Numerical simulation of compressible flows with immersed boundaries using Discontinuous Galerkin methods

Bachelor thesis by Simone Stange, Department of Computational Engineering
March 25, 2016
Supervisors: Prof. Dr.-Ing. Martin Oberlack, Dr.-Ing. Björn Müller
Department for Fluid Dynamics

TECHNISCHE
UNIVERSITÄT
DARMSTADT

fdy

Simone Katharina Stange
Matrikelnummer: 2012232
Studiengang: B.Sc. Computational Engineering

Bachelorarbeit
Thema: Numerical simulation of compressible flows with immersed boundaries using Discontinuous Galerkin methods

Eingereicht: March 25, 2016

Betreuer: Dr.-Ing. Björn Müller

Prof. Dr.-Ing. Martin Oberlack
Fachgebiet für Strömungsdynamik
Fachbereich Maschinenbau
Technische Universität Darmstadt
Otto-Berndt-Str. 2
64287 Darmstadt

Ehrenwörtliche Erklärung

Erklärung zur Abschlussarbeit gemäß § 22 Abs. 7 APB der TU Darmstadt

Hiermit versichere ich, Simone Katharina Stange, die vorliegende Bachelor-Thesis ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.
In der abgegebenen Thesis stimmen die schriftliche und elektronische Fassung überein.

Datum: Unterschrift:

Declaration of Academic Integrity

Thesis Statement pursuant to § 22 paragraph 7 of APB TU Darmstadt

I herewith formally declare that I have written the submitted thesis independently. I did not use any outside support except for the quoted literature and other sources mentioned in the paper. I clearly marked and separately listed all of the literature and all of the other sources which I employed when producing this academic work, either literally or in content. This thesis has not been handed in or published before in the same or similar form.
In the submitted thesis the written copies and the electronic version are identical in content.

Date: Signature:

Abstract

Informationen zu Inhalten der Zusammenfassung entnehmen Sie bitte Kapitel 6.1 des Skripts zur Veranstaltung *Wissenschaftliches Arbeiten und Schreiben für Maschinenbau-Studierende*.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Contents

Abbreviations

2D	2-dimensional
3D	3-dimensional
BoSSS	Bounded Support Spectral Solver
CFL	Courant-Friedrichs-Lewy
CNS	Compressible Navier-Stokes
DG	Discontinuous Galerkin
DG(·)CpD(·)	Degree (·), Cells per Direction (·)
FEM	Finite Element Method
FVM	Finite Volume Method
IBM	Immersed Boundary Method
ODE	Ordinary Differential Equation
RK	Runge-Kutta
RKDG	Runge-Kutta Discontinuous Galerkin

Nomenclature

Notation	Description
$\mathcal{A}$	Physical region of calculation area
$(\cdot)^*$	Dimensionless quantities
$(\cdot)_\infty$	Reference quantities
a	Speed of sound
$\mathcal{B}$	Void region of calculation area
$\mathbf{B}$	Source term
c	Concentration
C_D	Drag coefficient
C_L	Lift coefficient
c_p	Specific heat capacity at constant pressure
c_v	Specific heat capacity at constant volume
d	Drag force
e	Specific inner energy
$\mathbf{F}_i^c$	Convective flux
Fr	Froude number
$\mathbf{F}_i^v$	Viscous flux
γ	Heat capacity ratio
$\bar{h}$	Specific enthalpy
$\mathcal{I}$	Immersed boundary
$\mathcal{K}$	Cell domain
l	Lift force
μ	Dynamic viscosity
Ω	Problem domain
Ω_h	Discretised problem domain
P	Polynomial degree
p	Pressure
φ	Level set function
Φ	Cell-local test function
Pr	Prandtl number
q_i	Heat flux
r	Radius
Re	Reynolds number
ρ	Density

Notation	Description
ρE	Inner energy
s	Specific entropy
St	Strouhal number
T	Temperature
t	Time
τ_{ij}	Viscous stress tensor
W^*	Wake separation length
$\mathbf{x}$	Spatial coordinate vector, $\mathbf{x} = (x, y)^\intercal \in \Omega$

1 Introduction

The chair of fluid dynamics of the Technical University Darmstadt mainly deals with research in fluid mechanical problems, e.g. multiphase flows, turbulence or thermodynamics. In recent years they have been developing a Bounded Support Spectral Solver (BoSSS), that we will use in this thesis for the simulation of compressible flows with immersed boundaries. BoSSS is based on a Runge-Kutta Discontinuous Galerkin (RKDG) method that allows distinct grid types and dimensions (2-dimensional (2D), 3-dimensional (3D)) and arbitrarily defined polynomial order without losing the ability of being highly parallelisable.

1.1 Main Goals

This thesis aims at validating the Compressible Navier-Stokes (CNS) solver for both inviscid and viscid compressible flows using a grid defined through immersed boundaries. For inviscid flows it has already been verified thoroughly by [1], though not for the specific example we will concentrating on. Within this work, I will consider the flow around a cylinder in a 2D mesh. First, I will use the inviscid flow in order to validate BoSSS concerning robustness and convergence, using the entropy as an error criterion.

After having formed a firm basis for the inviscid cylinder flow simulation we can then consider the same example for viscid flows using different Reynolds numbers (20, 40, 100, 200) and compare the obtained results as has already been done for BoSSS by [2], who instead of an Immersed Boundary Method (IBM) used curved elements to define the grid. We will therefore comply with the structure given by [2].

1.2 Outline

In order to gain a fundamental background knowledge of the methods that are used during the simulation, I will give some theory in the first part of this work. The second part we will devote to the simulation and validation followed by a conclusion.

First, I will enumerate the important equations of fluid mechanics and their dimensionless forms in chapter 2. After that I will explain the Discontinuous Galerkin (DG) method using an example taken from [1] and the Runge-Kutta (RK) method in chapter 3. The methodological part of the thesis will then be completed with an introduction to the IBM in chapter 4.

The second part starts with the validation of BoSSS concerning the Euler equations or inviscid flow, respectively, considering the robustness and convergence of the computations in chapter 5. In chapter 6, I will first give some theory about the viscid flow around a cylinder followed by

the results and analysis of the main task of this work.

The thesis will then be closed by a short conclusion and outlook for following works.

2 Fundamentals

2.1 Essential Equations

In the following we will introduce the essential equations that form the basis of all methods and results discussed in this thesis.

2.1.1 Equations of State

First, I will enumerate the basic thermodynamic equations of state that describe the relations between the specific inner energy e, the local temperature T, the specific enthalpy $\bar{h}$, the pressure $p = p(\rho, e)$, the density ρ and the specific entropy s. Assuming that all material parameters are constant, we receive

$$e = c_v T \tag{2.1}$$

$$\bar{h} = c_p T = e + \frac{p}{\rho} \tag{2.2}$$

with the material parameters c_p and c_v as specific heat capacities at constant pressure and volume, respectively. Using these relations, we can define the heat capacity ratio

$$\gamma = \frac{c_p}{c_v} = \frac{\bar{h}}{e}, \tag{2.3}$$

e.g. $\gamma = 1.4$ for standard air.

Another essential equation is the relation for the specific entropy

$$T\,ds = de + p\,d\rho^{-1}. \tag{2.4}$$

2.1.2 Ideal Gas Law

The equations of state mentioned above are not yet complete as there is missing a law for the pressure $p = p(\rho, e)$. In this thesis we will limit ourselves to the ideal gas law as we are only

modelling standard air ($\gamma = 1.4$).
The ideal gas law is defined as

$$p = (\gamma - 1)\rho e \tag{2.5}$$

with $\rho e \in \mathbb{R}^+$ denoting the inner energy.

