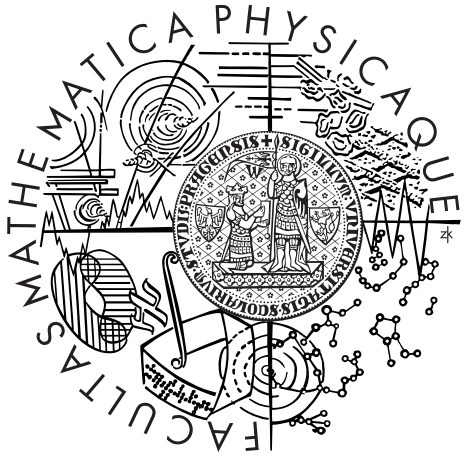
Univerzita Karlova v Praze
Matematicko - fyzikální fakulta

DIPLOMOVÁ PRÁCE



Tereza Zábojníková

Modelování proudění krve v geometrii aneuryzma

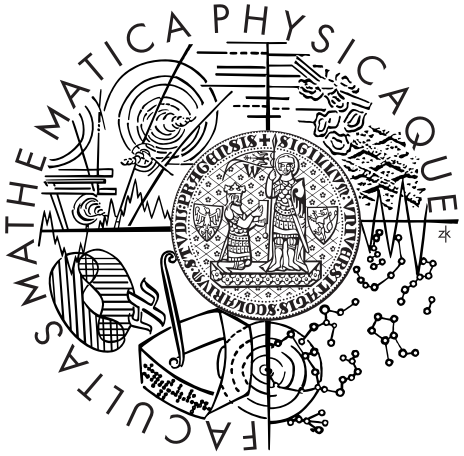
Matematický ústav

Vedoucí diplomové práce:	RNDr. Jaroslav Hron, Ph.D.
Studijní program:	Fyzika
Studijní obor:	Učitelství fyziky pro SŠ v kombinaci s odbornou fyzikou (Matematické modelování ve fyzice a technice)

Praha 2015

Charles University in Prague
Faculty of Mathematics and Physics

MASTER THESIS



Tereza Zábojníková

Modelling of blood flow in aneurysm geometry

Mathematical Institute

Supervisor of the master thesis: RNDr. Jaroslav Hron, Ph.D.
Study programme: Physics
Specialization: Mathematical Modelling in Physics and Technology

Prague 2015

I thank to Miguel Ángel Fernández, my consultant from Université Pierre et Marie Curie, Paris 6, who has spent a lot of time consulting my work during my Erasmus in Paris and who has gave me the original idea of this thesis, my supervisor Jaroslav Hron who helped me to finish my work started in Paris and my parents who have supported me during my studies.

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular the fact that the Charles University in Prague has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 paragraph 1 of the Copyright Act.

In ..... on ..... Author signature

Název práce: Modelování proudění krve v geometrii aneurysma

Autor: Tereza Zábojníková

Katedra: Matematický ústav UK

Vedoucí diplomové práce: RNDr. Ing. Jaroslav Hron, Ph.D., Matematický stav UK

Abstrakt: Cílem této práce bylo nalézt stabilní schéma, které by řešilo Stokesův problém tekutiny, ve které je ponořená elastická pevná látka. Narozdíl od většiny schémat řešících interakci pevné látky s tekutinou, naše schéma nevyžaduje, aby na sebe sítě pevné látky a tekutiny navazovaly. Omezili jsme se na dvoudimenzionální oblast pro tekutinu, ve které je ponořena jednodimenzionální elastická pevná látka. Pro popis interakce jsme použili metodu vnořené hranice (Immersed boundary method). Na začátku jsme považovali pevnou látku za nehmotnou. Upravili jsme již existující schéma řešící takovýto problém tak, aby bylo nepodmíněně stabilní, což jsme matematicky dokázali a numericky otestovali. Poté jsme navrhli modifikaci schématu tak, aby pevná látka již měla nějakou hmotnost, a též dokázali jeho nepodmíněnou stabilitu. Navržená schémata jsme implementovali v programu Freefem++ a otestovali jejich chování na geometrii podobné aneurysma. Vyzkoušeli jsme také chování navržených schémat v případě, kdy se rostoucí aneurysma dotkne překážky, například kosti (s no-slip podmínkou na okraji).

Klíčová slova: interakce tekutiny s pevnou látkou, metoda ponořené hranice, proudění krve

Title: Blood flow in geometry of aneurysm

Author: Tereza Zábojníková

Department: Mathematical Institute of Charles University

Supervisor of the master thesis: RNDr. Jaroslav Hron, Ph.D., Mathematical Institute of Charles University

Abstract: The aim of this work is to find a stable scheme which would solve the Stokes problem of the fluid flow, in which an elastic structure is immersed. Unlike most of the schemes solving fluid-structure interaction problems, in our scheme meshes of fluid and structure do not have to coincide. We have restricted ourselves to two-dimensional domain occupied by fluid with one-dimensional immersed structure. To describe a fluid-structure interaction, we have used an Immersed boundary method. At first we consider the strucure to be massless. We have modified an existing scheme and made it unconditionally stable, which was mathematically proven and numerically tested. Then we have proposed a modification where the structure is not massless and also proved the unconditional stability in this case. The proposed schemes were implemented using the Freefem++ software and tested on aneurysm-like geometry. We have tested the behavior of our scheme in case when the qrowing aneurysm touches an obstacle, for example a bone (with no-slip condition on the bone boundary).

The proposed schemes were implemented using the Freefem++ software and tested on aneurysm-like geometry

Keywords: fluid-structure interaction, immersed boundary method, blood flow

Contents

Chapter 1

Introduction

1.1 Aneurysms and computational hemodynamics

The motivation to study computational hemodynamics (a science about blood flow) is clear. Almost half of all deceases in Europe is caused by vascular deseases and 2/3 of all dispenses of the public health care are spent because of them.

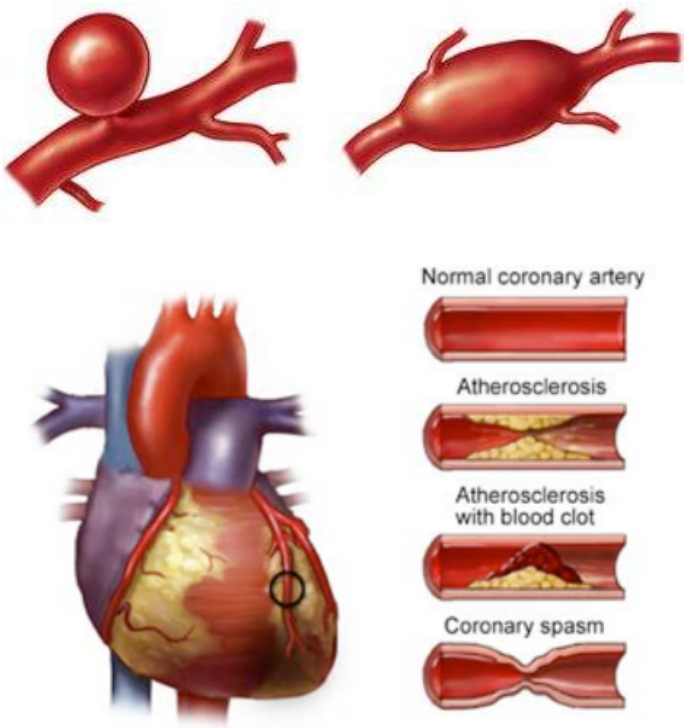


Figure 1.1: vascular deseases: aneurysme and atherosclerose [8]

In recent years, the blood flow modelling has matured greately. One reason is the development of imaging techniques, the second one is the huge development in the field of scientific computation (by that we mean the bigger performance of the computational clusters and also agorithmic progress).

The aim is to develop a computational method which would be quick (in some cases of vascular deseases the time is really important and we simply can't

wait one month for the cluster to show us the patient-specific result), stable and accurate. Once we would have it, we could scan the problematic place in the patient's vascular system, and the information about pressure and flow could help us to decide what kind of treatment we should use.

For example in case of aneurysm, there are two possibilities how to exclude the aneurysm from a vascular system and prevent it from bursting. We can either fill the space of aneurysm by coil, or clip the neck of aneurysm by a steel clip to seal it off from the parent vessel.

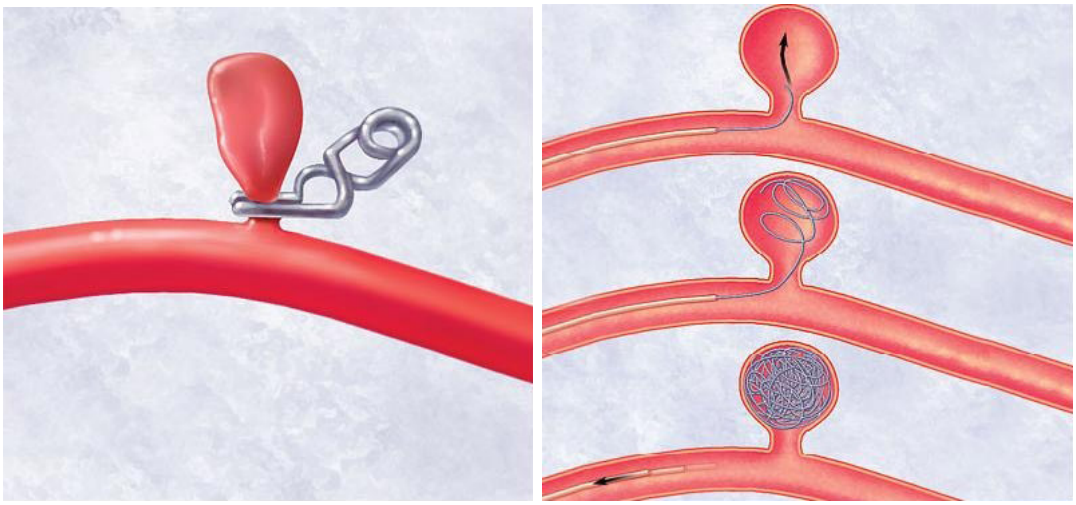


Figure 1.2: Two possibilities to treat aneurysms: clipping and coiling [1]

Deciding which treatment is more convenient in specific case could be simlified by running a simulation on patient-specific model.

Over the years not only the development of computers and clusters shortened the time needed to obtain a result. In 11 years, thanks to various algorithmic modifications, the computational time has reduced 50 times (solving the same problem on the same computer, but with some algorithmic modifications)! (see [8])

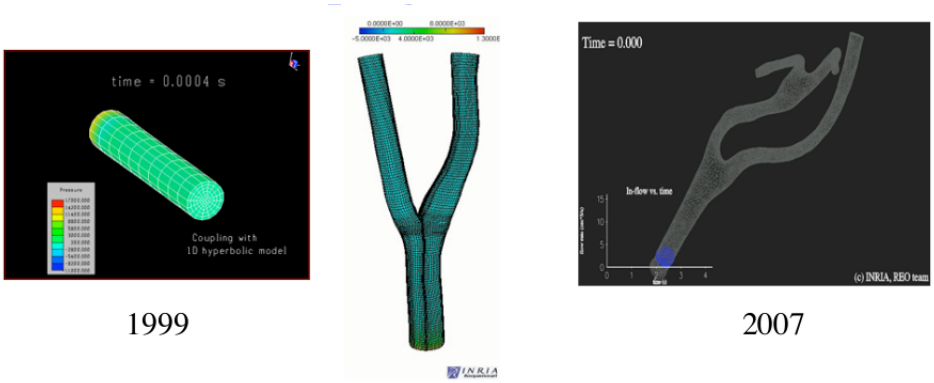


Figure 1.3: Progress of computational hemodynamics in INRIA, presented in course of École Polytechnique and Université Paris 6 [8]

Year	2001	2003	2007	2012
Computational time	50	20	4	1

Table 1.1: Reduction of a computational time in INRIA, presented in course of École Polytechnique and Université Paris 6 [8]

Apart from stability and time-consumeness, other problem of blood-flow modelling is mesurement of the boundary conditions. Having stable scheme and algorithm to solve it quickly doesn't mean that we can build a patient-specific model and solve it correctly. The problem is that we can't measure the boundary conditions on the borders of our model. We struggle either with the spatial resolution of in vivo velocimetry, or the invasiveness of the measuring procedure.

The velocity in arteries can be measured by laser doppler anemometry (LDA), which is non-invasive, but can be used only near the surface, on the skin capillars. Another possibility is magnetic resonance imaging (MRI), specifically magnetic resonance angiography, which can evaluate the velocity from the phase-shift of manetic signal, but has certain spatial resolution limitations.

On the other hand, measuring pressure on vessel limits is not a problem. Therefore some simplified 1D and 0D models (see [16]), which work with less variables, were developed to substitute absenting boundary conditions. Unlike a 3D model, in case of one-dimensional approximation, we work with flow, average pressure and area of the vessel cross-section. The Vessel is approximated as a cylinder. In case of zero-dimensional model, we work with pressure drop and flow, which is similar as working with voltage and electric current in case of an electric circuit. 1.4.

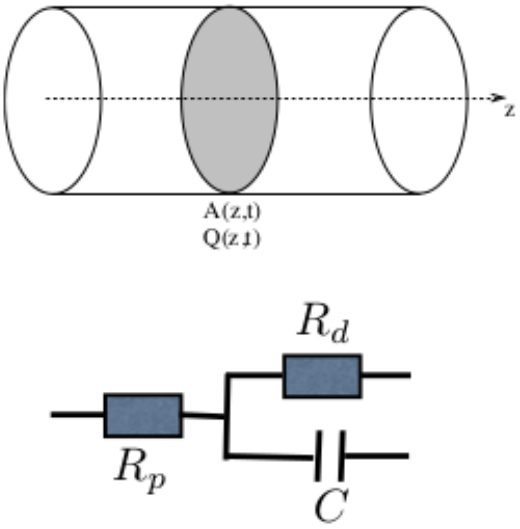


Figure 1.4: Simplified models [8]

Those simplified models can be used as a substitution of boundary conditions, or to model the whole human blood-circulation (see picture 1.5).

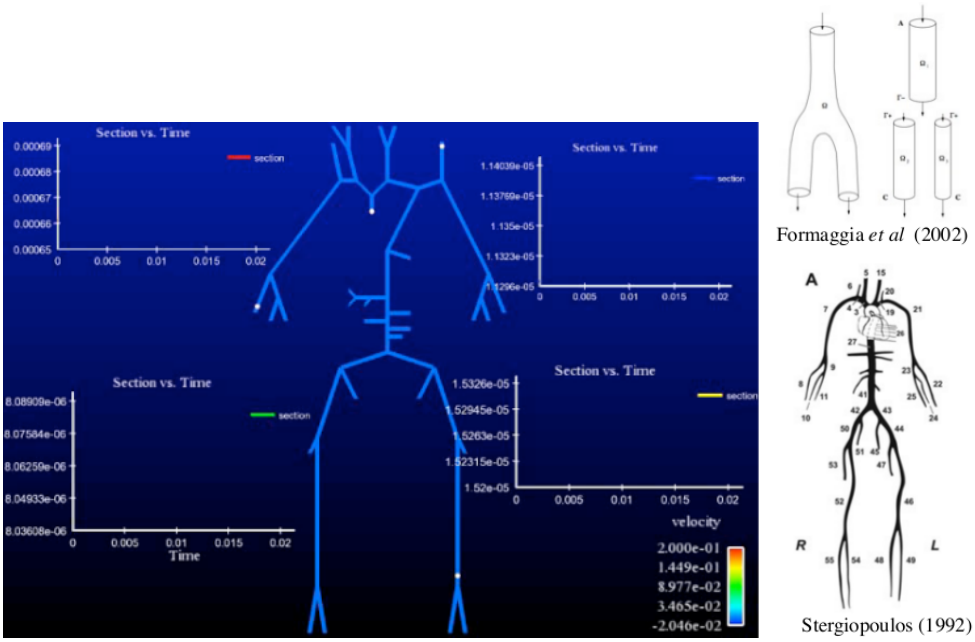


Figure 1.5: 1D Models from Stergiopoulos, Formaggia and Lombardi [7]

1.2 Blood flow modelling

1.2.1 Fluid-Structure interaction

The blood flow in aneurysms physical background consist from mechanics of the fluid (blood) and mechanics of solid materials (vessel's walls). While modelling this problem, we have to deal with the fact, that for fluids we use naturally Eulerian framework, while for the solid structure the Lagrangian framework is more natural. One possible solution is ALE (Arbitrary Lagrangian Eulerian) formulation of equations [12], or we can separate the problem to Euleruan and Lagrangian part as in the Immersed boundary method (IBM) [11].

1.2.2 Arbitrary Lagrangian Eulerian formulation

ALE formulation of the problem lie in the way how the fluid "feels" the structure deformation. The structure displacement causes the deformation of the fluid mesh, which presents new term in fluid equation. We can solve the problem monolithically (fluid and structure are solved at the same time) or separately (At first we find a fluid velocity and pressure which gives us the boundary conditions for structure, then we find structure displacement which gives the boundary conditions for fluid). The monolithical approach is numerically stable, but really computationally demanding. The separate way is less time-consuming. It is either strongly (multiple subiterations in each step) or loosely (only one fluid and structure iteration per time step) coupled. For certain combination of parametres (for example close fluid and structure densities, which is unfortunately the case of hemodynamics) the strongly coupled scheme demands a lot of subiterations, thus it is also time consuming. For the same combination of parameters the loosely coupled one does not converge at all.

