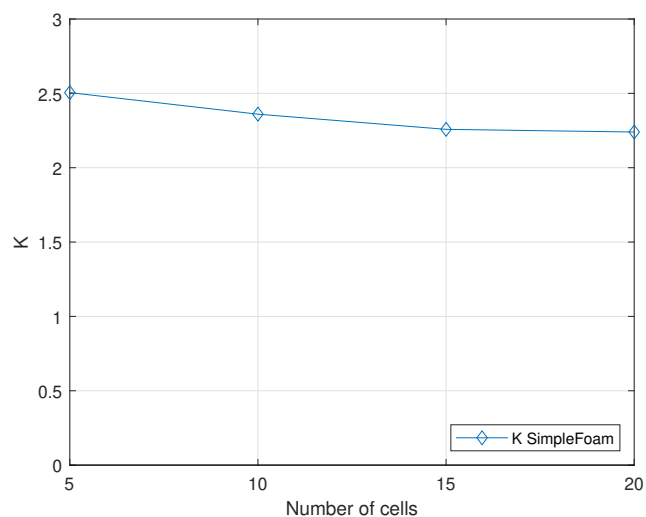

Figure 19: Woven Wire:Pressure Loss Coefficient/Speed Variation

6.4 Constant speed magnitude-varying speed orientation

This paragraph is dedicated to the second trance of explicit simulations with a velocity that now is inclined with a certain angle of incidence with the respect the orthogonal axis to the screen. Wind speed orientation (and intensity) is in fact extremely variable in the reality being the problem extremely unstationary. As

said in the first chapters, for this work stationary problems have been considered to deal with a simpler model, but the main aspects of wind behaviour, like the change in speed and orientation, have been considered. It is not of main interest the story evolution of the pressure loss coefficient, but much more its trend with the variation of the angle of incidence. In this section we don't expect to find a constant value of K as seen before, because even if the intensity of the velocity is kept fixed its orientation will be varied, and then the normal force acting on the screen will be reduced step by step, by observing at the same time the appearance of the shear forces. What is expected will be the reduction of the pressure loss coefficient by increasing the wind inclination with respect the normal axis, and as said the contemporary birth of the shear components.

6.4.1   R6T8 Results

To perform this trance of simulations the possible approaches were two: keep the screen fixed and change the wind orientation or keep the wind orientation fixed and vary the screen inclination. The second approach is used in general for wind tunnel experiments, in which the geometry of the channel is naturally fixed and the inclination of the model inside the test chamber can be varied. Of the two solutions the first one has been chosen in order to work with the same geometry of the problem for all the angles.
Velocity has been changed in xz plane and the vector of the forces has been investigated for the following angles: 0°, 10°, 20°, 30°, 45°, 60°, 70°, 80°, 90°.
By changing the speed only on the xz plane the vertical force f_y on the screen is null while the horizontal force f_x starts increasing.
The results of the simulations shows that the pressure loss coefficient decays progressively when the fluid flow is always more parallel to the screen. In this situation the flow that passes through the screen is less than the orthogonal case. Most of it slides on the screen surface and increases the shear force on it.
As said in the previous chapters, the pressure loss coefficient is the adimensional expression of the drag force acting on the porous screen. In this sense the drag force is mainly due to the blockage of the flow generated by the sheet. But when dealing with a variable wind orientation it has to be considered one side effect that gives a contribution placed inside the drag force. In fact when the flow is oriented with a certain angle of incidence with respect to the porous screen it tries to correct its deflection in order to pass through the open areas. By focusing on the velocity vector field, from the visualization of the streamlines it is possible to see that the flow with a certain angle of incidence starts to curve just before the screen. This phenomena ends with a different angle of orientation of the flow after the porous screen and quantitatively it is possible to be seen as a contribution in the drag coefficient. The effect of flow deflection is remarkable when the angle of

incidence is not so large with respect to the normal direction to the screen. It becomes negligible when the angle is so big that the flow is not able to deflect anymore and then its momentum is discharged on the screen not only with a net orthogonal force but also with a non negligible shear component. (flow deflection in porous screens/Aerodynamic characteristics of damping screens)

About the results, the pressure loss coefficient follows a decaying trend by increasing the angle of incidence. As can be seen in the figure below initially the decrease is slow, with an almost constant K, and then after the 30° it starts decreasing in a linear way. The idea is that of finding a theoretical curve that can describe with a good approximation the behaviour of the pressure loss coefficient. This curve has the shape of cosine wave. By starting form the value of the pressure loss coefficient detected at 0° to the screen, the following values are on the curve generated by the multiplication of the initial K and the cosine wave at the corresponding incidence angle.

$$K = K_0 \cdot cos(\theta) \tag{31}$$

In the existing literature there are proofs of this behaviour for the pressure loss coefficient. As cited in the introduction already Richards[13] found a relationship for the decaying trend of K expressed as a trigonometrical function .

One possible interpretation of this trend is that the porous sheet can be seen as a flat plate put at incidence with respect to an incoming wind. For the problem treated there is no flow that turns around the screen and then there are not the effects of separation on the back of it so the only acting term that concurs to the intensity of the drag and lift force is the fluid flow momentum. In a reference system coherent to the body, the drag force we are analyzing can be visualized as the lift force acting on the flat plate. The difference between the flat plate and the porous sheet is that in the second case the lift force has lower intensity due to the holes placed to the surfaces that let the flow recover its pressure behind the screen.

The generic polar curve of a flat plate goes up linearly from 0° incidence angle and then starts decaying when separation occurs. For the porous screen it is possible to see a similarity with this trend starting from the condition in which the flow is parallel to the screen. To see this similarity it is necessary to use the same reference system used for the flat plate, so we need to invert the wind orientation axis for the graphic of the pressure loss coefficient being the 0° angle the corresponding one of 90° angle for the flat plate and the other way round. Naturally the similarity is restricted only to the part in which there is a linear trend, after that there are phenomena for the flat plate like separation that are not related to the physics of our problem, so that cannot be taken under consideration.
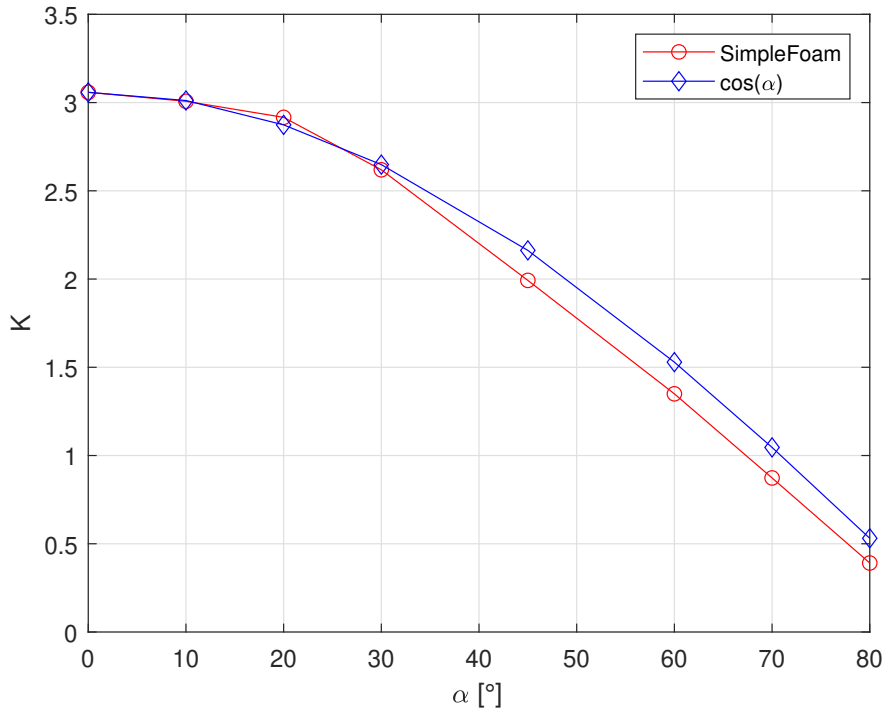
Figure 20: R6T8: SimpleFoam and theoretical trend comparison

6.4.2 R6.35T8 Results

For the R6.35T8 it is possible to assert the same conclusions. Being the diameter of holes bigger than the those in R6T8 the pressure loss coefficient is shifted down and its variation with the wind orientation follows as before the path drawn by a cosine curve.

Basing on the consideration made until now, only by the knowledge of the air density, the upcoming speed magnitude and the geometry of the screen it is possible to evaluate the pressure jump across it, and from this value, even the pressure losses for different incidence angle are accessible.
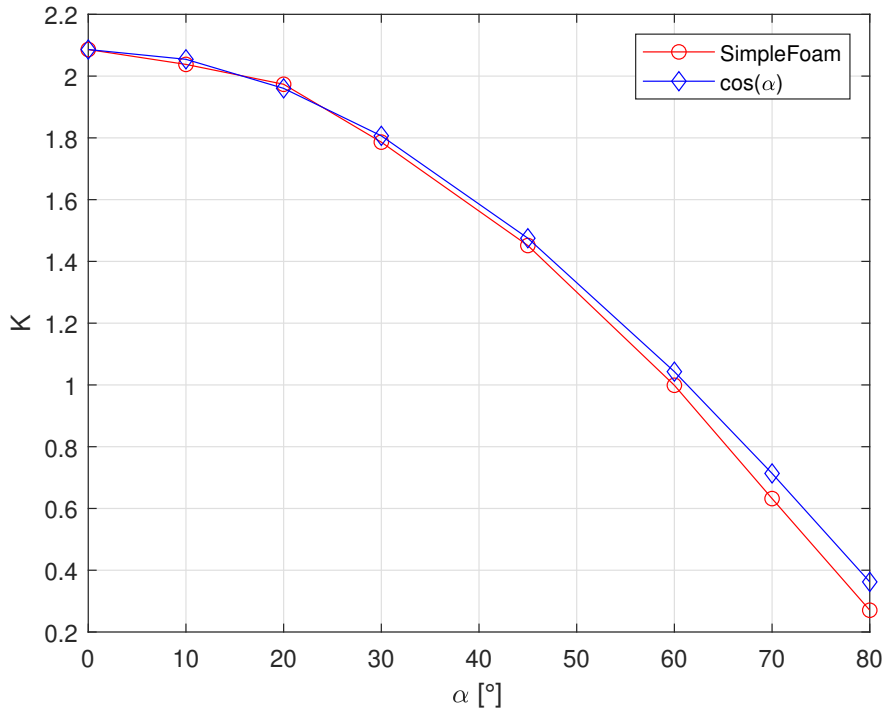
Figure 21: R6.35T8: SimpleFoam and theoretical trend comparison

6.4.3 Woven Wire Screen Results

Simulations for the woven wire screen follow the same procedure seen in the chapters before. However the trend seen in this case is different from the previous ones: the decaying of the pressure loss coefficient is sharper than the rounded holes cases and it is easy to find that its variation with the speed orientation follows a squared cosine wave. This trend is accordance with the results presented in [13], [14] and [10].

