# Numerical Simulation of Fluid-Structure Interaction Using Loose Coupling Methods

Vom Fachbereich Maschinenbau
an der Technischen Universität Darmstadt
zur
Erlangung des Grades eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte

D i s s e r t a t i o n

vorgelegt von

**M. Sc. Galina Sieber**
aus Pleven, Bulgarien

| | |
|---|---|
| Berichtersttater: | Prof. Dr. rer. nat. M. Schäfer |
| Mitberichterstatter: | Prof. Dr.-Ing. B. Stoffel |
| Tag der Einreichung: | 17.09.2001 |
| Tag der mündlichen Prüfung: | 20.11.2001 |

Darmstadt 2002
D 17

Hiermit erkläre ich an Eides Statt, dass ich die vorliegende Arbeit – abgesehen von den in ihr ausdrücklich genannten Hilfsmitteln – selbständig verfasst habe und dass ich noch keinen Promotionsversuch unternommen habe.

Darmstadt, 14. September 2001

Galina Sieber

# Zusammenfassung

Die Simulation einer Vielzahl technischer und naturwissenschaftlicher Prozesse basiert auf der Lösung von Problemen mit Fluid-Struktur-Interaktion (FSI). Die in einem Fluid auftretenden Druck- und Scherspannungen erzeugen auf der Oberfläche eines angrenzenden Festkörpers Kräfte, welche zu einer Deformation der Struktur führen können. Demzufolge wird das Strömungsgebiet ebenfalls verformt, was zu einer Änderung des Strömungszustands führt. Dies geschieht solange, bis sich ein Gleichgewichtszustand zwischen Strömung und Festkörperdeformation einstellt. Deshalb muss zur Beschreibung des strukturmechanischen bzw. des strömungsmechanischen Teilproblems die Lösung des jeweilig anderen Teilproblems als Randbedingung am gemeinsamen Interface berücksichtigt werden.

Das Ziel dieser Arbeit ist die Entwicklung von effizienten Kopplungsalgorithmen zur Berechnung verschiedenster Fluid-Struktur-Probleme mit dreidimensionalen Strömungsgebieten und beliebigen elastischen Körpern. In dieser Arbeit werden inkompressible und Newtonsche Fluide sowie isotropische elastische Festkörper betrachtet. Für die Modellierung der Materialeigenschaften der Festkörper werden sowohl kleine als auch finite Deformationen angenommen und somit lineare bzw. geometrisch nicht-lineare Modelle verwendet.

Eine explizite und eine schwache implizite Kopplungsmethode der strömungsmechanischen und strukturmechanischen Gleichungen wurde entwickelt. Zu diesem Zweck wurde der Finite-Volumen-Code FASTEST-3D für die Lösung der strömungsmechanischen Gleichungen und das Finite-Elemente-Programm FEAP wurde für die Lösung der strukturmechanischen Gleichungen verwendet. Die beiden Löser wurden schwach gekoppelt, so dass der Strömungs- und Strukturteil abwechselnd gelöst werden. Für die Modellierung der Fluid-Struktur-Interaktion werden die Druck- und die Scherkräfte des Fluids auf die Oberflächenelemente des Strukturgebiets projiziert und als Randbedingungen für die Lösung der Strukturgleichungen verwendet. Damit die durch die Strukturdeformation verursachte Veränderung des Strömungsgebiets berücksichtigt werden kann, muss der Strömungslöser so modifiziert werden, dass die Navier-Stokes-Gleichungen in einer Euler-Lagrange-Form betrachtet werden. Das numerische Gitter im Strömungsgebiet wird dabei dem neuen Gebietsrand mittels einer linearen Interpolation angepasst. Den Navier-Stokes-Gleichungen wird noch das Geometrieerhaltungsgesetz beigefügt, um bei einer auftretenden Verzerrung der finiten Volumen die globale Massenerhaltung zu gewährleisten. Bei der expliziten Kopplungsmethode wird die Information zwischen den beiden Lösern nur einmal pro Zeitschritt ausgetauscht, während die implizite Kopplung auf einem Prediktor-Korrektor-Verfahren zur Bestimmung des Fluid-Struktur-Gleichgewichtszustands innerhalb eines jeden Zeitschritts basiert.

Der explizite Kopplungsalgorithmus wurde für Probleme mit kleinen Deformationen verwendet. Er wird dazu benutzt, um zum Einen die laminare Strömung in einem elastischen Rohr, zusammengedrückt durch äußere Kräfte, zu simulieren und zum Anderen um den stationären Zustand einer laminare Strömung in einer 90° T-Verbindung von elastischen Rohren zu berechnen. Diese Methode ist jedoch nicht geeignet für die Berechnung dynamischer Probleme mit finiten Deformationen aufgrund der auftre-

tenden Zeitschrittweitenlimitierung bei expliziten Verfahren. Die schwache implizite Kopplung ist für diese Art von Problemen von Vorteil, da hier keine Begrenzung der Zeitschrittweite vorliegt. Das Prediktor-Korrektor-Verfahren wurde erfolgreich verwendet, um sowohl den stationären Zustand als auch das dynamische Verhalten von FSI Problemen mit finiten Deformationen zu berechnen. Es wurde das Verhalten einer zwei-dimensionalen laminaren Strömung um einen elastischen Zylinder bei den Reynoldszahlen 20 und 100 studiert. Die numerische Untersuchung der impliziten Kopplungsmethode zeigt sehr gute Konvergenzeigenschaften des Verfahrens. Wird der Zeitschritt klein genug gewählt, genügt ein Prediktor-Korrektor-Schritt um die Konvergenz innerhalb eines Zeitschritts zu erreichen. In diesem Fall degeneriert die implizite zu einer expliziten Methode. Am Ende wird gezeigt, dass die drei-dimensionale instationäre Strömung um einen elastischen Zylinder, eingebaut in einem quadratische Kanal, erfolgreich modelliert werden konnte.

Die präsentierten Kopplungstrategien haben die Testbeispiele erfolgreich beschrieben und können somit auf weitere praktische Probleme angewendet werden. In Abhängigkeit von den Fähigkeiten der verwendeten Fluid- und Strukturlöser können mit den entwickelten Kopplungsmethoden verschiedenste technische Probleme mit Fluid-Struktur Wechselwirkung simuliert werden.

# Preface

The present research was done at the Department of Numerical Methods in Mechanical Engineering, in the frames of the DFG Graduate College "Modelling and Numerical Description of Technical Flows".

First of all, I would like to thank Prof. Dr. rer. nat. Michael Schäfer for the given opportunity, supervision and interesting discussions throughout my work as well as for the valuable remarks for improving this dissertation.

I would like also to thank Prof. Dr.-Ing. Stoffel for refereeing this work.

I also thank Dr.-Ing. Sebastian Meynen for his help in getting along with the program FEAP as well as for the useful talks on structural mechanics and fluid-structure interaction themes.

Thanks also to my colleagues at the Department of Numerical Methods in Mechanical Engineering for the nice and friendly atmosphere of work.

I want specially to thank my husband Dr.-Ing. Rolf Sieber not only for his help in studying the solver FASTEST-3D and for the various interesting discussions on fluid dynamics and fluid-structure interaction topics but also for his understanding.

Special thanks to my parents and brother for encouraging me to come to Darmstadt University of Technology and always supporting me morally.

Finally, I would like to thank the German Research Foundation (DFG) for the provided financial support.

Galina Sieber

# Contents

# 1 Introduction

## 1.1 Motivation. Fluid-structure interaction (FSI)

The simulation of various processes in engineering and nature requires the coupled solution of different physical problems. Many examples of multi-field systems can be found in practice. In the present research one special case of coupled problems will be investigated – the fluid-structure interaction (FSI) problems.

Applications in various areas require the consideration of an elastic structure surrounded by or conveying a fluid. Examples can be found in:

- technical devices – membrane pumps, heat exchangers, pipe-systems, stirring techniques, turbomachinery, airbags, jet engines
- aeroelasticity – airfoil flutter
- civil engineering – wind-induced oscillations of high buildings and bridges
- hydroelasticity - water penetration of off-shore structures, submarines
- biology - the blood circulation in human body, modelling of the heart valves

Two main mechanisms of interaction between the fluid and the structural physical domains can be distinguished:

• The interaction between the flow and the structure takes place only at the interface. Due to the fluid pressure and shear stress, forces appear on the structural boundary. These forces deform the structure and consequently the fluid domain. Consequently, the fluid domain deformation leads to a change in the flow field. Therefore, the solution of each subproblem has to be considered as a boundary condition on the interface for the other subproblem.

• The flow and the structure are coupled through the temperature field. The fluid and the temperature interact between each other. This requires the simultaneous solution of the fluid dynamics and the energy equations in the flow domain. On the other hand the structural deformation depends on the temperature. Hence, the coupled solution of the structural dynamics and the energy equations is also required in the structural domain. If the thermal stresses at the interface are much bigger than the fluid pressure and shear stresses, the fluid-structure interaction would be due to the temperature exchange through the domains' boundaries.

Obviously, the fluid and the structure may interact in both mechanisms. The possible ways of fluid-structure interaction are depicted in Figure 1.1.

Further we will concentrate our attention on the first type problems, where the temperature exchange through the interface of the physical subdomains will be neglected.

The realistic modelling of these surface coupled problems requires not only the separate solutions of the fluid and the structural dynamics parts, but also:

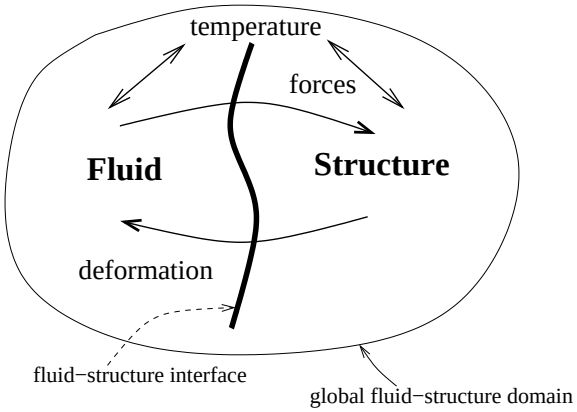



Figure 1.1: Fluid-structure coupling mechanisms

- taking into account the fluid forces within the structural dynamics subtask
- the update of the fluid domain so that it fits to its new boundaries depending on the structural displacements

Since on one hand the flow depends on the fluid-structure interface deformation and on the other the structural displacements are determined by the fluid forces at the interface, the interaction between the fluid and the structural fields is non-linear. The analytical solution of the whole coupled problem is not possible at all in nearly all cases. Therefore, the only possibility is to solve the FSI task numerically. Obviously, the numerical solution of the coupled FSI problem includes the numerical solutions of the fluid and the structural subtasks.

In the past decades the computational fluid dynamics (CFD) has developed many efficient methods for the numerical solution of various fluid dynamics problems. Thanks to this progress lots of commercial programs have been created and successfully applied to diverse complex fluid dynamics problems. Traditionally, the governing equations have been written using Eulerian (spatial) coordinates. From a numerical point of view the finite volume discretisation has been preferred because of its conservative properties.

On the other hand the computational structural dynamics (CSD) has also achieved a great advance independently from the CFD. Numbers of structural dynamics solvers have been developed to solve various structural dynamics tasks. The modelling of a wide range of material laws and structural properties has been made possible by creating special finite elements holding desired features. Contrarily to the fluid dynamics, the Lagrangian (material) coordinates have been selected for the description of the governing equations.

In many applications the structural answer to the fluid can be neglected and only the fluid dynamics part is enough to be modelled, i.e. only a fluid solver is needed. In others the fluid forces are very small compared to other external forces and hence, the

structural problem may be solved with the existing structural codes without taking into account the flow response.

However, in many processes neither the fluid forces nor the structural deformations can be neglected and special programs for the combined solution of fluid and structure dynamics problems are required as it is schematically shown in Figure 1.2.

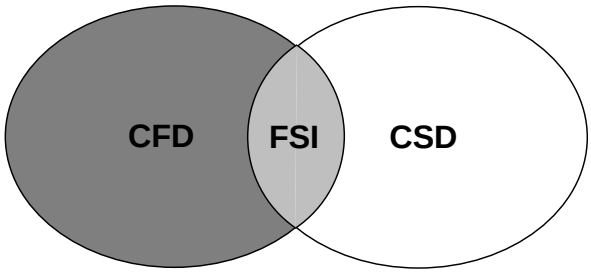



Figure 1.2: FSI problems - subset of fluid (CFD) and structural (CSD) dynamics problems

## 1.2 Methodology for solving the FSI problems

Though efficient solvers for both the fluid and the structural dynamics exist, the development of tools for modelling various fluid-structure interaction problems is still a challenge.

As it was already mentioned, to describe FSI problems certain information has to be transfered between the flow and structural fields. Based on the data exchange, the methods for solving fluid-structure interaction problems can be divided into weakly (loosely) and strongly coupled algorithms [4].

In the loose coupling methods the coupled problem is partitioned into fluid and structural parts, which are solved separately. The data exchange on the interface is done only once per time step and even not at every time-step. The partitioned analysis of coupled systems has been introduced by Park and Fellipa in [31] and further investigated in [31] and [50]. This solution approach has been applied for coupling different fluid and structural solvers [2], [42], [48]. First the fluid field at the current time-step is found. The received fluid forces are applied as boundary conditions for the structural subproblem and the structural displacements are obtained. The new structural position is considered on the next time-step. Then the fluid domain is modified to fit its new boundaries and the corresponding flow field is found. Since the new grid is taken into account at the next time-step, this staggered solution procedure can be considered as an explicit coupling method.

Contrarily to the loose coupling strategy, in a strongly coupled algorithm both parts of the FSI problem are solved simultaneously. For this purpose one system of equations is created after discretising the governing fluid and structural equations and taking into

account the boundary conditions on the interface. Hence, the whole FSI problem is solved at once using a monolithic scheme.

Obviously, both approaches have advantages and disadvantages. In Figure 1.3 they are compared with regard to their generality and stability as well as the needed programming efforts.

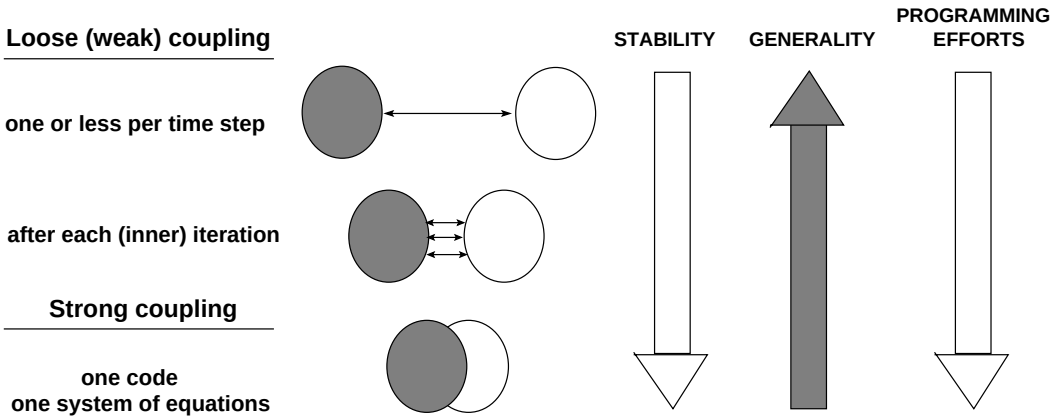



Figure 1.3: Coupling approaches for FSI

Since the fluid and the structural dynamics subproblems are based on different types of partial differential equations, the use of different numerical methods for their efficient solutions might be advantageous.

The main advantage of the loose coupling approach is that it allows already developed efficient and well validated solvers for each of the fluid and structure subtasks to be combined. Therefore, both parts of the FSI problem are solved in the best ways. Depending on the generality of the two codes, arbitrary complex flows and structures can be considered and successfully modelled. The only programming effort lies in creating suitable subroutines for information exchange between the solvers. Unfortunately, due to the explicit nature of this coupling convergence problems may arise. Consequently, there is a restriction on the choice of the time-step even if implicit time-stepping schemes are used by the two solvers.

In contrast to the loose coupling approach, the strong coupling algorithms are more difficult to create and to program. The simultaneous solution of the whole FSI problem normally requires reformulation of the systems of equations and sets restrictions on the choice of the numerical methods to be applied. Additionally, special strategies may be needed for modelling the non-linearities in each of the physical domains. This leads to a restriction on the range of tasks that a certain strong coupling algorithm is able to solve. Tremendous programming efforts are needed to create and to validate a new program applicable to various problems. However, because of the simultaneous solution of both parts of the FSI problem, there are no approximation errors and no convergence problems due to the data transfer between the fluid and structural domains.

Therefore, the strong coupling strategy is more stable but more difficult to program

than the loose coupling approach that is more general but connected with convergence problems.

The convergence properties of the loose coupling method can be easily improved by exchanging the data at the interface more than once per time-step. For this purpose at every time-step a predictor-corrector scheme can be used to find the equilibrium between the flow and the structure. Hence, this strategy will be considered as a loose coupling method of an implicit type. With regard to generality and stability it seems to be a good compromise between the loose and the strong coupling approaches. Therefore, this strategy will be preferred in the present study.

Nowadays, there are already investigations of fluid-structure interaction problems in which a simplified model of at least one of the subtasks is considered. For example in [30] slender structures are investigated. There, the governing equations for pipes conveying fluids are obtained and studied analytically, where the action of the fluid forces is considered as an added mass. Vibrating pipes conveying fluid have been studied experimentally in [19], [21]. Numerical results can be found in [44], [22], [29], [1].

Various experimental and numerical simulations [33], [34], [35], [26] study the fluid behaviour, when the movement of the structure is prescribed analytically. Other researchers [7], [25] concentrate on the fluid part, while a simple structural model for a rigid body is used. Further simplifications have been done in neglecting the dynamic effects and simulating static FSI [13], [24].

Some theoretical and numerical studies of loose coupling algorithms for one- and two-dimensional problems can be found in [37], [6]. Recently, commercial codes for combining existing solvers are also being created [23].

However, the most technical applications include fluids and structures with various material properties, complex three-dimensional geometries and dynamic interactions. The development of efficient numerical methods solving these different FSI problems is still a challenge.

## 1.3 Objectives of the research

The present research focuses on fluid-structure interaction problems in which the fluid and the structure act on each other only at the interface. Hence, the action of the fluid dynamic forces on the elastic boundaries and the deformation of the flow domain caused by the structural displacements are modelled. The fluid is assumed to be incompressible and Newtonian. The structure is made of an isotropic elastic material, where linear and geometrically non-linear models are used for small and finite deformations, respectively.