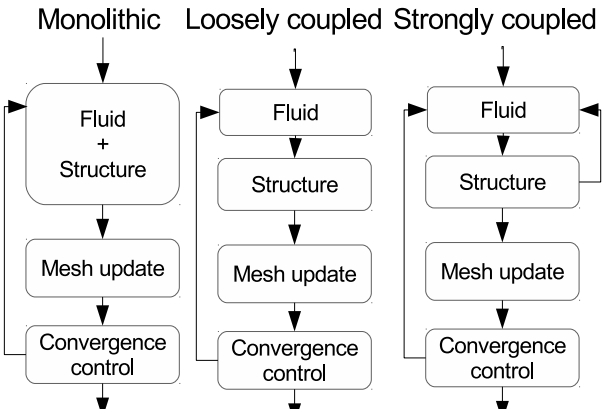


Figure 1.6: manolithic and separate approach

In [15] it was shown on really simple 2D model of large artery using the simplified Stokes equation, why these instabilities arise. It was shown, that loosely coupled scheme is unconditionally unstable if the fluid and structure density ρ_f, ρ_s, structure thickness h_s satisfy:

$$\frac{\rho_s h_s}{\rho_f \mu_{max}} < 1 \tag{1.1}$$

Where μ_{max} is the constant depending on the length and the radius of the artery-representing cylinder. The "instability" condition depends on the values, which seemed problematical from the numerical tests even with the more complicated models (Navier-Stokes).The scheme working for thick and short tube did not work for long and thin tubes (as $\frac{R}{L} \to 0 : \mu_{max} \approx \frac{2L^2}{\pi^2 R}$).

While in aerodynamics the loosely coupled schemes are perfectly usable because of the big $\frac{\rho_s}{\rho_f}$ ratio, in hemodynamics the ratio $\frac{\rho_s}{\rho_f} \approx 1$, so the instability condition may be satisfied.

As described above, studying simplified models can be really helpful when we want to understand the more complicated ones.

1.2.3 Immersed boundary

In Immersed Boundary Method (IBM), the fluid-structure interaction is not modeled by mesh deformation. The computational domaine Ω is divided to a Ω_f occupied by fluid and Ω_s occupied by solid structure. The fluid "feels" the structure via a force term, that could be expressed like:

$$F(x) = H_s(x) f(x) \tag{1.2}$$

Where $H_s(x)$ is a characteristic function of Ω_s domain.

$$H_s(x) = \begin{cases} 0 & \text{if } x \in \Omega_f, \\ 1 & \text{if } x \in \Omega_s. \end{cases}$$

In [11], a following model was used for computing a blood flow in the realistic 3D aneurysm geometry:

$$\begin{aligned} \frac{\partial \vec{u}}{\partial t} + \vec{u}.\nabla\vec{u} &= -\nabla p + \frac{1}{Re}\Delta\vec{u} + \vec{f} \\ \operatorname{div}\vec{u} &= 0 \\ \vec{f} &= -\frac{1}{\epsilon}H_s\vec{u} \end{aligned} \tag{1.3}$$

Where $\epsilon \approx 10^{-10}$ is a forcing parameter, u is a fluid velocity and p is a pressure. Re is the Reynolds number.

When implementing ALE formulation, the vertices of fluid mesh and the structure mesh has to coincide on the boundary. With immersed boundary method, we have two possibilities, the meshes may or may not match, it depends what scheme we use.

In [4], a scheme with simple structure update was used, but it was stable only under certain condition for time step, structure and fluid discretisation and elasticity model constant. We will propose two modifications, which make the scheme more stable.

Chapter 2

Basic Scheme

The immersed boundary method consist of adding a force effect to a fluid governing equation. While in elastic solid material the forces between the neighbour particles are big, this bond in fluid is looser, so the force term needs to compensate this difference at the place occupied by structure. When we work with a solid with a certain non-zero thickness, it can be quite complicated to express this additional force, but when we reduce the structure to a thin membrane, the mathematical expression of the force becomes easier and more rigorous. Following model was described in [4]. At first, we implemented the scheme of Daniele Boffi [4], and then modify it to obtain a scheme which would be more stable.

2.1 Massless membrane

Let Ω be 2 or 3 dimensional domain infilled by a fluid in which we immerse a thin solid elastic membrane. In 2-dimensional case the membrane is 1-dimensional line, in 3D case it is a 2-dimensional surface. We will consider the fluid as viscous, incompressible Newtonian fluid, so we can describe its behavior by Navier-Stokes equations:

$$\begin{aligned} \rho_f(\frac{\partial \vec{u}}{\partial t} + \vec{u}.\nabla\vec{u}) - \mu\triangle\vec{u} + \nabla p = \vec{F} \text{ in } \Omega \times [0,T] \\ \operatorname{div}\vec{u} = 0 \text{ in } \Omega \times [0,T] \end{aligned} \tag{2.1}$$

Where ρ_f is density of fluid, $\vec{u}$ its velocity, μ is viscosity, p is the pressure and $\vec{F}$ is density of the body force acting on the fluid. To simplify the problem, we neglect all body forces which are not related to the immersed boundary effects. We assume that ρ_f and μ are constants.

We will consider the structure as thin, incompressible, elastic membrane. Let's denote q the initial coordinates of the structure, thus labelling the solid particle.

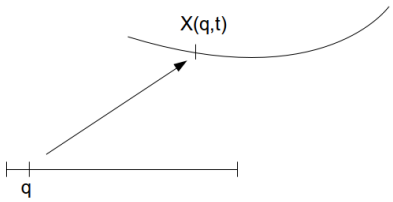


Figure 2.1: Parametrization of structure

$X(q,t)$ is its position in a time t. We can express an elastic additional force by following term:

$$\vec{F}(x,t) = \int_{\Omega_0} \vec{f}(q,t)\delta(x - X(q,t))dq \tag{2.2}$$

Where $q \in \Omega_0$ is the initial position of structure (which marks structure particles) $x \in \Omega$ is the position where we want to evaluate the force and $X(q,t) \in \Omega_s \subset \Omega$ is the current position of the particle marked by q in a time t.

δ is 2D or 3D Dirac distribution, which helps us to pass from Lagrangian to Eulerian formulation. The term above introduces a force generated by the particles marked as a part of a structure. The elastic properties of solid are expressed by $f(q,t)$.

Each structure particle is subjected to a "no-slip" condition, therefore the particles are moved along the streamlines of the fluid:

$$\frac{\partial X}{\partial t} = \vec{u}(X(q,t),t)(q,t) \text{ in } \Omega_0 \times [0,T] \tag{2.3}$$

Original formulation of immersed boundary method is as follows:

Definition 1 (Problem 1)**.** *Find* $\vec{u}(x,t), p(x,t), X(q,t)$ *which satisfy:*

$$\rho_f(\frac{\partial \vec{u}}{\partial t} + \vec{u}.\nabla\vec{u}) - \mu\triangle\vec{u} + \nabla p = \vec{F} \qquad in\ \Omega \times [0,T] \tag{2.4}$$

$$\text{div}\,\vec{u} = 0 \qquad in\ \Omega \times [0,T] \tag{2.5}$$

$$\frac{\partial X}{\partial t} = \vec{u}(X(q,t),t) \qquad in\ \Omega_0 \times [0,T] \tag{2.6}$$

$$\vec{F}(x,t) = \int_{\Omega_0} \vec{f}(q,t)\delta(x - X(q,t))dq \qquad in\ \Omega \times [0,T] \tag{2.7}$$

$$\vec{u}(x,t) = 0 \qquad on\ \partial\Omega \times [0,T] \tag{2.8}$$

$$\vec{u}(x,0) = \vec{u}_0(x) \qquad in\ \Omega \tag{2.9}$$

$$X(q,0) = X_0(q) \qquad in\ \Omega_0 \tag{2.10}$$

Equations (2.8), (2.9) are the boundary conditions and initial conditions for Navier-Stokes, (2.10) is an initial condition for structure-update equation (2.6).

2.2 Model with simplified elasticity

To see the stability of an immersed boundary model and to focus mainly on instabilities arising from immersed boundary formulation, we restrict ourselves to a very simple model of elasticity, just 2 dimensional domain Ω, containing the 1-dimensional immersed structure, which is represented as a closed parametrized curve Γ_t. And we follow the aproach from [4].

$$X(s,t) : 0 < s < L, X(0,t) = X(L,t) \tag{2.11}$$

Where L is length of the structure in the unstressed state. Here s represents the Lagrangian coordinate (marked as q above).

Now we can express the force term from a generalized Hook law for boundary tension:

$$T = \sigma\left(\left|\frac{\partial X}{\partial s}\right|; s, t\right). \tag{2.12}$$

The distance between two neigbour structure points on Γ_t are given by $|dX| = |\frac{\partial X}{\partial s}ds|$, in the reference configuration this distance is $|dX| = |ds|$. The elastic forces will be a function of relative prolongation, which is the ratio of these two values.

The direction of the tension is given by the direction of the curve, more precisely by the unit tangent τ.

$$\tau = \frac{\partial X/\partial s}{|\partial X/\partial s|} \tag{2.13}$$

The elastic force acting on a structure segment determined by a reference poits is defined:

$$(T\tau)(b,t) - (T\tau)(a,t) = \int_a^b \frac{\partial}{\partial s}(T\tau)(s,t)ds. \tag{2.14}$$

The structure being massless, there is no inertial term, therefore the force that is feeled by the fluid is exactly the elastic force. Considering the easiest model of elasticity with zero initial deformation and stress tensor proportional to $|\partial X/\partial s|$, we obtain elasticity force:

$$\vec{f} = \frac{\partial}{\partial s}\left(K|\partial X/\partial s|.\frac{\partial X/\partial s}{|\partial X/\partial s|}\right) = K\frac{\partial^2 X}{\partial s^2} \tag{2.15}$$

Even the model is really simple, it is applicable to a large class of problems, for example it was used by C.S. Peskin in [3] to model a blood flow in a heart.

2.3 Weak formulation

In order to discretise problem 1 with the finite element method, we need to express $\langle \vec{F}, \vec{v} \rangle$, where $\vec{v}$ is a test function from a chosen finite element space. We recall a lemma from [4]:

Lemma 1 (Force term). *Let's assume that* $\forall t \in [0,T]$ *the immersed boundary* Γ_t *is Lipchitz continuous and that* $\vec{f} \in L^2([0,L]\times(0,T))$. *Then* $\forall t \in (0,T)$ *the force density* $\vec{F}(t)$ *is a distribution in* $H^{-1}(\Omega)$ *defined:*

$$ {}_{H^{-1}}\langle \vec{F}(t), \vec{v} \rangle_{H_0^1} = \int_0^L \vec{f}(s,t).\vec{v}(X(s,t))ds \quad \forall t \in (0,T) \tag{2.16}$$

The variational formulation of problem 1 is then following:

Definition 2 (Problem 2). *Let* $\vec{u}_0 \in H_0^1(\Omega)^2$ *and* $X_0 : [0,L] \to \Omega$ *be given. Find* $(\vec{u}(t), p(t)) \in H_0^1(\Omega)^2 \times L_0^2(\Omega)$ *and* $X : [0,L]\times(0,T) \to \Omega$ $\forall t \in (0,T)$ *such that following equations are satisfied:*

$$\rho_f \left(\frac{\partial}{\partial t}(\vec{u}(t), \vec{v}) + (\vec{u}(t).\nabla\vec{u}(t), \vec{v}) \right) + \mu(\nabla\vec{u}(t), \nabla\vec{v})$$

$$-(\operatorname{div}\vec{v}, p) = \langle \vec{F}, \vec{v} \rangle \qquad \forall \vec{v} \in H_0^1(\Omega)^2 \tag{2.17}$$

$$(\operatorname{div}\vec{u}, q) = 0 \qquad \forall q \in L_0^2(\Omega) \tag{2.18}$$

$$\langle \vec{F}, \vec{v} \rangle = \int_0^L K \frac{\partial^2 X}{\partial s^2} \vec{v}(X(s,t))ds \qquad \forall \vec{v} \in H_0^1(\Omega)^2 \tag{2.19}$$

$$\frac{\partial X}{\partial t} = \vec{u}(X(q,t),t) \qquad \forall s \in [0,L] \tag{2.20}$$

$$\vec{u}(x,t) = 0 \qquad on\ \partial\Omega \times [0,T] \tag{2.21}$$

$$\vec{u}(x,0) = \vec{u}_0(x) \qquad \forall x \in \Omega \tag{2.22}$$

$$X(s,0) = X_0(s) \qquad \forall s \in [0,L] \tag{2.23}$$

$$X(0,t) = X(L,t) \qquad \forall t \in [0,T] \tag{2.24}$$

2.4 Finite elements discretisation

Let T_h be a triangulation of Ω with the maximal diameter of element denoted as h. We consider two finite-dimensional sub-spaces $V_h \subseteq H_0^1(\Omega)^2$ and $Q_h \subseteq L_0^2(\Omega)$. We wanted to use an easy-to-implement Lagrange P1-P1 finite elements:

Definition 3 (Discrete functional spaces of pressure and velocity). *The initial functional spaces, which we use when we pose the weak fromulation of Stokes problem:*

$$V = (H_0^1(\Omega))^2$$

$$Q = L^2(\Omega)$$

Discretised subspaces for P1-P1 Lagrange finite elements:

$$V_h = \{v = (v_x, v_y), v_x, v_y \in C(T_h), \forall T \in T_h : v_x, v_y|_T \in P_1(T)\}$$

$$Q_h = \{q \in C(T_h), \forall T \in T_h : q|_T \in P_1(T)\}$$

The Stokes problem with these two subspaces V_h, Q_h does not satisfy Lax-Milgram theorem, so we have to introduce a stabilization, which is described in the following section.

2.4.1 Stabilization

When we were deciding which finite elements we should use, we started from the analysis of the Stokes problem [9].

Definition 4 (Weak formulation of Stokes problem)**.** *Find $\vec{u} \in V, q \in Q$, so that $\forall \vec{v} \in V$ and $p \in Q$ and the following equations are satisfied:*

$$\mu \int_\Omega \nabla \vec{u} \nabla \vec{v} dx - \int_\Omega p \mathrm{div}\, \vec{v} dx = a(\vec{u}, \vec{v}) + b(p, \vec{v}) = \int_\Omega \vec{F} \vec{v} dx \tag{2.25}$$

$$\int_\Omega q \mathrm{div}\, \vec{u} dx = b(q, \vec{u}) = 0 \tag{2.26}$$

Theorem, which implies well-posedness of the weak formulation of certain problem is Lax-Milgram [9]:

Theorem 2 (Lax-Milgram)**.** *Let $(X, ||\cdot||_X)$ be a Hilbert space and $A(\cdot,\cdot)$ be a bilinear form on $X \times X$. Let $F \in X'$. If the following propositions hold true:*

- *$A(\cdot,\cdot)$ is continuous:*

$$\exists M \in \mathbb{R}, M > 0 : \forall u, v \in X, A(u, v) \leq M ||u||_X ||v||_X$$

- *$A(\cdot,\cdot)$ is coercive:*

$$\exists \alpha > 0 : \forall v \in X, A(v, v) \geq \alpha ||v||_X^2$$

then there exists unique $u \in X$ such that $\forall v \in X$

$$A(u, v) = \langle F, v \rangle$$

and

$$||u||_X \leq \frac{||F||_{X'}}{\alpha}$$

In the case of discretised 2D Stokes, $(X, ||\cdot||) = (V_h \times Q_h, ||\cdot||_X = \sqrt{|\cdot|^2_{(H^1)^2} + ||\cdot||^2_{L^2}})$,

$$A((\vec{u}_h, p_h), (\vec{v}_h, q_h)) = \mu \int_\Omega \nabla \vec{u}_h \nabla \vec{v}_h - \int_\Omega p_h \mathrm{div}\, \vec{v}_h + \int_\Omega q_h \mathrm{div}, \vec{u}_h$$

$$\langle F, (\vec{v}_h, q_h) \rangle = \langle f, \vec{v}_h \rangle$$

- continuity (From Cauchy-Schwarz)

$$A((u_h, p_h), (v_h, q_h)) \leq \sqrt{||u_h||_V + ||p_h||_Q} \sqrt{||v_h||_V + ||q_h||_Q}$$

- coercivity is not satisfied, because:

$$A((u_h, p_h), (u_h, p_h)) = \mu |u_h|^2$$

Since the term $||p_h||_0$ is absent in above identity, the problem is not coercive and we can't use the usual framework of Lax-Milgram. We would like to modify our problem so that it would be coercive. One possibility is to use Brezzi-Pitkaranta modification [14]:

Definition 5 (Weak formulation of stabilised Stokes problem)**.** *Find $\vec{u_h} \in V_h, p_h \in Q_h$, so that $\forall \vec{v_h} \in V_h$ and $q_h \in Q_h$.*

$$\mu \int_\Omega \nabla \vec{u_h} \nabla \vec{v_h} dx - \int_\Omega p \mathrm{div}\, \vec{v_h} dx = \int_\Omega \vec{F} \vec{v_h} dx \tag{2.27}$$

$$\int_\Omega q_h \mathrm{div}\, \vec{u}_h dx - \alpha_{mod} \frac{h^2}{\mu} \nabla p_h \nabla q_h dx = 0 \tag{2.28}$$

According to [14] the stabilization works with any α_{mod} positive. The stabilization term makes the Stokes problem coercive, therefore for the discrete version, we can use the framework of Lax-Milgram theorem.