$$K = K_0 \cdot cos^2(\theta) \tag{32}$$

Where K_0 is the pressure loss coefficient normal to the screen and θ is the angle of incidence of the incoming wind.
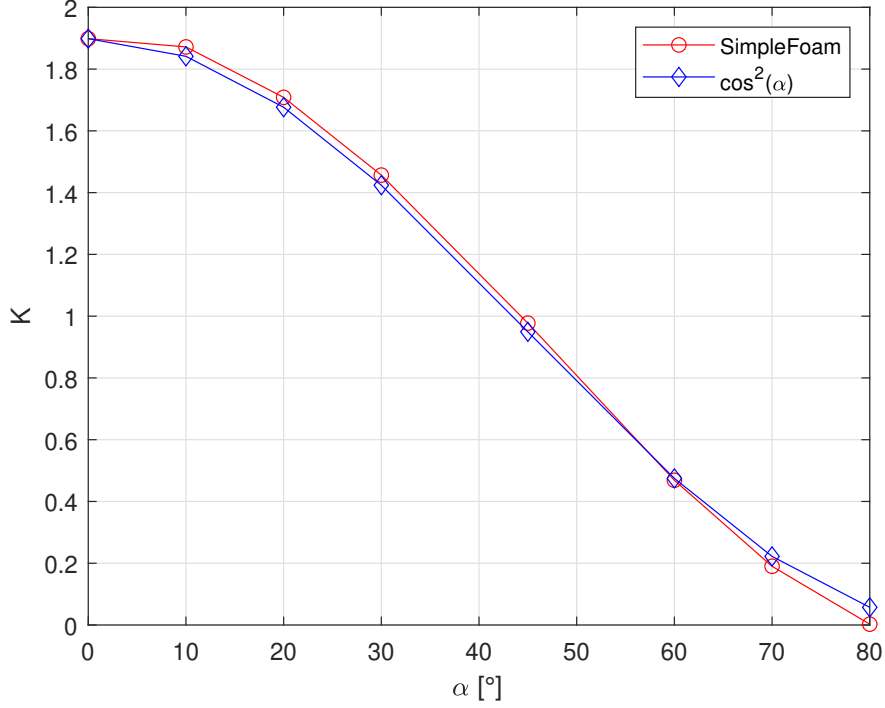
Figure 22: Woven Wire: SimpleFoam and theoretical trend comparison

6.5   Least Squares Method

Numerical simulations have been performed for all the grids seen in chapter 3. This required the use of LSM for all these test cases in order to define the components of the inertial tensor.

Except for the stretched screen, a linear 2D solver has been used for all the other grids. Most of them in fact are planar geometries, analyzed varying the angle orientation of the speed only in the xz plane. It is possible to observe then that only the normal component F_z and the horizontal component F_x are non null. For this reason the unique non zero components of the inertial tensor are that corresponding to the main diagonal and their coupling terms outside the main diagonal, namely f_{xx}, f_{zz}, f_{xz} and f_{zx}.

In the following subsections will be presented the values of the components of the inertial tensor for each screen. These values are based on the system of forces evaluated in the first trance of simulation made with simpleFoam at different incidence angles.

The algorithm constructs a vector of forces and the corresponding vector of incidence angles, and with them it implements the LSM. For each screen will be

presented the Inertial tensor and the predicted trend of forces using the LSM with respect to that of explicit simulations.

```matlab
%% Calcolo dei coefficienti del tensore
% caso bidimensionale - lamiera forata

%%
close all
clear all
clc

rho = 1.225; %kg/m3
Vel_ref = 10; %m/s
q = 0.5*rho*Vel_ref^2;
dx = 0.07;
dy = 0.07;
dz = 0.008;
V_ref = dx*dy*dz;   %volume di riferimento (setto poroso)

% Non-linear LSM
if 1 % weighted LSM
    xdata = T.angle;
    ydata = [T.fz'; T.fy']';

    x0 = [1,-1;-1,1];

    x = lsqnonlin(@mycurve,x0,[],[],[],xdata,ydata)
    F = (q*V_ref).*x*[cosd(T.angle)';sind(T.angle)'];

else    % tutti pesati allo stesso modo
    b=[];

    for ii = 1:length(T.angle)
        b(end+1,:) = T.fz(ii);
        b(end+1,:) = T.fy(ii);
    end

    alpha = T.angle;

    C = lin_solv2D(alpha, b, V_ref, q);
    F = (q*V_ref).*C*[cosd(T.angle)';sind(T.angle)'];
end
```

Listing 1: Application of Non-Linear and Linear LSM algorithm

Before using the algorithm it is important to know also the limits of this method. As a linear model, the linear Least Squares Method shows some limitations in the shapes that the model can assume over long ranges. This method has in fact huge

sensitivity to anomalies in the sample of data, in the sense that rapid changes in slope are difficult to be followed and reproduced. A complex distribution of data like a change in the slope can sometimes seriously skew the results of a least squares analysis.

6.5.1 R6T8 Inertial Tensor

For the screen with holes of 6mm, spacing of 8mm and 2 mm of thickness, namely the R6T8, the inertial tensor takes the following form:

$$F = \begin{bmatrix} 1493.600 & 0.000 \\ 0.000 & 262.020 \end{bmatrix} \tag{33}$$

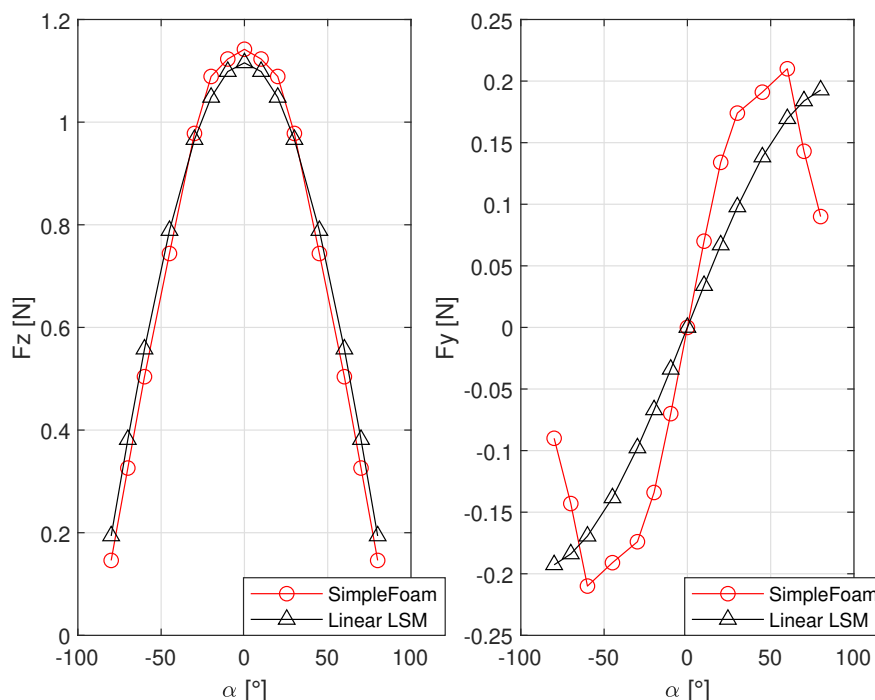For a reference volume $V_{ref}=1.219 \cdot 10^{-5}$. The main aspects to point out is that

Figure 23: R6T8: SimpleFoam and LSM comparison

LSM is pretty good in the estimation of the axial force F_z between -45° and 45°. After this interval, both for negative and positive angles the LSM underestimates the values around the 5%.
Different situation for the horizontal component F_y: the LSM prediction starts moving away from the explicit trend since we move from 0° incidence. F_y values are underestimated until we reach the tail of the curve in which the results and the trends are completely different.

6.5.2 R6.35T8 Inertial Tensor

For the screen with holes of 6.35mm, spacing of 8mm and 2 mm of thickness, namely the R6T8, the inertial tensor takes the following form:

$$F = \begin{bmatrix} 1030.000 & 0.000 \\ 0.000 & 284.500 \end{bmatrix} \tag{34}$$

For a reference volume $V_{ref}=1.219 \cdot 10^{-5}$. For both these screens the geometry doesn't change in the normal direction, being the screen essentially a flat plate oriented to the wind. For this reason we don't see the presence of non diagonal term. R6.35T8 is the screen for which numerical results and LSM prediction are
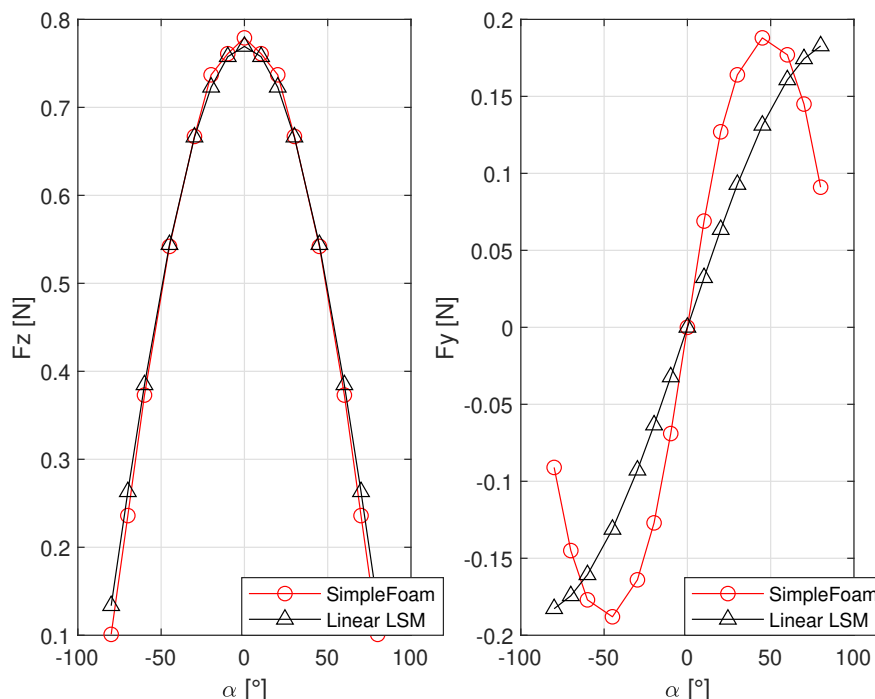

Figure 24: R6.35T8: SimpleFoam and LSM comparison

more in accordance. For what concerns the axial force the two graphs coincide until 60° and after this interval they differ with and error that is under the 5%. For the horizontal component F_y we can observe the same trend seen before.