A definition of the speed of sound is given by

$$a = \sqrt{\left.\frac{\partial p}{\partial \rho}\right|_s}. \tag{2.6}$$

Using this definition combined with the ideal gas law and an equation for the change of entropy as found in [1] leads to the useful relations @TODO: isentropy!

$$a^2 = \gamma \frac{p}{\rho} \tag{2.7}$$

$$\frac{p}{\rho^\gamma} = \text{const.} \tag{2.8}$$

2.1.3 Navier-Stokes Equations

The CNS equations in conservative forms read as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_i^c(\mathbf{U})}{\partial x_i} - \frac{\partial \mathbf{F}_i^v(\mathbf{U}, \nabla \mathbf{U})}{\partial x_i} = \mathbf{B}, \tag{2.9}$$

with $\mathbf{U}$ as the conserved flow variables, $\mathbf{F}_i^c$ and $\mathbf{F}_i^v$ as the convective and viscous fluxes and $\mathbf{B}$ as source terms:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho v_j \\ \rho E \end{pmatrix}, \quad \mathbf{F}_i^c = \begin{pmatrix} \rho v_i \\ \rho v_i v_j + p\delta_{ij} \\ v_i(\rho E + p) \end{pmatrix}, \quad \mathbf{F}_i^v = \begin{pmatrix} 0 \\ \tau_{ij} \\ \tau_{ij} + q_i \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ \rho F_j \\ \rho F_j v_j + Q_i \end{pmatrix}. \tag{2.10}$$

In addition to the denotations in section 2.1.1 we have F_j as body forces, Q_i as heat sources, the viscous stress tensor

$$\tau_{ij} = \mu \left[\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) - \frac{2}{3} \frac{\partial v_k}{\partial x_k} \delta_{ij} \right] \tag{2.11}$$

with the dynamic viscosity μ and the heat flux q_i modelled using Fourier's Law

$$q_i = k\frac{\partial T}{\partial x_i}. \tag{2.12}$$

Euler Equations

Regarding only compressible inviscid flow, the viscous fluxes and the source terms dissolve ($\mathbf{F}_i^v = \mathbf{B} = \mathbf{0}$) and the Navier-Stokes equations simplify to the so-called Euler equations:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_\mathbf{x}^\mathbf{c}(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{F}_\mathbf{y}^\mathbf{c}(\mathbf{U})}{\partial y} = 0. \tag{2.13}$$

2.2 Dimensionless Measures

For it is much easier to handle dimensionless PDEs, we will introduce some dimensionless measures as found in [3]. In order to derive these, we need some reference quantities: L_∞ as a reference length, a reference velocity V_∞, reference density ρ_∞, reference volume force g_∞, reference viscosity μ_∞, reference thermal conductivity coefficient k_∞ and the gas constant R. All other reference quantities can be derived from those. In the following all dimensionless quantities will be marked with an asterisk $(\cdot)^*$:

$$\begin{aligned}
&t^* = \frac{V_\infty}{L_\infty}\cdot t, \quad x_i^* = \frac{1}{L}\cdot x_i, \quad v_i^* = \frac{1}{V_\infty}\cdot v_i, \quad \rho^* = \frac{1}{\rho_\infty}\cdot \rho, \quad p^* = \frac{1}{\rho_\infty V_\infty^2}\cdot p, \\
&\mu^* = \frac{1}{\mu_\infty}\cdot \mu, \quad k^* = \frac{1}{k_\infty}\cdot k, \quad T^* = \frac{R}{V_\infty^2}\cdot T, \quad F_j^* = \frac{1}{g_\infty}\cdot F_j, \quad \rho E^* = \frac{1}{\rho_\infty V_\infty^2}\cdot \rho E, \\
&Q_i^* = \frac{L}{V_\infty^3}\cdot Q.
\end{aligned} \tag{2.14}$$

In order to derive the non-dimensional form of the CNS equation, we also need the dimensionless operators:

$$\frac{\partial}{\partial t} = \frac{\partial t^*}{\partial t}\frac{\partial}{\partial t^*} = \frac{V_\infty}{L}\frac{\partial}{\partial t^*}, \tag{2.15}$$

$$\frac{\partial}{\partial x_i} = \frac{\partial x_i^*}{\partial x_i}\frac{\partial}{\partial x_i^*} = \frac{1}{L}\frac{\partial}{\partial x_i^*}, \tag{2.16}$$

$$\nabla = \frac{1}{L}\nabla^*. \tag{2.17}$$

2.2.1 Non-dimensional Ideal Gas Law

For a closed system of equations we have to use the ideal gas law in dimensionless form:

$$p^* = \rho^*(\gamma - 1)e^* \tag{2.18}$$

$$= (\gamma - 1)\left(\rho E^* - \frac{1}{2}\rho^* \mathbf{v}^{*2}\right). \tag{2.19}$$

2.2.2 Dimensionless Navier-Stokes Equations

As we now have all required measures and operator we can use them to define the dimensionless relations

Reynolds Number $\mathrm{Re} = \dfrac{\rho_\infty V_\infty L}{\mu_\infty} \propto \dfrac{\text{inertia forces}}{\text{viscous forces}}$,

Froude Number $\mathrm{Fr} = \dfrac{V_\infty}{\sqrt{gL}} \propto \dfrac{\text{body inertia}}{\text{gravitational forces}}$,

Prandtl Number $\mathrm{Pr} = \dfrac{\mu_\infty c_p}{k_\infty} \propto \dfrac{\text{viscous diffusion rate}}{\text{thermal diffusion rate}}$.

Putting all together we receive the dimensionless Navier-Stokes equations. As they only depend on non-dimensional quantities we can drop the asterisk:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_i^c(\mathbf{U})}{\partial x_i} - \frac{\partial \mathbf{F}_i^v(\mathbf{U}, \nabla \mathbf{U})}{\partial x_i} = \mathbf{B} \tag{2.20}$$

with the dimensionless fluxes

$$\mathbf{F}_i^c = \begin{pmatrix} \rho v_i \\ \rho v_i v_j + p\delta_{ij} \\ v_i(\rho E + p) \end{pmatrix}, \quad \mathbf{F}_i^v = \frac{1}{\mathrm{Re}} \begin{pmatrix} 0 \\ \tau_{ij} \\ \tau_{ij} v_j + \dfrac{\gamma}{\mathrm{Pr}(\gamma - 1)} q_i \end{pmatrix}, \quad \mathbf{B} = \frac{1}{\mathrm{Fr}^2} \begin{pmatrix} 0 \\ \rho F_j \\ \rho F_j v_j \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ Q_j \end{pmatrix}. \tag{2.21}$$

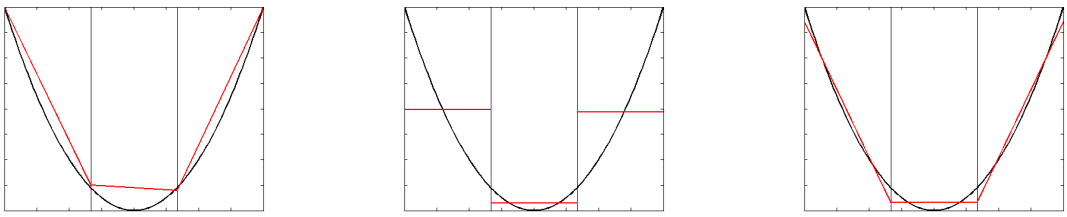
3 The Runge-Kutta Discontinuous Galerkin Method

This thesis deals with the software BoSSS that uses a RKDG method for the numerical approximation of compressible flows. The RKDG method is split into the DG method for space discretization and the RK method as an explicit time discretization. By using an explicit time-marching algorithm, the parallelization is made much easier.

In the following sections we will study the DG and RK methods separately considering simple examples and using the same notation as in [1].