2.4.2 Pseudo-compressibility method

While solving the Stokes problem, the pressure will be defined except for a constant. Use of the Pseudo-Compressibility method will cause, that the pressure will be well defined and we won't have the problem to determine the constant. This method is described in [17]. The idea is to relax the incompressibility constraint with perturbation-parameter ε_{pc} in certain way:

Definition 6 (Discretised weak formulation of Stokes problem with relaxed incompressibility constraint)**.** *Find $u^{\vec{\varepsilon_{pc}}}, p^{\varepsilon_{pc}}$, so that $\forall \vec{v} \in V, q \in Q$ the following equations are satisfied:*

$$\mu \int_\Omega \nabla \vec{u}^{\varepsilon_{pc}} \nabla \vec{v} dx - \int_\Omega p^{\varepsilon_{pc}} \mathrm{div}\, \vec{v} dx = a(\vec{u}^{\varepsilon_{pc}}, \vec{v}) + b(p^{\varepsilon_{pc}}, \vec{v}) = \int_\Omega \vec{F}, \vec{v} dx \tag{2.29}$$

$$\int_\Omega q \mathrm{div}\, \vec{u} dx + \varepsilon_{pc} \int_\Omega p^{\varepsilon_{PC}} q dx = b(q, \vec{u}) + stab(p^{\varepsilon_{pc}}, q) = 0 \tag{2.30}$$

There are more possible ways of the relaxation (as described in [17], the one above which we used is called the penalty method.) We will use both methods (Brezzi-Pitkaranta and pseudo-compressibility), so our final discretised Stokes problem will be:

Definition 7 (Weak formulation of stabilised Stokes problem with relaxed incompressibility constraint)**.** *Find $\vec{u_h} \in V_h, p_h \in Q_h$, so that $\forall \vec{v_h} \in V_h$ and $q_h \in Q_h$.*

$$\mu \int_\Omega \nabla \vec{u_h} \nabla \vec{v_h} dx - \int_\Omega p \mathrm{div}\, \vec{v_h} dx = \int_\Omega \vec{F} \vec{v_h} dx \tag{2.31}$$

$$\int_\Omega q_h \mathrm{div}\, \vec{u}_h dx - \alpha_{mod} \frac{h^2}{\mu} \nabla p_h \nabla q_h dx + \varepsilon_{PC} \int_\Omega p_h q_h dx = 0 \tag{2.32}$$

We have used $\alpha_{mod} = 0.01$ and $\varepsilon_{PC} = 10^{-10}$.

2.4.3 Force-term adjustment

The structure membrane discretisation will be the set of distinct points s_i, $i = 0..m$, $s_0 = 0, s_m = L$ with $h_s = max_{i<m}|s_{i+1} - s_i|$. Let S_h be a piecewise linear finite element space on $[0, L]$ interval.

$$S_h = \{v_s \in C^0([0, L] \to \Omega) : v_s|_{(s_i,s_i+1)} \in P_1(s_i, s_i + 1)^2, \tag{2.33}$$
$$i = 0, \ldots, m - 1, v_s(0) = v_s(L)\} \tag{2.34}$$

With $X_h \in S_h$ being a piecewise linear function, we can simplify the force expression, because the derivative of X_h is constant. By reordering the terms in the sum and using the fact that $v(X_{h0}) = v(X_{hm})$ we obtain a final expression for force, which is easier to implement.

$$\langle \vec{F}_h, \vec{v} \rangle = \int_0^L K \frac{\partial^2 X_h(s,t)}{\partial s^2} \vec{v}(X_h(s,t)) ds \tag{2.35}$$
$$= \int_0^L -K \frac{\partial X_h(s,t)}{\partial s^2} \frac{\partial \vec{v}(X_h(s,t))}{\partial s} ds \tag{2.36}$$
$$= -\sum_{i=1}^{m} K \frac{\partial X_{hi}}{\partial s}(t) \int_{s_{i-1}}^{s_i} \frac{\partial \vec{v}(X_h(s,t))}{\partial s} ds \tag{2.37}$$
$$= -\sum_{i=1}^{m} K \frac{\partial X_{hi}}{\partial s}(t)(\vec{v}(X_{hi}) - v(X_{hi-1})) \tag{2.38}$$
$$= -\sum_{i=1}^{m} K \frac{\partial X_{hi}}{\partial s}(t)\vec{v}(X_{hi}) + \sum_{i=0}^{m-1} K \frac{\partial X_{hi+1}}{\partial s}(t)\vec{v}(X_{hi}) \tag{2.39}$$
$$= -\sum_{i=0}^{m-1} K \left(\frac{\partial X_{hi+1}}{\partial s}(t) - \frac{\partial X_{hi}}{\partial s}(t) \right) \vec{v}(X_{hi}) \tag{2.40}$$

Now we can write a discrete version of problem 2.

Definition 8 (Problem 3)**.** *Let $\vec{u}_{0h} \in V_h$ and $X_{0h} \in S_h$ be given. Find $(\vec{u}_h(t), p_h(t)) \in$*

$V_h \times Q_h$ *and* $X \in S_h$ $\forall t \in (0.T)$ *such that following equations are satisfied:*

$$\rho_f \left(\frac{\partial}{\partial t}(\vec{u}_h(t), \vec{v}_h) + (\vec{u}_h(t).\nabla \vec{u}_h(t), \vec{v}_h) \right) + \mu(\nabla \vec{u}_h(t), \nabla \vec{v}_h)$$

$$-(\operatorname{div} \vec{v}_h, p_h) = \langle \vec{F}_h, \vec{v}_h \rangle \qquad \forall \vec{v}_h \in V_h \tag{2.41}$$

$$(\operatorname{div} \vec{u}_h, q_h) = 0 \qquad \forall q_h \in Q_h \tag{2.42}$$

$$\langle \vec{F}_h, \vec{v}_h \rangle = -\sum_{i=0}^{m-1} K \left(\frac{\partial X_{hi+1}}{\partial s}(t) - \frac{\partial X_{hi}}{\partial s}(t) \right) \vec{v}(X_{hi}) \qquad \forall \vec{v}_h \in V_h \tag{2.43}$$

$$\frac{\partial X_{hi}}{\partial t} = \vec{u}(X_{hi}(t), t) \quad \forall i = 0, \ldots, m \tag{2.44}$$

$$\vec{u}_h(x, 0) = \vec{u}_{0h}(x) \qquad \forall x \in \Omega \tag{2.45}$$

$$X_{hi}(0) = X_{h0}(s_i) \quad \forall i = 1, \ldots, m \tag{2.46}$$

2.5 Time discretisation and linearization

To simplify the problem 3, we will neglect the convective term. For time discretisation, we have two possibilities. Either to use implicit backward Euler method (BE) or explicit forward Euler method (FE). Even though we are neglecting the non-linear term in Navier-Stokes (2.41) and (2.42), the implicit scheme would demand to solve a non-linear system due to the coupling (2.44) and (2.41) through force term (2.43)

Definition 9 (Problem 4)**.** *Let* $\vec{u}_{0h} \in V_h$ *and* $X_{0h} \in S_h$ *be given. Set* $\vec{u}_h^0 = u_{0h}$ *and* $X_h^0 = X_{0h}$ *then for* $n = 0..N-1$ *compute*

- *compute force term*

$$\langle \vec{F}_h^{n+1}, \vec{v}_h \rangle = -\sum_{i=0}^{m-1} K \left(\frac{\partial Y_{hi+1}}{\partial s}(t) - \frac{\partial Y_{hi}}{\partial s}(t) \right) \vec{v}(X_{hi}) \quad \forall \vec{v}_h \in V_h \tag{2.47}$$

- *find* $(\vec{u}_h^{n+1}, p_h^{n+1}) \in V_h \times Q_h$

$$\rho_f \left(\frac{\partial}{\partial t}(\vec{u}_h(t), \vec{v}_h) \right) + \mu(\nabla \vec{u}_h(t), \nabla \vec{v}_h)$$

$$-(\operatorname{div} \vec{v}_h, p_h) = \langle \vec{F}_h, \vec{v}_h \rangle \qquad \forall \vec{v}_h \in V_h \tag{2.48}$$

$$(\operatorname{div} \vec{u}_h, q_h) = 0 \qquad \forall q_h \in Q_h \tag{2.49}$$

- *update structure*

$$\frac{X_{hi}^{n+1} - X_{hi}^n}{dt} = \vec{u}_h^{n+1}(Y_{hi}(t), t) \quad \forall i = 0, \ldots, m \tag{2.50}$$

Where Y_{hi} could be either X_{hi}^n (Forward Euler) or X_{hi}^{n+1} (Backwrd Euler). The scheme with BE is non-linear and quite hard to implement, but it would be unconditionally stable as shown in [4]. On the other hand FE is stable only if certain CFL condition is satisfied. From [4], we have following lemma:

Lemma 3. *Let's mark γ_i the structure segment $X_{hi-1}X_{hi}$. and T_i the set of triangles containing γ_i. By a trace inequality and inverse estimate we have:*

$$||\nabla \vec{u}^{n+1}||_{0,\gamma_i} \leq C_0(h_x^{-\frac{1}{2}})||\nabla \vec{u}^{n+1}||_{0,T_i} \tag{2.51}$$

If there existe positive non-zero K_0 such that for each n=0..N-1

$$\mu - \frac{CKdt}{2h_s H_x} L^n \geq K_0 \tag{2.52}$$

Where L^n is a maximum distance between X_{hi}^n and X_{hi+1}^n. $C = C_0^2 C_1$, C_1 is the maximum number of structure segments contained in a fluid element. Then we have a following energy estimate:

$$\frac{\rho_f}{2}||\vec{u}^n||_{0,\Omega}^2 + dt\sum_{k=1}^{n} K_0||\nabla \vec{u}^k||_{0,\Omega}^2 + \frac{K}{2}\left|\left|\frac{\partial X^n}{\partial s}\right|\right|_{0,S} \tag{2.53}$$

$$\leq \frac{\rho}{2}||\vec{u}_0||_{0,\Omega}^2 + \frac{K}{2}\left|\left|\frac{\partial X^0}{\partial s}\right|\right|_{0,S} \tag{2.54}$$

Chapter 3

Scheme Modification

We propose two modifications of the scheme of Problem 4 (see definition 9). Their purpose is to make the scheme more stable and to add an inertial term in case that the membrane is not massless. At first, we modify the way how the structure is updated.

3.1 Problem 5: Structure update modification

In Problem 4 (see definition 9), the structure update was based on non-slip condition and the update of the structure was based on the fact that the velocity of the structure and the velocity of the fluid has to be the same. Following modification of the structure update is in fact a force balance. The elasticity force is equal to the force acting on the fluid.

Definition 10 (Problem 5)**.** *Having computed the n-th iteration of velocity u_h^n and pressure p_h^n from initial conditions $u_h^0 \in V_h$ and $X_h^0 \in S_h$, we compute the (n+1) iteration. For the weak formulation we use the test functions $\vec{v} \in V_h$, $q \in Q_h$, $v_s \in S_h$. Find $\vec{u}_h^{n+1}(x,t), p_h^{n+1}(x,t), X_h^{n+1}(s,t)$ which satisfy (computational steps are in the same order as we compute them):*

- *Computation of the force term:*

$$\langle \vec{F}_h^{n+1}, \vec{v} \rangle = \sum_{i=0}^{m-1} K(\frac{\partial X_{i+1}^n}{\partial s} - \frac{\partial X_i^n}{\partial s})\vec{v}(X_i^n) \tag{3.1}$$

- *Fluid computation:*

$$\rho_f(\frac{\vec{u}_h^{n+1} - \vec{u}_h^n}{dt}, \vec{v}) + \mu(\nabla \vec{u}_h^{n+1}, \nabla \vec{v}) - (\mathrm{div}\,(\vec{v}), p_h^{n+1}) = \langle F_h^{n+1}, \vec{v} \rangle \tag{3.2}$$

$$(\mathrm{div}\,(\vec{u}_h^{n+1}), q) = 0 \tag{3.3}$$

- *Structure computation:*

$$\rho_s \varepsilon(\frac{\dot{X}_h^{n+1} - \vec{u}_h^{n+1}}{dt}, v_s)_s + K(\frac{\partial (X_h^{n+1} - X_h^n)}{\partial s}\frac{\partial v_s}{\partial_s})_s = 0 \tag{3.4}$$

Unlike Problem 4, we have one more finite element problem to solve. The part of computing the force term and fluid solving is the same as in problem 4. For solving the structure problem, we aproximate the structure membrane Σ by a curve $\chi : [0, l] \to X(s)$. We divide $[0, l]$ on n_X elements, while $X(0) = X(l)$, therefore $X(0) = X(n_X ds)$. For building a linear algebra problem, we introduce the vectors X_x, X_y with components X_{xi}, X_{yi} and U_x, U_y with components $U_{xi} = u_{xh}^{n+1}(X_{xi}, X_{yi})$ (analogically y component) where $i \in \{0, \ldots, (n_X - 1)\}$. U_i are the interpolations of the discrete velocity at the structure points. Then we introduce two matrices $\mathbb{G}$, $\mathbb{M}$.

$$\mathbb{M}_{ij} = \int_0^l v_{si} v_{sj}$$

$$\mathbb{G}_{ij} = \int_0^l \nabla v_{si} \nabla v_{sj}$$

Using them, we can formulate the structure problem:

$$\frac{\rho_s \varepsilon}{(dt)^2} \mathbb{M} X_x^{n+1} + K \mathbb{G} X_x^{n+1} = \frac{\rho_s \varepsilon}{(dt)^2} \mathbb{M} X_x^n + K \mathbb{G} X_x^n + \frac{\rho_s \varepsilon}{(dt)} \mathbb{M} U_x(X^n)$$

(And analogically for U_y and X_y). Assuming that we use P1 aproximation for the structure position X, the matrices $\mathbb{M}$ and $\mathbb{G}$ are following:

Mass matrix $\mathbb{M}$

$$\begin{pmatrix} 2/3 & 1/6 & 0 & 0 & \ldots & 1/6 \\ 1/6 & 2/3 & 1/6 & 0 & \ldots & 0 \\ 0 & 1/6 & 2/3 & 1/6 & \ldots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & 1/6 & 2/3 & 1/6 \\ 1/6 & \ldots & & 0 & 1/6 & 2/3 \end{pmatrix}$$

Elasticity matrix $\mathbb{G}$

$$\begin{pmatrix} 2 & -1 & 0 & 0 & \ldots & -1 \\ -1 & 2 & -1 & 0 & \ldots & 0 \\ 0 & -1 & 2 & -1 & \ldots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & -1 & 2 & -1 \\ -1 & \ldots & & 0 & -1 & 2 \end{pmatrix}$$

3.2 Problem 6 with inertial part

Secondly, we propose modification of the fluid equation, so that we would consider the the mass of the structure. The only difference between Problem 5 and Problem 6 is the inertial term integrated over the structure.

Definition 11 (Problem 6)**.** *Having computed the n-th iteration of velocity u_h^n and pressure p_h^n from initial conditions $u_h^0 \in V_h$ and $X_h^0 \in S_h$, we compute the (n+1) iteration. For the weak formulation we use the test functions $\vec{v} \in V_h$, $q \in Q_h$, $v_s \in S_h$. Find $\vec{u}_h^{n+1}(x,t), p_h^{n+1}(x,t), X_h^{n+1}(q,t)$ which satisfy (computational steps in same order as we compute them):*

- *Force term computation:*

$$\langle \vec{F}_h^{n+1}, \vec{v} \rangle = \sum_{i=0}^{m-1} K(\frac{\partial X_{i+1}^n}{\partial s} - \frac{\partial X_i^n}{\partial s}) \vec{v}(X_i^n) \tag{3.5}$$

- *Fluid computation:*

$$\begin{aligned} \rho_f(\frac{\vec{u}_h^{n+1} - \vec{u}_h^n}{dt}, \vec{v}) + \mu(\nabla \vec{u}_h^{n+1}, \nabla \vec{v}) - (\operatorname{div}(\vec{v}), p_h^{n+1}) \\ + \frac{\rho_s}{dt} \varepsilon(\vec{u}^{n+1}(X^n), \vec{v}(X^n))_s - \frac{\rho_s}{dt} \varepsilon(\dot{X}^n, \vec{v}(X^n))_s \\ = \langle F_h^{n+1}, \vec{v} \rangle \end{aligned} \tag{3.6}$$

$$(\operatorname{div} u_h^{n+1}, q) = 0 \tag{3.7}$$

- *Structure computation:*

$$\rho_s \varepsilon(\frac{\dot{X}_h^{n+1} - u_h^{n+1}}{dt}, v_s)_s + K(\frac{\partial (X_h^{n+1} - X_h^n)}{\partial s} \frac{\partial v_s}{\partial_s})_s = 0 \tag{3.8}$$

3.3 Stability of scheme 6

To see whether or not our modified scheme is stable, we define discrete energy and dissipation, and then we will prove that they are bounded by some finite number.

Definition 12. *For $n \geq 0$ we define discrete energy and dissipation at time level n:*

$$\begin{aligned} E_h^n &= \rho_f ||\vec{u}_h^n||_{0,\Omega}^2 + \rho_s \varepsilon ||\dot{X}_h^n||_{0,\Sigma}^2 + ||X_h^{n+1}||_e^2 \\ D_h^n &= \sum_{k=0}^{n} 2dt\mu ||\nabla \vec{u}_h^k||_{0,\Omega}^2 + dt^2 \rho_f ||\partial_t \vec{u}_h^k||_{0,\Omega}^2 + dt^2 \rho_s \epsilon ||\partial_t \dot{X}_h^k||_{0,\Sigma}^2 + dt^2 ||\partial_t X_h^k||_e^2 \end{aligned} \tag{3.9}$$

We want to prove the stability of scheme 6 (which is more complex). To do so, we will introduce the notation from [2]: elastic operator and Ritz-projector for evaluationg the value of fluid test function in the points of structure. Then we would be able to introduce a following theorem, which describes the unconditional stability of problem 6.