6.5.3 Woven Wire Inertial Tensor

For the screen with holes of 6.35mm, spacing of 8mm and 2 mm of thickness, namely the R6T8, the inertial tensor takes the following form:

$$F = \begin{bmatrix} 258.598 & 0.000 \\ -0.000 & 54.380 \end{bmatrix} \tag{35}$$

For a reference volume $V_{ref}=1.219 \cdot 10^{-5}$. This geometry has changes in shape not only in the xy plane but also in xz and yz. By construction the wires evolve with a sine wave trend on xz and yz plane being intertwined. So in this case it is possible to see also the presence of non-diagonal terms. Respect to the others, the
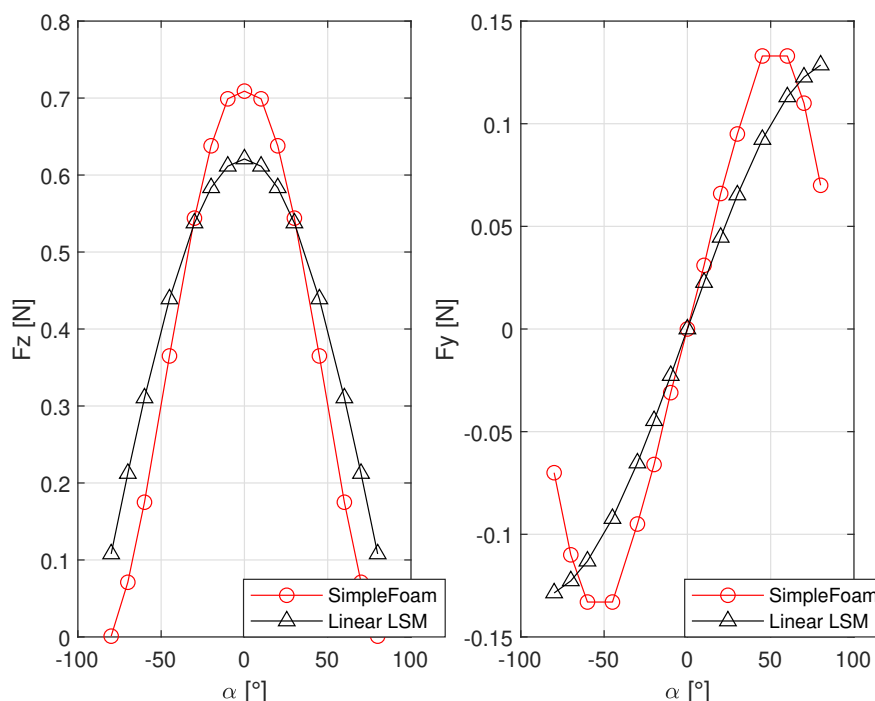

Figure 25: Woven Wire: SimpleFoam and LSM comparison

LSM prediction of forces for the woven wire screen is the one that more distances itself from the explicit calculations. The reason behind this discrepancy could be linked to the three-dimensionality of the screen that is not planar like R6T8 and R6.35T8 geometries. The prediction can be improved by giving some weight for low angles also for the axial force.

One common factor for all the three geometries analyzed is that the horizontal

force F_y predicted by the LSM doesn't follow the trend outlined by results of the numerical simulations, while the normal force F_z remains coherent until 45° except for the woven wire screen. A possible solution to correct this diversification between results and expectations is that of using a weighted Least Squares Method, with the aim of giving more weight to some angles with respect to others.

6.6   Weighted Least Squares Method

The problem of the previous results is the evident discrepancy between the explicit simulations and the curve based on the prediction made by the Least Squares Method. For what concerns the axial force the trend registered is the same with a non completely correct coincidence after 60° that is however under the 5% of error. The prediction of the tangential force instead is not the expected one and then it needs a correction.

To do so a possibility could be that of giving more weight to the angles in which a better prediction is requested. As anticipated before the aim of this section is to improve the quality of Least Squares Method for the prediction of the system of forces acting on the screen by giving more weight to an interval of angles rather than others. Numerically the effects of this modification affects the elements inside the inertial tensor.

The algorithm used is the Matlab function *lsqnonlin* which receives as input parameter the matrix of weights, forces and angle vectors and the error function. It returns back the values of the inertial matrix that goes inside eq.14 from which it is possible to evaluate the vector of forces.

In this sense the investigation is oriented on the correction of the F_y curve for which the error in the estimation arrives almost to 50% for some measures.

Two intervals of angles have been investigated:

- from -45° to 45°.

- from -80° to -60° and from 60° to 80°.

The reason under this choice is the willing of dividing the problem in two sub-problems and focusing on both of them separately, one regarding low angles and the other one regarding high angles.

For low angles the expectations are to correct the LSM curve slope and then to get next to the simulations values. For high angles the aim is to reach a better approximation that for this range of angles is totally absent.

Starting from the first interval, all the elements corresponding to the angles that belong to the interval -45°/45° are weighted by a factor 10 before being computed by the Least Squares Method. By increasing the weight given to the F_y the modifications on the inertial tensor results in a growth of the f_{yy} element. The modified

inertial tensors are reported below for each screen together with the representation of the new trend predicted by the Non Linear Least Squares Method.

6.6.1   R6T8 Screen: -45°/45° Weighing

The "new" inertial tensor registers an increase in the value of the f_{yy} component as said before:

$$F = \begin{bmatrix} 1493.600 & 0.000 \\ 0.000 & 412.200 \end{bmatrix} \tag{36}$$

This modification induces the LSM curve to reach a similarity in the slope with the curve representing the numerical results in a way similar to the F_z case.
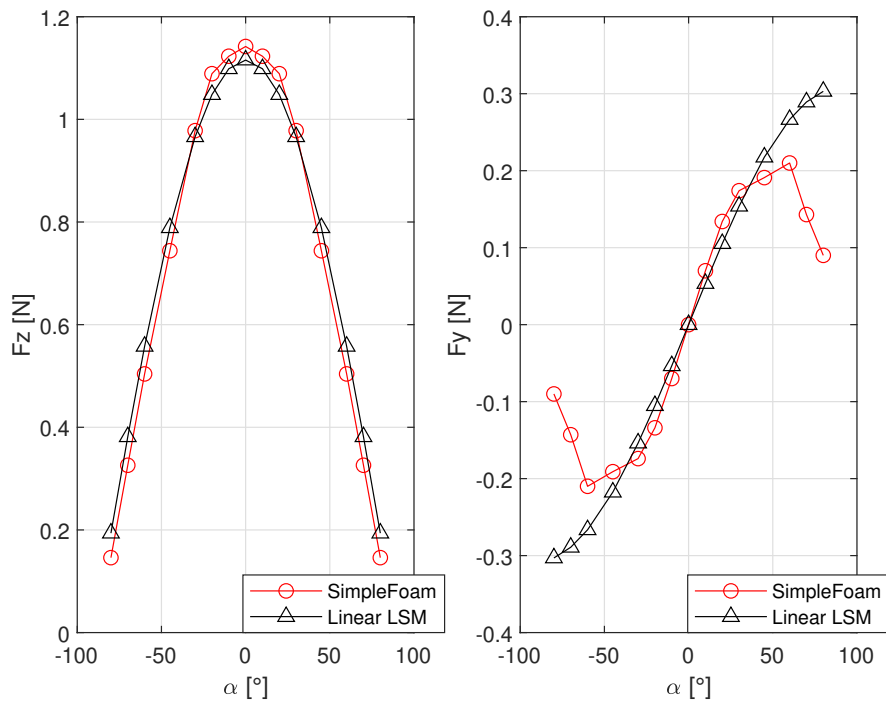

Figure 26: R6T8: SimpleFoam and Weighted Linear LSM comparison

By computing the percentage error of the results for each angle and by comparing it with the corresponding linear case seen in subsection 7.1 it is registered a reduction and most of all a trend that is very similar to the one we want to predict for the range of angles analyzed. However there is no coincidence in the results and still discrepancies remain.

6.6.2   R6.35T8 Screen: -45°/45° Weighing

Also for this case the correction is the same:

$$F = \begin{bmatrix} 1030.350 & 0 \\ 0 & 397.703 \end{bmatrix} \tag{37}$$

The situation is the same of the preceding case: the prediction of the trend of F_y is correct but the results are still not.
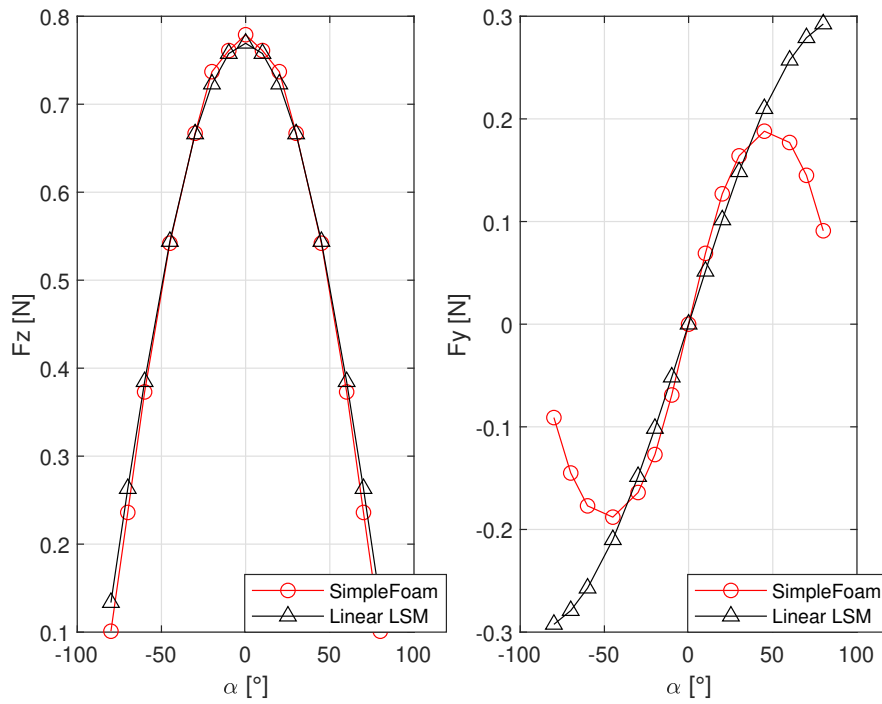

Figure 27: R6.35T8: SimpleFoam and Non Linear LSM comparison

The error registers a reduction too in the range of angles investigated as seen before.

6.6.3   Woven Wire Screen: -45°/45° Weighing

The woven wire screen needs one more consideration: the prediction made from the LSM is not accurate also for the normal force F_z. It is required then a weighing also for the axial component in order to predict the intensity of the force correctly especially because the angles involved are those for which the highest peaks of the F_z are registered.

$$F = \begin{bmatrix} 258.598 & 0.000 \\ -0.000 & 77.769 \end{bmatrix} \tag{38}$$
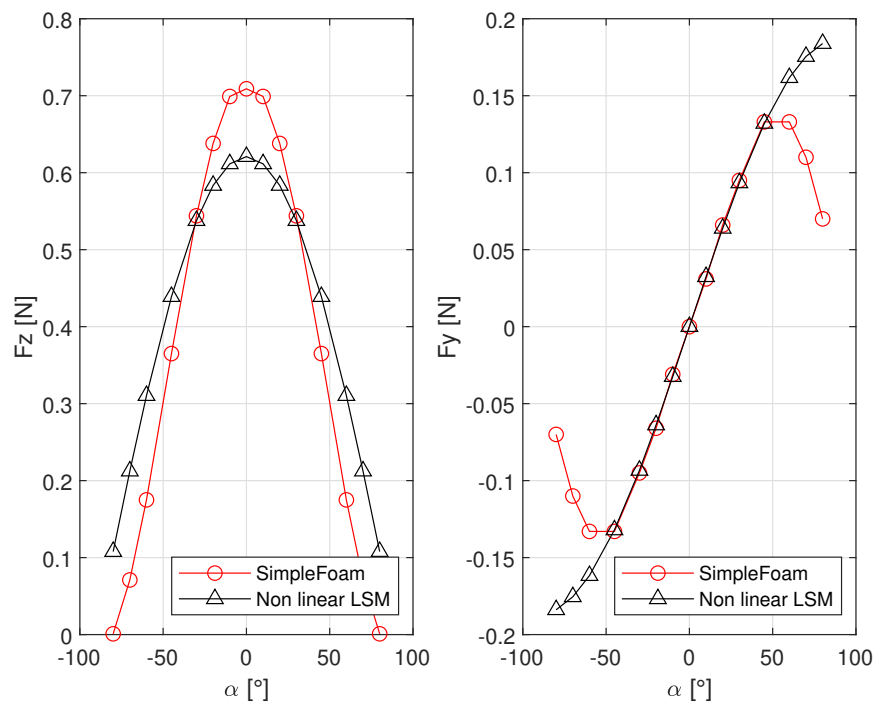

Figure 28: Woven Wire: SimpleFoam and Weighted Linear LSM comparison

For this case it is possible to see that the weighing of the tangential force is really effective not only in the prediction of the force's trend but also in the evaluation of the F_y for each angle. The error between the linear case and the weighted one is in fact reduced under the 1%.