3.1 DG Space Discretization

First, we will study the DG method which can be seen as combination of the Finite Volume Method (FVM) and the Finite Element Method (FEM). It aims at combining the advantages of both methods, namely high-order accuracy and hp-adaptivity (FEM) as well as conservativity (FVM), thus allowing the computation of higher order solutions with adjustable order on each element on a conservative grid. The main concept of the DG method is visualised in fig. 3.1.



(a) First order FEM (b) Zeroth order DG (FVM) (c) First order DG

Figure 3.1.: Comparison of FEM, FVM and DG

As a simple example that we can use to derive the DG formulation we will consider the scalar conservation law

$$\frac{\partial c}{\partial t} + \nabla \cdot \boldsymbol{f}(c) = 0 \tag{3.1}$$

for the concentration $c = c(\boldsymbol{x}, t)$ with $\vec{x} \in \Omega \subset \mathbb{R}^D$ and $t \in \mathbb{R}_0^+$ and a smooth function $\boldsymbol{f} : \mathbb{R} \to \mathbb{R}^D$ that also contains suitable initial and boundary conditions.[1]

3.1.1 Discrete Weak Formulation

Our first step of the DG method will be transferring the partial differential equation (3.1) into a weak formulation. Priorly to this we need a discretization Ω_h of Ω consisting of a tesselation of cells $\{\mathcal{K}_i\}_{i=1,\dots,N}$, where h represents a measure for the size of the cells. Each cell $\mathcal{K}_i$ is of dimension D with an outward unit normal vector $\boldsymbol{n}$.

After having discretised our geometry, we now need a set of cell-local test functions $\left\{\Phi_{i,j}\right\}_{j=1,\dots,M}$ with $\Phi_{i,j} = \Phi_{i,j}(\boldsymbol{x}) : \mathbb{R}^D \to \mathbb{R}$ that forms the basis of polynomials $P_{\mathcal{K}_i}(P)$ with the maximum degree P.

In order to obtain the discrete weak formulation we will now multiply equation (3.1) by $\Phi_{i,j}$, integrate over a cell $\mathcal{K}_i$ and then integrate by parts:

$$\begin{aligned}
\frac{\partial c}{\partial t} + \nabla \cdot \boldsymbol{f}(c) &= 0 \\
\frac{\partial c}{\partial t}\Phi_{i,j} + (\nabla \cdot \boldsymbol{f}(c))\,\Phi_{i,j} &= 0 \\
\int_{\mathcal{K}_i} \frac{\partial c}{\partial t}\Phi_{i,j}\,dV + \int_{\mathcal{K}_i} (\nabla \cdot \boldsymbol{f}(c))\,\Phi_{i,j}\,dV &= 0 \\
\int_{\mathcal{K}_i} \frac{\partial c}{\partial t}\Phi_{i,j}\,dV + \int_{\partial\mathcal{K}_i} (\boldsymbol{f}(c)\cdot\boldsymbol{n})\,\Phi_{i,j}\,dA - \int_{\mathcal{K}_i} \boldsymbol{f}(c)\,\nabla\Phi_{i,j}\,dV &= 0.
\end{aligned} \tag{3.2}$$

Considering that the cell's surface $\partial\mathcal{K}_i$ consists of internal or boundary edges $\left\{\mathcal{E}_{i,e}\right\}_{e=1,\dots,E_i}$ we can rewrite (3.2) as

$$\int_{\mathcal{K}_i} \frac{\partial c}{\partial t}\Phi_{i,j}\,dV + \sum_{e=1}^{E_i} \int_{\mathcal{E}_{i,e}} (\boldsymbol{f}(c)\cdot\boldsymbol{n})\,\Phi_{i,j}\,dA - \int_{\mathcal{K}_i} \boldsymbol{f}(c)\,\nabla\Phi_{i,j}\,dV = 0. \tag{3.3}$$

3.1.2 Numerical Fluxes

As the concentration c is unknown, we need to introduce a modal approximation

$$c(\mathbf{x},t)\,|_{\mathcal{K}_i} \approx \tilde{c}(\mathbf{x},t)\,|_{\mathcal{K}_i} = c_i(x,t) = \sum_{k=0}^{M} c_{i,k}(t)\Phi_{i,k}(\mathbf{x}) \tag{3.4}$$

with the Galerkin approach of identical Ansatz and test functions. For we do not enforce continuity on $\mathcal{E}_{i,e}$ and thus @TODO: $c_{n(i,e)}$ definieren

$$c_i\,|_{\mathcal{E}_{i,e}} =: c^- \neq c^+ := c_{n(i,e)}\,|_{\mathcal{E}_{i,e}} \tag{3.5}$$

we cannot simply insert the approximation into equation (3.3). Therefore we will introduce a monotone, Lipschitz continuous numerical flux function @TODO: erklären $f = f(c^-, c^+, \mathbf{n}) : \mathbb{R}^{D+2} \to \mathbb{R}$ satisfying the consistency property

$$f(c^-, c^+, \mathbf{n}) = -f(c^-, c^+, -\mathbf{n}). \tag{3.6}$$

By including these definitions into (3.3) we receive

$$\int_{\mathcal{K}_i} \frac{\partial c_i}{\partial t} \Phi_{i,j}\, dV + \underbrace{\sum_{e=1}^{E_i} \int_{\mathcal{E}_{i,e}} f\,(c^-, c^+, \mathbf{n})\, \Phi_{i,j}\, dA - \int_{\mathcal{K}_i} \boldsymbol{f}\,(c_i) \cdot \nabla \Phi_{i,j}\, dV}_{=:(\mathbf{f_i})_j} = 0 \tag{3.7}$$

with the discrete operator $\mathbf{f_i} = \mathbf{f_i}(t, \mathbf{c_i}) \in \mathbb{R}$.

Some well-known examples of numerical fluxes contain [4]:

- The Godunov flux
- The Engquist-Osher flux
- The Lax-Friedrichs flux
- The local Lax-Friedrichs flux
- The Roe flux with 'entropy fix',

whereby we will attend to the local Lax-Friedrichs or Rusanov flux, which is defined as

$$f(c^-, c^+, \mathbf{n}) = \frac{\mathbf{f}(c^-) + \mathbf{f}(c^+)}{2} \cdot \mathbf{n} - \frac{C_R}{2}(c^+ - c^-) \tag{3.8}$$

with the coefficient C_R based on a local stability criterion. In this thesis we will use an estimate based on the maximum local wave speed @TODO: cite Toro2009

$$C_R = \max(|\mathbf{u}^+ \cdot \mathbf{n}| + a^-, |\mathbf{u}^- \cdot \mathbf{n}| + a^+) \tag{3.9}$$

with $\mathbf{u}^\pm$ and $a^\pm$ denoting the one-sided normal velocity and the local speed of sound. As the Rusanov flux has a high stability it will be used disregarding that it is prone to numerical diffusion.

3.2 RK Time Discretization

After having studied the spatial discretization, we will now attend to the time discretization, using the RK method.