Definition 13. *As in [2] We introduce elastic operator L_h^e: $S \to S_h$. It also defines a norm on S_h:*

$$\begin{aligned} &\forall v \in S, L_h^e v \in S_h \\ &(L_h^e v, v_h)_\Sigma = a^e(v, v_h) = K(\frac{\partial v}{\partial s}\frac{\partial v_h}{\partial s})_s \\ &||v_h||_e = a^e(v_h, v_h) = (L_h^e v_h, v_h)_\Sigma \end{aligned} \tag{3.10}$$

Definition 14. *We define the Ritz-projector fluid to solid: π_h^{fs}: $V_h \to S_h$*

$$\begin{aligned} &\forall \vec{v_h} \in V_h, \pi_h^{fs}\vec{v_h} \in S_h \\ &\forall i \in \{0..m-1\} : \vec{v}_h(X_i) = \pi_h^{fs}\vec{v}_h(X_i) \end{aligned} \tag{3.11}$$

Theorem 4. (Stability of scheme 6) *Let $(\vec{u}_h^{n+1}, p_h^{n+1}, X_h^{n+1}, \dot{X}_h^{n+1})_{n\geq 0} \in V_h \times P_h \times S_h \times S_h$ given by problem 6. Then following estimates holdes:*

$$\begin{aligned} &E_h^{n+1} + 2D_h^{n+1} + dt^2||\dot{X}_h^{n+1}||_e^2 + \frac{dt^2}{\rho_s\epsilon}||L_h^e X^{n+1}||_{0,\Sigma}^2 \\ &+ \sum_{k=0}^{n+1} dt^2||\dot{X}_h^{k+1} - \dot{X}_h^k||_e^2 + \sum_{k=0}^{n+1}\frac{dt^2}{\rho_s\epsilon}||L_h^e X_h^{k+1} - X_h^k||_e^2 \\ = \\ &E_h^0 + dt^2||\dot{X}_h^0||_e^2 + \frac{dt^2}{\rho_s\epsilon}||L_h^e X^0||_{0,\Sigma}^2 \end{aligned} \tag{3.12}$$

which implies the estimate for the energy:

$$E_h^{n+1} < E_h^0 + dt^2||\dot{X}_h^0||_e^2 + \frac{dt^2}{\rho_s\epsilon}||L_h^e X^0||_{0,\Sigma}^2 \tag{3.13}$$

Proof. With the Ritz projector and the elastic operator, we can rewrite the formula for the force (3.5) and the structure problem (6.4). They also allow us to express $\vec{u}_h^{n+1}$:

$$\langle \vec{F}_h^{n+1}, \vec{v}\rangle = -a^e(X^n, \vec{v}(X^n))_s = (L_h^e X^n, \pi_h^{fs}\vec{v}_h)_s \tag{3.14}$$

$$\rho_s\varepsilon(\frac{\dot{X}_h^{n+1} - \vec{u}_h^{n+1}}{dt}, v_s)_s + (L_h^e(X_h^{n+1} - X_h^n), v_s)_s = 0 \tag{3.15}$$

$$\pi_h^{fs}\vec{u}_h^{n+1} = \dot{X}_h^{n+1} + \frac{dt}{\rho_s\epsilon}L_h^e(X^{n+1} - X^n) \tag{3.16}$$

We fix the test function $v_s = \pi_h^{fs}\vec{v}_h$, which is an element of S_h (the piecewise linear functions on Σ) and we sum the equations of problem 6: (6.3),(3.7),(6.4). The force component will be canceled out by a part of second member of the structure equation $(L_h^e(X_h^{n+1} - X_h^n), v_s)_s$, the inertial member $\frac{\rho_s}{dt}\epsilon(\vec{u}^{n+1}(X^n), \vec{v}(X^n))_s$ will be also canceled by the part of structure equation. That leaves us with the following equation:

$$\begin{aligned} &\text{STOKES: } \rho_f(\frac{\vec{u}_h^{n+1} - \vec{u}_h^n}{dt}, \vec{v}_h) + \mu(\nabla\vec{u}_h^{n+1}, \nabla\vec{v}_h) - (\operatorname{div}\vec{v}, p_h^{n+1}) + (\operatorname{div}(\vec{u}_h^{n+1}), q_h) \\ &\text{STRUC: } + \frac{\rho_s\varepsilon}{dt}(\dot{X}_h^{n+1} - \dot{X}^n, \pi_h^{fs}\vec{v}_h)_s + (L_h^e X_h^{n+1}, \pi_h^{fs}\vec{v}_h)_s = 0 \end{aligned} \tag{3.17}$$

Now we fix the test functions $\vec{v}_h = \vec{u}_h^{n+1}$ and $q_h = p_h$. Let's call the first line of equation (3.23) STOKES and the rest STRUC. The members with a divergence will be canceled out, and to major the others, we use a simple formula $ab = -\frac{1}{2}\big[(a-b)^2 - a^2 - b^2\big]$. So for the Stokes part, we get:

$$STOKES \leq \frac{\rho_f}{2dt}(||\vec{u}_h^{n+1}||^2_{\Omega,0} - ||\vec{u}_h^{n}||^2_{\Omega,0} + ||\vec{u}_h^{n+1} - \vec{u}_h^{n}||^2_{\Omega,0}) + \mu||\nabla\vec{u}_h^{n+1}||^2_{\Omega,0} \quad (3.18)$$

Now, we insert the expression (3.16) for $\pi_h^{fs}\vec{u}^{n+1}$ to the part STRUCT, which we want to major too:

$$\begin{aligned}&\underbrace{\frac{\rho_s\mathcal{E}}{dt}(\dot{X}_h^{n+1} - \dot{X}^n, \dot{X}_h^{n+1})_s}_{\text{T1}} \underbrace{+(\dot{X}_h^{n+1} - \dot{X}^n, L_h^e(X_h^{n+1} - X_h^n))_s}_{\text{T2}} \\ &+ \underbrace{(L_h^e X_h^{n+1}, \dot{X}_h^{n+1})_s}_{\text{T3}} + \underbrace{\frac{dt}{\rho_s\epsilon}(L_h^e X_h^{n+1}, L_h^e(X_h^{n+1} - X_h^n)_s}_{\text{T4}} = \text{STRUCT}\end{aligned} \quad (3.19)$$

$$\begin{aligned}T1 &= \frac{\rho_s\epsilon}{2dt}(||\dot{X}_h^{n+1}||^2_{0,\Sigma} - ||\dot{X}_h^{n}||^2_{0,\Sigma} + ||\dot{X}_h^{n+1} - \dot{X}_h^{n}||^2_{0,\Sigma}) \\ T2 &= dt\ a^e(\dot{X}_h^{n+1} - \dot{X}_h^{n}, \dot{X}_h^{n+1}) = \frac{dt}{2}(||\dot{X}_h^{n+1}||^2_e - ||\dot{X}_h^{n}||^2_e + ||\dot{X}_h^{n+1} - \dot{X}_h^{n}||^2_e) \\ T3 &= \frac{1}{2dt}(||X_h^{n+1}||^2_e - ||X_h^{n}||^2_e + ||X_h^{n+1} - X_h^{n}||^2_e) \\ T4 &= \frac{dt}{2\rho_s\epsilon}(||L_h^e X_h^{n+1}||^2_{0,\Sigma} - ||L_h^e X_h^{n}||^2_{0,\Sigma} + ||L_h^e(X_h^{n+1} - X_h^{n})||^2_{0,\Sigma})\end{aligned} \quad (3.20)$$

We sum the STOKES and STRUCT part over $k = 0..n+1$. We use the fact that the sum is telescopic in some members.

$$\begin{aligned}\sum \text{STOKES} &= \frac{\rho_f}{2dt}(||\vec{u}_h^{n+1}||^2_{\Omega,0} - ||\vec{u}_h^{0}||^2_{\Omega,0}) + \sum_{k=0}^{n+1}(\frac{\rho_f}{2dt}||\vec{u}_h^{k+1} - \vec{u}_h^{k}||^2_{\Omega,0}) + \mu||\nabla\vec{u}_h^{k+1}||^2_{\Omega,0}) \\ \sum \text{STRUCT} &= \frac{\rho_s\epsilon}{2dt}(||\dot{X}_h^{n+1}||^2_{0,\Sigma} - ||\dot{X}_h^{0}||^2_{0,\Sigma}) + \frac{dt}{2}(||\dot{X}_h^{n+1}||^2_e - ||\dot{X}_h^{0}||^2_e) \\ &+ \frac{1}{2dt}(||X_h^{n+1}||^2_e - ||X_h^{0}||^2_e) + \frac{dt}{2\rho_s\epsilon}(||L_h^e X_h^{n+1}||^2_{0,\Sigma} - ||L_h^e X_h^{0}||^2_{0,\Sigma}) \\ &+ \sum_{k=0}^{n+1}(\frac{\rho_s\epsilon}{2dt}||\dot{X}_h^{k+1} - \dot{X}_h^{k}||^2_{0,\Sigma} + \frac{dt}{2}||\dot{X}_h^{k+1} - \dot{X}_h^{k}||^2_e \\ &+ \frac{1}{2dt}||X_h^{k+1} - X_h^{k}||^2_e + \frac{dt}{2\rho_s\epsilon}||L_h^e(X_h^{k+1} - X_h^{k})||^2_{0,\Sigma})\end{aligned} \quad (3.21)$$

In terms of dissipation and descrete energy:

$$
\begin{aligned}
& 2dt \sum \text{STOKES+STRUCT} = \\
& E_h^{n+1} + 2D_h^{n+1} + dt^2 ||\dot{X}_h^{n+1}||_e^2 + \frac{dt^2}{\rho_s \epsilon} ||L_h^e X^{n+1}||_{0,\Sigma}^2 \\
& + \sum_{k=0}^{n+1} dt^2 ||\dot{X}_h^{k+1} - \dot{X}_h^k||_e^2 + \sum_{k=0}^{n+1} \frac{dt^2}{\rho_s \epsilon} ||Ł_h^e X_h^{k+1} - X_h^k||_e^2 \\
= & \\
& E_h^0 + dt^2 ||\dot{X}_h^0||_e^2 + \frac{dt^2}{\rho_s \epsilon} ||L_h^e X^0||_{0,\Sigma}^2
\end{aligned}
\tag{3.22}
$$

And that implies stability of scheme 6.

□

3.4 Stability of scheme 5

Proof. We can show the stability of Scheme 5 very similarily. After we fix the test function of the Structure finite element space: $v_s = \pi_h^{fs} \vec{v}_h \in S_h$, we get (we are using the same notation as above):

$$
\begin{aligned}
& \text{STOKES: } \rho_f (\frac{\vec{u}_h^{n+1} - \vec{u}_h^n}{dt}, \vec{v}_h) + \mu (\nabla \vec{u}_h^{n+1}, \nabla \vec{v}_h) - (\text{div } \vec{v}, p_h^{n+1}) + (\text{div } \vec{u}_h^{n+1}, q_h) \\
& \text{STRUC: } + \frac{\rho_s \varepsilon}{dt} (\dot{X}_h^{n+1}, \pi_h^{fs} \vec{v}_h)_s - \frac{\rho_s \varepsilon}{dt} (\pi_h^{fs} \vec{u}_h^{n+1}, \pi_h^{fs} \vec{v}_h) + (L_h^e X_h^{n+1}, \pi_h^{fs} \vec{v}_h)_s = 0
\end{aligned}
\tag{3.23}
$$

Then we fix fluid test function $\vec{v}_h = \vec{u}_h^{n+1}$ and we express from the structure equation: $\pi_h^{fs} \vec{u}_h^{n+1} = \dot{X}_h^{n+1} + \frac{dt}{\rho_s \epsilon} L_h^e (X^{n+1} - X^n)$. Then we get:

$$
\begin{aligned}
& \underbrace{(\dot{X}_h^{n+1}, L_h^e (X_h^{n+1} - X_h^n))_s}_{\text{T1}} \\
& \underbrace{- \frac{dt}{\rho_s \varepsilon} (L_h^e (X_h^{n+1} - X_h^n), L_h^e (X_h^{n+1} - X_h^n))}_{\text{T2}} \\
& + \underbrace{(L_h^e X_h^{n+1}, \dot{X}^{n+1})_s}_{\text{T3}} + \underbrace{\frac{dt}{\rho_s \varepsilon} (Ł_h^e X_h^{n+1}, L_h^e (X_h^{n+1} - X_h^n))}_{\text{T4}} \\
& = \text{STRUCT}
\end{aligned}
\tag{3.24}
$$

$$
\begin{aligned}
T1 &= -\frac{1}{dt} ||X_h^{n+1} - X_h^n||_e^2 \\
T2 &= -\frac{dt}{\rho_s \varepsilon} ||L_h^e (X_h^{n+1} - X_h^n)||_{0,\Sigma}^2 \\
T3 &= \frac{1}{2dt} (||X_h^{n+1}||_e^2 - ||X_h^n||_e^2 + ||X_h^{n+1} - X_h^n||_e^2) \\
T4 &= \frac{dt}{2\rho_s \epsilon} (||L_h^e X_h^{n+1}||_{0,\Sigma}^2 - ||L_h^e X_h^n||_{0,\Sigma}^2 + ||L_h^e (X_h^{n+1} - X_h^n)||_{0,\Sigma}^2)
\end{aligned}
\tag{3.25}
$$

We sum the STOKES (same as in problem 6) and STRUCT part over $k = 0..n+1$. We use the fact that the sum we get is telescopic in some members. Then we get:

$$\begin{aligned}\sum \text{STRUCT} = &\frac{2}{dt}(||X_h^{n+1}||_{0,\Sigma}^2 - ||X_h^0||_{0,\Sigma}^2) \\ &+ \frac{2dt}{\rho_s \varepsilon}(||L_h^e X_h^{n+1}||_{0,\Sigma}^2 - ||L_h^e X_h^0||_{0,\Sigma}^2) \\ &+ \sum_{k=0}^{n+1}(+\frac{2}{dt}||X_h^{k+1} - X_h^k||_e^2 + \frac{2dt}{\rho_s \varepsilon}||L_h^e(X_h^{k+1} - X_h^k)||_{0,\Sigma}^2) \\ &= 0\end{aligned} \tag{3.26}$$

When we sum this equation with result of STOKES part in (3.18), we get:

$$\begin{aligned}&\rho_f||\vec{u}_h^{n+1}||_{(\Omega,0)}^2 + \sum_{k=0}^{n+1}(||\vec{u}_h^{k+1} - \vec{u}_h^k||_{(\Omega,0)}^2 + \mu||\nabla \vec{u}_h^{k+1}||_{(\Omega,0)}^2) \\ &+ \frac{2}{dt}||X_h^{n+1}||_{0,\Sigma}^2 + \frac{2dt}{\rho_s \varepsilon}||L_h^e X_h^{n+1}||_{0,\Sigma}^2 \\ &+ \sum_{k=0}^{n+1}(\frac{2}{dt}||X_h^{k+1} - X_h^k||_e^2 + \frac{2dt}{\rho_s \varepsilon}||L_h^e(X_h^{k+1} - X_h^k)||_{0,\Sigma}^2) \\ &\leq \\ &\frac{2}{dt}||X_h^0||_{0,\Sigma}^2) + \frac{2dt}{\rho_s \varepsilon}||L_h^e X_h^0||_{0,\Sigma}^2 + \rho_f||u_h^0||_{(\Omega,0)}^2\end{aligned} \tag{3.27}$$

Which implies stability of scheme 5. □

3.5 Membrane with fixed borders

Above we have described the problems solving the situations where the membrane forms a closed shape. The force can have a special form due to this configuration. In a following section, we are going to describe the situation, where the membrane is a 1-dimensional line (in our 2D case), attached tightly on both ends.

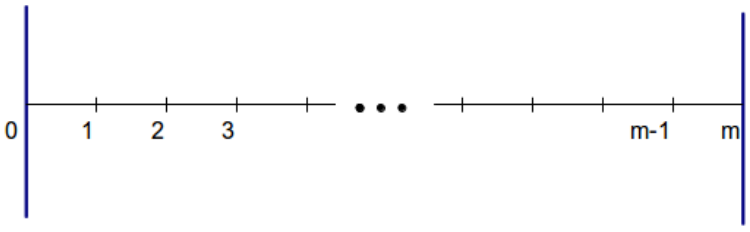


Figure 3.1: Structure numbering with fixed ends

In the part we derived the formula for force, we can't go from (2.39) to (2.40), because the points X_{h0} and X_{hm} do not coincide.

$$
\begin{aligned}
\langle F_h, v\rangle &= -\sum_{i=1}^{m} K\frac{\partial X_{hi}}{\partial s}(t)\vec{v}(X_{hi}) + \sum_{i=0}^{m-1} K\frac{\partial X_{hi+1}}{\partial s}(t)\vec{v}(X_{hi}) \\
&= -\sum_{i=0}^{m-1} K\left(\frac{\partial X_{hi+1}}{\partial s}(t) - \frac{\partial X_{hi}}{\partial s}(t)\right)\vec{v}(X_{hi}) \\
&\quad - K\frac{\partial X_{hm}}{\partial s}(t)\vec{v}(X_{hm}) \\
&\quad + K\frac{\partial X_{h1}}{\partial s}(t)\vec{v}(X_{h0})
\end{aligned} \tag{3.28}
$$

If the points X_{h0} and X_{hm} are on the boundary of a computational domain, where we postulate zero Dirichlet condition, we can assume that $\vec{v}(X_{h0}) = \vec{v}(X_{hm}) = 0$.

Given the fact that X_{h0} and X_{hm} are fixed, we can modify matrices for Problem 5: Mass matrix $\mathbb{M}$

$$
\begin{pmatrix}
1 & 0 & 0 & 0 & \dots & 0 \\
1/6 & 2/3 & 1/6 & 0 & \dots & 0 \\
0 & 1/6 & 2/3 & 1/6 & \dots & 0 \\
\vdots & & \ddots & \ddots & \ddots & \vdots \\
0 & \dots & 0 & 1/6 & 2/3 & 1/6 \\
0 & \dots & & 0 & 0 & 1
\end{pmatrix}
$$

Elasticity matrix $\mathbb{G}$

$$
\begin{pmatrix}
1 & 0 & 0 & 0 & \dots & 0 \\
-1 & 2 & -1 & 0 & \dots & 0 \\
0 & -1 & 2 & -1 & \dots & 0 \\
\vdots & & \ddots & \ddots & \ddots & \vdots \\
0 & \dots & 0 & -1 & 2 & -1 \\
0 & \dots & & 0 & 0 & 1
\end{pmatrix}
$$

And we set $U_x(X_{h0}) = U_y(X_{h0}) = 0$ and $U_x(X(hm)) = U_y(X(hm)) = 0$ in formulation of structure problem in problem 5:

$$
\frac{\rho_s \varepsilon}{(dt)^2}\mathbb{M}X^{n+1} + K\mathbb{G}X^{n+1} = \frac{\rho_s \varepsilon}{(dt)^2}\mathbb{M}X^{n} + K\mathbb{G}X^{n} + \frac{\rho_s \varepsilon}{(dt)}\mathbb{M}U(X^n)
$$

We can see that by solving this problem, X_{h0} and X_{hm} remains constant. For this configuration, we have run two kind of tests: the "tube" and the "aneurysm" test, which are described below.