6.6.4   R6T8 Screen: 60°/80° - -60°/-80° Weighing

In this subsection the weighing for high angles is investigated. It is an attempt to see if the Least Squares Method is able to catch the change in slope that the F_y curve has after 45°. The weight used is still 10 units for which the values of the forces corresponding to the interval considered are multiplied. Also in this case the change in the tensor corresponds in an increase of the f_{yy} component while the f_{xx} remains unaffected because there is not weight applied on the force F_z. The varied inertial tensor is then:

$$F_{60/80} = \begin{bmatrix} 1493.600 & 0.000 \\ -0.000 & 209.000 \end{bmatrix} \tag{39}$$
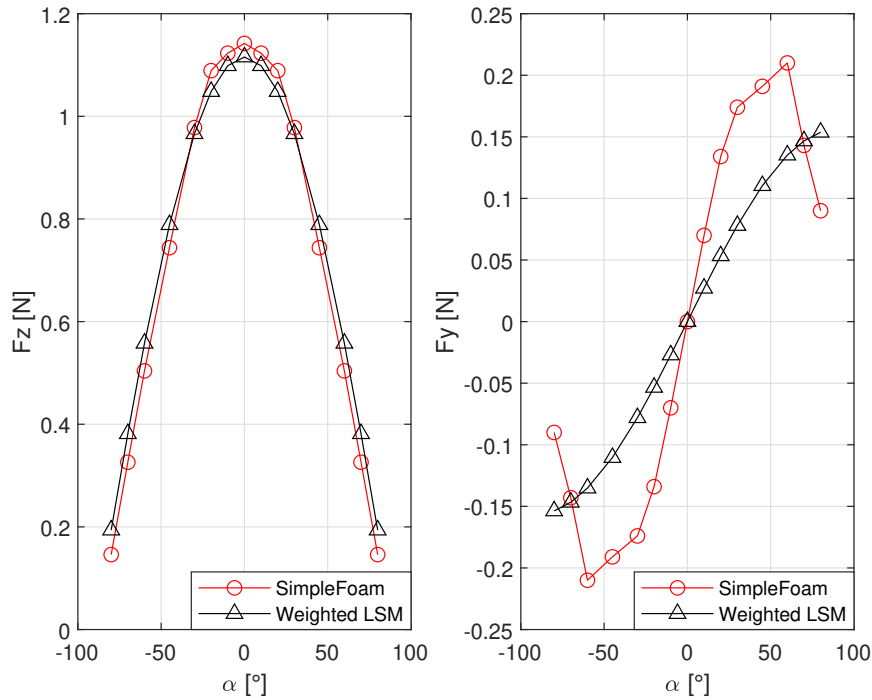

Figure 29: R6T8: SimpleFoam and Weighted LSM comparison

From a first esteem it is possible to see that the analytic curve is not able anymore to match with the curve of the numerical results. It seems instead that the weighted case is even more pendent than the original unweighted case.

58

6.6.5 R6.35T8 Screen: 60°/80° - -60°/-80° Weighing

In this subsection the weighing for high angles is investigated. It is an attempt to see if the Least Squares Method is able to catch the change in slope that the F_y curve has after 45°. The weight used is still 10 units in order to be able to compare the consistency of the results with that obtained for the previous weighting. Also in this case the change in the tensor corresponds in an increase of the f_{yy} component while the f_{xx} remains unaffected because there is not weight applied on the force F_z. The varied inertial tensor is then:

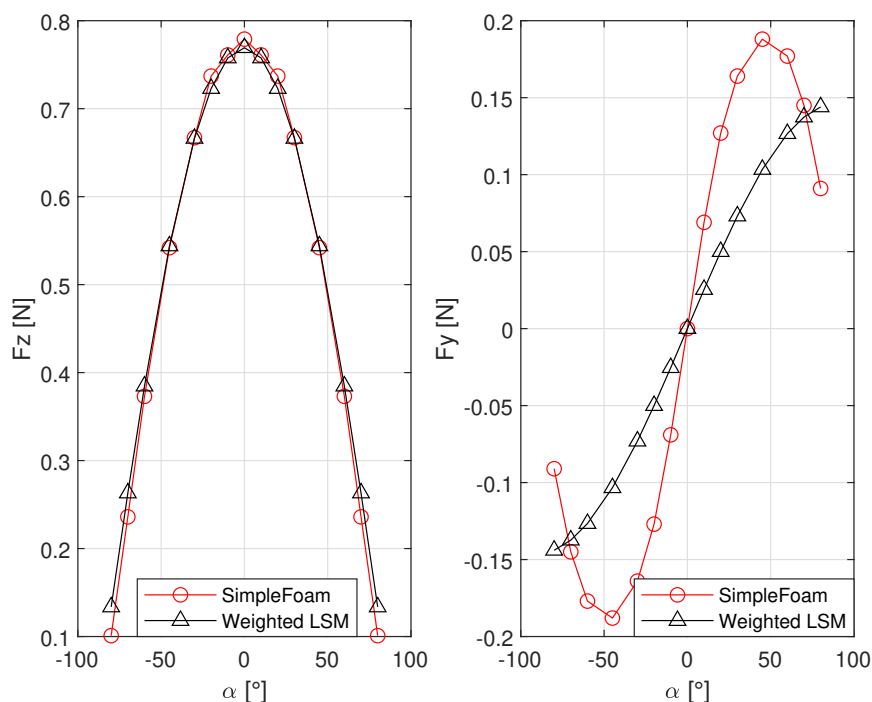$$F_{60/80} = \begin{bmatrix} 1030.0 & 0.000 \\ -0.000 & 195.800 \end{bmatrix} \tag{40}$$

Figure 30: R6.35T8: SimpleFoam and Weighted Linear LSM comparison

The same result as before can be seen here and for the following woven wire screen. In this sense it is possible to conclude that is preferable to choose a weighting for low angles.

6.6.6 Woven Wire Screen: 60°/80° - -60°/-80° Weighing

In this subsection the weighing for high angles is investigated. It is an attempt to see if the Least Squares Method is able to catch the change in slope that the F_y curve has after 45°. The weight used is still 10 units for which the values of the forces corresponding to the interval considered are multiplied. Also in this case the change in the tensor corresponds in an increase of the f_{yy} component while the f_{zz} remains unaffected because there is no weight applied on the force F_z. The varied inertial tensor is then:

$$F_{60/80} = \begin{bmatrix} 258.598 & 0.000 \\ -0.000 & 46.114 \end{bmatrix} \tag{41}$$
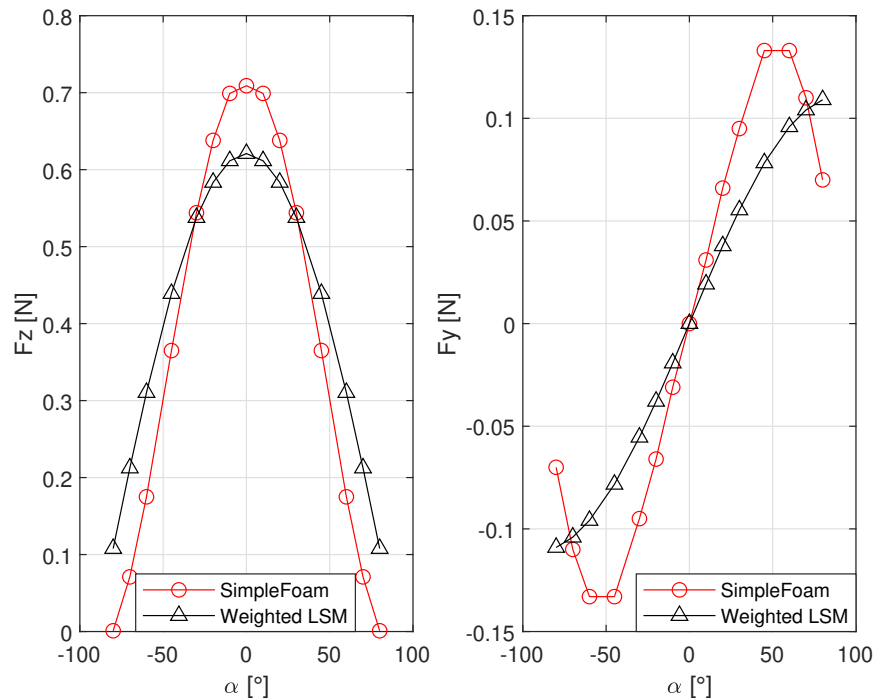

Figure 31: Woven Wire: SimpleFoam and Weighted Linear LSM comparison

6.6.7 Woven Wire Screen: Multiple Weighting

Until now the weighting has only been concentrated on the vertical component of the forces acting on the screen because the Least Squares Method predicted well the intensity of the axial component on its own. To be coherent with the analysis also for this case was considered only the weighting of the F_y. However for the woven wire case there is also a discordance for the axial force. Being the F_z the most intense load acting on the porous screen this component requires a weighting too.

The easiest way to start from is to consider the weighted case on the interval -45°/45° previous seen as starting point in order to maintain the precision obtained for the vertical force F_y. However the attempt to apply the same weighting to the axial component would add too many constrains to the problem having as result the not expected outcome. In fact the Least Squares Method is still not able to match the analytic curve as the previous cases.

In order to gain the precision desired the interval of weighting needs to be reduced or for the axial force or for the vertical one. The best choice is to give priority to the axial components for which the loads reach higher values and then reduce the interval for the F_y. However the LSM is still not able to reach the same values for the axial force. In this sense the weighting for the axial force needs to be performed for a restricted number of angles, precisely for the interval -10°/10°.