The main goal of this work is the development of an efficient coupling algorithm for solving various fluid-structure interaction problems in three-dimensional domains for arbitrary elastic structures.

**Overview**

Two loose coupling approaches from an explicit and an implicit type are developed and compared. The fluid part is solved using the finite volume code FASTEST-3D of Invent Computing GmbH. On the other hand the academic finite element program FEAP is used for the structural subproblem.

The governing equations and boundary conditions for a FSI problem are presented in section 2. First the equations for both fluid and structure are briefly pointed out. To consider the fluid domain deformation when solving the fluid dynamics, an arbitrary Lagrangian-Eulerian formulation [10] of the differential equations is used. Special attention is paid to the boundary conditions at the fluid-structure interface.

In section 3 the main properties of the chosen fluid solver and the implied numerical techniques are summarized. Since the fluid part is usually described using Eulerian (spatial) coordinates, the reuse of a fluid dynamics program requires the consideration of moving grids. Therefore, the necessary changes for modifying an Eulerian fluid code to a Lagrangian-Eulerian solver are pointed out. Further the implementation of moving grids into the finite volume code FASTEST-3D is elucidated and verified.

Section 4 shortly presents the structural solver FEAP. Since for dealing with FSI problems no changes are needed, only a few numerical methods with regard to dynamical FSI are pointed out.

The two loose coupling strategies are presented in detail in section 5, where their properties are also discussed.

In section 6 the proposed coupling methods are investigated on a few test examples. The explicit coupling approach is successfully applied to both transient and static problems with small deformations for the structural part. As an example of a dynamic FSI, the laminar flow in an elastic pipe with two pinching forces is investigated.The FSI steady state in a $90^0$ T-junction of elastic pipes conveying fluid is obtained.
The implicit coupling algorithm is used to solve dynamic FSI problems with finite deformations for the structure. As an example the laminar flow around an elastic cylinder is considered. The convergence properties and the advantages of the implicit coupling procedures over the explicit one are investigated.

Section 7 contains the numerical simulation of the three-dimensional laminar flow around an elastic thin-walled cylinder mounted in a channel with a square cross-section.

Finally, the obtained results are summarized in section 8, where some possible directions for future research are also given.

# 2 Fluid-structure interaction - governing equations

It was already pointed out in section 1.1 that a FSI problem is actually a two-field problem. Therefore, its mathematical description includes the governing equations of the fluid and the structural parts, which will be given in sections 2.1 and 2.2, respectively. The fluid-structure interaction is modelled via special boundary conditions on the interface for each subproblem which will be presented in section 2.3.

In the following investigations the fluid will be assumed to be incompressible and Newtonian. For the structure an isotropic linear elastic material law will be implied and geometrically non-linear and linear problems will be considered for finite and for small deformations, respectively.

## 2.1 Fluid dynamics

In the current research the fluid dynamics is a part of an FSI problem. Therefore, the new fluid domain boundary is the solution of a structural problem. Besides the FSI tasks, there are many mechanical problems in which the boundary of the fluid domain is moving. This movement can be given in advance - for example: piston-driven flows etc., or it may be a solution of another problem as it is the case of the fluid-structure interaction. The solution of the structural part of a coupled problem leads to a change of the fluid domain for the fluid dynamics problem. To account for the domain deformation, a consideration of moving grids within the solution of the fluid dynamics task is required.

The incompressible Newtonian fluids, which will be used in this study, are described by the Navier-Stokes equations. In order to handle the dynamics of the fluid domain, moving grids have to be considered in the Navier-Stokes equations. For this purpose Lagrangian-Eulerian (moving) coordinates are chosen.

### Momentum conservation

The momentum conservation law written in an integral form and in moving coordinates ([14]) is:

$$\frac{\partial}{\partial t}\int\limits_{V(t)} \rho \mathbf{v} dV + \int\limits_{S(t)} [\rho \mathbf{v}(\mathbf{v}-\mathbf{v}_g) - \mathbf{T}] \cdot \mathbf{n} dS = \int\limits_{V(t)} \rho \mathbf{f} dV \ , \tag{2.1}$$

where the following notations are used: V denotes an arbitrary volume with a boundary $S$, $\mathbf{n}$ is the outwards unit vector normal to the surface, $\rho$ is the fluid density, $\mathbf{v}$ is the fluid velocity vector, $\mathbf{v}_g$ is the velocity vector of the boundary of the volume and $\mathbf{f}$ is the vector of external forces. Here, $\mathbf{T}$ denotes the stress tensor for Newtonian fluids. It is defined by

$$\mathbf{T} = -\left(p + \frac{2}{3}\mu \ \mathrm{div}\mathbf{v}\right)\mathbf{I} + 2\mu\mathbf{D} \ ,$$

where $\mu$ is the fluid dynamic viscosity, $p$ is the fluid pressure, $\mathbf{I}$ is the unity matrix and

$$\mathbf{D} = \frac{1}{2}\left[\text{grad}\mathbf{v} + (\text{grad}\mathbf{v})^T\right]$$

is the rate of strain (deformation) tensor. Since for an incompressible fluid the velocity is divergence free, i.e. div$\mathbf{v} = 0$, the stress tensor simplifies to:

$$\mathbf{T} = -p\mathbf{I} + 2\mu\mathbf{D} \ . \tag{2.2}$$

A special attention deserves the convective term of equation (2.1). Here, to consider the domain movement, the convective flux through the volume surface has to be expressed by the relative velocity $\mathbf{v} - \mathbf{v}_g$. Further on fluids with a constant density in space and time will be considered.

### Mass conservation

The mass conservation law written in an integral form in moving coordinates is ([14]):

$$\frac{\partial}{\partial t}\int\limits_{V(t)} \rho dV + \int\limits_{S(t)} \rho(\mathbf{v} - \mathbf{v}_g)\cdot\mathbf{n}dS = 0 \ . \tag{2.3}$$

Again, the convective flux through the volume surface is found using the relative velocity $\mathbf{v} - \mathbf{v}_g$.

### Space conservation

Using a Lagrangian-Eulerian formulation, it turns out that an additional condition should be fulfilled for the conservativeness of the Navier-Stokes equations. This is the space (geometry) conservation law, which aim is to prevent the appearance of artificial mass sources due to the domain movement. Its importance has been recognised by many authors (for example [8], [9], [15], [45], [11] and [12]) and will be specially discussed in section 3.2.9.

The space conservation law may be obtained from the mass equation (2.3) assuming that the fluid is at a rest, i.e. that the fluid velocity is $\mathbf{v} = 0$. Since the density is assumed to be constant, the mass conservation yields

$$\frac{\partial}{\partial t}\int\limits_{V(t)} dV - \int\limits_{S(t)} \mathbf{v}_g\cdot\mathbf{n}dS = 0 \ . \tag{2.4}$$

Obviously, the control volume size is changing with time and can be different at different moments. To emphasise this time dependency in the above equations (2.1), (2.3) and (2.4) the volume and the surface are given as functions of the time, i.e. $V(t)$ and $S(t)$.

## 2.2 Structural mechanics

In this work elastic isotropic structures will be considered. The linear elastic material model is used for problems with small deformations, while the St. Venant-Kirchhoff elastic model is applied to structural dynamics tasks with finite deformations and small strains. The structural dynamics will be described using the weak form of the governing equations, which is needed for the application of finite elements methods.

### Kinematics

The structural domain $\Omega_0$ with boundary $\Gamma_0$ is considered. After a deformation its new position and boundary are denoted with $\Omega$ and $\Gamma$, respectively, and will be referred as a current configuration. For convenience the material points coordinates are $\mathbf{X}_s$ in $\Omega_0$ and $\mathbf{x}_s$ – in $\Omega$, see Figure 2.1. These coordinates are related through the displacement

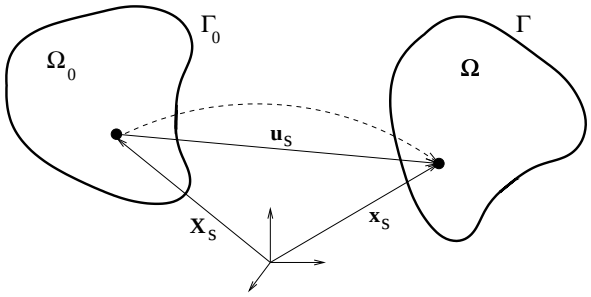



Figure 2.1: Deformation of a body $\Omega$

vector $\mathbf{u}_s$ by $\mathbf{x}_s = \mathbf{X}_s + \mathbf{u}_s$. Important deformation measures are the deformation gradient $\mathbf{F}$ defined by

$$\mathbf{F} = \text{Grad } \mathbf{x}_s = \mathbf{I} + \text{Grad } \mathbf{u}_s \tag{2.5}$$

and the Green-Lagrange strain tensor $\mathbf{E}$ given by

$$\mathbf{E} = \frac{1}{2}\left(\text{Grad } \mathbf{u}_s + \text{Grad}^T \mathbf{u}_s + \text{Grad } \mathbf{u}_s \text{ Grad}^T \mathbf{u}_s\right) \ . \tag{2.6}$$

If small deformations are considered, the last term in equation (2.6) can be neglected and the strain tensor simplified to the linear elasticity strain tensor $\epsilon$:

$$\epsilon = \frac{1}{2}\left(\text{Grad } \mathbf{u}_s + \text{Grad}^T \mathbf{u}_s\right) \ . \tag{2.7}$$

### Balance of momentum

The balance of the momentum includes the balance of the linear and the angular momentums. The balance of the linear momentum in the current configuration $\Omega$ reads:

$$\rho_s \dot{\mathbf{v}}_s = \text{div } \sigma + \rho_s \mathbf{b}_v \ , \tag{2.8}$$

where $\rho_s$ is the density, $\mathbf{v}_s = \dot{\mathbf{x}}_s$ is the structural velocity, $\mathbf{b}_v$ are the volume forces and $\sigma$ is the Cauchy stress tensor. Equation (2.8) represents the local equilibrium in the current configuration.

The balance of the angular moment gives the symmetry of the Cauchy stress tensor:

$$\sigma = \sigma^T \ . \tag{2.9}$$

In order to obtain the momentum conservation in the initial configuration (Lagrangian formulation), the first Piola-Kirchhoff stress tensor $\mathbf{P}$ is defined by

$$\mathbf{P} = \det\mathbf{F}\ \sigma\mathbf{F}^{-T} \ , \tag{2.10}$$

where $\det\mathbf{F}$ is the Jacobi determinant of $\mathbf{F}$. Therefore, the equilibrium in the initial configuration $\Omega_0$ reads:

$$\rho_{s0}\dot{\mathbf{v}}_s = \mathrm{Div}\ \mathbf{P} + \rho_{s0}\mathbf{b}_v \ , \tag{2.11}$$

where $\rho_{s0} = \det\mathbf{F}\ \rho_s$ is the density in the initial configuration $\Omega_0$.

Furthermore the balance of the angular momentum yields

$$\mathbf{P}\mathbf{F}^T = \mathbf{F}\mathbf{P}^T \ , \tag{2.12}$$

i.e. the unsymmetry of the tensor $\mathbf{P}$. Additionally, the second Piola-Kirchhoff stress tensor $\mathbf{S}$ is defined by

$$\mathbf{S} = \mathbf{F}^{-1}\mathbf{P} \ . \tag{2.13}$$

Then the balance of the angular momentum gives the symmetry of $\mathbf{S}$, i.e

$$\mathbf{S} = \mathbf{S}^T \ . \tag{2.14}$$

**Constitutive equations**

The material properties of the structures are described by the constitutive equations. These equations establish the relation between the strains $\mathbf{E}$ and the stresses $\mathbf{S}$. Here, the isotropic elastic material model of St. Venant–Kirchhoff will be considered. It is suitable for problems with large deformations and small strains. The St. Venant–Kirchhoff model assumes that a scalar function $\mathrm{W}(\mathbf{E})$ exists. $\mathrm{W}(\mathbf{E})$ is called a strain energy function and is such that the stresses are obtained by

$$\mathbf{S} = \frac{\partial W(\mathbf{E})}{\partial \mathbf{E}} \ . \tag{2.15}$$

Since the strains remain small, the strain energy function is presented by:

$$W(\mathbf{E}) = \frac{1}{2}\mathbf{E}^T\mathbf{D}_s\mathbf{E} \ , \tag{2.16}$$

where $\mathbf{D}_s$ is the elasticity matrix containing the appropriate material properties. For isotropic elastic material it is determined by the Young modulus $E$ and the Poisson

ratio $\nu$ of the material [52], [47]. From relation (2.15) it follows that the stress tensor $\mathbf{S}$ is represented by

$$\mathbf{S} = \mathbf{D}_s \mathbf{E} \ , \tag{2.17}$$

where the Green-Lagrange strain tensor $\mathbf{E}$ is defined by equation (2.6).

The St. Venant–Kirchhoff model gives identical results to the small deformation isotropic model when the displacements are truly infinitesimal. If small deformations are considered, the difference between the initial $\Omega_0$ and current $\Omega$ configurations can be neglected. Then using the linear elastic tensor $\epsilon$, equation (2.17) can be written as

$$\sigma = \mathbf{D}_s \epsilon \ , \tag{2.18}$$

which is the constitutive equation for linear elasticity.

### Variational description

As it was already mentioned in section 1.1 the finite elements methods are widely used in the structural mechanics. Therefore, the weak form of the equations is preferred. It can be obtained using variational methods. The elastic materials, that will be considered in this research, are such that a strain energy function W($\mathbf{E}$) exists and equation (2.15) holds. The principle for the minimum of the total potential $\Pi(\mathbf{u}_s)$ leads to the minimisation of the total potential:

$$\Pi(\mathbf{u}_s) = \int\limits_{\Omega_0} W(\mathbf{E}) dV - \int\limits_{\Omega_0} \rho_{s0}(\mathbf{b}_v - \dot{\mathbf{v}}_s) \cdot \mathbf{u}_s dV - \int\limits_{\Gamma_0} \mathbf{t} \cdot \mathbf{u}_s dS \longrightarrow \min \ . \tag{2.19}$$

The last two terms represent the potential energy of the volume and surfaces forces. To describe dynamic problems, the inertial forces are introduced through the body forces, i.e. the volume force $\mathbf{b}_v$ is replaced by $\mathbf{b}_v - \dot{\mathbf{v}}_s$.

The minimisation of (2.19) requires the solution of the variational problem

$$\delta\Pi = 0 \ , \tag{2.20}$$

where $\delta\Pi$ denotes the variation of the functional $\Pi$. To find $\delta\Pi$, the variation of the strain energy function $\delta W$ is considered at first. It holds that

$$\delta W(\mathbf{E}) = \frac{\partial W(\mathbf{E})}{\partial \mathbf{E}} \ \delta \mathbf{E} \ , \tag{2.21}$$

where $\delta\mathbf{E}$ is the variation of the strain $\mathbf{E}$. Taking into account equation (2.15), it follows that

$$\delta W(\mathbf{E}) = \mathbf{S} \ \delta \mathbf{E} \ . \tag{2.22}$$

If the variation of the displacement vector $\mathbf{u}_s$ is denoted with $\delta\mathbf{u}_s$, the equation to be solved is:

$$\delta\Pi(\mathbf{u}_s) = \int\limits_{\Omega_0} \mathbf{S} \ \delta \mathbf{E} \ dV - \int\limits_{\Omega_0} \rho_{s0}(\mathbf{b}_v - \dot{\mathbf{v}}_s) \cdot \delta\mathbf{u}_s dV - \int\limits_{\Gamma_0} \mathbf{t} \cdot \delta\mathbf{u}_s dS = 0 \ . \tag{2.23}$$

## 2.3 Fluid-Structure Interface

To complete the governing equations for the fluid-structure problems presented in sections 2.1 and 2.2, it remains to model the interaction between the flow and the structure, i.e. to set corresponding boundary conditions on the fluid-structure interface.

### Fluid part

In the FSI problem the fluid domain movement is prescribed by the structural displacements. Let us denote the time-dependent coordinate vectors of the fluid and structural grid points with $\mathbf{x}_f(t)$ and $\mathbf{x}_s(t)$, respectively. Therefore, the flow domain position should be such that the condition

$$\mathbf{x}_f(t) = \mathbf{x}_s(t) \tag{2.24}$$

is fulfilled on the interface boundary.

The boundary flow velocity $\mathbf{v}_g$ should be also the same as the structural velocity $\mathbf{v}_s$ on the interface, i.e.

$$\mathbf{v}_g(t) = \mathbf{v}_s(t) \tag{2.25}$$

Additionally, non-slip boundary conditions are applied on the moving walls. For viscous fluids they are simply:

$$\mathbf{v}(t) = \mathbf{v}_g(t) \ . \tag{2.26}$$

Equation (2.26) guarantees that the convective fluxes through the fluid-structure interface are zeros.

### Structural part

To consider the action of the flow on the structure, the fluid forces have to be taken into account when solving the structural subproblem. Due to relation (2.26) the fluid forces on the interface $S$ are:

$$\mathbf{F}_{fluid} = \int\limits_S \left[\rho \mathbf{v}(\mathbf{v} - \mathbf{v}_g) - \mathbf{T}\right] \cdot \mathbf{n} dS = - \int\limits_S \mathbf{T} \cdot \mathbf{n} dS \ .$$

Using definition (2.2) of the stress tensor $\mathbf{T}$, the relation

$$\mathbf{F}_{fluid} = \underbrace{\int\limits_S p\mathbf{I} \cdot \mathbf{n} dS}_{pressure} - \underbrace{\int\limits_S \mu \left[\mathrm{grad}\mathbf{v} + (\mathrm{grad}\mathbf{v})^T\right] \cdot \mathbf{n} dS}_{shear\ stress} \tag{2.27}$$

is obtained.

Therefore, there are two fluid forces acting on the surface of the structure - the pressure and the shear forces.

Considering structures (for example elastic pipes) conveying fluid or surrounded by it, additional external forces may also exist. Hence, all forces acting on the structure within the fluid-structure interaction have to be taken into account. So, the total force $\mathbf{F}_{struc}$ is:

$$\mathbf{F}_{struc} = \mathbf{F}_{struc}^{external} + \mathbf{F}_{fluid} \ , \tag{2.28}$$

where $\mathbf{F}_{fluid}$ is determined by (2.27) and $\mathbf{F}_{struc}^{external}$ denotes all other applied external forces.

Therefore, the system of equations necessary for the complete description of a fluid-structure interaction problem consists of equations (2.1), (2.3), (2.4) and (2.23) (or respectively its equivalent formulation for small deformations) completed with corresponding boundary conditions. Additionally, the special conditions (2.24), (2.25) and (2.28) have to be satisfied at the fluid-structure interface. The equilibrium between the fluid and the structure have to be also found. For this purpose information has to be exchanged between the two subproblems.