Chapter 4

Implementation

We have used the program FreeFem++ [10] to run all the computations. It has been created for an easy use of finite element method. It's syntax is close to c++, which makes it even easier to use.

4.1 Force assembly

Force calculation is same for all of the problems 4-6.

$$\langle F_h^{n+1}, \vec{v} \rangle = \sum_{i=0}^{m-1} K(\frac{\partial X_{i+1}^n}{\partial s} - \frac{\partial X_i^n}{\partial s})\vec{v}(X_i^n) \tag{4.1}$$

We use following notation:
structure points: $X_i\, i \in 1..nX$
x(y) component of structure point $X_{i,x}\,(X_{i,y})$
The finite element functions defined on the vertex of mesh `sx[number of vertex]`
step of Structure parametrization `dalpha` $= \frac{2\pi}{nX-1}$

Algorithm 1 Force calculation

set sx $\equiv 0$ ▷ zero in all vertexes
for i=1; i < nX; i++ **do**
  set `triang` the triangle containing X_i ▷ function in FreeFem++
  for k=0; k<3; k++ **do** ▷ Each vertex of triangle containing the point X_i
    set `vert` the number of k^{th} vertex of `triang`
    set sx[`vert`] = 1
    `sumx`=0;
    `sumy`=0;
    for j=1; j < nX; j++ **do** ▷ $sx(X_j)$ is an interpolation
      `sumx`+=$((\frac{X_{j+1,x}-2X_{j,x}+X_{j-1,x}}{dalpha}) * sx(X_j)$
      `sumy`+=$((\frac{X_{j+1,y}-2X_{j,y}+X_{j-1,y}}{dalpha}) * sx(X_j)$
    end for
    set sx[`vert`]=0
    set F[`vert`]=K*`sumx` ▷ force on vertex of a mesh
    set F[number of degrees of freedom Vh+`vert`]=K*`sumy`
  end for
end for

4.2 Fluid problem matrix definition

First, we assemble the matrix for problem 4, the inertial part in problem 6 will be added later. The fluid matrix $\mathbb{F}$ has following structure:

$$\begin{matrix} \mathbb{A}_{11} & \mathbb{A}_{12} & \mathbb{B}_x \\ \mathbb{A}_{12}^T & \mathbb{A}_{22} & \mathbb{B}_y \\ \mathbb{B}_x^T & \mathbb{B}_y^T & \mathbb{D} \end{matrix}$$

Fluid solving:

$$\begin{aligned} \mathbb{A} &= \rho_f \frac{\int_\Omega \vec{v}_h \vec{v}_h}{dt} + \mu \int_\Omega \nabla \vec{v}_h \nabla \vec{v}_h \qquad \vec{v}_h \in V_h = P_1^2(Th) \\ \mathbb{B} &= -\int_\Omega \operatorname{div}(\vec{v}_h) q_h \qquad \vec{v}_h \in V_h; q_h \in Q_h = P_1(Th) \\ \mathbb{D} &= \frac{\alpha h^2}{\mu} \int_\Omega \nabla p, \nabla q + \epsilon \int_\Omega pq \qquad p, q \in P_h \end{aligned} \tag{4.2}$$

Matrix $\mathbb{D}$ is a stabilization, which allow us to use simple P1-P1 finite elements both for velocity and for pressure. As stabilization parametres we have used $\alpha = 0.01$ and $\epsilon = 10^{-10}$. The scond part of the stabilization term $\epsilon \int_\Omega pq$ is de facto identity multiplied by a small epsilon. Without this term, the pressure would be determined by solving the fluid problem, except for the constant (if p is solution, $p+c, c \in \mathbb{R}$ is also solution).When we tried the computation without this part, as the computation proceeded the pressure was randomly jumping. With the identity part, the pressure constant $c = 0$ and the initial pressure is zero and pressure results corresponds with analytical solution (in case of the stationary solution).

Right hand side of the problem is formed of force calculated before and the already known part of the time derivative member (velocity from previous step).

$$RHS = F + \frac{\rho_f}{dt}(\vec{u}_n, \vec{v}_h) \tag{4.3}$$

Then the fluid problem is described as:

$$\mathbb{F}U^{n+1} = RHS \tag{4.4}$$

For both fluid and structure problem, we have used direct solver UMFPACK [5], because an iterative solver could cause an additional error in our computations.

4.3 Integration of inertial member

The inertial member in problem 6 will be added to the fluid matrix $\mathbb{F}$. The problem is, that we have to integrate the basis functions of the velocity space V_h on the one-dimensional structure.

$$\underbrace{\frac{\rho_s}{dt}\epsilon(u^{n+1}(X^n), \vec{v}(X^n))_s}_{\text{added to } \mathbb{F}} \quad \underbrace{-\frac{\rho_s}{dt}\epsilon(\dot{X}^n, \vec{v}(X^n))_s}_{\text{will be moved to right hand side}} \tag{4.5}$$

As described in the definition of the Ritz projector 14 $v_s = \pi_h^{fs}\vec{v}_h \in S_h$, where S_h is a P_1 finite element space on the structure parametrization, the projection v_s has to be linear on the structure segments. In the picture 4.1 we can see the two possibilities of piecewise linear functions to integrate. Due to the definition of the Ritz operator, the green one is the correct function which suits the theory.

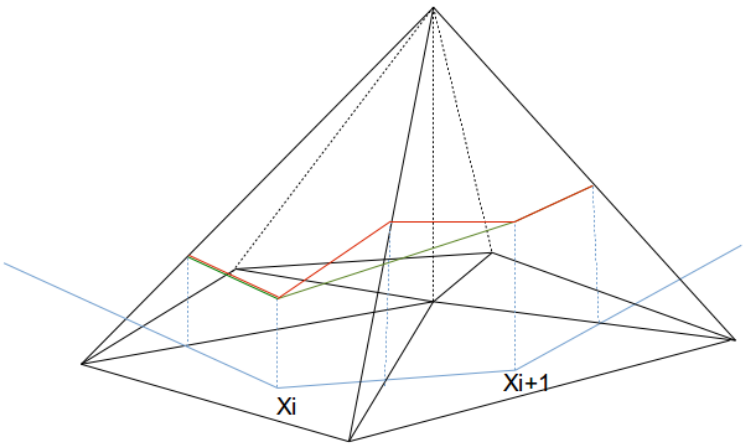


Figure 4.1: Integration of inertial member: The green line is the correct function which correspoonds to a definition of the Ritz operator

We will use the following notation: The last point to which we have integrated: `xlast`
The triangle containing `xlast` (there is an function in Freefem to find it): `trlast`
Field to store all of the triangles which might have an intersection with the next structure segment: `triToSearch`
Number of entries in the `triToSearch` field: `triToSearchCount`

Algorithm 2 Passing through the mesh and adding an inertial member to the Fluid matrix

1: xlast = X_0
2: xi = X_1
3: set trlast (number of triangle containing xlast)
4: set tri (number of triangle containing xi)
5: **while** !finished **do** ▷ finished=true if xlast is nXth vertex
6: Find the candidates of triangles which might have an intersection with current structure segment (and save it to triToSearch field) (algorithm 3)
7: Finding an intersection to which we will integrate (algorithm 5)
8: Integration
9: **if** xlast == X_{nX} **then**
10: finished = true
11: **end if**
12: **end while**

Algorithm 3 Filling the triToSearch field (triangles which might have potentially intersection with current xlast-xi segment)

```
set xlastOnBord = -1;
if xlast is on edge of trlast then
    set xlastOnBord = number of the edge containing xlast
    set xlastOnVertex = -1;
    if xlast is on vertex then
        set xlastOnVertex = number of vertex which coincide with xlast
        triToSearch ← triangles which contains the vertex xlastOnVertex
        set triToSearchCount = number of triangles containing xlastOnVertex
    else
        triToSearch ← triangles that share border xlastOnBord
        Set triToSearchCount = 2
    end if
else
    triToSearch ← trlast
    Set triToSearchCount = 1
end if
    ▷ We need to search an intersection only with edges not containing xlast
for trSearch = 0; trSearch<triToSearchCount; trSearch++ do
    tr=triToSearch(trSearch)
    xlastOnEdges(trSearch) = number of edge of tr cofcontaining xlast
end for
```

Algorithm 4 Finding an intersection of xlast-xi segment with selected triangles

```
Intersection = (-1,-1)
for trSearch = 0; trSearch<triToSearchCount; trSearch++ do
    tr=triToSearch(trSearch)
    if tr == tri or xi is on bord of tr then
        Intersection = xi
    else
        find intersection with the tr triangle's edges but not on xlas-
    tOnEdges(trSearch)
        if intersection found then
            Set Intersection
        end if
    end if
end for
if Intersection == (-1,-1) then
    Intersection not found!
    Abort
end if
```

Integration of the inertial member over the red segment in triangle trlast will contribute only to some matrix entries. Namely the entries with the numbers of vertexes of the triangle containing the segment (see the picture 4.2): $\mathbb{F}_{ii}$, $\mathbb{F}_{jj}$, $\mathbb{F}_{kk}$, $\mathbb{F}_{ij}$, $\mathbb{F}_{ik}$, $\mathbb{F}_{jk}$, $\mathbb{F}_{ji}$, $\mathbb{F}_{ki}$, $\mathbb{F}_{kj}$. Because the other basis functions (apart from i,j,k)

will have zero projection on the red segment.

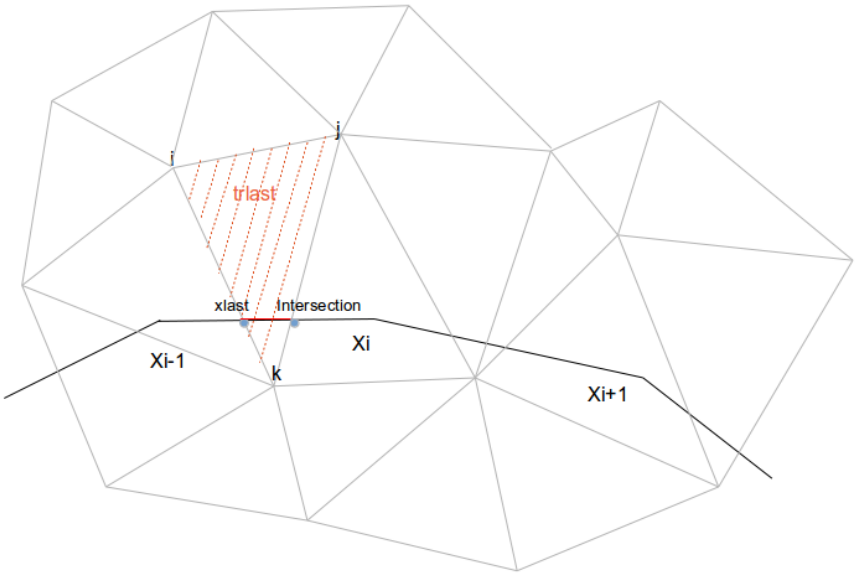


Figure 4.2: Nonzero contributions of inertial member integrated over the segment in triangle trlast

To integrate inertial member over the red segment, we need to know the $dX(xlast) = X(xlast) - Xold(xlast)$ and $dX(Intersection) = X(Intersection) - Xold(Intersection)$ - the difference of the structure position in previous and current step, respectively the x-component and y-component of this displacement. And we need to know the value of an interpolation of FEM basis function in the point xlast and Intersection (see the pictures 4.3 and 4.4).

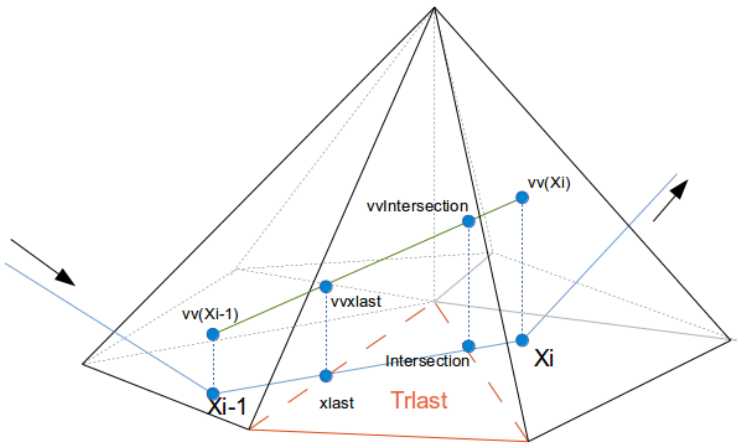


Figure 4.3: Integration of the projection of the fluid FEM basis function over a structure segment and interpolation of the basis function in the points which doesn't coincide with the points of the structure

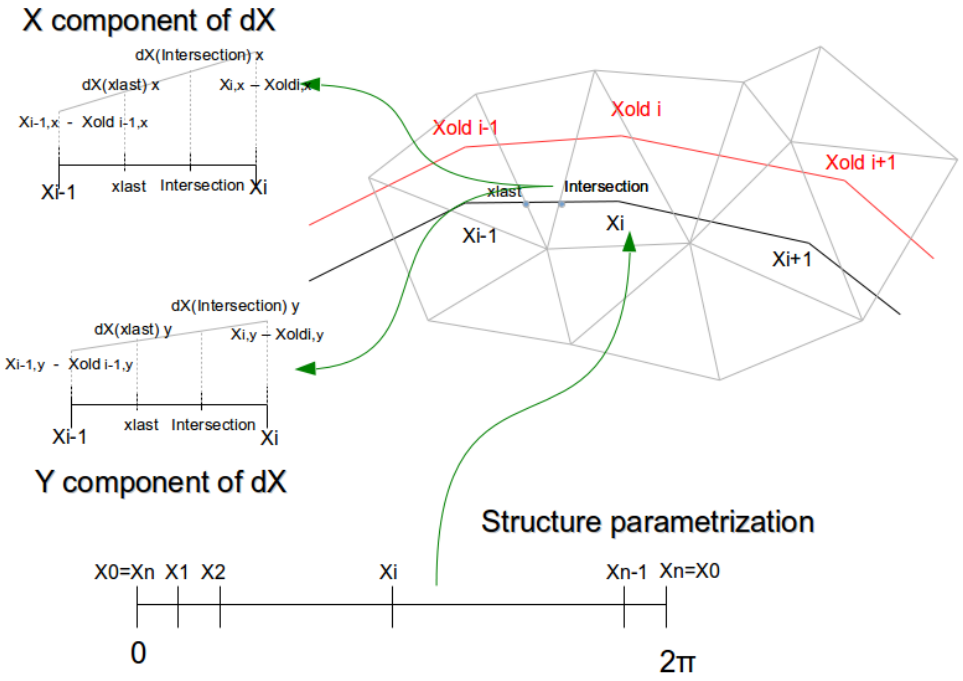


Figure 4.4: Counting dX (difference between the structure position in previous and current step) in xlast and Intersection

The integration and matrix modification will be done by following algorithm:

Algorithm 5 Integration and modification of matrix

set vv $\equiv 0$ ▷ zero in all vertexes
for k=0;k<3;j++ **do**
    ind(k)=Th[trlast][k] ▷ Save numbers of vertexes of the triangle trlast
    vv[ind(k)] = 1
    set vvxlast, vvIntersection
    $dX_x = X_{i,x} - Xold_{i,x}$ ▷ X_{old} from previous step of calculation
    $dX_y = X_{i,y} - Xold_{i,y}$
    $dXold_x = X_{i-1,x} - Xold_{i-1,x}$
    $dXold_y = X_{i-1,y} - Xiold_{i-1,y}$

    ▷ Difference between structure position in this and previou step in point xlast
    $dX(xlast)_x = \frac{distance(xlast,X_{i-1})}{distance(X_i,X_{i-1})}(dX_{i,x} - dXold_{i,x}) + dXold_{i,x}$
    $dX(xlast)_y = \frac{distance(xlast,X_{i-1})}{distance(X_i,X_{i-1})}(dX_{i,y} - dXold_{i,y}) + dXold_{i,y}$
    ▷ Difference between structure position in this and previou step in point Intersection
    $dX(Intersection)_x = \frac{distance(Intersection,X_{i-1})}{distance(X_i,X_{i-1})}(dX_{i,x} - dXold_{i,x}) + dXold_{i,x}$
    $dX(Intersection)_y = \frac{distance(Intersection,X_{i-1})}{distance(X_i,X_{i-1})}(dX_{i,y} - dXold_{i,y}) + dXold_{i,y}$

        ▷ Now we will integrate from xlast to Intersection two linear functions defined by the values in those two points

    rhsIntegx = integrateij(xlast,Intersection,vvxlast,vvIntersection,$dX(xlast)_x$, $dX(Intersection)_x$)
    rhsIntegy = integrateij(xlast,Intersection,vvxlast,vvIntersection, $dXxlast_y$,$dX(Intersection)_y$)

    RHSx(ind(k))+=1./(dt*dt)*rhos*epsilon*rhsintegx ▷ adding member to the right hand side
    RHSy(ind(k))+=1./(dt*dt)*rhos*epsilon*rhsintegy

            ▷ Now we will integrate the bilinear part which will be added to a matrix $\mathbb{F}$
    for j=0;j<3;j++ **do**
        ww(ind(j))=1;
        set wwxlast, wwIntersection ▷ See the picture above
        integ=integrateij(xlast,I1, vvxlast, vvI1, wwxlast, wwI1)
                ▷ Vh is the space basis functions of one component of velocity
                    ▷ Vh.ndof is its number of degrees of freedom
        $\mathbb{F}$(ind(j),ind(k)) += $\frac{1}{dt}\rho_s\epsilon$;
        $\mathbb{F}$(Vh.ndof+ind(j),Vh.ndof+ind(k)) += $\frac{1}{dt}\rho_s\epsilon$;
        if ind(k)!=ind(j) **then**
            $\mathbb{F}$(ind(k),ind(j)) += $\frac{1}{dt}\rho_s\epsilon$ integ
            $\mathbb{F}$(Vh.ndof+ind(k),Vh.ndof+ind(j)) += $\frac{1}{dt}\rho_s\epsilon$ integ
        end if
        ww(ind(j))=0;
    end for
    vv(ind(k))=0;
end for

4.4 Algorithm Complexity

We have compared the time consumed by the computation for problem 4-6. We have used the parametres dt=0.01, T=3s (therefore we had 300 iterations), $\frac{h_x}{h_s} = 2, K = 2$. From the graph 4.5, we can see the proportion of time-consumeness of each computational step:

- Fluid problem solving and force calculation (t_{pb4})
- Structure problem solving (problem 5,6) ($t_{pb5} - t_{pb4}$)
- inertial member assembly (and modification of fluid matrix) ($t_{pb6} - t_{pb5}$)

We can see that the complexity of inertial member assembly is almost linear.