The resulting inertial tensor has the f_{zz} component increased in its value while the f_{yy} remains still the same as expected:

$$F = \begin{bmatrix} 294.521 & 0.000 \\ -0.000 & 77.769 \end{bmatrix} \tag{42}$$

The comparison of the analytic curve with the explicit one is presented below.
It is possible to see that the vertical force trend remains unvaried while the axial force is shifted upwards. The vales in the interval -10°/10° are perfectly matched while for the remaining angles are overrated.
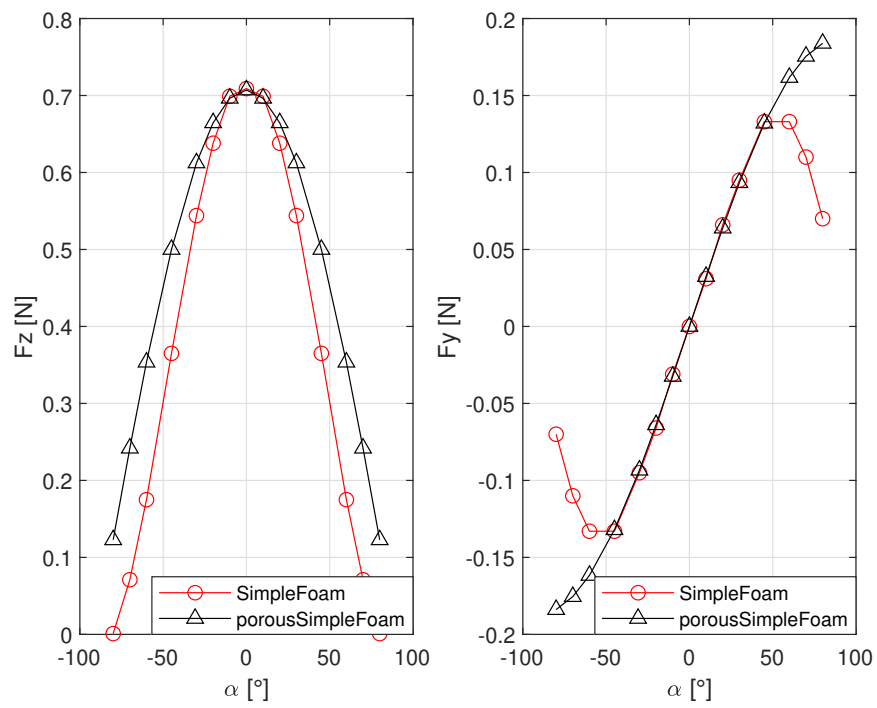
Figure 32: Woven Wire: SimpleFoam and Weighted Linear LSM comparison

6.7 PorousSimpleFoam Results

Porous simulations regard the last part of this work. Once obtained the values of the inertial tensor for each grid it is possible to fill the *porosityProperties* dictionary with them and run the simulations. The dimensions of the numerical domain are kept untouched for this second trance of simulations. The problem analyzed is in fact the same, what changes is only the way to represent the porosity inside the channel flow. For this purpose the porous septum is replaced with a block which has the same characteristic dimensions of it.

The channel flow is exactly the same of the starting one: it has been divided in a sequence of connected blocks meshed with an increasing number of cells until the porous zone location, after that the blocks discretization starts decreasing. The passage from one block to another implies the doubling in the number of cells. Not being the presence a physical grid anymore, the necessity to recur to the *snappy-HexMex* process misses too, resulting in a huge reduction of the number of cells required for the numerical simulation. In fact this entire trance of simulation is characterized by far lower computational costs being the convergence speed very fast.

Before looking at the results, convergence analysis have been conducted in order to make the problem be independent from the mesh discretization. The number of cells for which this independence of the results is found has been around 20.000 units.

RANS simulations still has been used to solve the fluid flow inside the channel, choosing the k-ϵ turbulence model as closure model. The k-ϵ model is one of the possible choices that OpenFoam guarantees in the turbulence dictionary.

In the figures below are presented the results obtained by using the modified *porousSimpleFoam* solver. Two different cases have been considered for each screen: the general case in which the components of the inertial tensor are obtained by the implementation of the Least Squares Method, and a second case in which a Weighted Least Squares Method is used. In this last case it will be put more weight on the angles belonging to the interval between -45° and 45°.

For each page will be reported the difference between the two cases. For what concerns the normal force F_z the trend is the same for both the situations. What changes is the slope of the curves for the tangential force F_y: by using the LSM the foreseen trend (black curve) is not coincident with the numerical results of the SimpleFoam simulations, as stated in the previous chapter, and the results obtained with PorousSimpleFoam slide on the black curve too. By using a non linear approach the results get better in the sense that the foreseen trend provided by the non linear LSM and the results of porousSimpleFoam (blu dotted curve) follows the red curve of the numerical results of SimpleFoam simulations.

It is possible to see that shifting from the "linear" case to the "non-linear" one

the *porousSimpleFoam* curve approaches very well the reference red one with a reduction of the error in the prediction of the result.

6.7.1 R6T8: porousSimpleFoam Results
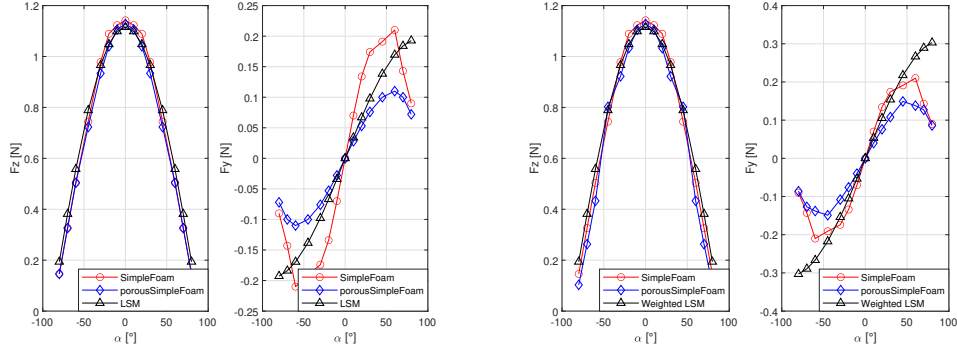
(a) R6T8: SimpleFoam, PorousSimpleFoam and LSM comparison

(b) R6T8: SimpleFoam, PorousSimpleFoam and Weighted LSM comparison
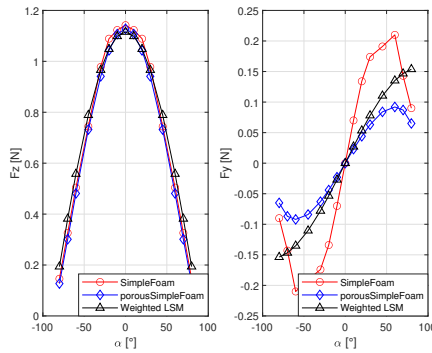
(c) R6T8: SimpleFoam, PorousSimpleFoam and 60°/80° Weighted LSM comparison

Figure 33: R6T8: results comparison

All the three cases guarantee a good prediction of the axial force. The best prediction of the vertical one is instead given in the case b. The weighting in the interval -45°/45° permits not only to follow the curve of the explicit simulations for low angles but also to get closer to the red curve for high angles.
The weighting for high angles doesn't have the hoped effects: the implicit curve is not able in fact to follow that of the explicit results.

6.7.2 R6.35T8 porousSimpleFoam Results

(a) R635T8: SimpleFoam, PorousSimple-Foam and LSM comparison

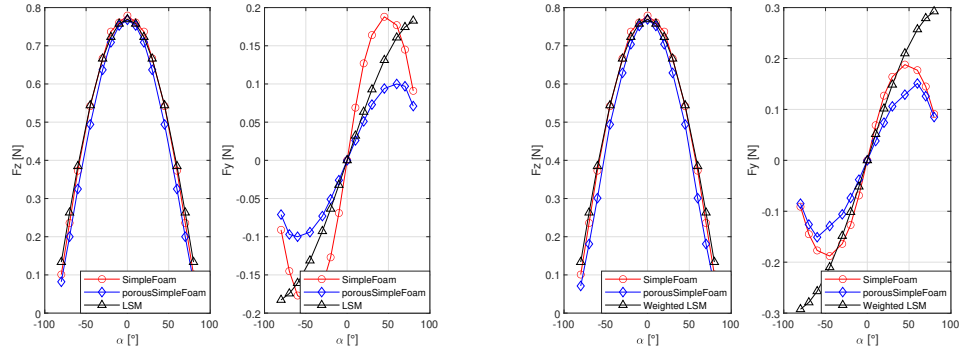(b) R635T8: SimpleFoam, PorousSimple-Foam and Weighted LSM comparison

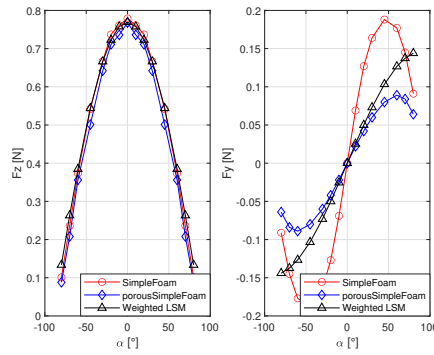(c) R635T8: SimpleFoam, PorousSimple-Foam and 60°/80° Weighted LSM comparison

Figure 34: R635T8: results comparison

6.7.3 Woven Wire porousSimpleFoam Results
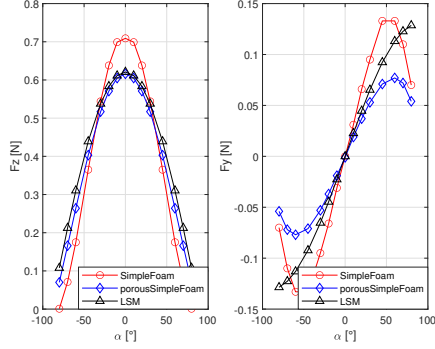
(a) Woven Wire: Non Weighted LSM
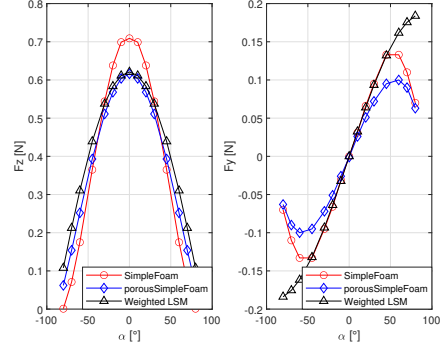
(b) Woven Wire: -45°/45° Weighted LSM
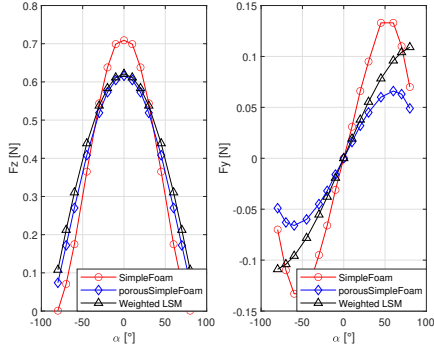
(c) Woven Wire: 60°/80° Weighted LSM
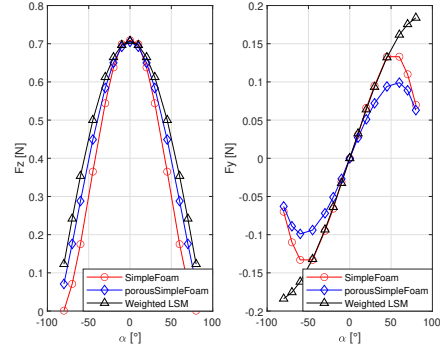
(d) Woven Wire: Weighted LSM on F_z and F_y

Figure 35: Woven Wire Screen: SimpleFoam, PorousSimpleFoam and LSM results comparison

As seen for the previous cases, the application of the weighting inside the range -45°/45° is the better solution to have a good prediction. The addition of the weighting also for the axial force is able to improve the quality of the results also for this case. To reach this result is however necessary to reduce the interval of weighting for the axial force to have an efficient effect. Otherwise the problem would have had too many constraints applied on it and no improvements would have been achieved.