First of all, we need to reformulate equation (3.7) in order to achieve a system of coupled Ordinary Differential Equation (ODE)s.

$$\int_{\mathcal{K}_i} \frac{\partial c_i}{\partial t} \Phi_{i,j}\, dV + \underbrace{\sum_{e=1}^{E_i} \int_{\mathcal{E}_{i,e}} f\left(c^-, c^+, \mathbf{n}\right) \Phi_{i,j}\, dA - \int_{\mathcal{K}_i} \boldsymbol{f}\left(c_i\right) \cdot \nabla \Phi_{i,j}\, dV}_{=:(\mathbf{f_i})_j} = 0.$$

The first term of the equation above can be reformulated as

$$\begin{aligned}
\int_{\mathcal{K}_i} \frac{\partial c_i}{\partial t} \Phi_{i,j}\, dV &= \int_{\mathcal{K}_i} \frac{\partial}{\partial t} \left(\sum_{k=0}^{M} c_{i,k}(t) \Phi_{i,k}(\mathbf{x}) \right) \Phi_{i,j}\, dV \\
&= \sum_{k=0}^{M} \frac{\partial c_{i,k}}{\partial t} \underbrace{\int_{\mathcal{K}_i} \Phi_{i,k} \Phi_{i,j}\, dV}_{=:(\mathbf{M_i})_{k,j}} \\
&= \mathbf{M_i} \frac{\partial \mathbf{c_i}}{\partial t}
\end{aligned}$$

thus leading to

$$\mathbf{M_i} \frac{\partial \mathbf{c_i}}{\partial t} + \mathbf{f_i} = 0 \tag{3.10}$$

with $\mathbf{M_i} \in \mathbb{R}^{M,M}$ being a cell-local symmetric mass matrix associated with $\mathcal{K}_i$. As we have assumed an orthonormal basis $\left\{\Phi_{i,j}\right\}_{j=1,\ldots,M}$ thus reducing the mass matrix to the identity matrix I, the ODEs simplify to

$$\frac{\partial \mathbf{c_i}}{\partial t} + \mathbf{M_i^{-1}} \mathbf{f_i} = \mathbf{0} \tag{3.11}$$

$$\frac{\partial \mathbf{c_i}}{\partial t} + \mathbf{f_i} = \mathbf{0}. \tag{3.12}$$

Using an explicit RK method of order S we can now advance this system of ODEs and calculate the new coefficients from

$$\mathbf{c_i}(t_1) = \mathbf{c_i}(t_0) - \Delta t \sum_{s=1}^{S} (\alpha)_s \mathbf{k_s}, \tag{3.13}$$

with a known solution at t_0 to a new instant t_1 and $\Delta t = t_1 - t_0$, where

$$\mathbf{k}_s = \mathbf{f}_i \left(t_0 + (\beta)_s \Delta t, \mathbf{c}_i(t_0) + \Delta t \sum_{t=1}^{S} (\mathbf{\Gamma})_{s,t} \mathbf{k}_t \right). \tag{3.14}$$

The coefficients $\alpha \in \mathbb{R}^S$, $\beta \in \mathbb{R}^S$ and $\Gamma \in \mathbb{R}^S$ are specific for each RK method. Those of the most common RK methods are displayed in the Butcher Tableaus in table 3.2. They determine the stability and accuracy of the time integration scheme.
@TODO: cite Butcher (1987), Gottlieb and Shu (1998)

$$\begin{array}{c|ccccc} 0 & & & & & \\ \beta_2 & \Gamma_{21} & & & & \\ \beta_3 & \Gamma_{31} & \Gamma_{32} & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ \beta_s & \Gamma_{s1} & \Gamma_{s2} & \cdots & \Gamma_{s,s-1} & \\ \hline & \alpha_1 & \alpha_2 & \cdots & \alpha_{s-1} & \alpha_s \end{array}$$

Table 3.1.: Butcher Tableau for the Explicit Runge–Kutta Method.

A well-known stability criterion according the explicit Euler time discretization for linear, hyperbolic PDEs, namely the Courant-Friedrichs-Lewy (CFL) criterion, restrains the temporal step-size Δt:

$$\Delta t \leq c_{CFL} \frac{h}{\underline{u}} \tag{3.15}$$

Explicit Euler (first order)

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}$$

Trapezoidal rule (second order)

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

Third order TVD (third order)

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{3} & \frac{1}{3} & & \\ \frac{2}{3} & 0 & \frac{1}{3} & \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array}$$

Classical RK (fourth order)

$$\begin{array}{c|ccccc} 0 & & & & & \\ \frac{1}{2} & \frac{1}{2} & & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & & \\ 1 & 0 & 0 & 1 & & \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{2}{6} & \end{array}$$

Table 3.2.: Butcher Tableaus for different orders of RK

with $\underline{u} \in \mathbb{R}^+$ denoting the largest propagation velocity and a positive constant $c_{CFL} \leq 1$ depending on the applied spatial discretization procedure.

Concerning the Euler equations the largest propagation velocity is given by $\underline{u} = ||\mathbf{u}|| + a$ and by taking the influence of the approximation order P into account we can use @TODO: cite Cockburn and Shu 1991

$$\Delta t \leq \frac{c_{CFL}}{2P+1} \frac{h}{||\mathbf{u}|| + a} \tag{3.16}$$

as a sufficiently accurate estimate for the stability criterion in this thesis.

4 The Immersed Boundary Method

In the following chapter we will study the DG method with immersed boundaries. This chapter is based on [5]

IBMs are characteristic in the way of creating the calculation mesh as they do not rely on body fitted grids but on a level set function φ that cuts the cells into the physical and the void region. It therefore makes the mesh generation much easier, as it only needs a cartesian mesh and a function that approximates the level set. Brought along with the cartesian mesh, it is easily parallelisable, thus rendering it convenient for more complex structures that shall be computed on several processors.

4.1 The DG Scheme with Immersed Boundaries

We regard an implicit representation of an immersed boundary using the level set function φ that parts the calculation area Ω_h into @TODO: align

the physical region $\mathcal{A} = \{\vec{x} \in \Omega_h : \varphi(\vec{x}) > 0\}$,

the void region $\mathcal{B} = \{\vec{x} \in \Omega_h : \varphi(\vec{x}) < 0\}$,

and the immersed boundary $\mathcal{I} = \{\vec{x} \in \Omega_h : \varphi(\vec{x}) = 0\}$

as can be seen in fig. 4.1. In our next step we use the definitions above in (3.7) considering

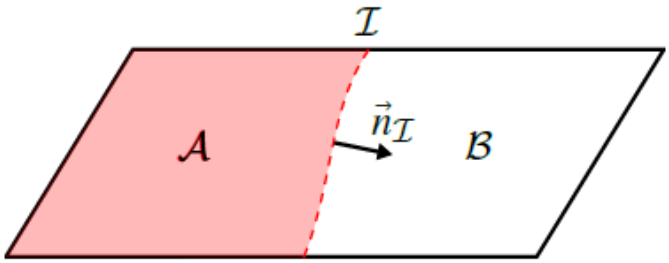


Figure 4.1.: Cut cell with physical (red) and void region (white) [5]

cell $\mathcal{K}_i$ with the sub-domain $\mathcal{A}_i = \mathcal{K}_i \cap \mathcal{A}$ and the surface $\partial\mathcal{A}_i$. As in cut cells the surface

$\partial\mathcal{A}_i$ consists not only of the edges $\left\{\mathcal{E}_{i,e}^{\mathcal{A}}\right\}_{e=1,\dots,E} = \left\{\mathcal{E}_{i,e} \cap \bar{\mathcal{A}}_i\right\}_{e=1,\dots,E}$ but also of the boundary segment $\mathcal{I}_i = \mathcal{K}_i \cap \mathcal{I}$, the discrete weak formulation using an IBM follows as

$$\int_{\mathcal{A}_i} \frac{\partial c_i}{\partial t} \Phi_{i,j}\, dV + \sum_{e=1}^{E_i} \int_{\mathcal{E}_{i,e}^{\mathcal{A}}} f\left(c^-, c^+, \mathbf{n}\right) \Phi_{i,j}\, dA + \int_{\mathcal{I}_i} f\left(c^-, c^+, \mathbf{n}_{\mathcal{I}}\right) \Phi_{i,j}\, dA - \int_{\mathcal{A}_i} \boldsymbol{f}\left(c_i\right) \cdot \nabla \Phi_{i,j}\, dV = 0 \tag{4.1}$$

with $\mathbf{n}_{\mathcal{I}} = -\dfrac{\nabla\varphi}{\|\nabla\varphi\|}$. In intersected cells the mass matrix is defined by

$$\left(\mathbf{M_i}\right)_{k,j} := \int_{\mathcal{A}_i} \Phi_{i,k} \Phi_{i,j}\, dV \tag{4.2}$$

and the discrete operator by

$$\left(\mathbf{f}_i\right)_j := \sum_{e=1}^{E_i} \int_{\mathcal{E}_{i,e}^{\mathcal{A}}} f\left(c^-, c^+, \mathbf{n}\right) \Phi_{i,j}\, dA + \int_{\mathcal{I}_i} f\left(c^-, c^+, \mathbf{n}_{\mathcal{I}}\right) \Phi_{i,j}\, dA - \int_{\mathcal{A}_i} \boldsymbol{f}\left(c_i\right) \cdot \nabla \Phi_{i,j}\, dV. \tag{4.3}$$

The difficulty of the IBM lies in the correct evaluation of $\mathcal{A}_i$ and $\mathcal{I}_i$ and in the agglomeration of intersected cells with very small volume fractions

$$\mathrm{frac}(\mathcal{A}_i) = \frac{\mathrm{meas}(\mathcal{A}_i)}{\mathrm{meas}(\mathcal{K}_i)} \tag{4.4}$$

as we will discuss in section section 4.3.