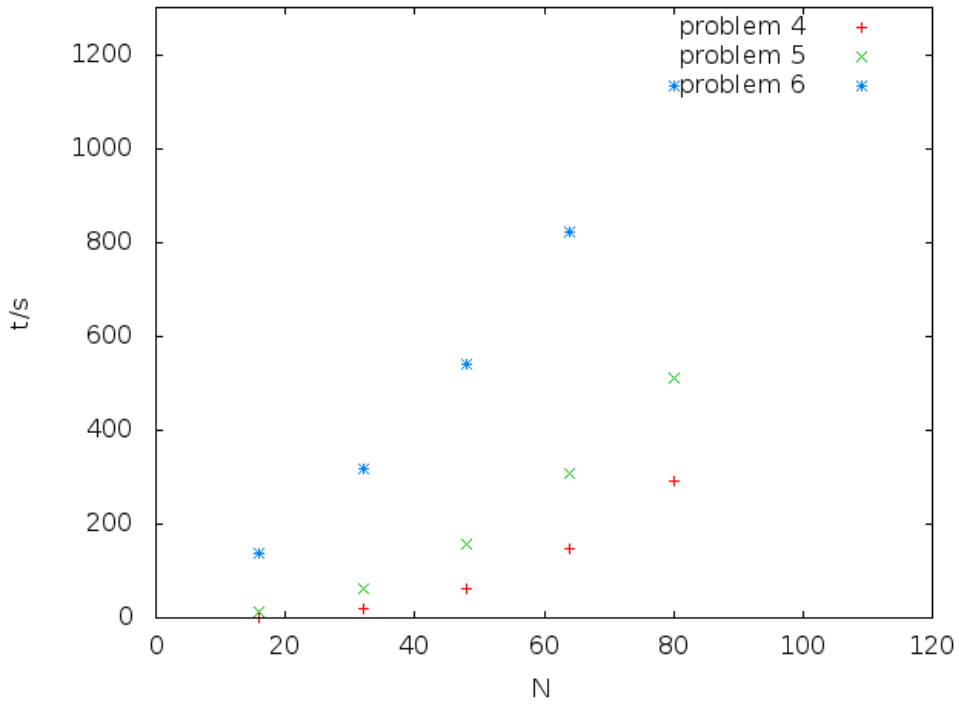


Figure 4.5: Consumed time for various finesses of fluid mesh. dt=0.01, T=3s, $\frac{h_x}{h_s} = 2, K = 2$

Chapter 5

Numerical tests

5.1 Steady state - analytical solution

As the first and the simpliest test, we have used a steady state of infilled baloon in a fluid that does not move. As an initial condition, we have used:

$$\begin{aligned} & u_h^0 = 0 \\ & s \in [0, 2\pi r] \\ & X_x^0(s) = r\cos(s/r) + 0.5 \\ & X_y^0(s) = r\sin(s/r) + 0.5 \end{aligned} \tag{5.1}$$

Problem 4 in this case gives us the exact solution according to [4]:

$$\begin{aligned} & p(x,t) = \begin{cases} K(\frac{1}{r} - \pi r) & \text{if } |x| \leq r \\ -K\pi r & \text{otherwise} \end{cases} \\ & u(x,t) = 0 \in \Omega \end{aligned} \tag{5.2}$$

Actually, the steady state of Problem 5(structure update modification) and 6(with inertial part) are the same, because of the zero initial velocity, the structure update equation reduces to the same one as in problem 4, and therefore the inertial part added in problem 6 will be zero as well.

We have been evaluating the stability of the scheme by counting the energy of the system at each step.

$$E^n = \frac{\rho_f}{2}\|\vec{u}_h^n\|_{0\Omega}^2 + \frac{K}{2}\|\frac{\partial X_h^n}{\partial s}\|_{0\Sigma}^2$$

5.2 Stability

For all of our schemes, we have run the same tests of stability as in the article [4]. At first set of tests (A) we kept the ratio of fluid and structure $\frac{h_x}{h_s} = 2$, at second (B) one we have increased it, $\frac{h_x}{h_s} = 4$, test (C) compares the stability for different ratios $\frac{h_x}{h_s}$ and finally at (D), we kept the K=2 and $\frac{h_x}{h_s} = 2$ and we tested it for various mesh finess.

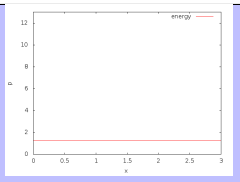
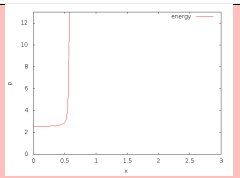
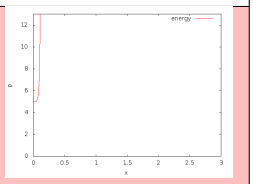
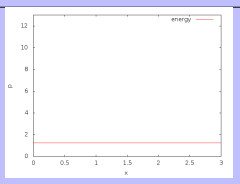
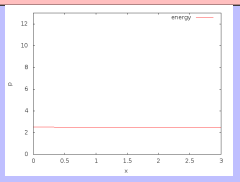
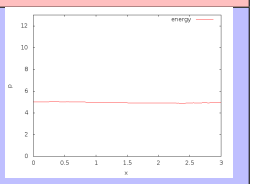
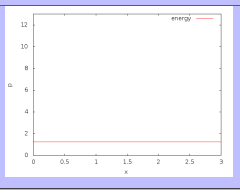
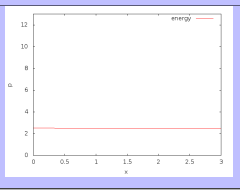
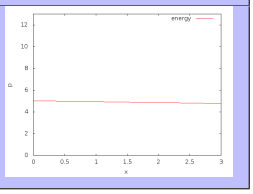
dt K	1	2	4
0.025			
0.01			
0.005			

Table 5.1: test (A): Energy for scheme 4.

A	pb4			pb5			pb6		
K / dt	1	2	4	1	2	4	1	2	4
0.025	x	x	x						
0.01		x	x						
0.005			x						

Table 5.2: test (A): N=64, M=322, various K

B	pb4			pb5			pb6		
K / dt	1	2	4	1	2	4	1	2	4
0.025		x	x						
0.01			x						
0.005									

Table 5.3: test (B): N=32, M=322, various K

C	pb4			pb5			pb6		
M	161	322	643	161	322	643	161	322	643
dt — $\frac{h_x}{h_s}$	1	2	4	1	2	4	1	2	4
0.025	x	x	x						
0.01		x	x						
0.005									

Table 5.4: test (C): K=2, N=64, various ratio $\frac{h_x}{h_s} = 1, 2, 4$

D	pb4			pb5			pb6		
N	16	32	64	16	32	64	16	32	64
dt M	80	161	322	80	161	322	80	161	322
0.025	x	x	x						
0.01		x	x						
0.005			x						

Table 5.5: test (D): $\frac{h_x}{h_s} = 2$, K=4, various mesh finess

At tables 5.12-5.4.1 you can see the results of stability test A-D. The red cell indicates instability, the blue one stability. We can see that scheme 4 is conditionally stable. The results are similar as in the article [4], our implementation seems a bit more stable (it can be caused by Brezzi stabilisation).For quick comparison, we have placed an x mark in cells where the scheme in [4] is unstable.

We have confirmed by test, that scheme 5 is unconditionally stable in the range of our parameters.

5.3 Precision

For three schemes, we display the pressure at $y = 0.5$ (red line) with the analytical solution for precision comparison (green line).

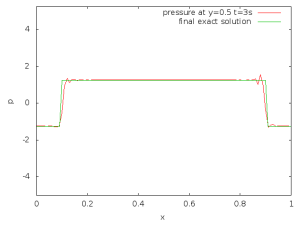
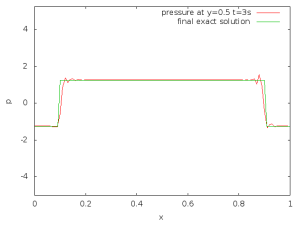
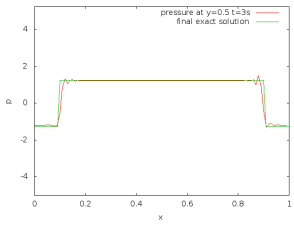
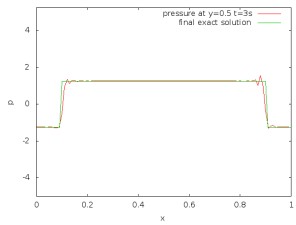
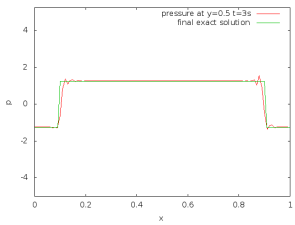
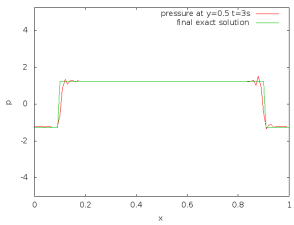
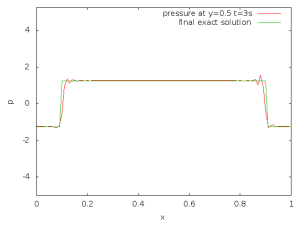
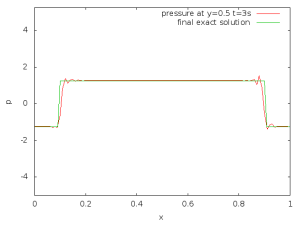
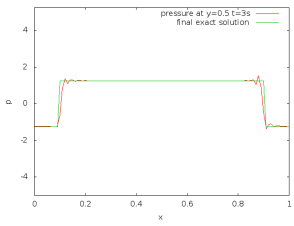
variables	Scheme 4	Scheme 5	Scheme 6
K=1 dt=0.005			
K=1 dt=0.01			
K=1 dt=0.025			

Table 5.6: various K,dt N=64 M=322

We have confirmed for all test cases, that schemes with the modifications we have proposed (problems 5 and 6) are stable.

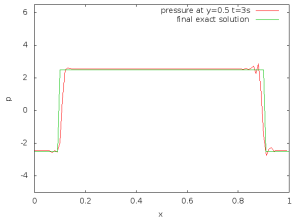
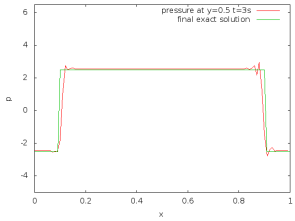
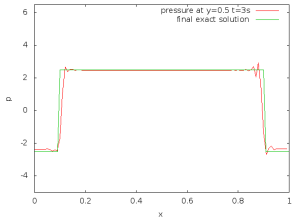
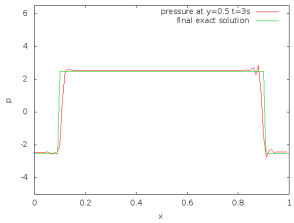
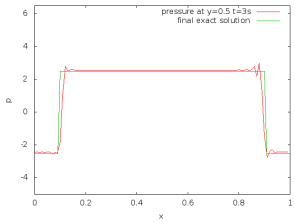
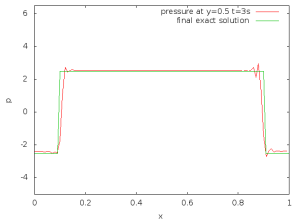
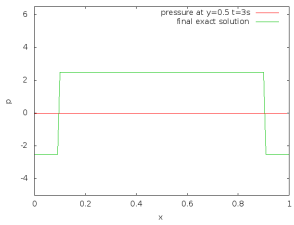
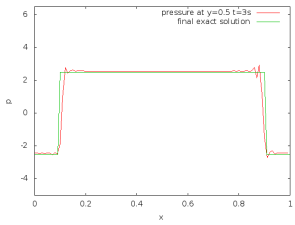
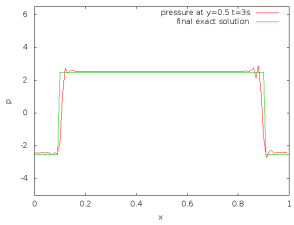
variables	Scheme 4	Scheme 5	Scheme 6
K=2 dt=0.005			
K=2 dt=0.01			
K=2 dt=0.025			

Table 5.7: various dt K=2 N=64 M=322

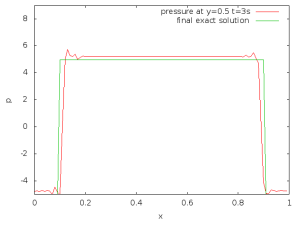
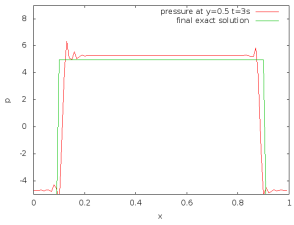
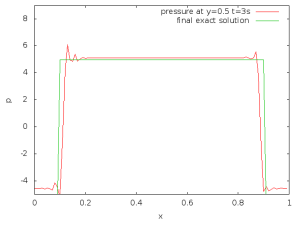
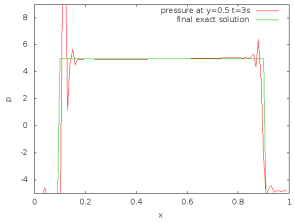
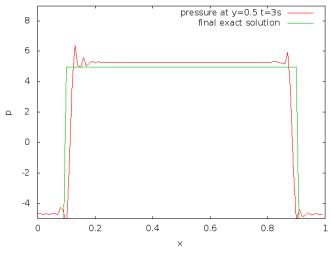
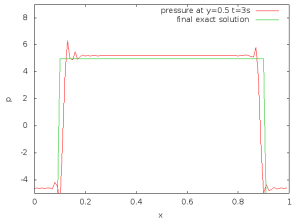
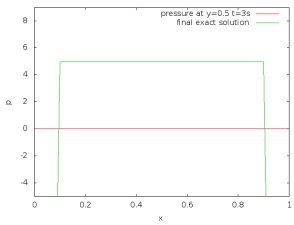
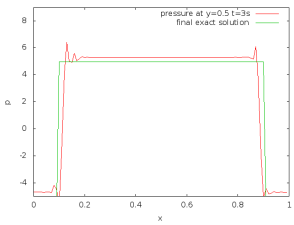
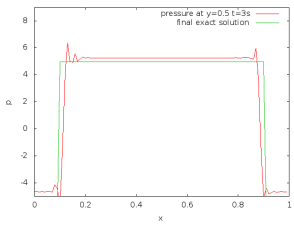
variables	Scheme 4	Scheme 5	Scheme 6
K=4 dt=0.005			
K=4 dt=0.01			
K=4 dt=0.025			

Table 5.8: various dt K=4 N=64 M=322

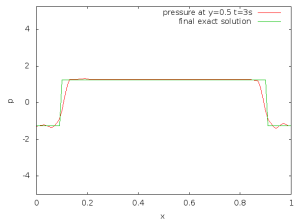
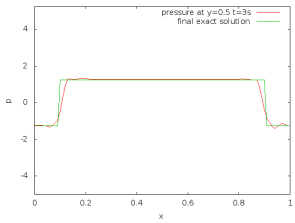
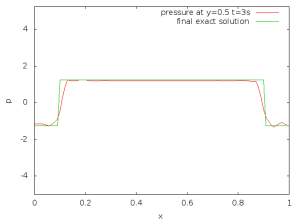
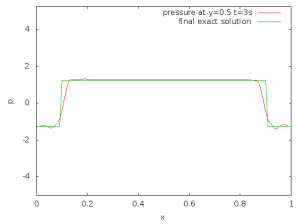
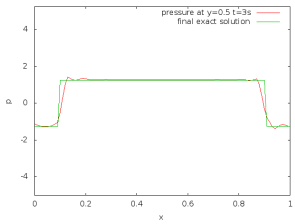
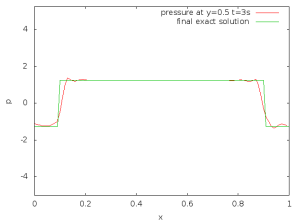
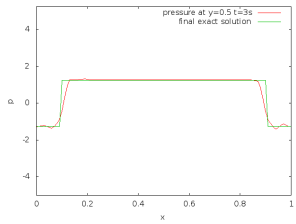
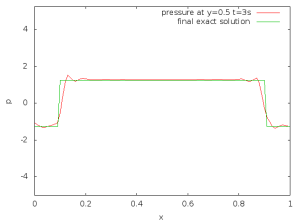
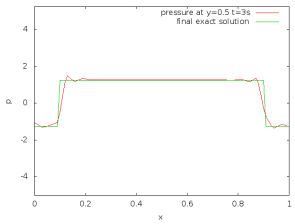
variables	Scheme 4	Scheme 5	Scheme 6
K=1 dt=0.005			
K=1 dt=0.01			
K=1 dt=0.025			