6.7.4   R6.35T8 Variable Thickness Simulations

Until now the procedure followed for all the types of screens has been that of computing their forces with the explicit simulations, using their results to evaluate the components of the inertial tensor through the Least Squares Method and then put them in the working directory of myporousSimpleFoam to perform the last simulations. In this third phase of the work the thickness of the porous block that replaces the screen has been kept equal to the one of the screen itself. The aim was in fact that of seeing how well a porous zone with the same dimension of the screen could replace the physical presence of the it inside the domain. Once having verified the validity of this approach and having obtained satisfying results it is interesting then to analyze what happens if the extension of the porous zone inside the domain in increased maintaining at the same time its properties. In other words we want to see if it is possible to vary the thickness of the porous zone maintaining the same results seen before.

For example, the user wants to represent a porous region which has the same porosity properties of a certain screen in order to obtain the same results seen for that screen but in a length that is double the thickness of the screen itself. To do that it is necessary to re-compute another time the components of the inertial tensor taking into account in the calculation of the elements the porous block desired thickness.

Three different values of thickness will be inspected: 2mm, 4mm and 8mm. The thickness of 2mm has been analyzed in the previous chapters. It lasts to see if by doubling the value of the thickness each time it is possible to obtain the same results.

The Least Squares Method is then used to compute the new tensor for each test case. Being the system linear, the multiplication of the tensor affects directly the elements of the tensor itself, in the sense that for a doubling of the thickness a halving of the components is registered. The same happens for the 8mm test case. The porous properties of the screen are linearly distributed on the length of the block.

The test case considered for this section is the R6.35T8, one of the most coherent in the results. In order to deal with results principally similar to that of the explicit analysis the set of simulations with weight on the interval -45°/45° has been considered. The inertial tensor for the R6.35T8 screen with 2mm of thickness was:

$$F_{2mm} = \begin{bmatrix} 1030.350 & 0 \\ 0 & 397.703 \end{bmatrix} \tag{43}$$

The algorithm of the Least Squares Method has been modified changing the value of the control volume inside the expression for the computation of the forces. Its

value has been doubled being the thickness modified from 2mm to 4mm. This results in a new inertial tensor like the one presented below:

$$F_{4mm} = \begin{bmatrix} 515.013 & 0 \\ 0 & 198.809 \end{bmatrix} \tag{44}$$

And the same procedure goes for the 8mm case with the new inertial tensor of:

$$F_{8mm} = \begin{bmatrix} 257.506 & 0 \\ 0 & 99.404 \end{bmatrix} \tag{45}$$

The simulations performed for the three test cases analyzed show that the results are almost coincident for all the angles as it is possible to see in fig.36. This
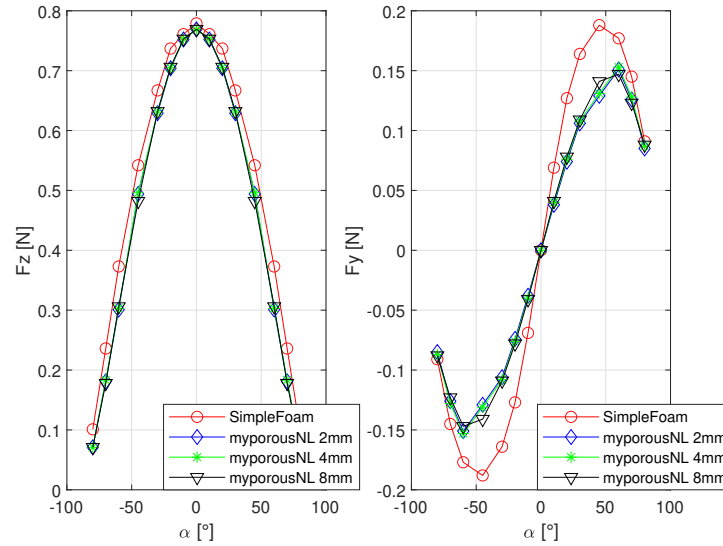
Figure 36: Stretched Screen: F_z and F_y comparison on zx and zy plane with thickness variation

confirms the fact that, knowing the properties and the effects of a screen in terms of the forces generated on its surface, it is possible to define a porous region in place of the real screen whose thickness can be freely chosen by the user without affecting the quality and the accuracy of the results.

A direct consequence of this fact is that also the error doesn't vary with the variation of the thickness but depends only on the LSM prediction:

Considerations about figure36 have been already treated in the previous chapter, instead some words can be said for the errors. In fig.37 are represented the percentage errors for the two components F_z and F_y.

69
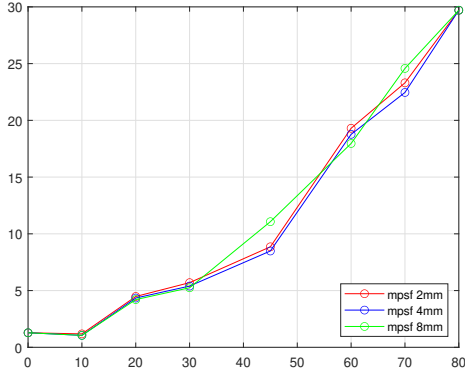
(a) R6.35T8 Screen: F_z Percentage error (b) R6.35T8 Screen: F_y Percentage error
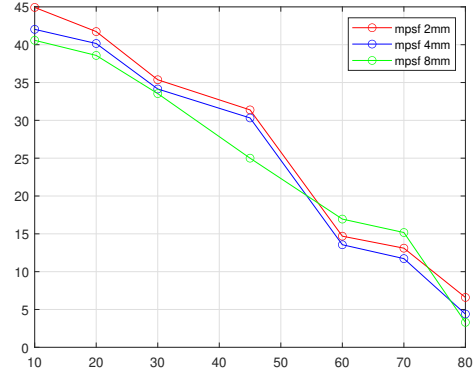
Figure 37: R6.35T8 screen: percentage error representation

The error of the axial component is acceptable until 45° where it reaches a maximum of 10% of difference. After 45° the measures are not trustworthy anymore. However in this analysis the reliability of the results after 45° can be in some sense neglected because the peaks of intensity of the axial force are reached for low angles for which the error is under the 5%. For the remaining values in the interval from 45° the prediction of the force trend is still captured. So, even if the error increases a lot after some point it is possible to consider this model a valid solution to reproduce porosity. The error of the tangential component instead suffers of low reliability for low angles but recovers its precision for high angles, for which the Least Squares Method has difficulties in following the variation of their corresponding values. In this sense, the error for the F_y component is not acceptable but regards quantities that are small with respect to the ones observed for the axial component. Besides the trend of the tangential component is also captured very well and gains precision at high angles, where the trend of the curve changes its slope.

In conclusion, the reproduced values of the axial force are acceptable at low angles where the maximum intensities are reached, while for the tangential force even if the data are not precise, the trend of the curve is captured very well.

7 Stretched Screen

The procedure used for the stretched screen is different from the previous cases. A 3D linear solver has been used to predict the entity of the forces on the surface in fact.

The geometry has been recovered by a previous thesis work done by A.Buscemi (cita letteratura), in which the behaviour of the screen was analyzed for a range of angles between -45° and 45°.

Its characteristic dimensions are:

- length: 250 mm

- height: 84 mm

- thickness: 50 mm

The choice of using a 3D solver instead of a 2D one is related to the fact that the screen is not symmetric. This means that the vector of forces recorded for a variation of the wind orientation in the zx plane is different from that registered for a variation in the zy plane.

The utility of this screen is that it is able to grant a great blockage of the flow and at the same time redistribute the load not only on z axis but also on y one. For example at 0° the axial load is in fact decomposed in two components with similar intensity on z axis as a normal stress and on y axis as shear stress, reducing then the contribution of the normal force that is the one we want to dissipate more. The contribution on the x axis for all the wind orientation is instead negligible in both cases on zx and zy plane and then it is not reported.

By considering a variation of the wind orientation on the xz plane it is possible to see that both F_z and F_y trend decreases almost linearly.

The variation on the zy plane instead is first of all asymmetric. The geometry of the screen in fact is asymmetric itself and this results in a different distribution of the forces. For a positive variation of the wind orientation the F_z and F_y components decrease faster than the previous case reaching even negative values for high incidence angles. This happens because each single stretched sheet sticks out downwards from the screen. When the flow starts having a positive incidence angle it is going to be always more parallel to the inclination of the sheets avoiding to be arrested by the geometry.

For a negative variation of the wind orientation the effects is instead the opposite. The fluid flow finds itself always more orthogonal to the outgoing sheets by resulting in an increase of the resistance in both the directions.

For what concerns the implementation of the least squares method for this case, the procedure depicted in section 6 has been used. To predict the vector of forces,

Figure 38: Stretched Screen Element

the analytic trend of them with wind orientation is required both on zy and zx plane, and in addition, forces on the bisector plane are required too. Compute the forces on the bisector plane means to project them on a reference system that is rotated of 45° with respect to the normal axis z. Actually, the consideration of all the three components on the bisector plane could lead to an over-determined problem. In this sense only one component for each angle is sufficient to predict the forces behaviour and this is why we consider only the F_y force on the bisector plane.

Then the vector of known terms can be rewritten in the following way:

$$
\mathbf{b} = \begin{bmatrix} F_{x\alpha_1}^{xy} F_{y\alpha_1}^{xy} F_{z\alpha_1}^{xy} \cdots F_{x\alpha_n}^{xy} F_{y\alpha_n}^{xy} F_{z\alpha_n}^{xy} F_{x\alpha_1}^{xz} F_{y\alpha_1}^{xz} F_{z\alpha_1}^{xz} \cdots F_{x\alpha_n}^{xz} F_{y\alpha_n}^{xz} F_{z\alpha_n}^{xz} F_{y\alpha_1}^{bis} \cdots F_{y\alpha_n}^{bis} \end{bmatrix}
$$
(46)

With this purpose we are going to expect a fully determined inertial tensor that now has 9 components. Even for this screen are reported both the linear LSM and the non linear LSM with the latter performed with weights applied for different intervals of angles.

7.1 Stretched Screen: Linear LSM ZX plane

For the LSM with no weight applied the inertial tensor takes the following form:

$$F = \begin{bmatrix} 40.8598 & -18.8984 & 0.0000 \\ -35.1627 & 25.1813 & 0.0013 \\ 1.1393 & 0.0018 & 0.0000 \end{bmatrix} \tag{47}$$

Now the tensor is non symmetric and almost all of its components are non zero. The unique zero components are those of the third column that are representative of the contribution given by the horizontal force F_x. This result could be predictable from the fact that the forces in the horizontal direction are approximately zero and then the effects of the terms associated with the F_x are absent. The corresponding forces evaluated on zx plane are reported below, where for zx plane is intended the plane in which the wind orientation is varied.

Figure 39: Stretched Screen: SimpleFoam, LSM comparison - zx plane

The prediction made by the Least Squares Method for the F_z is similar to the one seen for the woven wire screen. The axial force is underrated even if the trend represented is coherent. Nevertheless as we have seen for the woven wire screen when the evolution of the force is too sharp the LSM is not able to follow perfectly it, resulting in a wider curve from 45° on.