4.2 RK Time Discretisation with Immersed Boundaries

In this thesis we only use explicit Euler time discretisation for immersed boundary problems as we are only interested in the steady state:

$$\mathbf{c_i}(t_1) = \mathbf{c_i}(t_0) - \Delta t \mathbf{M}_i^{-1} \mathbf{f}_i(c). \tag{4.5}$$

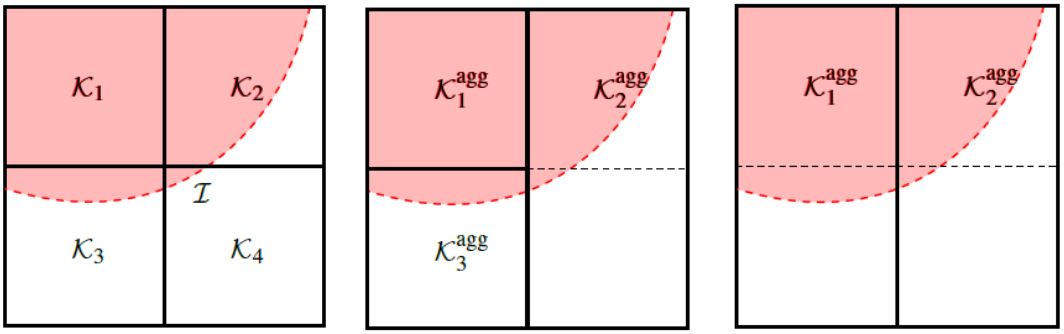
Using IBMs we have to modify the stability criterion and therefore use the modified step restriction

$$\Delta t \leq \frac{c_{CFL}}{2P+1} \frac{\sqrt[D]{\mathrm{meas}(\mathcal{A}_i)}}{\|\mathbf{u}\| + a} \tag{4.6}$$

which is strongly influenced by the sub-cell $\mathcal{A}_i$ with the smallest volume.

4.3 Cell Agglomeration

As can be seen in (4.6) the time step size is strongly restricted in cells with very small volume fractions. This leads to an elongated calculation process thus rendering the method impractical. Therefore we need to agglomerate those small cells to larger ones using a cell agglomeration factor $0 \leq \alpha \leq 1$.
The cell agglomeration strategy depends on finding the source cells $\left\{\mathcal{K}_s^{\text{src}}\right\}_{s=1,\dots,S}$ with $\text{frac}(\mathcal{A}_i) \leq \alpha$ and agglomerating them to the neighboring cell with the highest volume fraction, namely target cell $\mathcal{K}_s^{\text{tar}}$. In fig. 4.2 you can see the cell agglomeration for a smaller (b) and a bigger (c) agglomeration factor.



(a) Initial mesh partitioning (b) Cell agglomeration with small agglomeration threshold (c) Cell agglomeration with bigger agglomeration threshold

Figure 4.2.: Cell agglomeration, taken from [5]

For the neighboring cells are weakly coupled via fluxes, the basis $\vec{\Phi}_i$ can be extended from the target cell into the source cell. Therefore the source cell can formally be deleted from the discretisation mesh, reducing it to $\left\{\mathcal{K}_s^{\text{agg}}\right\}_{i=1,\dots,N-S}$. As can be found in [5], it however does not reflect the actual implementation in BoSSS which only requires few cell-local matrix-vector products per time-step, thus not affecting the parallel efficiency.

4.4 Simulation Parameters Concerning Stability and Runtime in BoSSS

In the following we will describe some of the preferences for the CNS solver that can be modified in order to stabilise the computations.

LevelSetQuadratureOrder

The first preference is the *LevelSetQuadratureOrder*. It describes the quadrature order that is used for the evaluation of volume and edge operators. It is set to an integer; we will use a constant

```
c.LevelSetQuadratureOrder = 8;
```

in chapter 5 as it is important to have comparable results for the verification and a degree dependent order

```
c.LevelSetQuadratureOrder = 3*dgDegree;
```

in chapter 6 as it is more important to produce efficient and fast though stable results.

SIPGPenaltyConstant

The second preference that we used is the *SIPGPenaltyConstant*. It is set to a double that should be greater than one and describes the weighting of the viscous terms. For the stability of the computation it is essential that the numerical flux functions that are used for the discretisation of the viscous terms are consistent. This can be achieved by different methods; one of those being the Symmetric Interior Penalty Method (SIPG) which we are using which introduces the penalty factor. In chapter 6 we will set

```
c.SIPGPenaltyConstant = 5.0;
```

for our calculations.

NodeCountSafetyFactor

The last factor that we will vary for a better stability is the *NodeCountSafetyFactor*. This factor, that is also set to a double, indicates the number of the nodes with respect to the number of modes that will be used during the quadrature. By standard BoSSS sets it to 1.0; in our calculations we mostly used a NodeCountSafetyFactor of 2.0 in chapter 5 and of 5.0 in chapter 6 though it has been modified for few calculations in order to get them more stable.

All of these factors should be set just high enough in order to ensure a accurate and stable solution and the shortest possible runtime at the same time.

5 Verification of BoSSS for Inviscid Flows

In the following chapter we will regard a flow at Mach 0.2 around a frictionless cylinder with adiabatic slip walls at changing parameters. We will compute a domain with $-40 \leq x \leq 40$, $0 \leq y \leq 40$, a cylinder radius of $r = 1$ and the consequential level set $\varphi = x^2 + y^2 - 1$. We are only considering the upper half of the domain as we can assume a symmetric flow.

For we will regard an isentropic inviscid flow with

$$\frac{p}{\rho^\gamma} = \text{const} \tag{5.1}$$

we can compare our results for the entropy to the analytical solution $s = 0$. Using this comparison we aim at verifying BoSSS for inviscid flows and IBMs considering robustness and convergence.

5.1 Robustness Study

In the first study regarding the frictionless cylinder, we compare the absolute error of entropy for a polynomial degree from 1 to 3 along a shift of the centre point of the cylinder from -0.075 to 0.075 at steps of 0.015. By shifting the cylinder we can consider several cases where the cells would be cut differently and therefore cause different cell agglomerations. The cell agglomeration threshold is at a constant level of 0.5 in a mesh of 64×64 cells thus causing different cell agglomerations with every shift. In this example we aim at proving the robustness of the solver as for each position of the cylinder the error of entropy should not vary too much thus making it independent of the way the border cells are cut.

As we look at fig. 5.1, first of all we will note that the absolute error of entropy decreases with increasing polynomial degree. As a higher polynomial degree implies a better approximation this can be explained very easily.

Secondly, we can observe that the error of entropy behaves roughly symmetrically to the ordinate which is unsurprising as we shifted the cylinder symmetrically.