In order to present the solution algorithm for the complete fluid-structure interaction problem in section 5, the solution methods for the flow and for the structural dynamics governing equations will be given in sections 3 and 4, respectively.

# 3 Fluid dynamics subproblem - numerical solution

In section 1.2 it was mentioned that the use of available programs may be advantageous for the simulation of the fluid dynamics problem. Since these commercial solvers have been widely used and tested on various problems, they are well verified. Therefore, for the purposes of the current research the fluid dynamics code FASTEST-3D ([18]) is used.

In the following section 3.1 the main properties of this solver are summarised. Section 3.2 contains the discretisation of the fluid equations and the changes required in the code FASTEST-3D for modelling a flow described in Lagrangian-Eulerian (moving) coordinates. The used boundary conditions are considered in section 3.3. The possibilities for the fluid grid update are given in section 3.4. Consequently, the main solution methods and the changes in them needed for moving grids are presented in section 3.5. Finally, the implementation of moving grids in FASTEST-3D is verified in section 3.6.

## 3.1 The finite volume code FASTEST-3D

To model the fluid subproblem the code FASTEST-3D from INVENT Computing GmbH has been used. This solver uses the conservative form of the Navier-Stokes equations written in Eulerian coordinates, i.e. it solves the equations (2.1) and (2.3), where the grid is fixed and $\mathbf{v}_g = 0$. The computational domain is discretised into finite volumes so that the obtained grid is block-structured and boundary-fitted. For the velocity and pressure variables a collocated arrangement is chosen.
Some of the numerical methods used by FASTEST are:

- pressure-correction approach of SIMPLE type for solving the coupled non-linear system of equations for the velocity and the pressure
- strongly implicit incomplete lower-upper decomposition (ILU) method of Stone for solving the linear system of equations (received after discretisation)
- non-linear multigrid scheme for reducing the computational time
- grid partitioning for parallelisation for computational acceleration
- upwind (UDS) and central (CDS) differencing schemes for achieving second-order spatial accuracy
- fully implicit first- and second-order time discretisations

In the following section 3.2 the discretisation of the fluid governing equations and the used numerical methods will be presented in detail. Moreover, the extra terms containing the grid velocity $\mathbf{v}_g$ will be specially considered.

## 3.2 Discretisation - Moving grids

To model the flow in a moving domain, the Lagrangian-Eulerian formulation [10] of the governing equations is required. Since the fluid dynamics in FASTEST-3D is described using Eulerian coordinates, the solver cannot be used unless moving grids are implemented in it.

Obviously, the main difference between the fluid dynamics equations in Eulerian and those in Lagrangian-Eulerian coordinates is that in the latter the boundary velocity is not zero anymore. In the following the governing equations (2.1) and (2.3) will be discretised, where a special attention will be paid to the terms containing the boundary velocity $\mathbf{v}_g$ and to the space conservation law (2.4). Actually, these are the points that should be implemented in the solver FASTEST-3D to model fluid dynamics problems described by the system of partial differential equations (2.1), (2.3) and (2.4).

Before discretising the fluid governing equations, they will be written in more suitable forms. Let us denote the components of the fluid velocity $\mathbf{v} = (v_1, v_2, v_3)$, the components of the body force vector $\mathbf{f} = (f_1, f_2, f_3)$ and the vector-line components of the tensor $\mathbf{T} = (\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3)$. The momentum conservation law (2.1) is considered at first. For each velocity component it reads:

$$\frac{\partial}{\partial t}\int\limits_{V(t)} \rho v_i dV + \int\limits_{S} \rho v_i(\mathbf{v}-\mathbf{v}_g)\cdot\mathbf{n}dS - \int\limits_{S} \mathbf{t}_i\cdot\mathbf{n}dS = \int\limits_{V} \rho f_i \, dV \qquad \text{for} \quad i=1,2,3. \tag{3.1}$$

Referring to equation (2.2)

$$\mathbf{t}_i = \mathbf{T}\mathbf{e}_i = -p\mathbf{e}_i + \mu\left[\mathrm{grad}v_i + (\mathrm{grad}\mathbf{v})^T\mathbf{e}_i\right] \qquad \text{for} \quad i=1,2,3$$

where $\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{e}_3$ denote the coordinate unit vectors.

Therefore, the momentum conservation law for each velocity component $\phi = v_i$ for $i = 1, 2, 3$ can be generalised as

$$\underbrace{\frac{\partial}{\partial t}\int\limits_{V(t)} \rho\phi dV}_{rate\ of\ change} + \underbrace{\int\limits_{S} \rho\phi(\mathbf{v}-\mathbf{v}_g)\cdot\mathbf{n}dS}_{convection} - \underbrace{\int\limits_{S} \mu\ \mathrm{grad}\phi\cdot\mathbf{n}dS}_{diffusion} = RHS_\phi\ , \tag{3.2}$$

where the right hand side $RHS_\phi$ for $\phi = v_i$ and $i = 1, 2, 3$ of the momentum equation is

$$RHS_{v_i} = \int\limits_{S}\left[-p\mathbf{e}_i + \mu(\mathrm{grad}\mathbf{v})^T\mathbf{e}_i\right]\cdot\mathbf{n}dS + \int\limits_{V}\rho f_i dV. \tag{3.3}$$

Obviously, the mass conservation equation (2.3) can be also represented in the form (3.2) for $\phi \equiv 1$ and a zero diffusion term as well as $RHS \equiv 0$.

To solve these fluid governing equations numerically, a finite volume discretisation method will be applied. For this purpose the fluid domain should be discretised into a finite volume computational grid.

### 3.2.1 Fluid domain discretisation

A block-structured grid is created for the discretisation of the fluid domain. Here, the computational domain is divided into a limited number of non-overlapping blocks. A structured grid is used for every block. For convenience some notations are adopted and shown in Figure 3.1. The center of an arbitrary control volume (CV) is denoted with P. Its West, East, North, South, Top and Bottom faces are named respectively $S_w$, $S_e$, $S_n$, $S_s$, $S_t$ and $S_b$ and the center points on them - w, e, n, s, t and b. Correspondingly, the centers of the West, East, North, South, Top and Bottom adjacent control volumes are denoted with W, E, N, S, T and B.

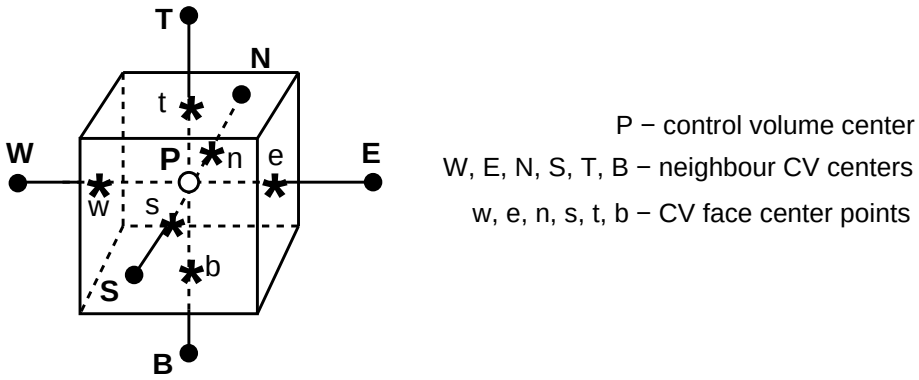



Figure 3.1: Control volume discretisation - notations.

A collocated arrangement of the unknown variables $\mathbf{v}$ and $p$ is chosen, i.e. their values are searched at all control volumes centers (named $P$ in Figure 3.1). The boundary nodes required for the specification of boundary conditions are placed at the centers of the boundary control volume faces.

However, since the fluid domain boundaries are moving, the computational domain will change its form and position. Respectively, the control volumes will move to new positions. An example of an arbitrary control volume movement is presented in Figure 3.2.

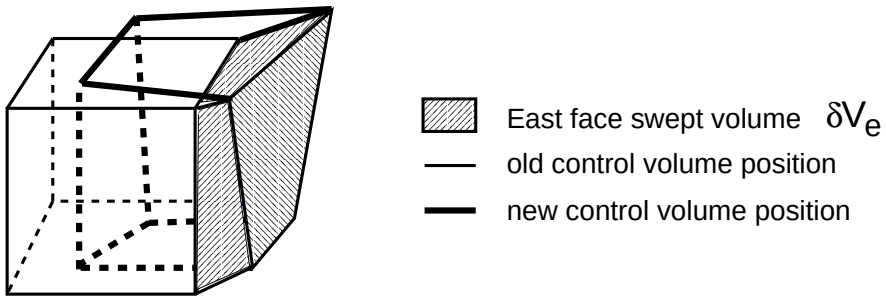



Figure 3.2: Control volume movement - old and new positions.

Due to this motion the surface $S_j$ of every control volume "sweeps" a certain volume, which will be called a "swept" volume and denoted as $\delta V_j$ for every $j \in \{e, w, n, s, t, b\}$. In Figure 3.2 the swept volume $\delta V_e$ by the East face $S_e$ of the control volume is depicted (the volume with the dashed surface).

### 3.2.2 CV volume computation

In this work complex three-dimensional fluid domains are considered, that are deformed to an arbitrary shape. Moreover, the CVs into which they are discretised may deform and take various forms. Therefore, special attention is required to the control volumes and swept volumes computation. The simplest approach for finding the volume of an arbitrary CV is its presentation as a sum of five tetrahedrons [14] as it is shown in Figure 3.3.

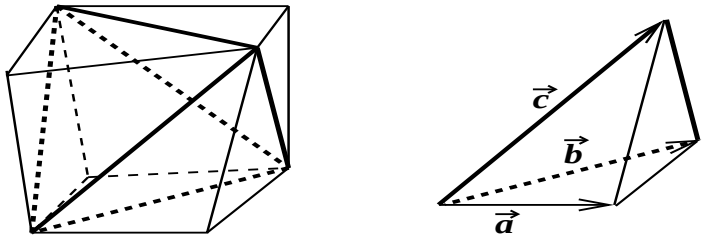



Figure 3.3: Arbitrary CV divided into five tetrahedrons

The volume of each tetrahedron can be exactly computed using the CV vertices coordinates. For example the volume $V$ of the tetrahedron depicted in Figure 3.3 (right) is simply

$$V = \frac{1}{6}\, \vec{a}\ \vec{b}\ \vec{c}\,,$$

where $\vec{a}$, $\vec{b}$ and $\vec{c}$ denote three edge-vectors with a common starting point. Analogically, the swept volumes can be evaluated using the old and the new positions of the vertices.

### 3.2.3 Rate of change

The Euler backwards discretisation scheme for the time discretisation and the midpoint rule for the spatial approximation of the rate of change terms at time $t_n$ in equation (3.2) are used. The discretised form is given by

$$\frac{\partial}{\partial t} \int\limits_{V(t)} \rho\phi dV \approx \frac{(\rho\phi V)_P^n - (\rho\phi V)_P^{n-1}}{\Delta t}\ . \tag{3.4}$$

Here, the subscript $P$ denotes that the variables are taken at the center of each control volume $V(t)$ and $\Delta t$ is the time step. This approximation is first-order accurate in

time and second-order accurate in space. It uses the values at the two time levels $n$ and $n-1$.

Another discretisation scheme is the three level scheme

$$\frac{\partial}{\partial t}\int\limits_{V(t)} \rho\phi dV \approx \frac{3(\rho\phi V)_P^n - 4(\rho\phi V)_P^{n-1} + (\rho\phi V)_P^{n-2}}{2\Delta t} \ , \tag{3.5}$$

which uses the values at the three time levels $n$, $n-1$ and $n-2$. It is second-order accurate both in time and space.

### 3.2.4 Convective terms

In order to show how the convective fluxes through the CV faces in equation (3.2) are approximated, we will consider in detail the convective flux $C_e$ through the East face:

$$C_e = \int\limits_{S_e} \rho\phi(\mathbf{v}-\mathbf{v}_g)\cdot\mathbf{n}dS_e \ .$$

First the approximation $F_e$ of the mass flux $\int\limits_{S_e} \rho(\mathbf{v}-\mathbf{v}_g)\cdot\mathbf{n}dS_e$ through the East face $S_e$ in equation (3.2) for $\phi \equiv 1$ will be found. The application of the midpoint rule gives

$$\int\limits_{S_e} \rho(\mathbf{v}-\mathbf{v}_g)\cdot\mathbf{n}dS_e \approx \rho_e(\mathbf{v}\cdot\mathbf{n})_e\delta S_e - \underbrace{\rho_e(\mathbf{v}_g\cdot\mathbf{n})_e\delta S_e}_{grid\ \ flux\ \ F_{g,e}} = \rho_e(\mathbf{v}\cdot\mathbf{n})_e\delta S_e - F_{g,e} = F_e \ , \tag{3.6}$$

where $\delta S_e$ denotes the surface of the East face $S_e$ and the underlined term $F_{g,e}$ is the necessary correction for moving grids. Since it represents the flux due to the grid movement, it will be called "grid flux". Its computation will be considered in detail in section 3.2.9. Therefore, using the midpoint rule the convective flux through the East face is approximated with

$$C_e \approx \phi_e\rho_e\left[(\mathbf{v}-\mathbf{v}_g)\cdot\mathbf{n}\right]_e\delta S_e = F_e\phi_e \ , \tag{3.7}$$

where $F_e$ is calculated with equation (3.6). The variables at the center $e$ of the control volume face $S_e$ are approximated with their values at the neighbour CV centers using central (CDS) or upwind (UDS) differencing schemes. The CDS is based on a linear interpolation given by

$$(\phi_e)^{CDS} = \left(1 - \frac{\overline{Pe}}{\overline{Pe}+\overline{eE}}\right)\phi_P + \frac{\overline{Pe}}{\overline{Pe}+\overline{eE}}\ \phi_E \ . \tag{3.8}$$

Here, $\overline{Pe}$ denotes the distance between the points $P$ and $e$. The convective flux discretisation using UDS is

$$(F_e\phi_e)^{UDS} = \max(F_e,0)\phi_P + \min(F_e,0)\phi_E \ , \tag{3.9}$$

where $F_e$ is the mass flux through the East face of the CV with a center $P$.

The CDS is second-order accurate. Unfortunately, it may lead to unphysical oscillations, when the convective fluxes are dominating. On the other hand the UDS is unconditionally stable, but only first-order accurate in space. To use the advantages of the two schemes, a combination of them is used as given by

$$F_e\phi_e = (F_e\phi_e)^{UDS} + \gamma_\phi \left[(F_e\phi_e)^{CDS} - (F_e\phi_e)^{UDS}\right]^{old} , \tag{3.10}$$

where $\gamma_\phi$ is a blending parameter $0 \leq \gamma_\phi \leq 1$. Obviously, $\gamma_\phi = 1$ gives the pure CDS, which has a second-order accuracy. Here, for additional stability the second term is computed explicitly and may be added to the right hand side of equation (3.2). The fact that the variables in this term are taken at the previous time step (iteration) is denoted by the superscript *old*.

The convective fluxes through the other CV boundaries are approximated analogically.

### 3.2.5 Diffusive terms

The discretisation of the diffusive fluxes in equation (3.2) will be demonstrated for the diffusive flux $D_e$ through the East face of an arbitrary control volume:

$$D_e = -\int_{S_e} \mu \ \mathrm{grad}\phi \cdot \mathbf{n} dS_e \ .$$

Since the diffusive fluxes are not affected by the grid movement, no special care has to be taken. Again, the midpoint rule gives

$$D_e \approx -\mu_e \left(\mathrm{grad}\phi \cdot \mathbf{n}\right)_e \ \delta S_e$$

Further the diffusive flux can be approximated using the following deferred correction[14]

$$D_e \approx -\mu_e \left(\frac{\partial\phi}{\partial\xi}\right)_e \delta S_e - \left(\mu_e \left[\left(\frac{\partial\phi}{\partial n}\right)_e - \left(\frac{\partial\phi}{\partial\xi}\right)_e\right] \delta S_e\right)^{old} , \tag{3.11}$$

where $\xi$ denotes the local coordinate along the line through the nodes P and E. If the grid is orthogonal, the line through the nodes P and E will be parallel to the vector $\mathbf{n}$ normal to the face $S_e$ and the second term in equation (3.11) is zero. The second term represents the deferred correction. If the grid non-orthogonalities are not severe, it is small compared to the first term. Therefore, it is treated explicitly (denoted by the superscript *old*) and is added to the right hand side of equations (3.2). The first term is implicitly approximated by

$$\mu_e \left(\frac{\partial\phi}{\partial\xi}\right)_e \delta S_e \approx \mu_e \frac{(\delta S_e)^2}{V_e} \left(\phi_E - \phi_P\right) , \tag{3.12}$$

where $V_e$ is the volume defined by the vectors $\overrightarrow{PE}$ and the vector $(\mathbf{n}\ \delta S_e)$ normal to the East face $S_e$, i.e. $V_e = \overrightarrow{PE} \cdot \mathbf{n}\delta S_e$.

The other diffusive fluxes are treated analogically.

### 3.2.6 Right hand side ($RHS$)

The midpoint rule is also used to discretise the right hand side $RHS_{v_i}$ of equation (3.2) for each velocity component $v_i$. Therefore,

$$RHS_{v_i} = \int_V \rho f_i dV + \int_S \left[-p\mathbf{e}_i + \mu(\text{grad}\mathbf{v})^T\mathbf{e}_i\right] \cdot \mathbf{n} dS \approx$$
$$\rho_P f_{iP} V + \sum_j \left\{ \left[-p\mathbf{e}_i + \mu(\text{grad}\mathbf{v})^T\mathbf{e}_i\right] \cdot \mathbf{n} \right\}_j \delta S_j \ , \tag{3.13}$$

where $V$ is the volume of the considered CV.

Since $RHS_{v_i}$ will be treated explicitly regardless of the used time-stepping technique, all variables are taken at the old time step $t_{n-1}$. Again, the CDS is used to approximate the values at the boundary face centers $j \in \{e, w, t, b, n, s\}$.

For convenience the discretised right hand sides of the equations will be denoted with $R_\phi$ and will include all explicitly evaluated terms, i.e. $RHS_\phi$ and the explicit treated terms from equations (3.10) and (3.11).