Table 5.9: various K,dt N=32 M=322

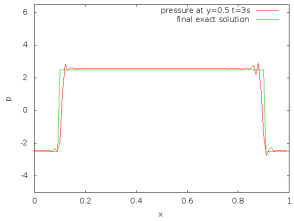
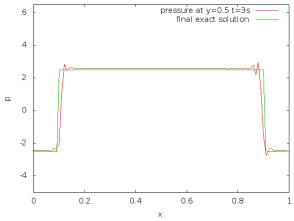
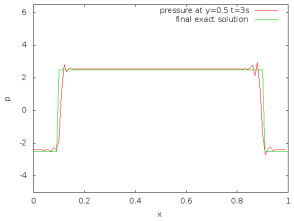
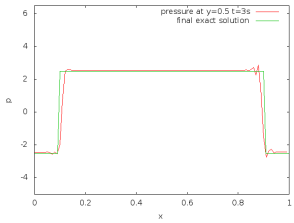
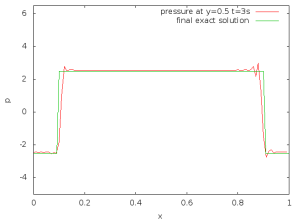
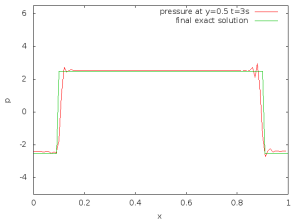
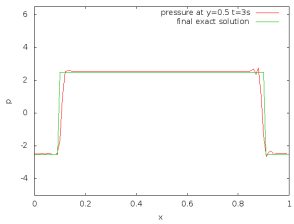
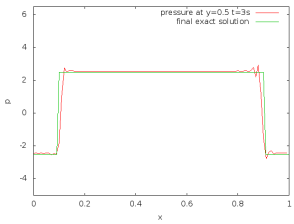
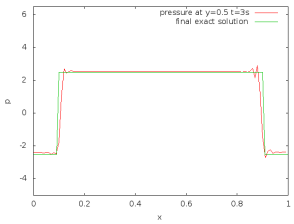
variables	Scheme 4	Scheme 5	Scheme 6
M=161 dt=0.01 K=2			
M=322 dt=0.01 K=2			
M=643 dt=0.01 K=2			

Table 5.10: various M K=2 dt=0.01 N=64

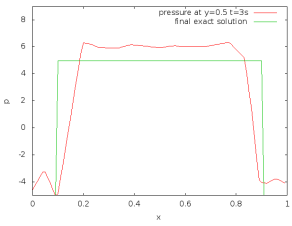
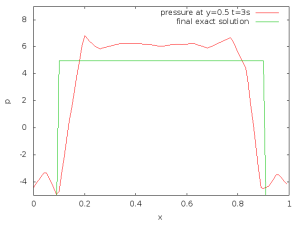
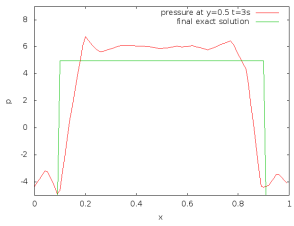
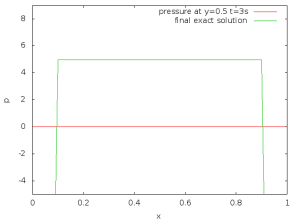
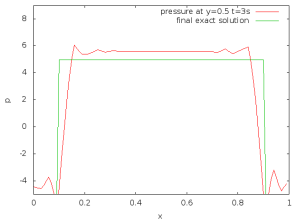
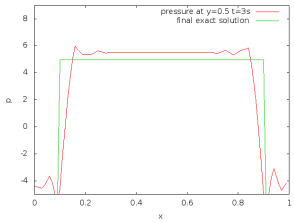
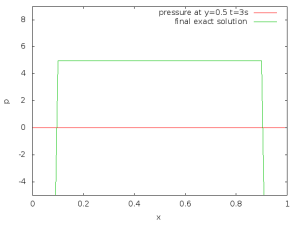
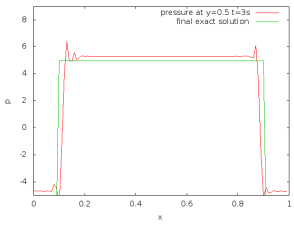
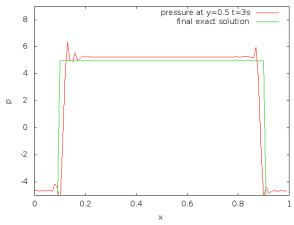
variables	Scheme 4	Scheme 5	Scheme 6
N=16 M=80 dt=0.025 K=4			
N=32 M=161 dt=0.025 K=4			
N=64 M=322 dt=0.025 K=4			

Table 5.11: various mesh finess $\frac{h_x}{h_s}$ K=4 dt=0.025

5.4 Unsteady state - Ellipse

To test our schemes in a dynamical case, we have started from the structure in the shape of an ellipse and we have let it relax to a form of circle. With an impervious membrane, the volume of fluid inside a membrane should be the same at the beginning of the test as in the end of the test, therefore if we start with an ellipse:

$$\begin{aligned} r_a &= 0.3 \\ r_b &= 0.4 \\ S &= \pi r_a r_b \end{aligned} \tag{5.3}$$

we should end up with a circle with the same volume and the radius $r = \sqrt{r_a r_b}$. We have run the same test of stability as in the circle case. Unfortunately the scheme doesn't conserve the mass inside of the structure, so the final solution after 5s doesn't coincide with the expected solution.

5.4.1 Stability

The stability in dynamical case does not differ much from the stability of the statical case. We have computed the development of the system in 5s. In one case ($N = 16, M = 80, dt = 0.005, K = 4$), after some time (around 0.7s of computational time) the instabilities have arisen, but we think that this is due to the fact that the mesh was too rough.

A	pb4			pb5			pb6		
K / dt	1	2	4	1	2	4	1	2	4
0.025	x	x	x						
0.01		x	x						
0.005			x						

Table 5.12: test (A): N=64, M=322, various K

B	pb4			pb5			pb6		
K / dt	1	2	4	1	2	4	1	2	4
0.025		x	x						
0.01			x						
0.005									

Table 5.13: test (B): N=32, M=322, various K

C	pb4			pb5			pb6		
M	161	322	643	161	322	643	161	322	643
dt — $\frac{h_x}{h_s}$	1	2	4	1	2	4	1	2	4
0.025	x	x	x						
0.01		x	x						
0.005									

Table 5.14: test (C): K=2, N=64, various ratio $\frac{h_x}{h_s} = 1, 2, 4$

D	pb4			pb5			pb6		
N	16	32	64	16	32	64	16	32	64
dt M	80	161	322	80	161	322	80	161	322
0.025	x	x	x						
0.01		x	x						
0.005			x						

Table 5.15: test (D): $\frac{h_x}{h_s} = 2$, K=4, various mesh finess

5.4.2 Comparison of geometries of results using problem 4-6

We have compared the structure position for the problem 4-6 after 1s and 5s of computation. On the picture 5.1 we can see, that schemes 4 and 5 give after 1s of computation (the ellipse hasn't relaxed yet to the form of circle) almost same results, while the relaxation of the scheme with inertial member (Problem 6) is much slover (as we would expect, when we encounter inertia).

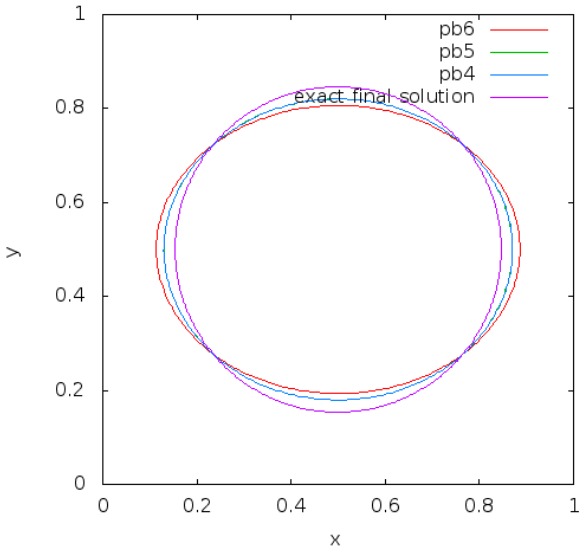


Figure 5.1: K=2 dt=0.005 N=64 M=322 T=1s

On the picture 5.2 we can see, that all results almost coincide, when we arrive to the steady state. That is not surprising, because when we were introducing

the analytical solution of the case when the structure is a circle, we found out, that in steady state the problems 4-6 has the same analytical solution.

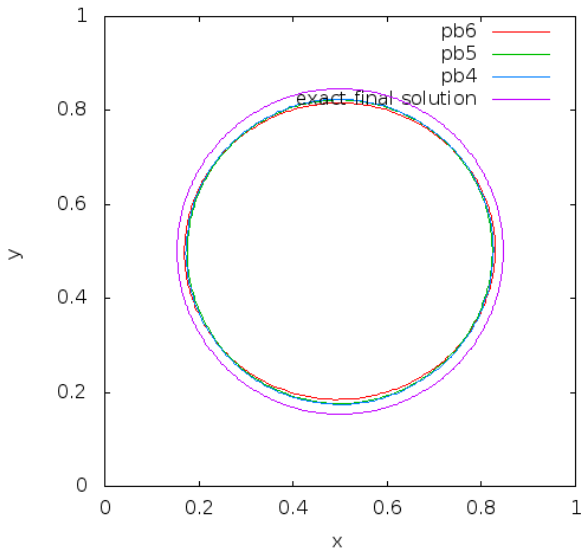


Figure 5.2: K=2 dt=0.01 N=32 M=322 T=5s

We can also see, that the loss of mass inside of the structure is smaller, when we use finer mesh (see 5.3, 5.4).

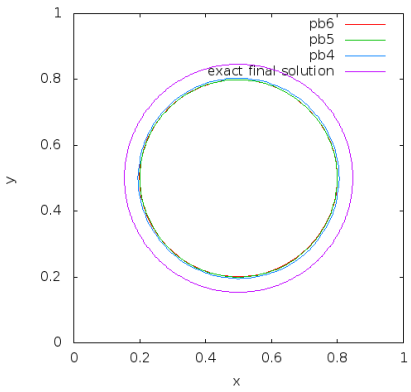


Figure 5.3: K=4 dt=0.001 N=32 M=161 T=5s

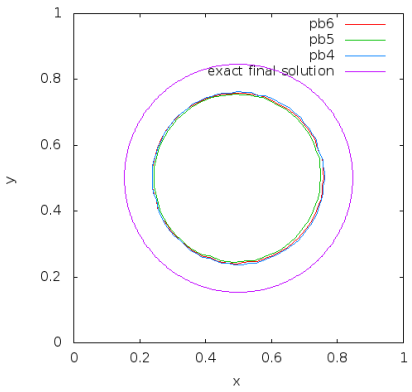


Figure 5.4: K=4 dt=0.001 N=16 M=80 T=5s

5.5 Test - tube

The structure in this case are two lines. The boundary conditions for fluid domain are fixed Dirichlet for velocity (quadratic profile) on the side of inlet, the outlet is free.

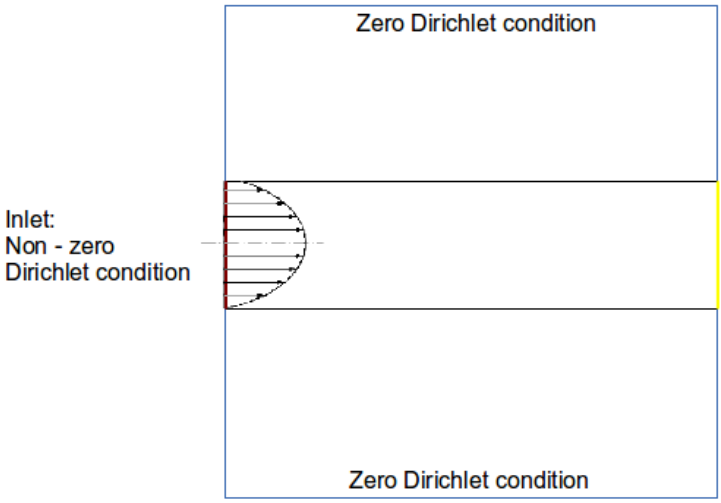


Figure 5.5: Domain with boundary conditions for tube test

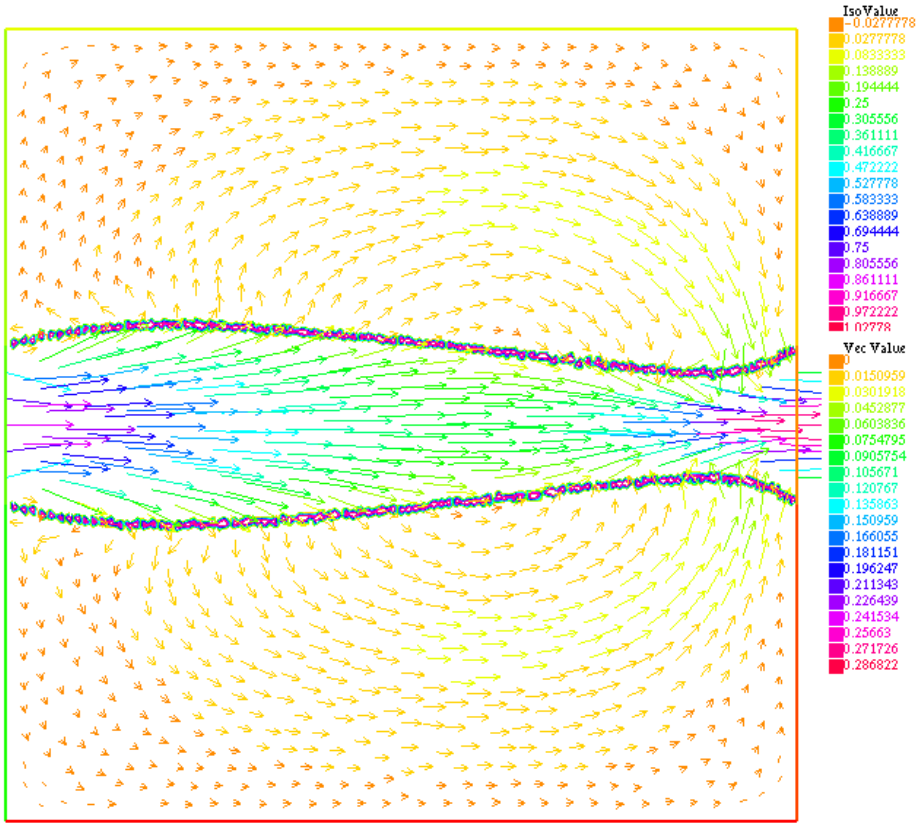


Figure 5.6: fluid velocity in tube (problem 6, dt=0.01, K=2, N=32, M=100)

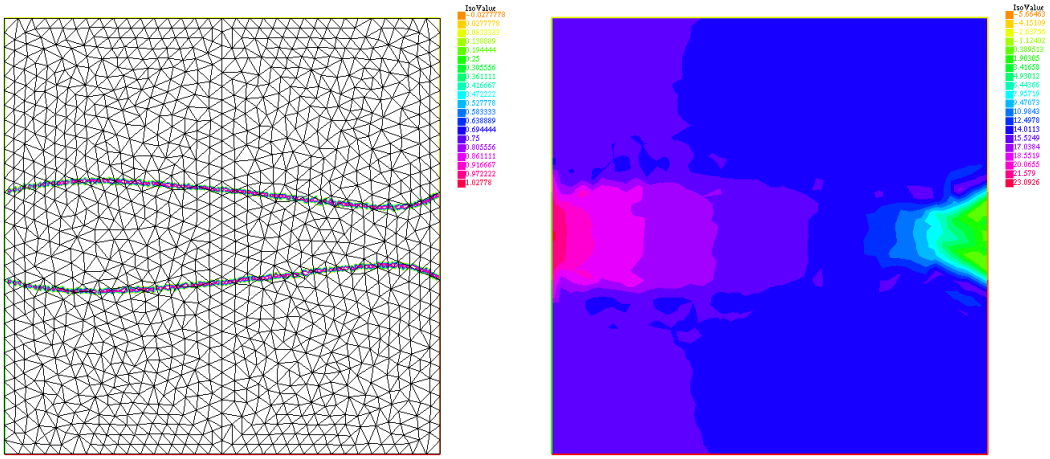


Figure 5.7: Mesh of problem, fluid pressure(problem 6, dt=0.01, K=2, N=32, M=100)

When we compare the results for different schemes, we can see, that while the shemes 4 and 5 give almost same results, the structure computed by scheme 6 with inertial member is deformed by flow a bit more quickly. It almost seems to be in conflict with the result of an Ellipse test. In fact, it is not, because in the case of ellipse, the cause of the motion is the structure itself. Elastic force is deforming it and the inertial member is weakening this effect. Fluid moves as a result of the structure deformation. In the tube case, the cause of the motion is the fluid, structure's elasticity is opposing this tendecy and this opposing is weakened by the inertial effect.

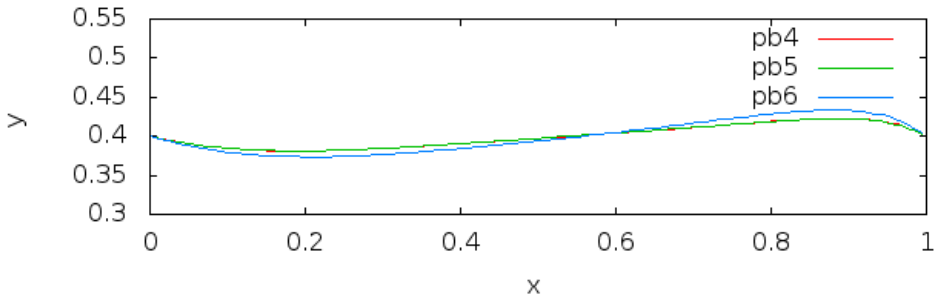


Figure 5.8: Comparison of pb4-pb6 results (bottom part of structure) after 1s of computation

5.6 Geometry of aneurysma

We wanted to design a situation simmilar to growth of aneurysm in vessel. Here, the vessel is not elastic, the elastic structure is only on the place of aneurysm. If we wouldn't let the top boundary of the domain as free outlet, due to the incompressibility of the fluid, we wouldn't see any growth.