For degree 1 and 2 the error is at a fairly constant value throughout the cylinder shift; at the polynomial degree 3 it is very irregular. The two most discordant values appear at a shift of ± 0.03. There the calculation was very unstable. By varying the parameters explained in section 4.4 we made them as stable as possible but still produced the inaccurate values shown

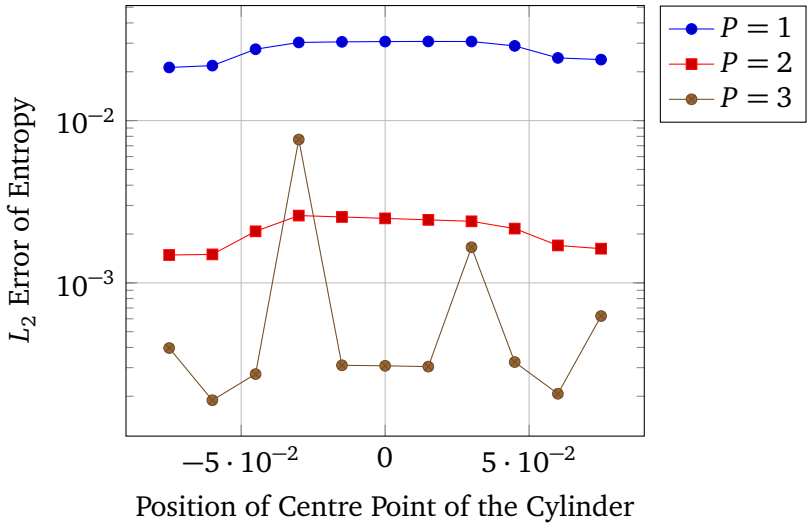
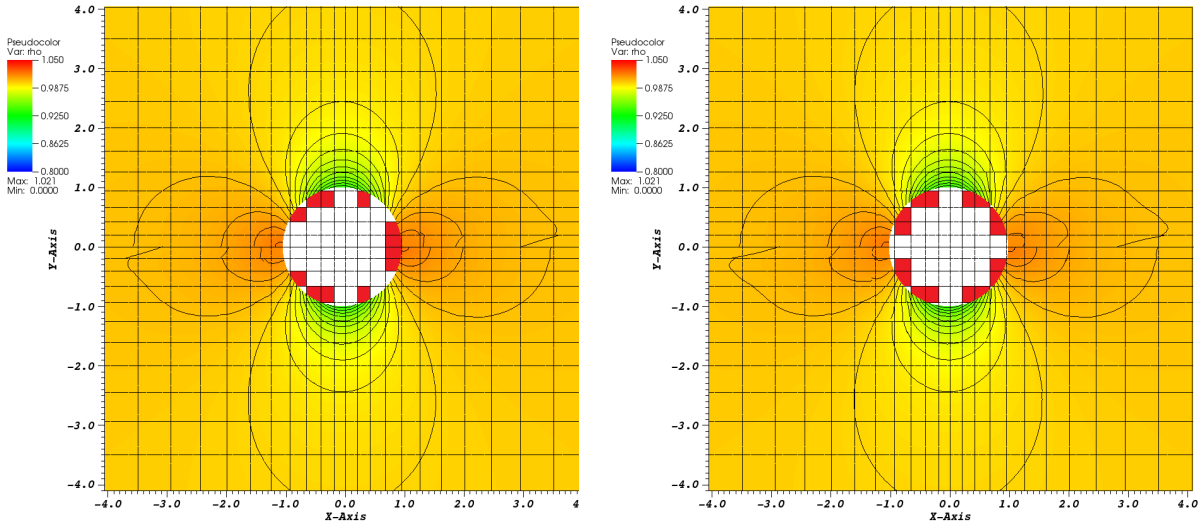


Figure 5.1.: Convergence Plot

in fig. 5.1.

We can therefore infer that with increasing polynomial degree at a mesh that course the calculation gets very unstable; a finer mesh should be used.

Now we will consider two cases which have the largest error difference apart from degree 3 - degree 2 at the shifts −0.06 and −0.03 - more closely.



Degree 2, shift −0.06 Degree 2, shift −0.03

Figure 5.2.: Isolines of pressure

In fig. 5.2 you can see the two mentioned cases with highlighted isolines of pressure and pseudocolored density. As only the upper half of the cylinder has been calculated, we reflected the results through the centre point of the cylinder. Therefore you can easily see that the results are not flawless, as the flow before and after the obstacle should be identical. Furthermore you can see that in the left picture the isolines are smoother than in the right one. We also highlighted the cells that have been agglomerated in red; in the right case almost every cut cell has been agglomerated while in the left one it were fewer. It may be assumed, that each agglomeration causes an error which results into the higher error that we remarked in fig. 5.1. Unfortunately we did not find a proper explanation for the agglomeration error; this could be subject of future research.

Except for the polynomial degree 3, the error of entropy changes very little for the different cases. We can therefore assume that the solver is good enough validated concerning the way the agglomerated cells influence the calculation as long as we consider a fine enough mesh for higher degrees.

5.2 Convergence Study of Mesh Size and Polynomial Degree

In the second study we vary the mesh size of our geometry from 32×32 by 64×64 to 128×128 cells. Additionally we also vary the polynomial degree from 0 to 4, consequently regarding fifteen cases in total. The agglomeration factor is at a constant moderate level $\alpha = 0.3$.
Our aim is the validation of the convergence of the RKDG method based solver for the inviscid cylinder. Therefore we hope to achieve an experimental order of convergence that is near the optimal rate $O(h^{P+1})$. In fig. 5.3 we compared the absolute error entropy to the mesh size logarithmically for each polynomial degree and added the linear regressions R_P with its slopes to each graph.

As you can see in fig. 5.3 each graph has a quite constant slop that is higher with increasing polynomial degree. For $1 < P < 4$ it is approximately of the order $P + 1$ as we hoped, only for $P = 0$ the computations converge much more slowly.

Again it was necessary to adjust some of the parameters explained in section 4.4 in order to stabilise the calculations with higher polynomial degrees.

As an example for the correction of a critical calculation we visualised the case with mesh size 128×128 and polynomial degree $P = 3$ before and after the correction in fig. 5.4. Again, we visualised entropy and pressure. The picture in the middle shows a zoomed view of the critical cell where a high amount of entropy was produced and lead to the breakup. Please remark that differently coloured entropy ranges had to be used before and after the correction in order to point out the critical cell.

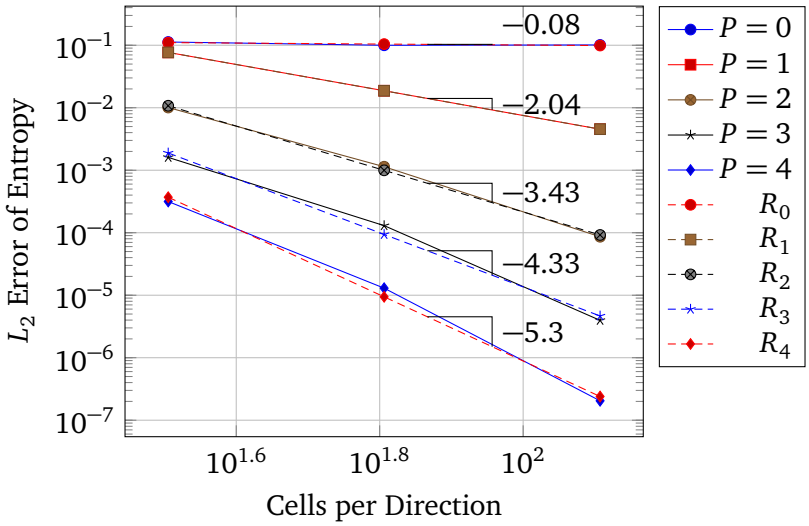
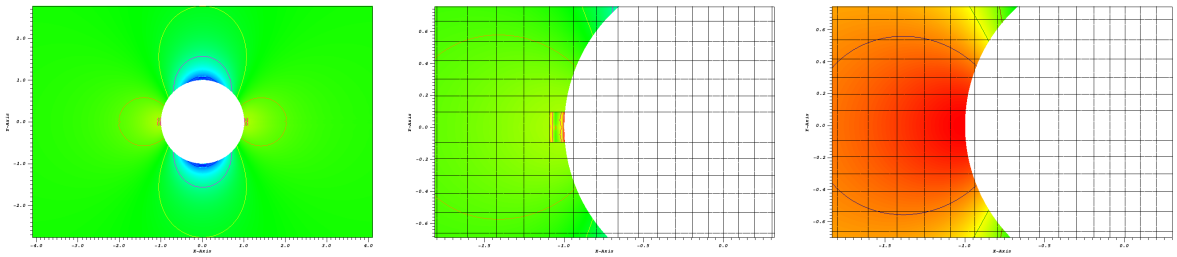