### 3.2.7 Time-stepping schemes

In order to achieve unconditional stability only implicit methods are chosen to advance in time. The used time-stepping schemes for the momentum equation (3.2) can be generally written in the form:

$$\frac{1}{\Delta t}\left[(1+\beta)(\rho\phi V)_P^n - (1+2\beta)(\rho\phi V)_P^{n-1} + \beta(\rho\phi V)_P^{n-2}\right] +$$
$$\alpha_\phi \sum_j \left(F_j\phi_j + D_j\right)^n + (1-\alpha_\phi)\sum_j \left(F_j\phi_j + D_j\right)^{n-1} = RHS_\phi^{n-1} \tag{3.14}$$

and for the SCL equation (2.4):

$$\frac{1}{\Delta t}\left[(1+\beta)V_P^n - (1+2\beta)V_P^{n-1} + \beta V_P^{n-2}\right] =$$
$$\alpha_m \sum_j (\mathbf{v}_g \cdot \mathbf{n})_j^n \ \delta S_j^n + (1-\alpha_m)\sum_j (\mathbf{v}_g \cdot \mathbf{n})_j^{n-1} \ \delta S_j^{n-1} \tag{3.15}$$

where as usual $j \in \{e, w, t, b, s, n\}$. Here $\alpha_\phi$ and $\alpha_m$ are blending parameters between the explicit and implicit treated parts of the spatial discretisations of the convective and diffusive fluxes $0 < \alpha_\phi \leq 1$ and $0 < \alpha_m \leq 1$. The parameter $\beta$ blends the first- and second- order time-discretisations for the rate of change term. Because of the used second-order spatial discretisations for each term of the equations, all these methods are second-order accurate in spatial discretisation.

In regard to the time accuracy the following schemes will be considered:

- $\beta = 0$, $\alpha_\phi = 1$ and $\alpha_m = 1$ - implicit Euler method
  It is unconditionally stable and first-order accurate in time discretisation. Since this method allows large time-steps, it is especially suitable for solving steady-state problems.

- $\beta = 0$, $\alpha_\phi = 0.5$ and $\alpha_m = 0.5$ - Crank-Nicolson method
  It is also unconditionally stable, however, oscillatory solutions may appear for very large time-steps. The time-step restriction is problem-dependent. Although, the method requires slightly more computational efforts than the implicit Euler method, it gives a second-order accuracy in time-discretisation.

- $\beta = 0.5$, $\alpha_\phi = 1$ and $\alpha_m = 1$ - implicit second-order Euler or three-time-level method. This approximation is unconditionally stable and second-order accurate in time. It needs a computational effort similar to the implicit first-order Euler method, however, a lot of additional memory to store the variables at time-level (n-2).

Therefore, if second-order in time-discretisation is desired, the Crank-Nicolson approach will be preferred in the following investigations.

### 3.2.8 Mass conservation

It is interesting to point out that despite the time-derivative term in the mass conservation law (2.3), no time discretisation is required. Taking into account the SCL (2.4) the time derivative term in the mass conservation equation disappears:

$$\underbrace{\frac{\partial}{\partial t}\int\limits_{V(t)} \rho dV - \int\limits_{S} \rho \mathbf{v}_g \cdot \mathbf{n} dS}_{0} + \int\limits_{S} \rho \mathbf{v} \cdot \mathbf{n} dS = 0 \ . \tag{3.16}$$

Therefore, the mass equation actually gives that the sum of the global mass fluxes for every CV as well as for the whole computational domain should be zero at every time-step. Let us remember that for incompressible fluids with a constant density described in Eulerian coordinates exactly the same relation (3.16) holds.
Further depending on the type of the domain boundaries, the last equality can be written for the whole computational domain as:

$$\int\limits_{S_{inlet}} \rho \mathbf{v} \cdot \mathbf{n} dS + \int\limits_{S_{outlet}} \rho \mathbf{v} \cdot \mathbf{n} dS + \int\limits_{S_{wall}} \rho \mathbf{v} \cdot \mathbf{n} dS = 0 \ ,$$

where $S_{inlet}$, $S_{outlet}$ and $S_{wall}$ are the inlet, outlet and wall domain boundaries, respectively.

Since non-slip conditions are applied on the walls, $\mathbf{v}_{wall} = \mathbf{v}_g$. The total inlet mass flux $F_{inlet}^{total}$ is prescribed by the problem. Therefore, to have global mass conservation

the total outlet mass flux $F_{outlet}^{total}$ should satisfy

$$F_{outlet}^{total} = F_{inlet}^{total} - \sum_{walls} F_g \ , \tag{3.17}$$

where the sum of the grid fluxes $F_g$ is over all wall faces.

### 3.2.9 Space conservation law (SCL) and swept volumes

Special attention has to be paid to the discretisation of the space conservation law given by equation (2.4). The importance of this equation was shown by Demirdžić and Perić [8], where they demonstrated that the violence of the SCL produces errors in the form of artificial mass sources. These errors can become negligible if a small enough time-step is taken. However, this would lead to unnecessarily big computational time especially in the case of large deformations of the domain.

For the spatial discretisation of equation (2.4) the second-order central difference scheme based on the midpoint rule is used. Therefore, since the surface of every control volume is $S = \bigcup S_j$ for $j \in \{e, w, n, s, t, b\}$ the total grid flux through the CV boundary is the sum of all boundary grid fluxes, i.e

$$\int_S \mathbf{v}_g \cdot \mathbf{n} \ dS = \sum_j \int_{S_j} \mathbf{v}_g \cdot \mathbf{n} \ dS_j \approx \sum_j (\mathbf{v}_g \cdot \mathbf{n})_j \ \delta S_j = \sum_j F_{g,j} \tag{3.18}$$

where $\delta S_j$ denotes the area of the surface $S_j$ and $j \in \{e, w, t, b, n, s\}$.

The grid velocities $\mathbf{v}_g$ at the centers of the control volume faces may be computed using the old and new grid points coordinates. However, in order to guarantee the SCL they should be approximated so that the discretised SCL equation (3.15) holds. Otherwise, artificial mass fluxes in the mass equation appear and lead to instability and wrong results. According to equation (3.6) not the grid velocity itself but the grid flux is actually needed for the convective flux computation. Here, an approximation for the grid fluxes will be derived so that it is consistent with the SCL discretisation.

Let us notice that at every time-step $n$ the volume change of an arbitrary CV with center $P$ is always equal to the sum of the volumes swept by its boundary faces, i.e.

$$V_P^n - V_P^{n-1} = \sum_j \delta V_j^n \ . \tag{3.19}$$

Taking equality (3.19) into account and leaving on the left side only the sum of the grid fluxes at the new time-step $n$, the discretised SCL equation (3.15) may be written as:

$$\sum_j (\mathbf{v}_g \cdot \mathbf{n})_j^n \ \delta S_j^n =$$

$$\frac{1}{\alpha_m} \left[ \sum_j \left[ (1+\beta) \frac{\delta V_j^n}{\Delta t} - \beta \frac{\delta V_j^{n-1}}{\Delta t} \right] - (1-\alpha_m) \sum_j (\mathbf{v}_g \cdot \mathbf{n})_j^{n-1} \ \delta S_j^{n-1} \right] \tag{3.20}$$

Therefore, when the fluid density is constant, the SCL will be obviously satisfied if the grid fluxes are computed with the help of the respective swept volumes

$$F_{g,j}^{n} = \frac{1}{\alpha_m}\left[(1+\beta)\,\frac{\rho_j \delta V_j^n}{\Delta t} - \beta\,\frac{\rho_j \delta V_j^{n-1}}{\Delta t}\right] - \frac{(1-\alpha_m)}{\alpha_m}\,F_{g,j}^{n-1} \tag{3.21}$$

This flux representation is especially advantageous for three-dimensional and complex geometries. The swept volumes can be easily and exactly computed knowing the old and the new grid positions, while the grid velocities at the CV face centers may be only approximated. Hence, the grid fluxes will be evaluated using equality (3.21). In this way no approximation of the grid velocities at the CV face centers is necessary. If the implicit first-order Euler method is chosen for the time discretisation, i.e. $\beta = 0$ and $\alpha_m = 1$, equation (3.21) simplifies to

$$F_{g,j}^{n} = \rho_j \frac{\delta V_j^n}{\Delta t}\ . \tag{3.22}$$

Then only the swept volumes at time-step $n$ are needed.
For the three-time-level discretisation scheme with $\beta = 0.5$ and $\alpha_m = 0$, which is second-order accurate in time-step, equality (3.21) results in

$$F_{g,j}^{n} = \rho_j (\mathbf{v}_g \cdot \mathbf{n})_j\ \delta S_j = \rho_j \frac{3\delta V_j^n - \delta V_j^{n-1}}{2\Delta t}\ . \tag{3.23}$$

Again, the grid velocities in the grid fluxes at time-step $n$ can be replaced by the swept volumes at the two previous time-steps $n-1$ and $n-2$.

On the other hand for the Crank-Nicolson discretisation ($\beta = 0$ and $\alpha_m = 0.5$) relation (3.21) leads to a recursive connection between the new and the old grid fluxes:

$$F_{g,j}^{n} = 2\rho_j \frac{\delta V_j^n}{\Delta t} - F_{g,j}^{n-1}\ . \tag{3.24}$$

In this case initial conditions for the grid fluxes are required.

The main advantage of the presented grid fluxes computations is that no approximations of the grid velocities at the CV faces are needed. The grid fluxes are evaluated using only the swept volumes. Since the grid points coordinates are known at each time-step, the swept volumes can be easily computed using the approach described in section 3.2.1. Therefore, no approximation errors are introduced by the grid fluxes computations.

### 3.2.10 Discretised governing equations

Knowing the approximations for the convective and diffusive parts as well as the right hand side of the momentum equation (3.2), here, its final discretised form will be given. Since some of the variables are needed at the CV centers and others at the CV faces, the following indices are used: $P$ is the center of the CV, $nb \in \{E, W, T, B, S, N\}$ are

the centers of the neighbour CVs and $j \in \{e, w, t, b, s, n\}$ are the centers of the CV boundaries. Therefore, the discretised momentum equation (3.14) can be written in the general form:

$$a_P \phi_P = \sum_{nb} a_{nb} \phi_{nb} + R_P \ , \tag{3.25}$$

where $R_P$ denotes the sum of all terms treated explicitly and hence, not depending on the new time-step values $\phi$.
Taking into account equations (3.9) and (3.12) it follows that:

$$a_P = (1+\beta)\frac{\rho_P V_P^n}{\Delta t} + \alpha_\phi \left[ \sum_j \max(F_j^n, 0) + \sum_j \mu_j \frac{(\delta S_j)^2}{V_j} \right] \tag{3.26}$$

and

$$a_{nb} = \alpha_\phi \left( -\min(F_j^n, 0) + \mu_j \frac{(\delta S_j)^2}{V_j} \right) \ . \tag{3.27}$$

The coefficient $a_P$ deserves a special attention. Using the definitions (3.27) of the coefficients $a_{nb}$ and applying the mass conservation equation (3.16) into (3.26), it consequently follows that

$$\begin{aligned} a_P &= (1+\beta)\frac{\rho_P V_P^n}{\Delta t} + \sum_{nb} a_{nb} + \alpha_\phi \sum_j \left[ (\max(F_j^n, 0) + \min(F_j^n, 0) \right] \\ &= (1+\beta)\frac{\rho_P V_P^n}{\Delta t} + \sum_{nb} a_{nb} + \alpha_\phi \sum_j F_j^n \\ &= (1+\beta)\frac{\rho_P V_P^n}{\Delta t} + \sum_{nb} a_{nb} - \alpha_\phi \sum_j F_{g,j}^n \ . \end{aligned} \tag{3.28}$$

Therefore, the coefficient $a_P$ depends on the coefficients $a_{nb}$, the CV volume and the grid fluxes at the searched time-step. This would lead to additional computational efforts and memory if an iterative method is applied for solving the non-linear system of equations (3.25). To reduce the computational time, expression (3.28) will be modified. Since for the consistency of the approximations $\alpha_m = \alpha_\phi$, then applying equations (3.19) and (3.21) into (3.28), the coefficient $a_P$ is

$$a_P = \sum_{nb} a_{nb} + (1+2\beta)\frac{(\rho V)_P^{n-1}}{\Delta t} - \beta \frac{(\rho V)_P^{n-2}}{\Delta t} + (1-\alpha_m) \sum_j F_{g,j}^{n-1} \ . \tag{3.29}$$

Hence, the coefficient $a_P$ is the sum of the coefficients $a_{nb}$ and terms depending on the previous time-steps CV volumes and grid fluxes. The sum of the last three terms in equation (3.29) may be computed and stored at the beginning of each time-step. In this way the computational time is reduced.

Finally, the discretised system of equations to be solved is (3.25) and its coefficients are defined by (3.29) and (3.27).

## 3.3 Boundary conditions

To solve the discretised system of equations (3.25) and to find the velocities at the centers of the CVs, their values on the fluid domain boundaries are required. There are principally two types of boundary conditions that may be set:

• Dirichlet boundary conditions - the variables at the boundary points are prescribed. For example - at the inlet, on the fixed and moving walls. Here, we will concentrate on the moving boundaries of the fluid domain. For a known time-dependent movement $\mathbf{x}_f(t)$ of the walls, the grid velocities are also available $\mathbf{v}_g(t) = \dot{\mathbf{x}}_f(t)$. Using non-slip boundary conditions the corresponding fluid velocity is $\mathbf{v}(t) = \mathbf{v}_g(t)$. Moreover, the mass fluxes through the walls are known - they are zeroes.

• Neumann boundary conditions - the gradients of the variables at the boundaries are known. For example - at an outlet or at planes of symmetry. In the case of moving grids the outlet boundary condition deserves special attention. Here, the outlet flux is not simply the same as the inlet flux as it is when the fluid domain is fixed. The total volume change has to be considered to assure the mass conservation in the whole domain. Therefore, equation (3.17) obtained in section 3.2.8 should be satisfied.

## 3.4 Moving grids

As it was emphasized in section 2.1 flows in domains with moving boundaries are considered. Since at every new time-step the fluid boundary moves to a new position, the whole computational domain has to be deformed so that it fits to its new boundaries. From a numerical point of view, this means that the fluid grid has to be modified so that it continues to be a suitable discretisation of the new fluid domain. An example of a fluid domain deformation can be seen in Figure 3.4, where the boundary-fitted grid at two time-steps $t_1$ and $t_2$ is depicted.

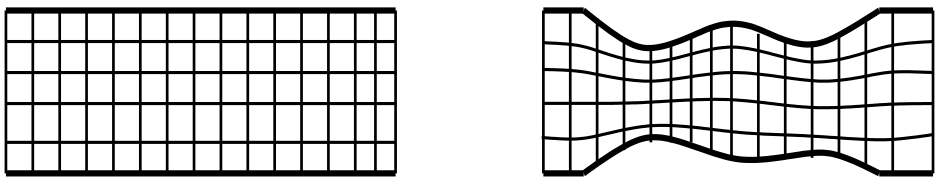



Figure 3.4: Numerical grid change from time $t_1$ to time $t_2$

Depending on the physical problems there are two types of grid movements:

• **Discontinuous grid movement** in which the grid topology at every time-step is changed due to very large displacements. Typical mechanical problems describe a body moving relatively to another one, for example rotating parts in turbines and stirrers, moving cars and trains. Hence, no grid blocks remeshing is required, the movement is done on the block boundaries with special procedures for information transfer between

the blocks moving relatively to each other. Such grid discretisation techniques are the clicking and sliding meshes [41].

• **Continuous grid movement** in which the grid topology and connections are fixed, but the block boundaries are deformed in an arbitrary continuous manner. This deformation may be prescribed explicitly with some functions or may be the result of another problem as it is in the case of FSI.

Here, the second type grid movement will be considered. This means that the positions of all grid points on the walls are known. The new numerical grid which fits these new domain boundaries is searched. Let us denote with $\mathbf{x}_f$ and $\mathbf{x}_b$ respectively the coordinates of the fluid and the boundary grid nodes. The boundary nodes positions are known $\mathbf{x}_b(t)$ - they may be in advance prescribed or obtained from the solution of another problem (in the case of FSI - a structural problem). Therefore, the positions of the inner grid points are searched so that the new computational grid is still a reasonable discretisation of the fluid domain. Two main continuous approaches for a grid update can be pointed out:

• **Pseudo-structural methods** in which the moving grid is treated like a pseudo-structure [3], [49]. Every grid update requires the solution of a differential system of equations. The moving boundaries are used for boundary conditions and the "pseudo-structure deformation" gives the searched fluid domain grid displacements. This method is very general and can be applied to various applications. Unfortunately, big computational efforts may be needed.

• **Algebraic methods** in which the grid modification is done using prescribed functions [20],[4], for example a linear interpolation. Hence, it needs less computational effort and time than the pseudo-structural methods. Additionally, for problems with a moderate boundary movement, the linear interpolation grid update gives suitable computational grids.

Therefore, the last approach is preferred in the present work.

### 3.4.1 Moving grids - update

The algebraic method based on a linear interpolation will be used for grid update and will be described below. A block-structured grid can be modified blockwise. Hence, it is enough to consider one-block grid.

**Problem formulation**: For a given numerical grid $\mathbf{x}_f^{n-1}$ at time-step $t^{n-1}$ and known positions of the grid wall boundaries $\mathbf{x}_b^n$ at the new time-step $t^n$, the new grid position $\mathbf{x}_f^n$ is searched.
Obviously, this is equivalent to finding the "incremental displacements" $\mathbf{d}^n$ for every grid point so that the new grid position $\mathbf{x}_f^n = \mathbf{x}_f^{n-1} + \mathbf{d}^n$ is such that the wall nodes are "displaced" with $\mathbf{d}_b^n = \mathbf{x}_b^n - \mathbf{x}_b^{n-1}$. Since topologically a block is equivalent to a cube, the problem may be schematically depicted as in Figure 3.5.