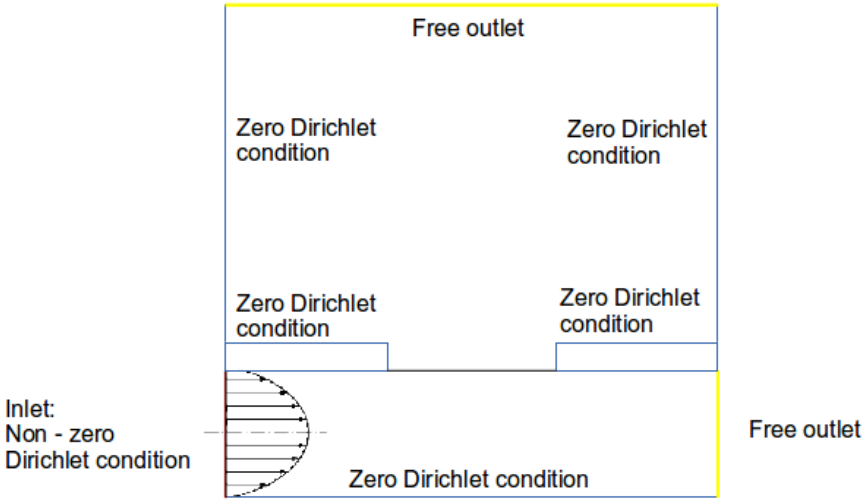


Figure 5.9: Domain with boundary conditions for Aneurysm test

On the picture 5.12 we can see the aneurysm growth during 10s for different ratios of vessel and aneurysm diameter. The structure was captured in interval of 0,25s of comutational time. The bigger the aneurysm neck is compared to the vessel diameter the bigger it grows. Because the Immersed boundary does not conserve mass inside of the structure (we can say that it slithly "leaks", we won't arrive to the absolutely stationary state).

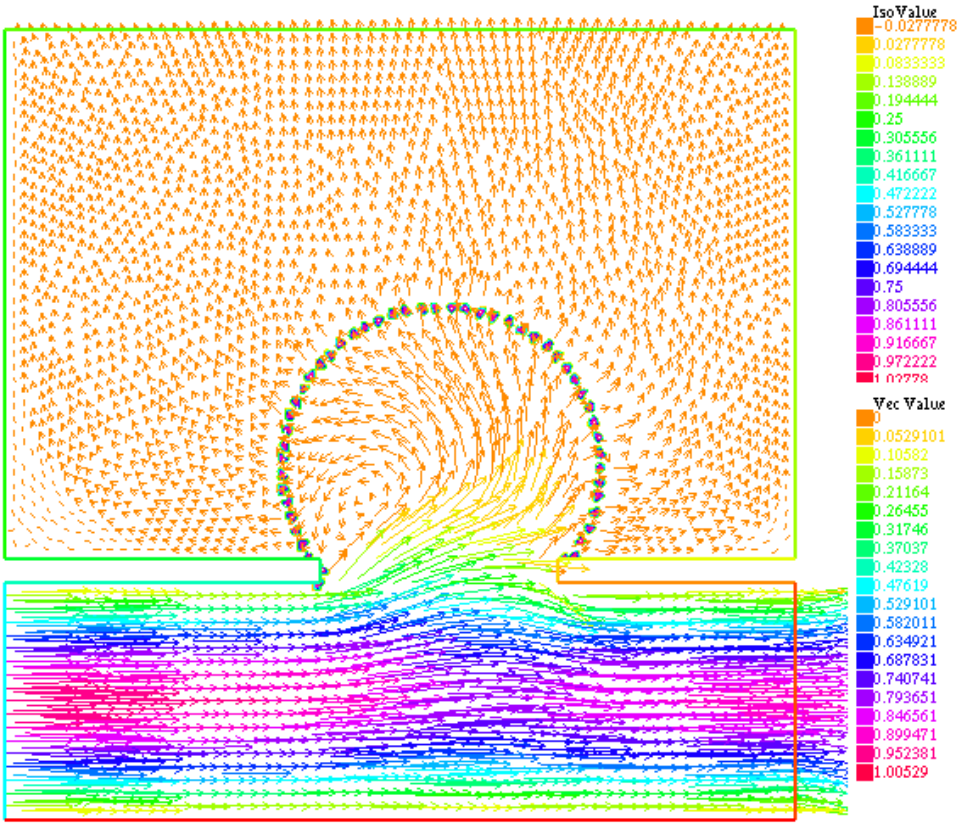


Figure 5.10: Pb5 - almost stationary state, velocity

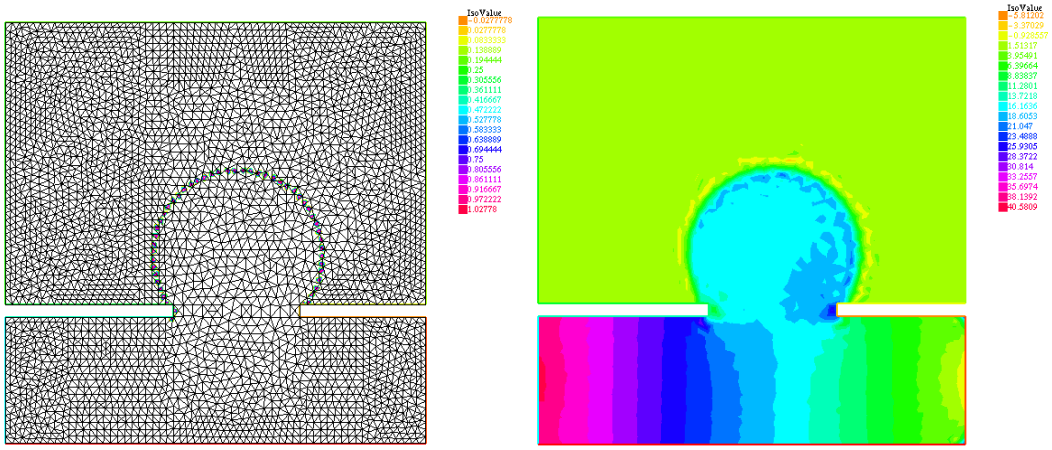


Figure 5.11: Pb5 - almost stationary state, structure points in mesh and pressure inside an aneurysm

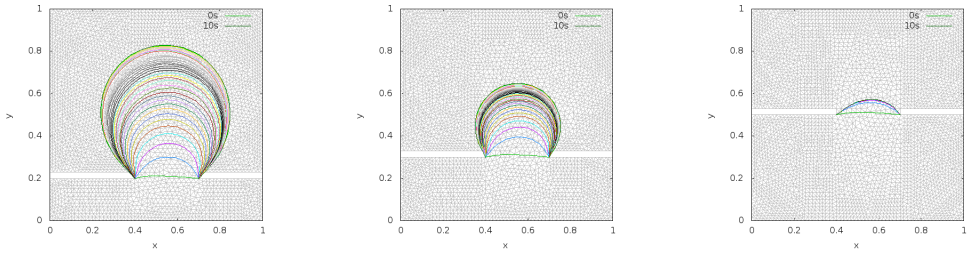


Figure 5.12: problem 5 - aneurysm growth for different ratios of vessel and aneurysm diameter

The advantage of the fact, that our fluid and structure meshes do not have to coincide, is that it is quite easy to implement the growth of anurysma against some barrier, for example a bone. While most of the hemodynamics models are computed in vacuum, our model is immersed in the fluid, in which we can put some solid obstacle simply by making a "hole" in our computational domain. We would like our aneurysm to slide on that obstacle, which we haven't implemented yet (zero Dirichlet condition will cause, that when the structure touches the obstacle, it sticks there, because the interpolation of fluid velocity in that place would be zero). But even with the no-slip condition, the results look interesting (see 5.13).

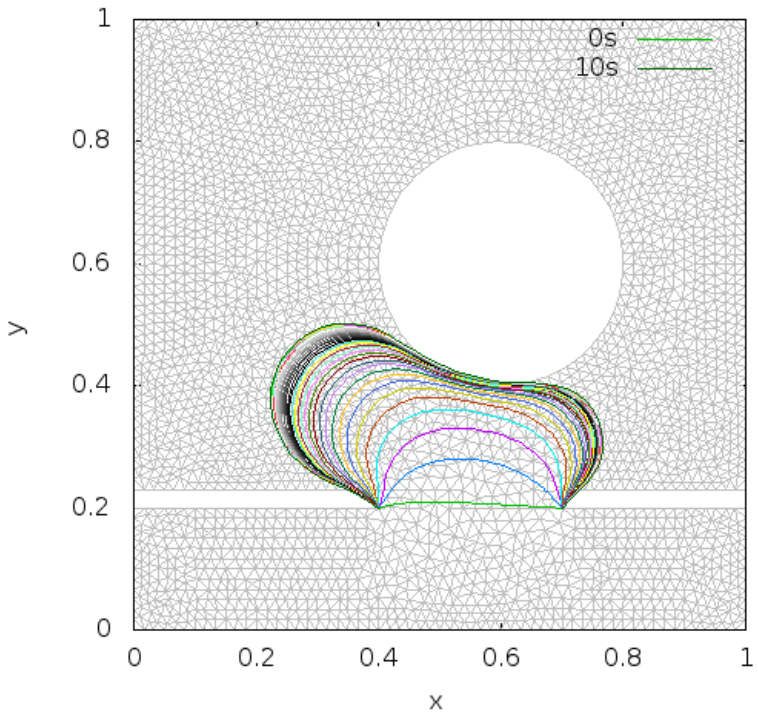


Figure 5.13: problem 5 - aneurysm growth near a bone

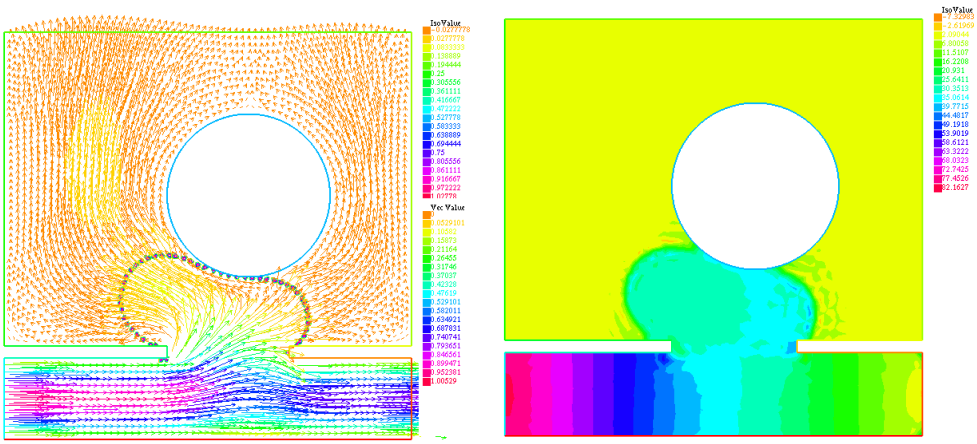


Figure 5.14: problem 5 - aneurysm near a bone - velocity and pressure

Chapter 6

Conclusion

We have implemented the interaction of 2-dimensional fluid with 1-dimensional structure membrane with simple elasticity model. At first we have used a scheme from [4], which is conditionally stable under certain CFL condition (3) (depending on fluid mesh finess and structure mesh finess, time step and viscosity and the elasticity parameter). Then we have proposed two possible modifications. The first scheme had a simple structure update method, only based on velocity of the fluid in the structural point (see definition (9)):

$$\frac{X_{hi}^{n+1} - X_{hi}^{n}}{dt} = \vec{u}_h^{n+1}(X_{hi}^n(t), t) \quad \forall i = 0..m \tag{6.1}$$

The first step of modification lies in additional structure problem solving, instead of simple structural update (see definition(10)):

$$\rho_s \varepsilon(\frac{\dot{X}_h^{n+1} - \vec{u}_h^{n+1}}{dt}, v_s)_s + K(\frac{\partial X_h^{n+1} - X_h^n)}{\partial s}\frac{\partial v_s}{\partial_s})_s = 0 \tag{6.2}$$

The second step of modification (while we preserve the modified structure position update) lies in adding the inertial term to the fluid equation, so that we wouldn't consider the structure membrane massless (see definition (11)):

$$\begin{aligned} \rho_f(\frac{u_h^{n+1} - u_h^n}{dt}, v) + \mu(\nabla u_h^{n+1}, \nabla v) - (div(v), p_h^{n+1}) \\ + \frac{\rho_s}{dt}\epsilon(u^{n+1}(X^n), v(X^n))_s - \frac{\rho_s}{dt}\epsilon(\dot{X}^n, v(X^n))_s \\ = \langle F_h^{n+1}, v\rangle \end{aligned} \tag{6.3}$$

$$\rho_s \varepsilon(\frac{\dot{X}_h^{n+1} - u_h^{n+1}}{dt}, v_s)_s + K(\frac{\partial (X_h^{n+1} - X_h^n)}{\partial s}\frac{\partial v_s}{\partial_s})_s = 0 \tag{6.4}$$

We have proved the unconditional stability of the two proposed modified schemes, then we have implemented the computation off all three shemes, using several computational methods:

- Immersed boundary method (to simulate fluid-structure interaction)
- Brezzi-Pitkaranta stabilization (to make Stokes Problem Coercive)
- Pseudo-Compressibility method (to make the pressure unique even for a constant)

We have ran two kinds of tests to show the stability on numerical test. The results are in sections: 5.1 (test of stationary cicrcle), 5.4 (dynamic test of ellipse). We have verified the stability in many combinations of fluid and structure mesh finess, time step and elasticity constant. Then we have implemented the computation of more "physiological" geometries. We have tried tube (in section 5.5) and aneurysm-like geometry (in section 5.6), to see the evolution of the structure and to compare it with our expectation. We have simulated the aneurysm growth for various ratios of aneurysm and vessel diameter. The smaller the aneurysm diameter was compared to the vessel diameter, the less it grew (as was expected) or even stopped the growth in some stationary state, which grew further (very slowly, compared to the original growth) only because of the fact, that Immersed boundary method does not conserve mass inside of the structure.

We have gone through many steps of computational hemodynamics: from scheme-design, proof of its stability and implementation of its computation to application onto various geometries, but some further analysis still remain:

- add the non-linear member of Navier-Stokes Equations
- generalize the scheme for 3 dimensional meshes
- try more complex elasticity models
- modify the boundary conditions of the domain, so that the behaviour of the structure when it touches an obstacle would be more interesting

The last one is very interesting. Many hemodynamics models are computed in a vacuum, so the problem of the growing aneurysm touching a bone is hard to simulate. In our case, the structure mesh does not have to coincide with the fluid mesh, but it can move independently, therefore it can hit the end of the fluid computational domain (or hole in it - representing bone) and slide on it. In our implementation, we use zero dirichlet condition on the domain boundary, so when the structure meets the fluid domain limits, the interpolation of fluid velocity in the structure point would be zero and the structure won't move anymore (stays stuck on the bone). But in future we could try to modify the boundary condition, so that only the normal component (normal to the domain boundary) of the velocity would be zero and the structure could slide.

Bibliography

[1] *the American Center for Spine and Neurosurgery aneurysm clipping/coiling.* `http://www.acsneuro.com/surgeries/brain_detail/aneurysm_clipping_coiling`, 2015. Accessed: 2015-01-10.

[2] M. A.Fernández, *Incrmental displacement-correction schemes for incompressible fluid-structure interaction*, Numer. Math., 123 (2013), pp. 21–65.

[3] C.S.Peskin and D. McQueen, *A three-dimensional computational method for blood flow in the heart*, J. Comput. Phys, 82 (1989), pp. 372–405.

[4] L. D. Boffi, L.Gastaldi, *Numerical stability of the finite element immersed boundary method*, World Scientific, 17 (2007), pp. 1479 – 1505.

[5] T. A. Davis, *Umfpack v4.3—an unsymmetric-pattern multifrontal method*, ACM Transactions on Mathematical Software (TOMS), 30 (2004), pp. 196–199.

[6] M. D.Boffi, F. Brezzi, *Finite elements for the stokes problem.* `http://www.imati.cnr.it/brezzi/papers/BBF-Stokes-CIME.pdf`.

[7] D.Lombardi and J.-F. Gerbeau, *Le systéme cardiovasculaire en vue de la simulation (cours modélisation directe et inverse en hemodynamique)*, Université Paris 6 and École Polytechnique, (2014).

[8] J.-F. Gerbeau, *Cours modélisation directe et inverse en hemodynamique*, Université Paris 6 and École Polytechnique, (2013-2014).

[9] ——, *A digest of finite element method for incompressible fluid dynamics for cours modelisation directe et inverse en hemodynamique*, Université Paris 6 and École Polytechnique, (2014).

[10] F. Hecht, *New development in freefem++*, J. Numer. Math., 20 (2012), pp. 251–265.

[11] B. G. J. Mikhal, *Immersed boundary method for pulsatile transitional flow in realistic cerebral aneurysms*, Computers and Fluids, 91 (2014), pp. 144 – 163.

[12] M. J.Hron, *Fluid-structure interaction with applications in biomechanics*, Nonlinear Analysis: Real World Application, 8 (December 2007), pp. 1431–1458.

[13] O. J. J.T. Oden, *Stability of some miced finite element methods for stokesian flows*, Comp. Meth. in Applied Mech. and Engineer., 43 (1984), pp. 231–247.

[14] D. Kuzmin and J. Hamalainen, *Finite Element Methods for Computational Fluid Dynamics: A Practical Guide*, Computational Science and Engineering, Society for Industrial and Applied Mathematics, 2014.

[15] F. N. P. Causin, J.F. Gerbeau, *Added-mass effect in the design of partitioned algorithms for fluidstructure problems*, Elsevier, (December 2005).

[16] A. Quarteroni and col., *A reduced computational and geometrical framework for inverse problems in hemodynamics*, Int. J. Numer. Meth. Biomed. Engng., 29 (2013), pp. 741–776.

[17] J. Shen, *Pseudo-compressibility methods for the unsteady incompressible navier-stokes equations.*

[18] Y.Bazilevs, *A fully coupled fluid-structure interaction simulation of cerebral aneurysms*, Comput. Mech., 46 (2010), pp. 3–16.

List of Figures

List of Tables