Figure 5.3.: Convergence Plot



Overview of flow before correction

Detailed view of critical cell before correction

Detailed view of critical cell after correction

Figure 5.4.: Mesh size 128×128, $P = 3$

5.3 Conclusion

After having studied the behaviour of BoSSS concerning the Euler equations with immersed boundaries, we can conclude that is sufficiently validated. The robustness studied showed that the results are mostly independent from the exact position of the grid cells. During the convergence study we remarked that the convergence behaves as desired with an order close to the optimal rate of $O(h^{P+1})$.

Nevertheless in order to receive the correct results one sometimes needs to put a large effort into adjusting parameters for stable calculations.

6 Evaluation of BoSSS for Viscid Flows

In this chapter we aim at validating BoSSS for viscid flows with IBMs. In order to have a good comparable result, we will once again regard the flow around a cylinder as in chapter chapter 5. The viscous flow around a cylinder has been approached by many papers both experimentally and numerically, e.g. [6], [7], [8], though very few numerical approaches use a RKDG method combined with immersed boundaries. In order to verify the BoSSS code with immersed boundaries not only for the Euler equations as we did in chapter chapter 5 but also for the viscous case we will now consider different Reynolds numbers for the steady and unsteady flow and compare our results to those of other studies.

6.1 Theory

The flow around a viscous cylinder can be divided into different sections depending on the Reynolds number as shown in fig. 6.1. The first section applies for Reynolds numbers $0 < \mathrm{Re} < 40 - 50$ characterised by a laminar steady flow. In that regime a recirculation region with two symmetric vortices with opposite directions is comprised by the wake. The flow can be described using the wake separation length W^*.

The second section contains all other Reynolds number $\mathrm{Re} > 40 - 50$ and thus describes

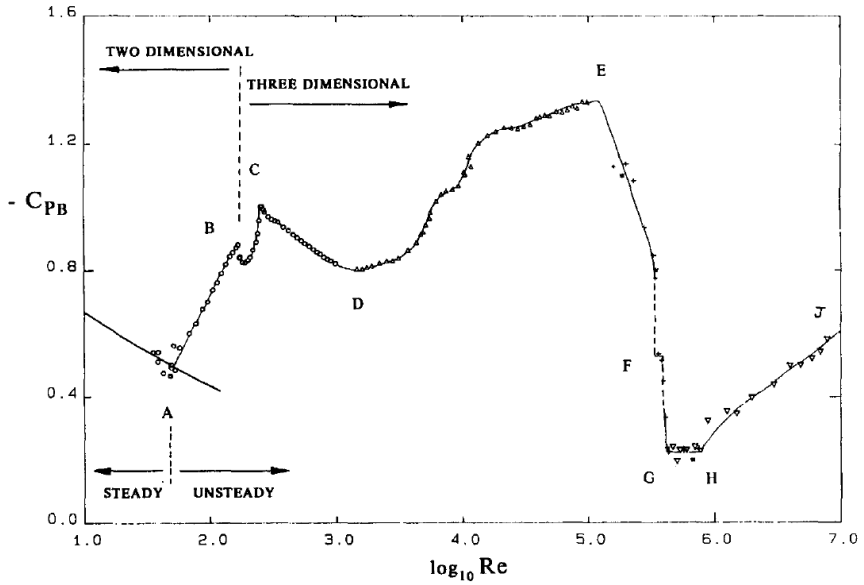


Figure 6.1.: Overview of Base Suction Coefficients over Reynolds Number [6]

the unsteady flow. It can be subdivided in several subsections [6]:

$40 - 50 < \mathrm{Re} < 190$ laminar vortex shedding,

$190 < \mathrm{Re} < 260$ 3D wake-transition regime,

$260 < \mathrm{Re} < 1000$ increasing disorder in the fine-scale three dimensionalities,

$1000 < \mathrm{Re} < 200000$ shear layer transition regime,

$200000 < \mathrm{Re}$ critical transition, supercritical regime and post-critical regime.

As we will only discuss Reynolds numbers up to $\mathrm{Re} = 200$ the important phases for us are the laminar steady regime and the laminar vortex shedding. At around $\mathrm{Re} = 190$ the three dimensionality of the system has an incrementing influence on the flow; for we only analyse the 2-D model of the experiment we stop at $\mathrm{Re} = 200$ expecting slight deflection in our results.

6.1.1 The Laminar Steady Regime

At Reynolds numbers below 50 the flow forms a steady recirculation region, characterised by the wake separation length W^*. It is built by two symmetrically placed vortices on each side of the wake as can be seen in fig. 6.2. It has been shown experimentally as well as numerically, that the wake separation length increases with increasing Reynolds number.

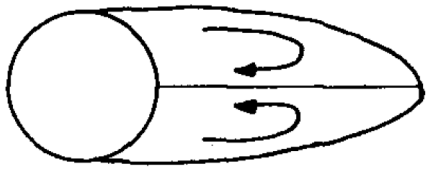


Figure 6.2.: Recirculation Region [6]

6.1.2 Laminar Vortex Shedding

For Reynolds numbers of between 50 and 200 the recirculation region develops instabilities leading to the development of turbulence in the wake. This results into fully periodic vortex shedding, known as the Kármán vortex street, as can be seen in fig. 6.3. With increasing Reynolds number the amplitudes of drag and lift coefficients increases while the Strouhal number and frequency, respectively, decreases.

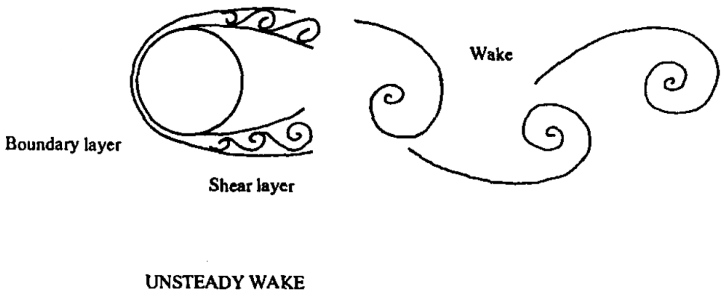


Figure 6.3.: Kármán Vortex Street [6]

6.2 Simulations

@TODO: DoF In this section we will compare the lift and drag coefficients C_L and C_D at different Reynolds numbers and mesh sizes at a constant agglomeration threshold of 0.3, different polynomial degrees of 1, 2 and 3 and meshes of 40×40, 60×60 and 80×80 cells. The meshes can be found in the appendix.

The different simulation properties will be abbreviated as DG + *polynomial degree* + CpD + *number of cells per direction*, e.g. DG2CpD80 for a simulation with polynomial degree 2 and 80×80 cells.

In table 6.1 you can see the total Degrees of Freedom (DoF) for each simulation taking into account the DoFs produced by order 1, 2 and 3 as 3, 6 and 10 DoFs per cell.