Regardless of the type of the boundary conditions on the block faces, the corner grid nodes are always determined. Additionally, the new positions of the nodes on all walls

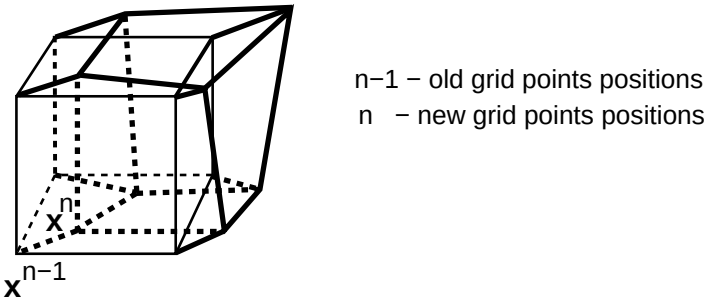
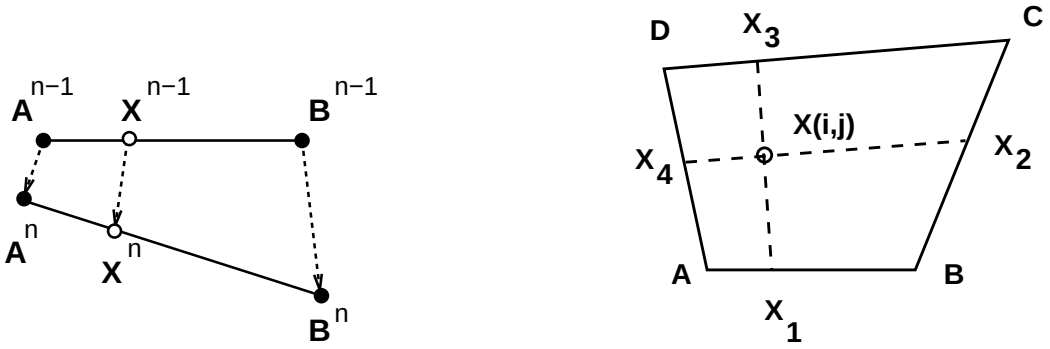



Figure 3.5: Block grid movement

boundaries are known. If the wall is elastic, the new node position is obtained from the structural solver. On the other hand, if no FSI at this wall is considered, the grid points on it do not move. Therefore, the new positions of the nodes on the other boundaries and inside the computational domain remain to be found. In order to find the new grid position, these grid points can be divided into three groups:

• the points on the block edges.

For finding their incremental displacements, it is enough to know the new positions of the corner points of the block edges.




(a) Segment (b) Patch

Figure 3.6: Computation of the grid points displacements

If the displacements $\mathbf{d}_A^n$ and $\mathbf{d}_B^n$ of two nodes $A$ and $B$ are known, then the displacements of a point $X$ of the segment $AB$ is

$$\mathbf{d}_X^n = \frac{|XA|}{|AB|}\mathbf{d}_A^n + \frac{|BX|}{|AB|}\mathbf{d}_B^n \ ,$$

where $|AB|$ denotes the length of the segment $A^{n-1}B^{n-1}$ at the previous time step $t^{n-1}$.

• the points on the non-walls block faces.

Since the new positions of the nodes on all edges can be determined, the incremental displacements only for the inner points of the face remain to be found.

If the new positions of a square patch $ABCD$ boundary are known, then the displacements of an arbitrary inner point $X$ may be found as

$$\mathbf{d}_X^n = \frac{1}{2}\left[\frac{|XX_1|}{|X_1X_3|}\mathbf{d}_{X_3}^n + \frac{|XX_3|}{|X_1X_3|}\mathbf{d}_{X_1}^n + \frac{|XX_2|}{|X_2X_4|}\mathbf{d}_{X_4}^n + \frac{|XX_4|}{|X_2X_4|}\mathbf{d}_{X_2}^n\right]$$

where $X_1$ and $X_3$, respectively $X_2$ and $X_4$ denote the grid points in which the grid lines through the point $X$ intersect the boundaries of the patch.

• the inner grid points of the block.

Knowing the new positions of the block surface, the displacements of the inner grid points is easily found using a linear interpolation of the displacements of the grid points on the block faces in which the respective grid planes pass through.

The linear interpolation gives a very fast method for a grid update. It is easy to implement and works successfully on problems in which the grid deformations are moderate and do not lead to very big distortions. Then the obtained new grid is still a "good" domain discretisation. However, this grid update strategy is not suitable for problems with very complex geometries in which big distortions of the domain boundaries take place. The application of the method on such domains may lead to grid irregularities and negative control volumes. In this case the pseudo-structural methods are advantageous.

## 3.5 Solution methods

For the numerical solution of the discrete system of equations (3.2) at every time-step the Semi-Implicit Pressure-Linked Equations (SIMPLE) algorithm [32] is used. Below, this method will be shortly presented and generalised so that it can be applied to the written in Eulerian-Lagrangian coordinates equations. Additionally, a multigrid method is applied to accelerate the convergence and to help finding a grid-independent solution.

### 3.5.1 SIMPLE algorithm

The fluid behaviour when its domain is deformed in a prescribed way, is found at every time-step using the prediction-correction SIMPLE algorithm depicted in Figure 3.7.

At the beginning of each time-step the new grid position is found. Then the grid fluxes are calculated using equation (3.21). The velocities and the mass fluxes at the inlet and wall boundaries are also set.

As an initial guess for the fluid variables, the previous time-step values are used.

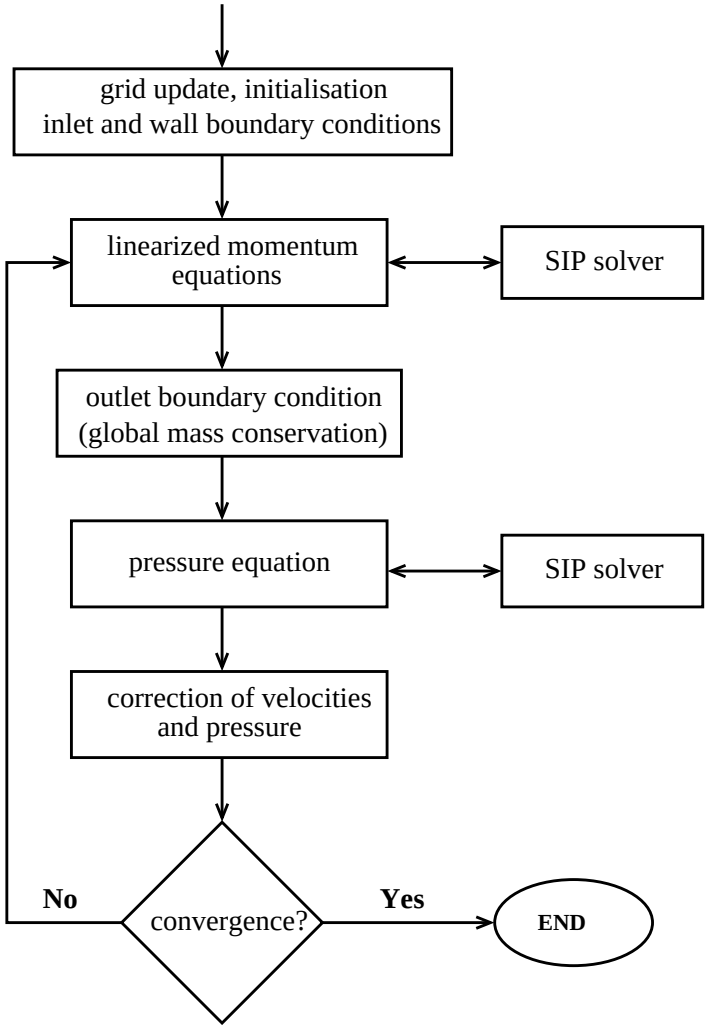



Figure 3.7: SIMPLE algorithm

The solution at the new time-step is found using prediction-correction iterations. For a convergence criterion the condition

$$\sum_{CVs} |\mathrm{Res}_{v_i}| \leq \varepsilon \ , \quad i = 1, 2, 3 \tag{3.30}$$

is chosen. Here, $|\mathrm{Res}_{v_i}|$ denotes the absolute value of the residual for the $v_i$-velocity equation for some CV normalised by the total inlet mass flux. The sum is over all CVs and $\varepsilon$ is the allowed error.

First the momentum conservation equations are solved and the new fluid velocities are found. In order to assure a global mass conservation the outlet velocities and mass fluxes are corrected using equation (3.17).

To guarantee mass conservation in each control volume, the continuity equation (3.16) is transformed into a pressure-correction equation using a selective interpolation [36]. Solving it, the necessary pressure and velocity corrections are found. These corrections

are used to modify the fluid variables. The obtained pressure and velocities are used to build the coefficients of the momentum equations at the next outer iteration.

To improve the convergence properties, underrelaxations of the variables are applied.

The iteration process continues until the convergence criterion (3.30) is satisfied.

At each iteration the discretised velocity and pressure-correction equations result into linear systems of equations. Each linear system of equations is solved using the method of Stone called also strongly implicit procedure (SIP) [43]. This iterative algorithm uses an incomplete lower-upper decomposition technique.

### 3.5.2 Multigrid method

The solution of the Navier-Stokes equations using the iterative SIMPLE algorithm requires the more computer power the finer the numerical grid is. Moreover, the number of the outer iterations linearly depends on the number of the grid points in each direction. Hence, in 3-dimensional domains if the numbers of grid points in all directions are doubled, the number of outer iterations on the refined grid will increase with factor $2^3$ and will lead to very large computational time.

Unfortunately, most technical flows have complex behaviours and fine numerical grids are required for their correct simulation. The development of multigrid methods, for which the number of iterations does not depend on the number of the grid points has been a research area for many scientists [16]. Here, we will briefly present the multigrid idea for finite-volume methods for structured grids and obtain the necessary modifications for moving grids.

The finest grid is created at first. Then a number of consequently coarsened grid levels is obtained so that every coarse grid control volume consists of eight fine grid control volumes as shown in Figure 3.8.

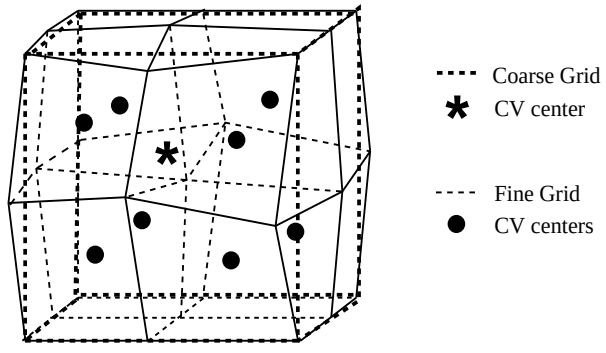



Figure 3.8: One coarse grid CV consisting of eight fine grid CVs

To present the multigrid idea, stationary flow problems will be discussed at first. The solution process starts on the coarsest grid level. After the fluid variables are found on this grid, they are interpolated to the control volumes centers of the next finer grid

level and used as initial approximations. To solve the problem on the second grid level a two-grid procedure is applied in the following manner:

1. unless a converged solution is obtained, do a certain number of SIMPLE iterations on the second grid
2. restrict the intermediate solution to the coarse grid CV centers and do a given number of iterations on this level to find the corrections for the fine grid solution
3. interpolate the coarse grid nodes corrections to the fine grid nodes, correct the fine grid solution and repeat step 1.

In this way the converged solution on the second grid is found. The loop consisting of steps 1, 2 and 3 is called a V-cycle. The received solution is prolongated to the CV centers of the next finer grid. The iterative process continues until a converged solution is obtained on the desired finest grid. The described procedure is known as a full multigrid scheme and is schematically depicted in Figure 3.9. In this way the errors with wave-lengths corresponding to the different grid-spacings are consequently removed and finally, the solution on the finest grid is received.

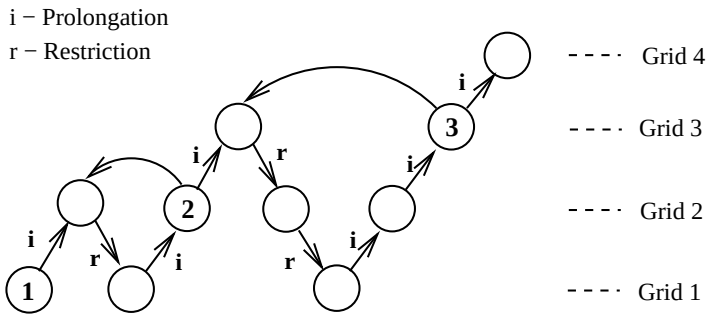



Figure 3.9: Full multigrid approximation scheme

In order to use the multigrid method for fluid domains with moving grids, not only the finest grid has to be moved to its new position, but also all coarser grid levels should be consistently updated. Additionally, the grid fluxes have to be found on all coarser grid levels as well. Since only the necessary corrections are searched, no global conservation on the coarser levels is needed.

## 3.6 Validation of the implementation of moving grids

To validate the implementation of moving grids and the SCL into the solver FASTEST-3D, the flow in a channel with a moving indentation is simulated. This channel flow may be found in various practical problems in biomechanics such as flows in arteries and veins. Hence, it has already been studied experimentally [33] and numerically [34], [9], [49].

### Problem Formulation

The problem geometry is schematically depicted in Figure 3.10. In order to compare with the available results, the same parameters as in [9] are set. The channel is chosen to be $\mathbf{L}_1 + \mathbf{L}_2$ long, where $\mathbf{L}_1 = 0.0985$m and $\mathbf{L}_2 = 0.18$m and to have a square cross-section with a length b = 0.01m. It is assumed that the part of its wall in the range $[-x_3, x_3]$ is oscillating with a period T in the following way:

$$y(x) = \begin{cases} h & \text{for} \quad 0 < x < x_1 \\ 0.5h\{1 - \tanh\left[a(x - x_2)\right]\} & \text{for} \quad x_1 < x < x_3 \\ 0 & \text{for} \quad x > x_3 \end{cases}$$
$$\text{with} \quad h = h(t) = 0.5h_{max}\left[1 - \cos(2\pi(t - t_0)/\mathrm{T})\right],$$

where the maximum wall displacement is $h_{max} = 0.38$ b and the rest of the parameters are $x_1 = 4$b, $x_3 = 6.5$b, $x_2 = (x_1 + x_3)/2$, $a = 414$. Since this is a two-dimensional problem, in order to use it for the verification of our 3-dimensional solver, symmetry boundary conditions are applied on the Top and Bottom boundaries of the channel.

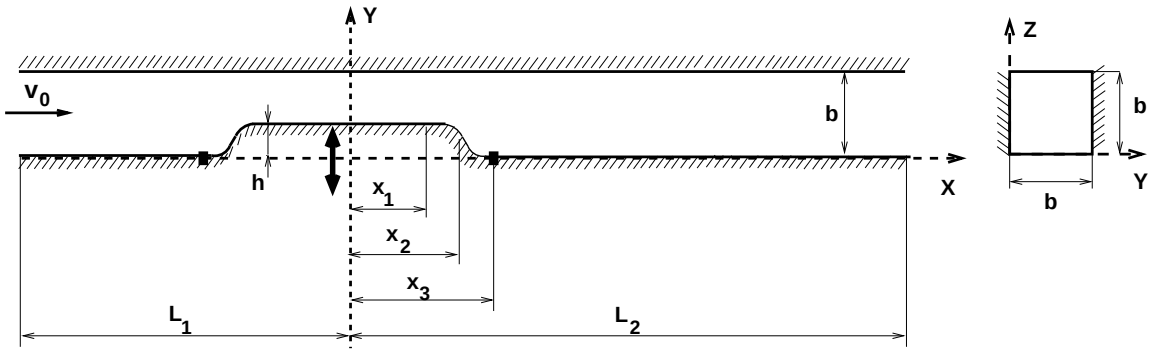



Figure 3.10: Flow in a channel with a moving indentation - problem formulation

At the inlet boundary a parabolic velocity profile constant in time with an average velocity $v_0 = 1$ m/s is given, which is the same as the analytical solution for a corresponding channel flow. The fluid viscosity is $\mu = 10^{-3}$ m s/kg and the fluid density $\rho$= 50.7 kg/m$^3$ leading to Reynolds number

$$Re = \frac{\rho v_0 \mathrm{b}}{\mu} = 507 \ .$$

The experimental investigations in [33] show that the flow behaviour strongly depends on the frequency of the wall movement and on the corresponding Strouhal number

$$St = \frac{\mathrm{b}}{v_0 \ \mathrm{T}} \ .$$

Two regimes can be distinguished:

- $St \leq 0.005$ – It is characterised by two quasi-steady vortices appearing behind the wall indentation and then disappearing as the wall returns to its original position.
- $St > 0.005$ – Here, as the wall is deformed a street of vortices alternatively close to the two walls appears behind the indentation. The vortices are travelling to the outlet and slowly disappear as the channel wall returns to its start position. In the second part of the period a second corotating eddy develops upstream of the first one in the same separated-flow region, known as 'eddy doubling'.

The second regime has been studied in detail for Strouhal number $St = 0.037$ in [9], where enough data are presented that can be used for comparison. Additionally, the flow pattern is more complicate than the one in the first regime, because there are many particular features which should be modelled by the numerical scheme. Therefore, this flow is chosen to be investigated and in this way to validate the moving grids implementation.

**Numerical simulation (single grid)**

The fluid domain is discretised using a structured grid with 221x40x3 control volumes. The time-step $\Delta t = \mathrm{T}/200$ is chosen for time discretisation. The developed steady state channel flow is taken as initial flow condition. The fluid behaviour is investigated when a wall of the channel is oscillating as described above. To obtain a second-order accuracy in spatial discretisation, the CDS is applied. Two implicit time-stepping schemes are used – Euler backwards (first-order) and Crank-Nicolson (second-order) schemes.

In order to verify the implementation of moving grids for both time-discretisation methods, the obtained results are compared to the results received by the numerical simulation in [9]. The maximum values of the horizontal velocity for all three computations are shown at different time-steps within one time period in Figure 3.11. It can be seen that the results received from both simulations are in a very good agreement with the results obtained in [9]. In all data sets the velocity reaches its maximal value 2.66 m/s at time 0.4 T. It is about 77% bigger than the maximal velocity 1.5 m/s at the inlet. The comparison in Figure 3.11 also shows that the fluid behaviour is not influenced by the chosen time-discretisation scheme. Therefore, to present the flow pattern, the results of the simulation with Euler backwards scheme will be used.

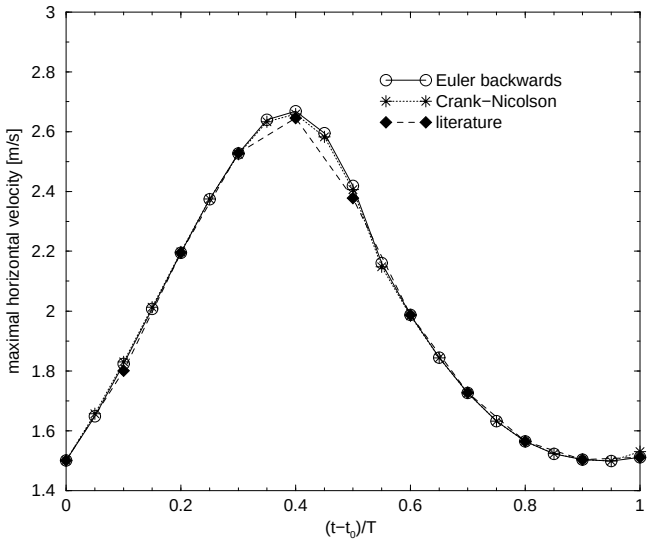



Figure 3.11: Maximal horizontal velocity

In Figure 3.12 the horizontal velocity and the streamtraces behind the wall indentation are presented at different time-steps within one time period. The corresponding pressure distribution is depicted in Figure 3.13. Here, for a better visualisation the domain is scaled in $Y$- direction with factor 2.