Below the same graph is presented with the addiction of porous simulations. As seen before the implicit simulations are in accordance with the prediction made by the Least Squares Method, however it is necessary a weighting for the range of angles between -45° to 45° where the forces F_z and F_y are underrated.
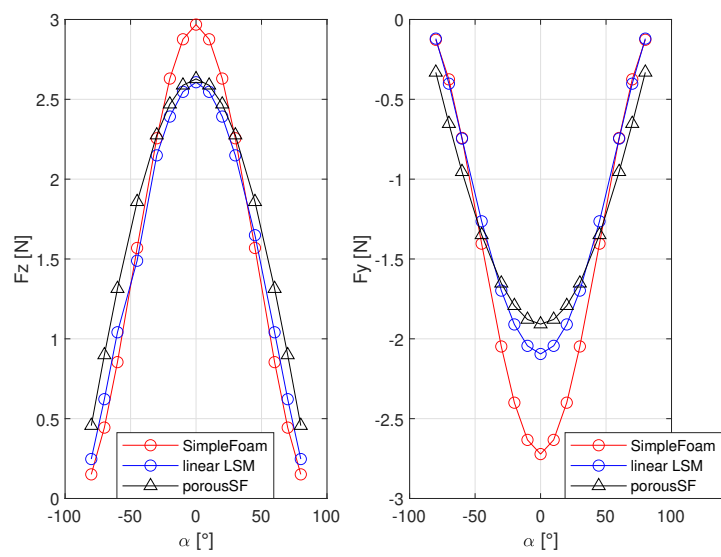
Figure 40: Stretched Screen: SimpleFoam, LSM and PorousSimpleFoam comparison - zx plane

7.2 Stretched Screen: Linear LSM ZY plane

The algorithm for the computation of the Least Squares Method on the zy plane gives also in this case a non symmetric matrix with almost all the elements different from zero.

$$F = \begin{bmatrix} 40.8598 & -18.8984 & 0.0000 \\ -29.661 & 23.013 & 0.0013 \\ 1.1393 & 0.0018 & 0.0000 \end{bmatrix} \tag{48}$$

The unique zero components are still those of the third column that are representative of the contribution given by the horizontal force F_x. The reason is the same depicted in the previous section, that is the horizontal force has a negligible intensity with respect to the other two components. The corresponding forces evaluated on zy plane are reported below, where for zy plane is intended the plane in which the wind orientation is varied. For a variation of the speed orientation
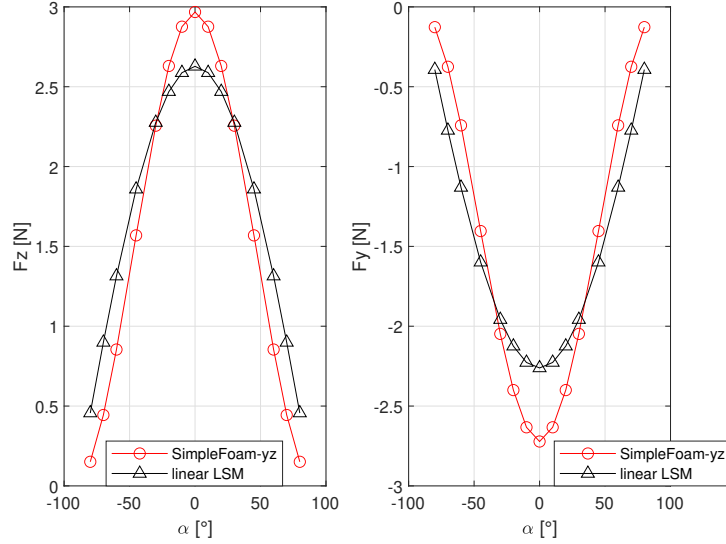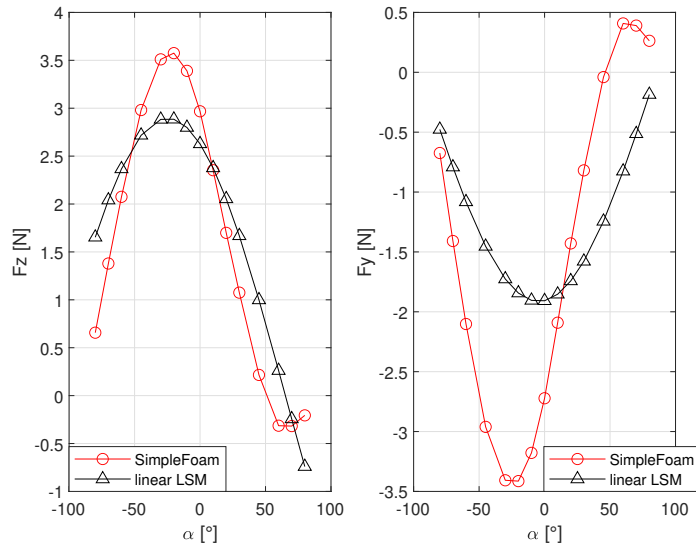

Figure 41: Stretched Screen: SimpleFoam, LSM and PorousSimpleFoam comparison - zy plane

on the zy plane the distribution of the forces intensity is not symmetric anymore on the interval that goes from -80° to 80°. The Least Squares Method follows the trend of the forces on both the axes, however a weighting is needed also in this case to improve the prediction at least for the range of angles for which the forces reach their highest intensities.

8 Porous Media Visualizations

In the figures below the pressure and velocity field of the R6.35T8 screen are reported. The thickness of the porous media is of 2mm and it corresponds to the thin yellow region in fig.42a. It is possible to see how the implicit model permits a fast transition from a given pressure field to a lower one. Due to the fact that the

(a) R6.35T8 Screen: Pressure Visualization   (b) R6.35T8 Screen: Speed Visualization
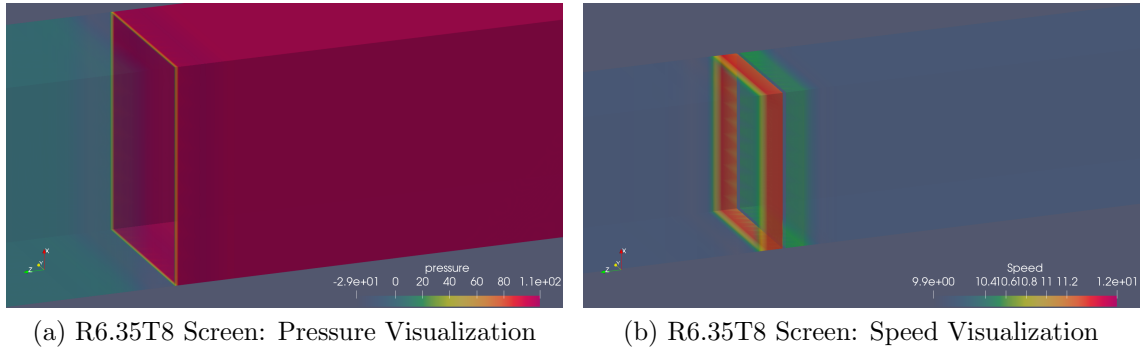
Figure 42: R6.35T8 screen: Channel Flow Visualizations

real screen is absent, the porous media is not able to represent precisely the local effects produced to the flow. However the speed visualization in fig.42b shows how the model can reproduce the local increase of speed in proximity of the porous media.

By increasing the thickness naturally the transition is more gradual and the evolution of the flow is clearer to be visualized.

Below the visualizations for the stretched screens are presented. A comparison between the explicit and implicit simulations is presented both for speed and pressure profile. It is possible to see that the porous media approach is not able to reproduce the local effects generated by the presence of the porous screen but the global effects resumed in the change in intensity of the physical quantities are well captured.
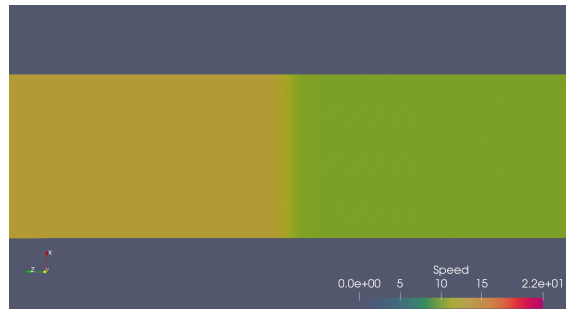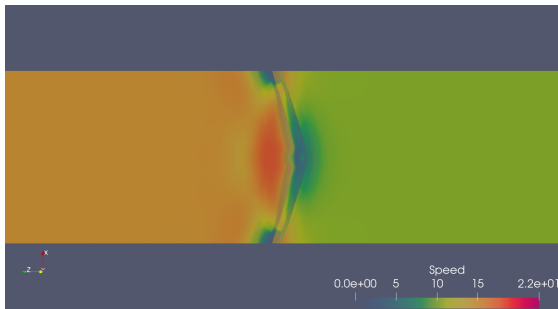
(a) Stretched Screen: Explicit Speed Visualization

(b) R6.35T8 Screen: Implicit Speed Visualization
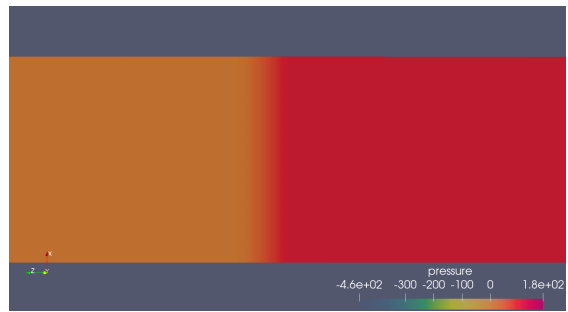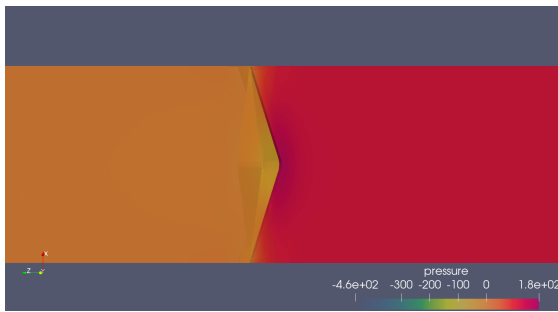
(c) Stretched Screen: Explicit Pressure Visualization

(d) Stretched Screen: Implicit Pressure Visualization

Figure 43: Stretched Screen: Channel Flow Speed and Pressure Visualizations

9 Conclusions

Over the years porous screens have increased their range of applicability essentially because their effects can be used for different purposes that space from civil engineering to environmental one. The interaction between these structures and the wind motion is the argument of this work whose aim is the definition and the prediction of the aerodynamic effects provided by them. The geometries of the screens have been generated with CAD software and tested with the CFD. The performance of each screen has been quantified through the pressure jump imposed to the flow. To deal with a quantity that is independent from the dimensions of the screen the pressure jump has been normalized with respect to the area of the screen and the intensity of the wind speed, obtaining an adimensional coefficient commonly known as pressure loss coefficient in the existing literature.

It is found that for an incoming normal flow the pressure loss coefficient is independent from the intensity of the speed in the range between $5\frac{m}{s}$ and $20\frac{m}{s}$ confirming the same results obtained in wind tunnel tests. The variation of the speed orientation shows instead an evolution of the pressure loss coefficient similar to trigonometric curves for screens characterized by a symmetric trend of the forces on both the axes laying on the screen.