The flow behaviour presented in Figures 3.12 and 3.13 is in a very good qualitative agreement with other authors computations [33], [35], [9], [49].

As the wall starts moving, the volume of the domain reduces. Due to the constant inlet flow, the global mass conservation leads to a bigger outlet mass flux. Accordingly, the pressure difference between the inlet and the outlet increases. Therefore, the fluid speed also magnifies and reaches its maximum (2.66 m/s) at time 0.4 T. At this moment the first two vortices formed behind the indentation can also be seen. The alternative building up of vortices continues till about 0.7 T, when four vortices close to each wall of the channel are formed. The vortex street travels downstream together with the flow. The observation of the streamtraces at times 0.6 T and 0.7 T, presented in Figure 3.12, shows that another physical phenomenon - the double vortex formation - is also successfully modelled. After 0.7 T the vortices start decreasing their size and finally, they totally disappear. So, the flow pattern at the end of the period is similar to the steady initial flow.

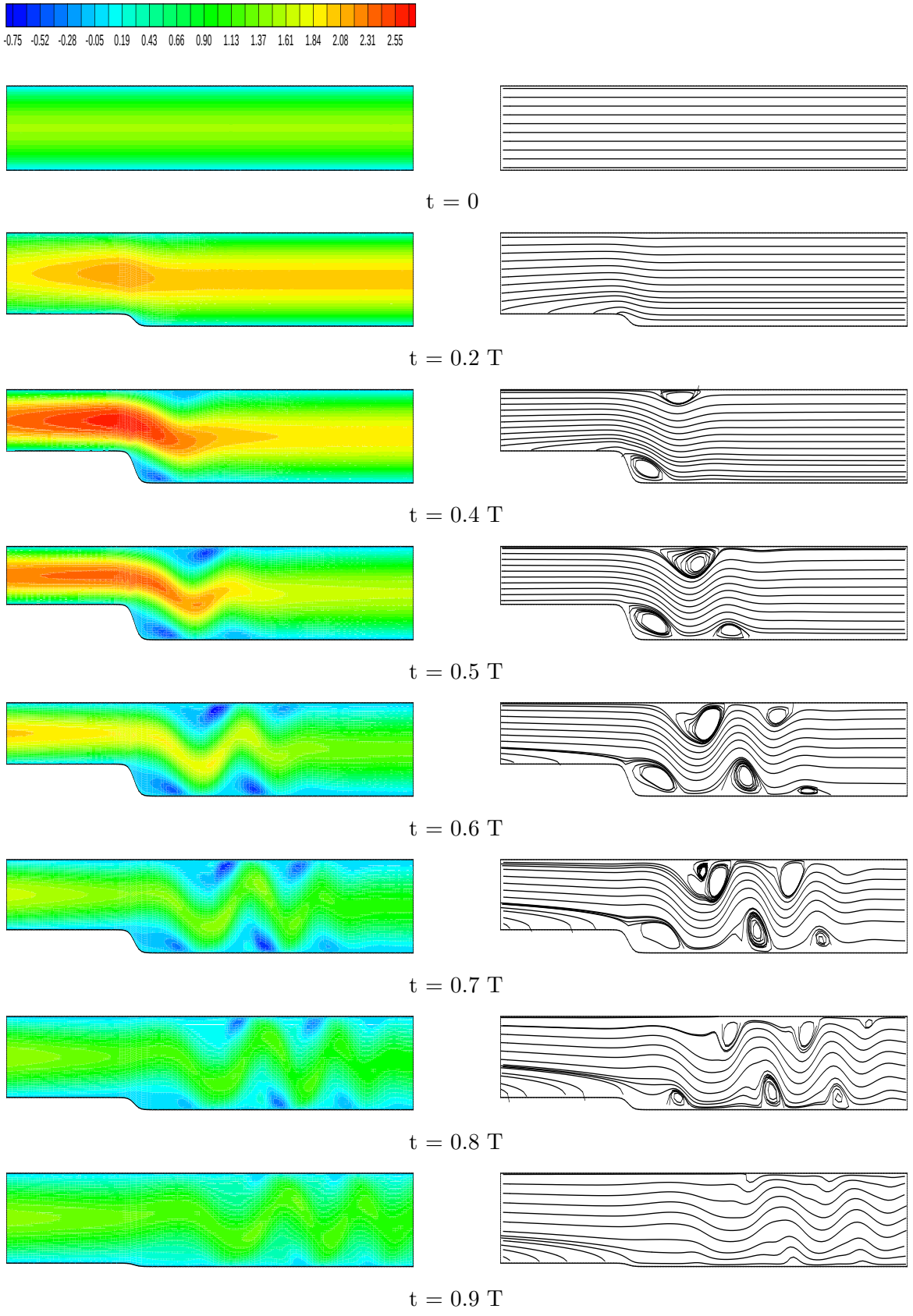



Figure 3.12: Horizontal velocity (left) and streamtraces (right)

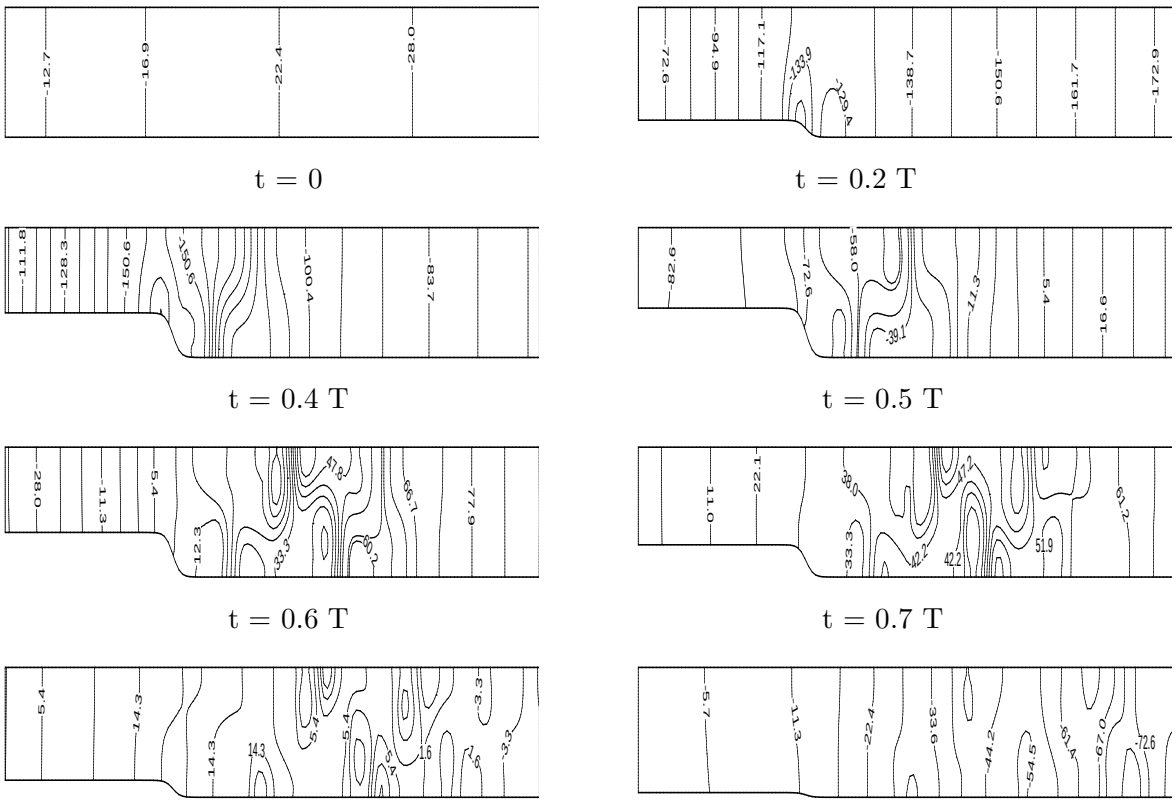



Figure 3.13: Pressure distribution

### Numerical simulation (multigrid)

For modelling complex fluid problems it is very important that the multigrid solver also works for problems with moving grids. To study and to verify this feature, the channel problem described above is simulated using a multigrid method with 3 grid levels. For this purpose a numerical grid (Grid 3) consisting of 220x40x12 = 105600 control volumes is created. The other two coarser grid levels are respectively Grid 2 with 110x20x6 = 13200 CVs and Grid 1 with 55x10x3 = 1650 CVs.

The computations have been performed on an AMD Athlon 1.1GHz processor.

Again, two different time-discretisations are applied - the first-order accurate Euler backward and the second-order accurate Crank-Nicolson discretisations. The average computing time per time-step on Grid 3 with and without using multigrid is compared and given in Table 3.1.

| | computational time | | acceleration |
|---|---|---|---|
| time discretisation | single grid | multigrid | factor |
| Euler backwards | 892 s | 127 s | 7.02 |
| Crank-Nicolson | 1380 s | 160 s | 8.625 |

Table 3.1: Computational time for one time-step - comparison

Hence, an acceleration factor of about 7 for the Euler backwards and 8.6 for the Crank-Nicolson schemes are achieved. This is consistent with the expected acceleration factor 5–20 in [38] for a 3-level multigrid applied to an insteady laminar flow.

### Conclusion

The provided verification shows that moving grids are correctly implemented into the code FASTEST-3D for Euler backwards and Crank-Nicolson time discretisations. Additionally, the multigrid efficiency in the solver is preserved and also works successfully for moving fluid domains.

# 4 Structural dynamics subtask - numerical solution

Similar to the fluid dynamics task, for solving the structural dynamics problem the advantages of the already developed and well validated program FEAP are chosen. Since a Lagrangian formulation of the governing equations is used (see section 2.2), no changes in the solution algorithms are required. Special care has to be taken only at the interface boundaries to the fluid domain. In the following section the main features of the structural solver will be presented. Additionally, the used numerical methods will be briefly pointed out. In regard to our final goal - dynamic simulation of FSI, the employed time-stepping techniques will be elucidated.

## 4.1 The finite element code FEAP

The program FEAP has been developed to solve various structural dynamics problems [46]. It uses a finite-element discretisation of the computational domain. A variety of element formulations have been created to model the static and transient behaviour of structures with various material laws for both small and finite deformations. The discretisation with finite elements of the equations presented in section 2.2 results in a system of equations in the form:

$$\mathbf{M}\ddot{\mathbf{u}}_s + \mathbf{N}(\mathbf{u}_s) = \mathbf{f} \ , \tag{4.1}$$

where $\mathbf{u}_s$ denotes the structural displacements, $\mathbf{M}$ is the mass matrix and $\mathbf{f}$ consists of the body and surface traction terms. The term $\mathbf{N}(\mathbf{u}_s)$ represents the internal forces of the structure. For a linear elastic behaviour $\mathbf{N}(\mathbf{u}_s) = \mathbf{K}\mathbf{u}_s$, where $\mathbf{K}$ is the usual stiffness matrix. Therefore, the system of equations (4.1) is linear. On the other hand if a non-linear elastic behaviour is considered, $\mathbf{K}$ depends on the displacements and $\mathbf{N}(\mathbf{u}_s) = \mathbf{K}(\mathbf{u}_s)\mathbf{u}_s$. In this case the system of equations (4.1) is non-linear.

The discretised problem (4.1) is solved using the following numerical methods:

• Different methods for transient solution based on the classical Newmark method [27] are available. The Newmark method uses approximations of the displacements $\mathbf{u}_s^n \approx \mathbf{u}_s(t^n)$, the velocities $\mathbf{v}_s^n \approx \dot{\mathbf{u}}_s(t^n)$ and the accelerations $\mathbf{a}_s^n \approx \ddot{\mathbf{u}}_s(t^n)$. It is a one-step algorithm solving the equation

$$\mathbf{M}\ \mathbf{a}_s^{n+1} + \mathbf{N}(\mathbf{u}_s^{n+1}) = \mathbf{f}^{n+1} \tag{4.2}$$

at time-step $t^{n+1}$, where the displacements and the velocities are defined by

$$\mathbf{u}_s^{n+1} = \mathbf{u}_s^n + \Delta t \mathbf{v}_s^n + \Delta t^2 \left[(0.5-\beta)\mathbf{a}_s^n + \beta \mathbf{a}_s^{n+1}\right] \tag{4.3}$$

$$\mathbf{v}_s^{n+1} = \mathbf{v}_s^n + \Delta t \left[(1-\gamma)\mathbf{a}_s^n + \gamma \mathbf{a}_s^{n+1}\right] \tag{4.4}$$

These equations are completed with appropriate initial conditions.

Using formula (4.3) and (4.4), equation (4.2) can be written in terms of the displacements as

$$\mathbf{M}\ \mathbf{u}_s^{n+1} + \beta \Delta t^2 \mathbf{N}(\mathbf{u}_s^{n+1}) = \hat{\mathbf{f}}^{n+1} \tag{4.5}$$

with

$$\hat{\mathbf{f}}^{n+1} = \beta \Delta t^2 \mathbf{f}^{n+1} + \mathbf{M}\left(\mathbf{u}_s^n + \Delta t \mathbf{v}_s^n + \Delta t^2 (0.5-\beta)\mathbf{a}_s^n\right) \ .$$

The numerical parameters $\beta$ and $\gamma$ control the stability and the dissipation of the method. If $\beta$ is non-zero the Newmark algorithm is implicit and requires a linearisation of the momentum equation (4.2) and may be combined with the iterative Newton method. Investigations of its convergence properties and discretisation error can be found in [50], [17] and [53]. In the general case it is first-order accurate and the condition $2\beta \geq \gamma \geq 0.5$ should be fulfilled for stability. Second-order time-accuracy is obtained only if $\gamma$=0.5, which in absence of a natural damping means that the modes of all frequencies are preserved. However, the finite element discretisations model very well the lower frequencies, but give a bad approximation for the higher frequencies [50]. Therefore, for better accuracy the time-stepping scheme should damp the higher frequencies.

To introduce a numerical damping without reducing the temporal accuracy, the Newmark method can be modified to the $\alpha$-method (Hilbert-Hughes-Taylor (HHT) method) [17], [50]. Using the energy-conserving form of this method, the time-discrete equation of motion (4.1) is replaced by

$$\mathbf{R}^{\mathbf{n}+\alpha} = \mathbf{f}^{n+\alpha} - \mathbf{N}(\mathbf{u}_s^{n+\alpha}) - \mathbf{M}\ \mathbf{a}_s^{n+\alpha} = 0 \ , \tag{4.6}$$

where $\alpha$ is a parameter between zero and 1. Here, $t^{n+\alpha} = (1-\alpha)\ t^n + \alpha\ t^{n+1}$ and the displacements, velocities and accelerations at this intermediate time are:

$$\begin{aligned} \mathbf{d}_s^{n+\alpha} &= (1-\alpha)\ \mathbf{d}_s^n + \alpha\ \mathbf{d}_s^{n+1} \\ \mathbf{v}_s^{n+\alpha} &= (1-\alpha)\ \mathbf{v}_s^n + \alpha\ \mathbf{v}_s^{n+1} \\ \mathbf{a}_s^{n+\alpha} &= \frac{1}{\Delta t}\left(\mathbf{v}_s^{n+1} - \mathbf{v}_s^n\right) \ . \end{aligned} \tag{4.7}$$

In this way no starting conditions for the accelerations are required.

Obviously, if $\alpha = 0$, this algorithm coincides to the Newmark method.

If $\alpha$=0.5 the conserving $\alpha$-method is equivalent to the Crank-Nicolson method. The momentum and energy conservations are achieved by a special selection of $\beta$ and $\gamma$. For example, for a linear elastic behaviour without damping, a conservation is obtained choosing $\alpha = \beta = 0.5$ and $\gamma = 1$.

With regard to our final goal - creating a method for modelling fluid-structure interaction problems, the energy conserving $\alpha$-method will be adopted for time-stepping within the structural subtask. In this way since the fluid solver is conservative, a global conservation within the coupled problem will be also achieved.

• To solve at every time-step the non-linear system of equations obtained by the Newmark or the $\alpha$-methods, the well known Newton method is applied. After a linearization, the following system of equations

$$\mathbf{K}_t\ \Delta \mathbf{u}_s = \mathbf{R} \tag{4.8}$$

is received. Here, $\Delta\mathbf{u_s}$ denotes the incremental displacements and $\mathbf{R}$ is the residual for equation (4.5). $\mathbf{K}_t$ stays for the stiffness matrix, which is the sum of the geometric stiffness and the material tangent matrices.

Let us notice that the Newton method is not needed when small deformations are considered. In this case the system of equations (4.1) is linear.

• The algebraic system of equations is solved by a sparse direct solver using Cholesky triangulation.

# 5 Coupling algorithms for fluid-structure interaction

Due to the fluid flow pressure and shear stress, forces appear on the wall boundaries of the fluid domain. Therefore, the structural behaviour is determined not only by the applied external forces, but also by the fluid forces. All these forces deform the elastic walls which leads to a change of the computational domain of the fluid. In this way the fluid movement is also affected. Hence, both the flow and the structure act on each other.

In section 2 the governing equations of fluid-structure interaction problems have been presented. The existing strategies for their solution have been discussed in section 1.2.

Here, the available well-validated codes FASTEST-3D and FEAP will be applied to the fluid and the structure subproblems, respectively. For this purpose moving grids have been implemented into the fluid solver (section 3). The aim of this section is to propose a suitable algorithm that couples the two programs into one powerful code able to solve various FSI problem.

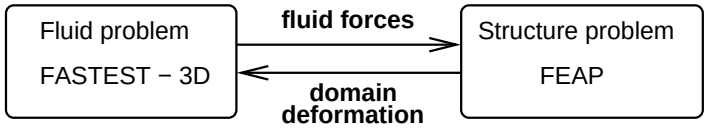



Figure 5.1: Necessary data exchange

At the beginning of the numerical solution spatial and temporal discretisations of the whole computational domain are needed. To model the interaction between the flow and the structure special procedures for data exchange between the codes are required:

- The fluid pressure and shear forces have to be considered as boundary forces for the structural subproblem.
- The structural deformations lead to displacements of the fluid boundaries. Hence, the fluid grid has to be updated correspondingly.
- Additionally, at every time-step the equilibrium between the flow and the structure should be found.

## 5.1 Spatial discretisation - Surface tracking

In order to apply the finite-volume solver to the fluid subtask, the fluid domain has to be discretised using a block-structured finite volume grid as described in section 3.2. Accordingly, the structure has to be discretised into the available in FEAP finite elements. With regard to the FSI problem special attention deserves the fluid-structure interface.

In the general case the fluid and the structure boundaries may be arbitrarily discretised. Then a suitable surface tracking procedure which passes the structural displacements to the fluid domain is required. General strategies for grids non-matching at the interface can be found in [4], [5] and [3].

In the present research we will choose the fluid grid points and structural nodes in a special way. First the fluid domain is discretised into finite volumes. The structure is divided into finite elements so that the nodes at the interface coincide to the corresponding fluid grid points as shown in Figure 5.2.

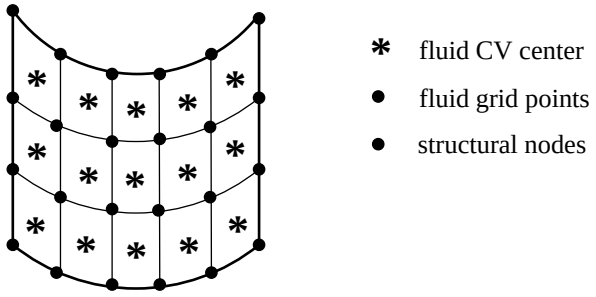



Figure 5.2: Fluid-structure interface: Coinciding fluid and structural grids

In this way no transformation of displacements from the structural nodes to the fluid grid-points is needed. The incremental displacements obtained from the structural computation are directly used as displacements of the fluid domain interface. Using these grid boundary conditions and the strategy given in section 3.4.1, the fluid grid is updated correspondingly.

## 5.2 Time discretisation

The fluid behaviour depends on the fluid domain movement. The bigger the domain deformation is, the more different the flow field will be. On the other hand the structural deformation strongly depends on the material properties as well as on the magnitude of the applied forces. Hence, different time-steps may be required for the solutions of the two FSI parts.

However, to assure the energy conservation of the time-stepping scheme, the fluid and the structure should be in an equilibrium at every time-step. Therefore, here, one fixed time-step $\Delta t$ is chosen for the whole FSI problem. If the flow is generally much more sensitive to the time resolution than the structure, the time-step is determined from the fluid dynamics subproblem. Contrarily, when the structure requires smaller time-steps, $\Delta t$ is chosen in accordance to the structural subtask.

## 5.3 Fluid forces at the fluid-structure interface

The fluid pressure and shear forces given by equation (2.27) are needed at the structural nodes on the interface for solving the structural subproblem. Because of the collocated

variable arrangement in the fluid code, the pressure is known at the wall CV centers and the shear forces may be computed using the velocity from the wall and CV centers next to it. Assuming that the forces are constant on each boundary face, the fluid forces into the grid nodes can be found from the values at the CV centers depicted in Figure 5.3 using a linear interpolation. This force projection is conservative because it

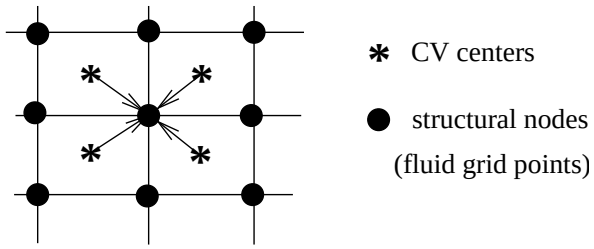



Figure 5.3: Fluid-structure interface: Fluid forces projection

assures that the total force at the fluid boundary is the same as the total force on the structural boundary.

## 5.4 Explicit coupling algorithm

After these preparations, the loose (weak) coupling algorithm will be presented at first. In this approach information will be exchanged between the solvers only once per time-step. At the beginning of the simulation the fluid and the structural parts are discretised as described in section 5.1. The time-step is also chosen as well as the desired convergence criteria. All material and fluid properties parameters are set and the fluid and the structural variables are initialised.

After the initialisation the iteration process starts. At every time-step at first the fluid domain is modified so that it fits to its new boundaries. The corresponding swept volumes due to this grid movement are also found. The code FASTEST-3D is applied to solve the fluid subtask. As a result the pressure and the velocity field are obtained. Then the fluid forces are projected into the structural nodes as it was proposed in section 5.3. These forces are applied as boundary conditions for the structural dynamics subtask. Using the program FEAP the displacements of the interface boundary are obtained. During the iteration process these displacements are always computed relatively to the initial position of the structure. Therefore, only the incremental displacements are actually used for the fluid grid update.

Due to the weak coupling approach the deformed mesh is taken into account at the next time-step. Hence, this strategy will be referred as 'explicit' coupling. Then the fluid domain grid is changed by distributing linearly the incremental displacements among the internal fluid grid points (see section 3.4.1).

The elucidated 'explicit' coupling algorithm is schematically presented in Figure 5.4.

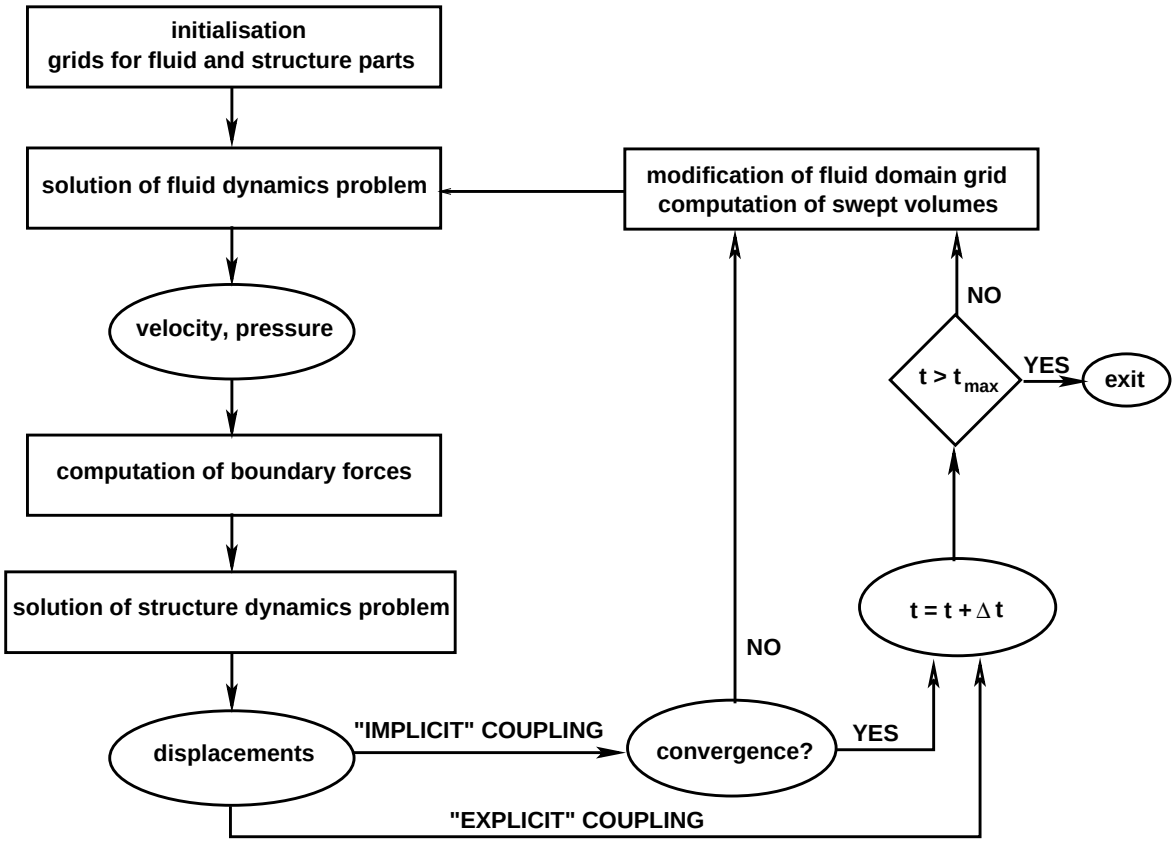



Figure 5.4: Coupling algorithms for fluid-structure interaction

Because of the explicit treatment of the fluid grid, convergence problems may be expected. This results into restrictions of the time-step even if implicit schemes are applied to both fluid and structure parts.

To preserve the conservative properties of the solvers, care has to be taken for energy conservation also at the interface. This means that at every time-step the fluid and the structure should be at equilibrium. Hence, the data can be exchanged between the fields just once per time-step, only if both the fluid and the structure variables are constant within the time-step. This assumption additionally restricts the choice of the time-step size.

If big time steps are used, the fluid program predicts the new flow field based on an approximation of the structural position without considering the movement of the structure caused by the fluid forces. However, the structure will change its place due to the fluid action. This leads to an overprediction of the fluid forces by the flow solver. Therefore, the stability may be improved by suitable underrelaxation of the forces or the displacements computed by the program FEAP.

In the general case, the 'explicit' approach does not assure energy conservation. Therefore, it is not suitable for problems with large structural deformations.

However, when small deformations are considered this strategy can be used. Since

the small fluid grid movement leads to minor changes in the flow field, the energy conservation is not significantly affected.

Let us notice that this strategy can be successfully applied to solve steady state FSI problems. The explicit coupling method will be further investigated in section 6.1, where some applications will be also presented.

## 5.5 Implicit coupling algorithm

As it has just been pointed out in the previous section the explicit approach has many disadvantages as well as a restricted area of applications. However, it can be easily modified so that the energy conservation and the equilibrium between the flow and the structure are guaranteed at every time-step. In this way its convergence properties are also improved.

To obtain the equilibrium of the whole system at every time-step a predictor-corrector iterative scheme is applied.

In the beginning initial values for the displacements are provided. For this purpose a linear interpolation can be used. In this case the incremental displacements received at the previous time-step are taken as starting approximations. Let us notice that such a prediction is consistent with the one used by the explicit coupling algorithm. It would also give the searched displacements, if the structure consists of a linear elastic material. Therefore, this prediction will be used by our coupling algorithm. Using these incremental displacements, a grid prediction for the flow subproblem is found.

Further the iteration scheme follows the one of the explicit coupling algorithm. Using the code FASTEST-3D the new fluid velocity and pressure are obtained. The fluid forces are evaluated and with the help of the program FEAP the new domain displacements are received. The latter are used to modify the current spatial discretisation. Hence, a new prediction for the fluid grid displacements is found and the iterative process continues until the convergence conditions are fulfilled.

### Conditions for convergence

Special attention deserves the choice of the convergence criterion for the predictor-corrector scheme for solving a FSI problem. Let us notice that if the flow field stops varying there will be also no change in the fluid forces and consequently in the structural displacements. Since the fluid solver is iterative, at every predictor-corrector iteration the fluid convergence conditions are automatically checked. The satisfaction of these conditions means that the variations of the structural deformations are too small to influence the flow field. Therefore, the convergence conditions of the fluid solver may be used as convergence criteria for the whole FSI problem.

Due to the achievement of a fluid-structure equilibrium, the implicit coupling strategy is energy conserving and may be successfully applied to time-dependent problems with finite deformations.

If the fluid part is solved in every predictor-corrector step until the convergence criterion is fulfilled, the implicit coupling would need much more computational effort than the explicit one. It also requires additional memory for the storage of the old time-step grid. However, the computational performance can be improved. Since the final converged fluid-structure equilibrium is searched, it is not necessary at every predictor-corrector step to solve exactly both subtasks. Just a few iterations in FASTEST-3D may be used and hence, a significant reduction of the computational time is achieved.

The presented implicit coupling algorithm will be investigated in details and compared with the explicit one in section 6.2.

# 6 Investigation of the coupling algorithms

## 6.1 'Explicit' coupling algorithm - FSI problems with small deformations for the structural part

As it was already mentioned in section 5.4 the explicit coupling algorithm may be successfully applied to problems with small deformations. Here, two different examples will be investigated and used to study the properties of the coupling method.

### 6.1.1 Example 1: Flow in an elastic pipe with supported ends and two pinching forces

Here, the laminar flow in an elastic pipe will be studied. A schematic problem description is given on Figure 6.1. For computational convenience the pipe is fixed along a certain area at the inlet and the outlet.

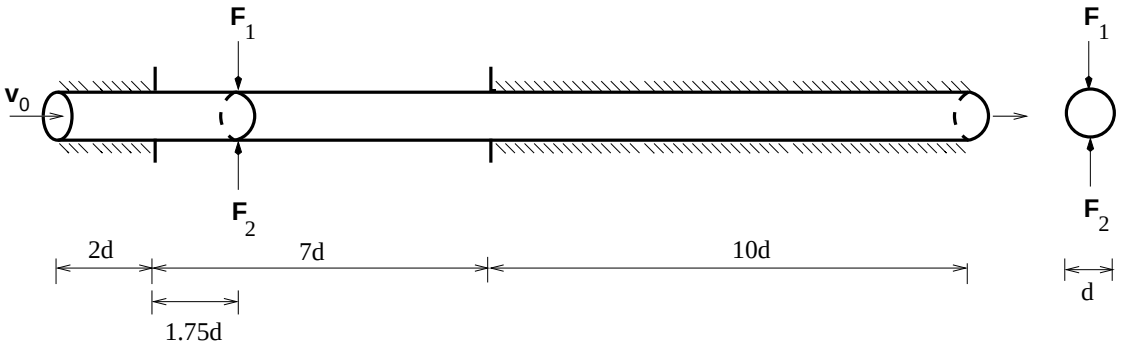



Figure 6.1: One pipe - problem definition

The inlet velocity is assumed to have a constant parabolic profile with a maximal value $\mathbf{v}_0 = 0.002$ m/s. The pipe diameter is d = 0.1 m and the Reynolds number is Re = 100. The pressure at the outlet is assumed to be the same as the static pressure, hence the pressure along the pipe is positive throughout the computations. The pipe is made of a linear elastic isotropic material with elasticity module $\mathrm{E} = 10^7$ kN/m$^2$ and Poisson's ratio $\nu = 0.4$. The thickness of the pipe wall is 0.0004 m.

Two pinching forces $\mathbf{F}_1$ and $\mathbf{F}_2$ pointing to the center of the cross-section are applied at two opposite points on the surface of the elastic pipe. The forces are taken to be time-periodic with a period $T = 200$ s and an amplitude of 50 N. Hence, it is enough to consider the fluid-structure interaction within one time period.

At the beginning the fluid domain is discretised into finite volumes. For the needs of the solver FEAP, the finite element grid is extracted from the fluid grid so that its nodes coincide to the fluid grid points on the elastic walls. As finite elements the 4-node shell elements with 6 degrees of freedom (3 translations and 3 rotations per node) are chosen.

Since small deformations are considered, the terms of inertia for the structure may be neglected. Hence, at every time-step the structural subproblem is solved quasi-steady, where the fluid forces are applied as boundary conditions. Let us notice that because of the used elements and fluid forces projections, the structural solution is second-order accurate in spatial discretisation.

To achieve a second-order spatial accuracy also for the fluid task, the convective fluxes are approximated using the CDS. On the other hand the implicit first-order Euler backwards time-stepping scheme is selected for the fluid part of the FSI problem.

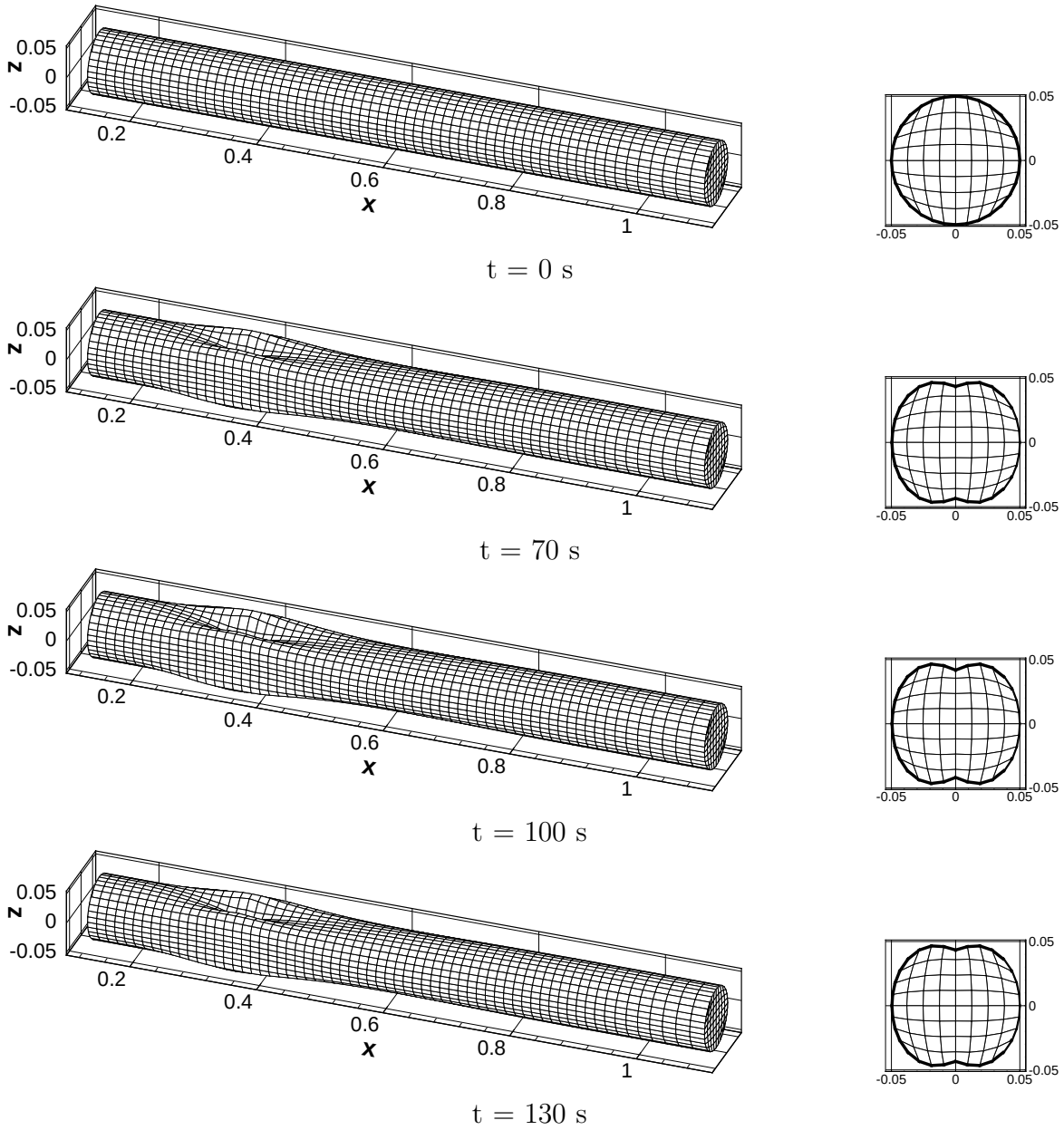



Figure 6.2: Grid (left) and cross-section (right) deformations at different times

For the given fluid and material properties the deformation of the boundaries is more effected by the external forces than by the pressure and shear stresses. The action of the structural deformations on the fluid dominates the response of the fluid to the structure. Therefore, this example allows us to investigate the fluid behaviour when the structure is being deformed due to external forces. Nevertheless, the fluid forces are not neglected and there is a small interaction.