The aim to reduce computational costs leads this work to investigate a new model based on *PorousSimpleFoam* that is able to predict the aerodynamic effects provided by porous screens without representing them in the numerical domain. This new model introduces the generation of a porous media that obeys to a modified Darcy-Forchheimer model which takes into account the anisotropy of the screen. The outcome consists in a region that provides the same global effects of the screen. The results obtained have been compared with the analytic simulations of the Least Squares Method and the explicit ones computed with SimpleFoam. It is found that the results of the implicit simulations are consistent with those of the explicit ones: the trend of the forces with the wind orientation is well captured for both the axial and the tangential case while the estimation of the values of the forces until 45 remains inside an acceptable interval of confidence.

A   Least Squares Method Implementation

For the explicit numerical simulations the obstacle inside the domain was represented by the physical presence of the slab, producing a pressure jump over the screen area. As already said, the objective of this work is that of finding an alternative way to reproduce porosity without having the physical presence of the screen. The OpenFoam solver porousSimpleFoam has been identified as a good tool to do that. With respect to the SimpleFoam case, Navier-Stokes momentum equation now contains a sink term whose role is to take into account the porosity introduced by the slab. The position of the porous region will be coincident with that one previously occupied by the screen. SnappyHexMesh dictionary doesn't need to be used anymore because the porous zone is directly defined in the blockMesh directory: once chosen the position, the inlet and outlet areas of the interested block are specified with a porosity patch in order to be recognized by OpenFoam. At the same time the porosity properties need to be specified by the user itself in the porosityProperties dictionary presented below.

```
/*--------------------------*- C++ -*----------------------------*\
  =========                 |
  \\      / F ield           | OpenFOAM: The Open Source CFD
   Toolbox
   \\    /   O peration       | Website:  https://openfoam.org
    \\  /    A nd             | Version:  6
     \\/     M anipulation   |
\*---------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      porosityProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

porosity
{
    type            myDarcy;

    cellZone        porosity;

    d   (0 0 0);

    f   (0 0 0);

```

```
28    coordinateSystem
29    {
30        type    cartesian;
31        origin  (0 0 0);
32        coordinateRotation
33        {
34            type    axesRotation;
35
36          e1      (1 0 0);
37        e2      (0 1 0);
38
39        }
40    }
41 }
42
43 // ******************************************************* //
```

Listing 2: PorosityProperties directory

As can be seen it is possible to specify the main directions of the porosity by defining the components of two versors e_1 and e_2 while the third one is computed as the vector orthogonal to the plane generated by e_1 and e_2. How the porous zone behaves is instead defined by filling the components of Darcy and ForchHeimer tensors. For the range of speed of interest in this problem viscosity has negligible effects, then Darcy term can be neglected and its components be put to 0. However in this file can be defined only three components of the tensor, corresponding to the ones placed on the main diagonal of the matrix. This corresponds to a problem in which the effect of porosity acts only on three main axis of the global reference system basically.

With the procedure depicted in appendix.B the main code of *porosityProperties* dictionary changes as illustrated in the following figure:

```
1 /*--------------------------*- C++ -*--------------------------*\
2   =========                 |
3   \\      /   F ield         | OpenFOAM: The Open Source CFD
    Toolbox
4    \\    /    O peration     | Website:  https://openfoam.org
5     \\  /     A nd           | Version:  6
6      \\/      M anipulation  |
7 \*---------------------------------------------------------------*/
8 FoamFile
9 {
10    version     2.0;
11    format      ascii;
12    class       dictionary;
13    location    "constant";
14    object      porosityProperties;
```

```
15 }
16 // * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
17
18 porosity
19 {
20     type              myDarcy;
21
22     cellZone          porosity;
23
24     d    (0   0   0
25          0   0   0
26          0   0   0);
27
28     f    (0   0   0
29          0   0   0
30          0   0   0);
31
32     coordinateSystem
33     {
34         type    cartesian;
35         origin  (0 0 0);
36         coordinateRotation
37         {
38             type    axesRotation;
39
40           e1        (1 0 0);
41         e2        (0 1 0);
42
43         }
44     }
45 }
46
47 // ********************************************************** //
```

Listing 3: PorosityProperties Directory

By compiling the new solver it is given to the user the possibility to specify all the components of the two tensors and not only their main diagonal. As already said this means that the solver is capable to take into account also the anisotropy (if present) of the porous structures.

B Modified porousSimpleFoam Code

Below is displayed part of the code that belongs to the compiler file for the computation of the elements of the DarcyForchheimer tensor. The one presented is the basic file located in porousSimpleFoam, for which only the main diagonal terms are defined.

```
1  // * * * * * * * * * Member Functions  * * * * * * * * * * //
2
3  void Foam::porosityModels::DarcyForchheimer::
       calcTransformModelData()
4  {
5      // The Darcy coefficient as a tensor
6      tensor darcyCoeff(Zero);
7      darcyCoeff.xx() = dXYZ_.value().x();
8      darcyCoeff.yy() = dXYZ_.value().y();
9      darcyCoeff.zz() = dXYZ_.value().z();
10
11     // The Forchheimer coefficient as a tensor
12     // - the leading 0.5 is from 1/2*rho
13     tensor forchCoeff(Zero);
14     forchCoeff.xx() = 0.5*fXYZ_.value().x();
15     forchCoeff.yy() = 0.5*fXYZ_.value().y();
16     forchCoeff.zz() = 0.5*fXYZ_.value().z();
17
18  // ********************************************************** //
```

Listing 4: DarcyForchheimer Compiler Code

The modifications, as stated in section.3 consist in the definition of the non diagonal terms too. For the range of speeds considered in this analysis the viscous tensor is not necessary, in fact during numerical simulations only the inertial one was specified. Below is written the part of the code (renamed as myDarcy.C) modified with the addition of the non diagonal terms.

```
1  // * * * * * * * * * * * * * * * Member Functions  * * * * * * * *
       * * * * * //
2
3  void Foam::porosityModel3Ds::myDarcy::calcTransformModelData()
4  {
5      // The Darcy coefficient as a tensor
6      tensor darcyCoeff(Zero);
7      darcyCoeff.xx() = dXYZ_.value().xx();
8      darcyCoeff.xy() = dXYZ_.value().xy();
9      darcyCoeff.xz() = dXYZ_.value().xz();
10
11     darcyCoeff.yx() = dXYZ_.value().yx();
12     darcyCoeff.yy() = dXYZ_.value().yy();
13     darcyCoeff.yz() = dXYZ_.value().yz();
```

```
14
15      darcyCoeff.zx() = dXYZ_.value().zx();
16      darcyCoeff.zy() = dXYZ_.value().zy();
17      darcyCoeff.zz() = dXYZ_.value().zz();
18
19      // The Forchheimer coefficient as a tensor
20      // - the leading 0.5 is from 1/2*rho
21      tensor forchCoeff(Zero);
22
23      forchCoeff.xx() = 0.5*fXYZ_.value().x();
24      forchCoeff.yy() = 0.5*fXYZ_.value().y();
25      forchCoeff.zz() = 0.5*fXYZ_.value().z();
26
27      forchCoeff.yx() = 0.5*fXYZ_.value().yx();
28      forchCoeff.yy() = 0.5*fXYZ_.value().yy();
29      forchCoeff.yz() = 0.5*fXYZ_.value().yz();
30
31      forchCoeff.zx() = 0.5*fXYZ_.value().zx();
32      forchCoeff.zy() = 0.5*fXYZ_.value().zy();
33      forchCoeff.zz() = 0.5*fXYZ_.value().zz();
34
35  // ************************************************************
    //
```

Listing 5: Modified DarcyForchheimer Compiler Code

Changing the names of the compilers files naturally means to operate also to the headers ones. In this sense it is necessary to enter inside the .h files and change the names of the modified compilers file with the new names given to them.

References

[1]   A.R.Collar. "The Effect of a Gauze on the Velocity Distribution in a Uniform Duct". In: (1939).

[2]   Øystein Patursson; M. Robinson Swift; Igor Tsukrov; Knud Simonsen; Kenneth Baldwin; David W. Fredriksson; Barbaros Celikkol. "Development of a porous media model with application to flow through and around a net panel". In: *Ocean Engineering* 37 (2010), pp. 314–324. DOI: www.elsevier.com/locate/oceaneng.

[3]   Mitchell Cohen. "Modelling of Airflow through Wire Mesh Security Screens". In: (2015).

[4]   G. De Vahl Davis. "The Flow Of Air Through Wire Screens". In: ().

[5]   R.M.Davies G.I.Taylor. "The Aerodynamic of Porous Sheets". In: *A.R.C.* ().

[6]   Stefan Velikov Klementina Gerova and Kevin P. Garry. "The Effects of Porosity and Inclination on the Pressure Drop across Porous Screens and Honeycombs Used for Heat Exchanger Simulations in Wind Tunnel Studies". In: *SAE Int. J. Passeng. Cars - Mech. Syst. 6(2):2013* 6 (2013). DOI: doi:10.4271/2012-01-2340saepcmech.saejournals.org.

[7]   Kheir Al-Kodmany. "Sustainability and the 21st Century Vertical City: A Review of Design Approaches of Tall Buildings". In: *Buildings* 102 (2018), p. 11. DOI: https://www.mdpi.com/2075-5309/8/8/102.

[8]   L.Buscemi. "A CFD's modeling of a screen considering as a porous media". In: (2019).

[9]   L.Prandtl. "Attaining a Steady Air Stream In Wind Tunnels". In: *N.A.C.A.* 726 (1933).

[10]   C.W. Letchford. "Wind loads on rectangular signboards and hoardings". In: *Journal of Wind Engineering and Industrial Aerodynamics 89 (2001) 135–151* 89 (2001), pp. 135–151. DOI: http://elsevier.com/locate/jweia.

[11]   J. Tanny; I. Seginer M. Teitel; D. Dvorkin; Y. Haim. "Comparison of measured and simulated flow through screens: Effects of screen inclination and porosity". In: *Biosystems Engineering* 104 (2009), pp. 404–416. DOI: www.elsevier.com/locate/issn/15375110.

[12]   Mao Xu; Luca Patruno; Yuan-Lung Lo; Stefano de Miranda. "On the use of the pressure jump approach for the simulation of separated external flows around porous structures:A forward facing step". In: *Journal of Wind Engineering and Industrial Aerodynamics* 207 (2020). DOI: www.elsevier.com/locate/jweia.

[13] M. Robinson P.J. Richards. "Wind loads on porous structures". In: *Journal of Wind Engineering and Industrial Aerodynamics 83 (1999) 455-465* 83 (1999), pp. 455–465. DOI: http://elsevier.com/locate/jweia.

[14] Yizhong Xu; Mohamad Ya. Mustafa; Rajnish K. Calaya; Bjørn S. Sørensena. "Numerical simulation of non-normal wind load on porous fences. [Sustainable Solutions for Energy and Environment, EENVIRO 2016, 26-28 October 2016, Bucharest, Romania]". In: *Energy Procedia 112 (2017) 382 – 389* 112 (2017), pp. 382–389. DOI: hettepr:-/r/ecrveiaetwiveucnodmemrroenssp.orngs/liibcielintsyeso/fbyth-neco-nrgd/a4n.0iz/).

[15] O. Bistoni; G. Pomaranzi; P. Schito; A. Zasso. "Numerical characterization of three-dimensional screens, treated as porous media". In: (2020).