To achieve a good starting condition for the numerical simulation a steady flow in the elastic pipe without external forces is computed. For the presentation of the results in the following only the part of the fluid domain grid containing the elastic part of the pipe is shown.

As it is expected, the pipe is the most deformed when the pinching forces are the biggest, i.e. at time 100 s. The deformation of the grid at different times is presented in Figure 6.2 (left) in comparison with the undeformed state at t = 0 s. Here, for better visualisation the displacements are scaled with factor 5. The unscaled change of the cross-section, where the pinching forces are applied, is given next to the corresponding deformed pipe in Figure 6.2 (right).

The variation of the applied forces and the corresponding outlet mass flux are shown in Figure 6.3. For comparison the inlet mass flux, which is constant during the simulation, is also given.

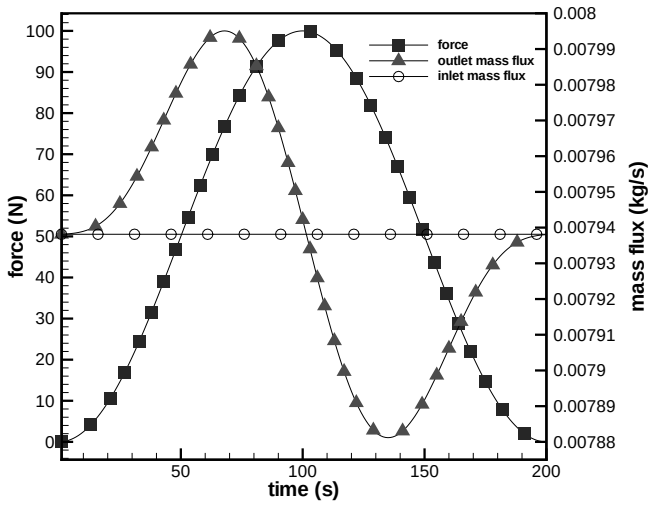



Figure 6.3: Applied force, inlet and outlet mass fluxes

It can be noticed that the outflow mass flux does not strictly follow the applied forces magnitude. This effect can be explained with observing the behaviour of the increment displacements for the fluid dynamics grid at the load point of $\mathbf{F}_1$ and the global change of the volume shown in Figure 6.4.

Obviously, the most significant displacement is in z-direction - the direction of the applied forces. However, it can be seen that the incremental z-displacement and the global change of the volume vary in different ways. The change of the area of the

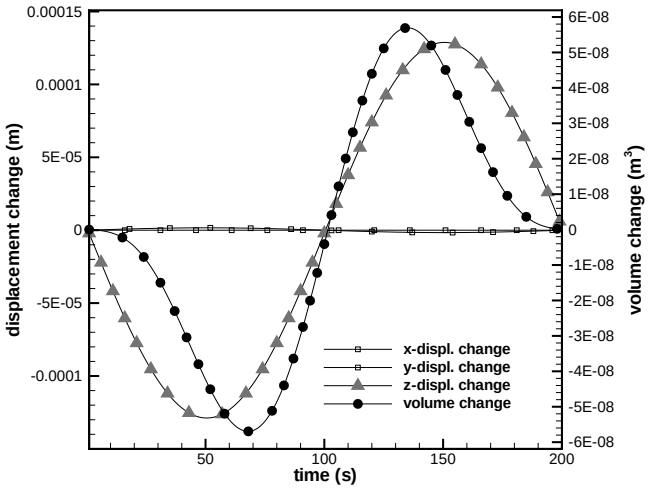



Figure 6.4: Increment displacements at the load point of force $\mathbf{F}_1$, global volume change

cross-section and hence the change of the outlet mass flux are not linearly connected to the change of the displacements. At the beginning and at the end of the pinching the change of the volume is much smaller than the change of the displacements. This is opposite to the case near time 100 s, when the increment displacements are smaller than the volume change. At time 100 s the increment displacements are zero as well as the change of the volume. Therefore, at this moment the outlet mass flux is equal to the inlet mass flux.

In Figure 6.5 (left) the change of the fluid velocity z-component is presented at different time steps in the intersection of the pipe with the plane y = 0. As it is expected, when the fluid-structure interaction is taken into account, the fluid flow follows the grid movement. In the beginning the z-velocity component is nearly zero. However, due to the elastic walls movement two areas with opposite signs develop. Later, these exchange their signs, because the pipe shape starts returning to its initial state (after the time 100 s).

The change of the pressure in the pipe can be seen in Figure 6.5 (right). It slowly increases as the pinching forces increase and is highest when the global change of the volume is the biggest (time 70 s). Then it starts decreasing until the forces become maximal (time 100 s). With the reduction of the forces the pressure again begins increasing until it reaches the initial distribution (at time 200 s the same as that at time 0 s).

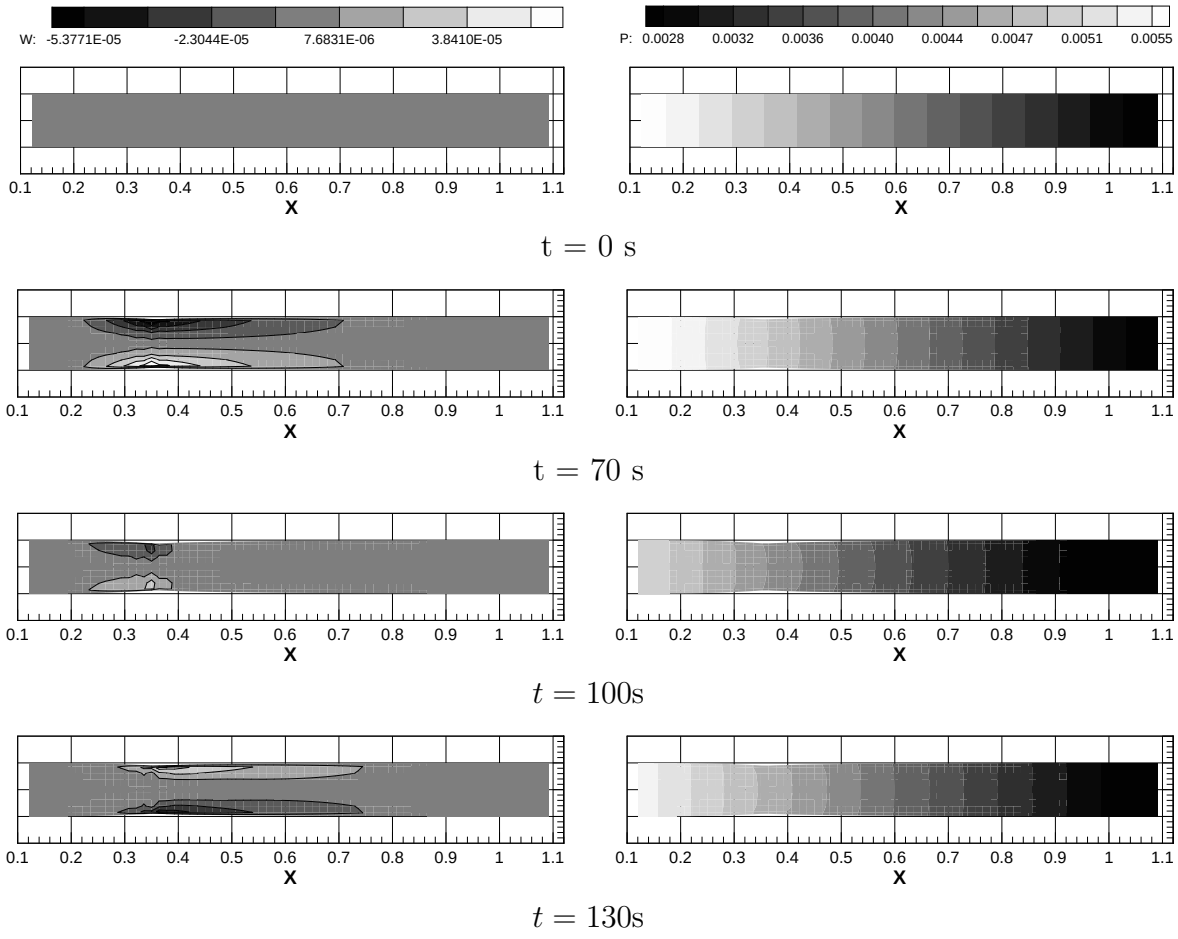



Figure 6.5: z-component of fluid velocity (left) and pressure (right) in the plane y=0 at different times

### 6.1.2 Example 2: Flow in a T-junction of elastic pipes

The presented coupling method can also be used to model the fluid-structure interaction in complex fluid domains. Here, a steady state of a laminar flow within a $90^0$ T-junction of two pipes with fluid-structure interaction is considered. The schematic description of the problem is given on Figure 6.6.

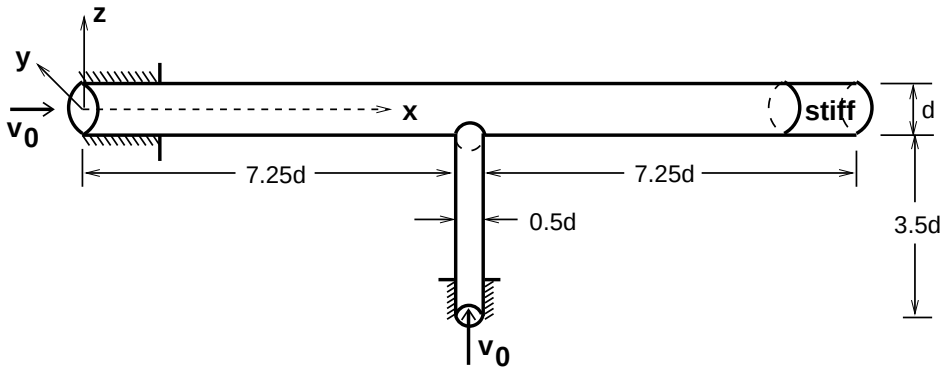



Figure 6.6: T-junction of two pipes - problem description

The cross-section diameter of the first pipe is d = 0.1 m and the one of the connected second pipe is 0.5 d. Additionally, the thickness of the pipe-walls is chosen to be 0.0004 m. The fluid velocity is assumed to have a constant parabolic profile with a maximal value $\mathbf{v}_0 = 0.02$ m/s at both inlets and the fluid viscosity is chosen to be 2.E-5 Pa/s. These flow parameters lead to Reynolds number Re = 100 for the first pipe.

The following boundary conditions are applied. Constant inlet flows are obtained by fixing the pipes along a certain distance at the inlets. To assure a constant cross-section shape at the outlet, a small ring at the end of the first pipe is assumed to be made of a stiff enough material with elasticity module $E = 10^9$ kN/m$^2$. The rest of the pipes consists of a linear isotropic elastic material with elasticity module E = 10 kN/m$^2$ and Poisson's ratio $\nu = 0.4$. In addition the pressure at the outlet is chosen to be the same as the static pressure throughout the computations. To investigate the pure fluid-structure interaction the external forces acting on the T-junction are neglected.

In order to use the fluid solver FASTEST-3D the fluid domain is discretised into finite volumes. A second-order spatial accuracy is achieved by applying the CDS to the convective terms. The finite elements grid for the elastic part of the T-junction is extracted from the surface fluid grid so that the grid nodes of the two grids coincide. Similar to the example presented in section 6.1.1 the 4-nodes shell elements with 6 degrees of freedom are adopted. Hence, a second-order spatial discretisation for the FSI problem is achieved. Because a steady state is searched, a first-order time-stepping scheme for the fluid is enough. The structure is solved quasi-steady due to the linear elasticity and small displacements. For the fluid part time-step 1 s is chosen.

Since the equilibrium between the position of the pipes and the fluid inside is desired,

at first the fluid is developed to a steady flow into the T-junction. Then the fluid–structure interaction is taken into account. The main forces acting on the structure are determined by the pressure on the elastic pipe walls. They are dominating the shear forces that are small because of the small fluid velocity. In Figure 6.7 the initial pressure distribution for the fluid-structure interaction problem is presented.

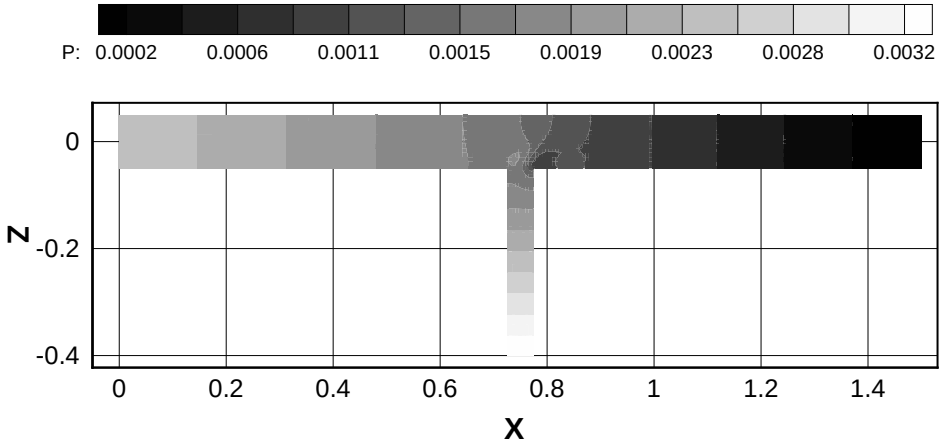



Figure 6.7: Pressure in the plane y = 0

An area of low pressure appears exactly after the two inlet flows have mixed. This results in a momentum which benches the first pipe and moves its end up. Hence, the most displaced points on the elastic walls are the ones at the outlet. In Figure 6.8 the displacements of the point at the outlet with the biggest z-coordinate during the simulation are given.

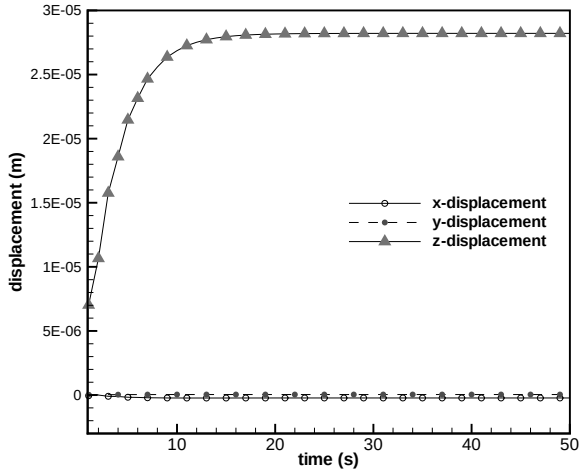



Figure 6.8: Displacements at a point at the outlet

It can be seen that this point moves mainly in z-direction, which coincides with the direction of the flow within the thinner pipe. The outlet part slowly moves up to

$2.8{\cdot}10^{-5}$m and then the equilibrium between flow and structure is achieved. The deformed shape of the T-junction can be seen in Figure 6.9. Here, the displacements are scaled with a factor 2000 for better visualisation.

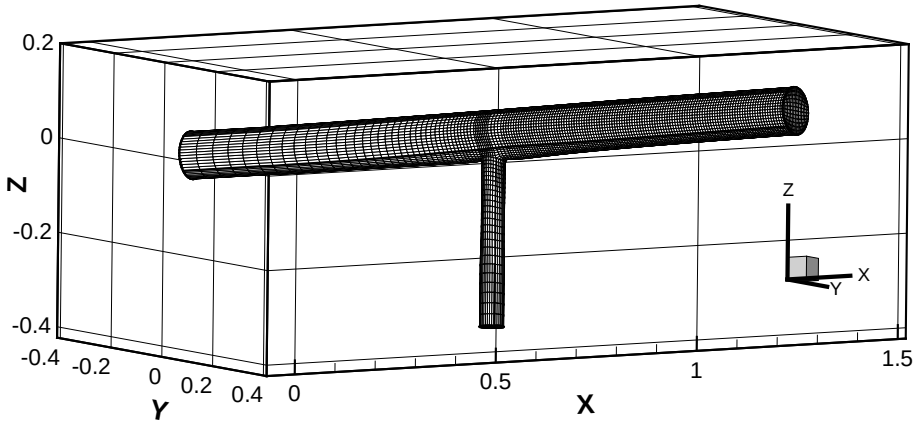



Figure 6.9: Deformed T-junction of two pipes

These displacements lead to a change in the fluid behaviour. The absolute value of the fluid-velocity vector increases, while the pressure decreases compared to the corresponding values obtained without considering fluid-structure interaction. However, qualitatively the values are distributed along the pipes in a similar way. The received absolute value of the fluid velocity can be seen in Figure 6.10.

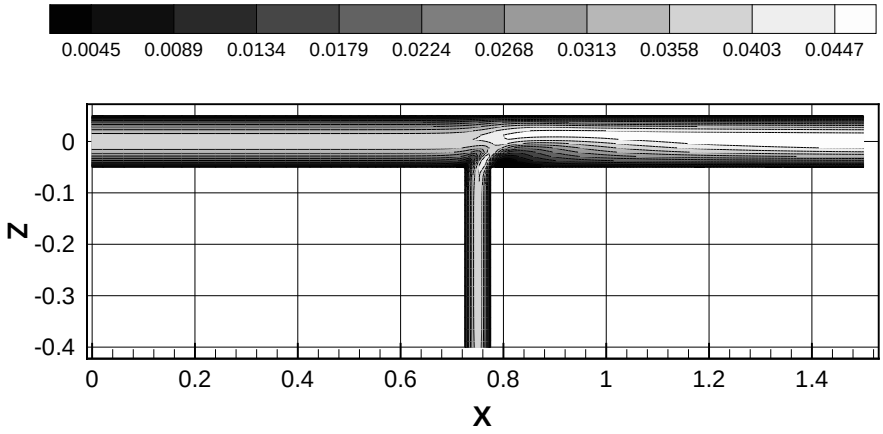



Figure 6.10: Fluid velocity absolute value in the plane y = 0