DoF		CpD		
		40	60	80
DG	1	4800	10800	19200
	2	9600	21600	38400
	3	16000	36000	64000

Table 6.1.: Degrees of Freedom for Different Simulation Properties

The drag and lift coefficients are defined as

$$C_D = \frac{d}{q_\infty L_\infty} \tag{6.1}$$

$$C_L = \frac{l}{q_\infty L_\infty} \tag{6.2}$$

with the dynamic pressure $q_\infty = \frac{1}{2}\rho_\infty V_\infty^2$. For we set $L_\infty = \rho_\infty = V_\infty = 1$ in our boundary and initial conditions, we can assume

$$C_D = 2 \cdot d \tag{6.3}$$

$$C_L = 2 \cdot l, \tag{6.4}$$

with the drag and lift forces d and l provided from the calculation.

During the simulations we now compute the complete domain with $-40 \leq x \leq 40$, $-40 \leq y \leq 40$ and the cylinder radius $r = 1$ as we expect asymmetrical oscillations of the flow with higher Reynolds numbers. We will use a rectilinear mesh that is finer near the cylinder, an isothermal wall boundary condition at the cylinder wall and supersonic inlet boundary conditions for the domain borders that shall prevent the reflection of the initial wave.

6.2.1 Steady State Simulations (Re $< 40 - 50$)

For the steady state simulations we can use the wake separation length W^* as an additional variable to compare to other simulations. It can be found from examining the x-velocity U at

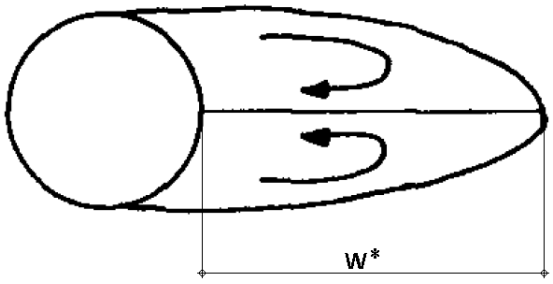


Figure 6.4.: Wake separation length, from [6] modified

$y = 0$; the x-position where U changes its sign should be the end position of the wake.

Simulation at Reynolds Number 20

We will now simulate the flow at Re $= 20$ and compare our results to several experimental and numerical results as shown in table 6.2. The results are divided into three categories: experimental, numerical incompressible and numerical compressible in order to coincide with the arrangement given by [2].

Re = 20	Source	2D/3D	W^*	C_D
Numerical - Incompressible	Dennis et al [1970]	2D	0.94	2.05
	Forberg [1980]	2D	0.91	2.00
	Linnick et al. [2005]	2D	0.93	2.06
Experimental	Coutanceau et al. [1978]	-	0.93	-
	Tritton [1959]	-	-	2.09
Numerical - Compressible	Brehm et al. [2015] (Ma = 0.1)	3D	0.96	2.02
	Ayers [2015]	2D	0.975	2.06
	Present Results:	2D	d	4

Table 6.2.: Comparison of Results for W^* and C_D, taken from [2], modified

C_D		CpD		
		40	80	160
DG	1			
	2			
	3			

Table 6.3.: C_D Values for each simulation

W^*		CpD		
		40	80	160
DG	1			
	2			
	3			

Table 6.4.: Wake Separation Lengths for each simulation

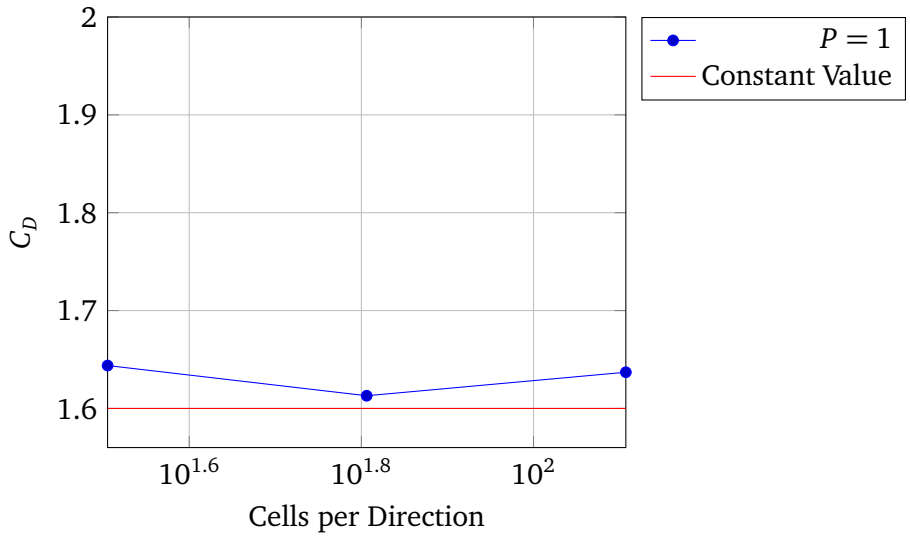


Figure 6.5.: Graphical presentation of table 6.3 for Re=20

Simulation at Reynolds Number 40

Re = 40	Source	2D/3D	W^*	C_D
Numerical - Incompressible	Dennis et al [1970]	2D	2.35	1.52
	Forberg [1980]	2D	2.24	1.50
	Linnick et al. [2005]	2D	2.28	1.54
Experimental	Coutanceau et al. [1978]	-	2.13	-
	Tritton [1959]	-	-	1.59
Numerical - Compressible	Brehm et al. [2015] (Ma = 0.1)	3D	2.26	1.51
	Ayers [2015]	2D	2.250	1.605
	Present Results:	2D	d	4

Table 6.5.: Comparison of Results for W^* and C_D, taken from [2], modified

6.2.2 Unsteady Simulations (Re $> 40 - 50$)

In order to compare the unsteady simulations we need the Strouhal number

$$\mathrm{St} = \frac{f L_\infty}{V_\infty}. \tag{6.5}$$

As our initial and boundary conditions give $V_\infty = L_\infty = 1$, we can calculate $\mathrm{St} = f$ with f found from examining the oscillation of C_L over time. For developing vortex shedding, the flow needs small perturbations that destabilize the flow towards a symmetry breaking state [7]. In reality those are given by the structure of the cylinder, the influence of the walls or the not completely straight inflow; in our simulations they come from small truncation errors and the computer's round-off errors. In order to accelerate the process until the wake begins to oscillate one could also start the flow with a vortex that induces a high perturbation much earlier. For it did not take long until the wake began to oscillate it was not needed in our simulations.

Simulation at Reynolds Number 100

Simulation at Reynolds Number 200

Re = 100	Source	2D/3D	St	C_D	C_L
Numerical - Incompressible	Gresho et al. [1984]	2D	0.18	1.76	-
	Linnick et al. [2005] ($\lambda = 0.056$)	2D	0.169	1.38 ±010	±337
	Linnick et al. [2005] ($\lambda = 0.023$)	2D	0.169	1.34 ±009	±333
	Persillon et al. [1998]	2D	0.165	1.253	-
	Saiki et al. [1996]	2D	0.171	1.26	-
	Persillon et al. [1998]	3D	0.164	1.240	-
	Liu et al. [1998]	3D	0.165	1.35 ±012	±339
Experimental	Berger et al. [1972]	-	0.16-0.17	-	-
	Clift et al. [1978]	-	-	1.24	-
	Williamson [1996]	-	0.164	-	-
Numerical - Compressible	Brehm et al. [2015] (Ma = 0.1)	3D	0.165	1.32 ±01	±32
	Ayers [2015]	2D	0.167	1.371 ±011	±333
	Present Results:	2D	d	4	

Table 6.6.: Comparison of Results for St, C_D and C_L, taken from [2], modified

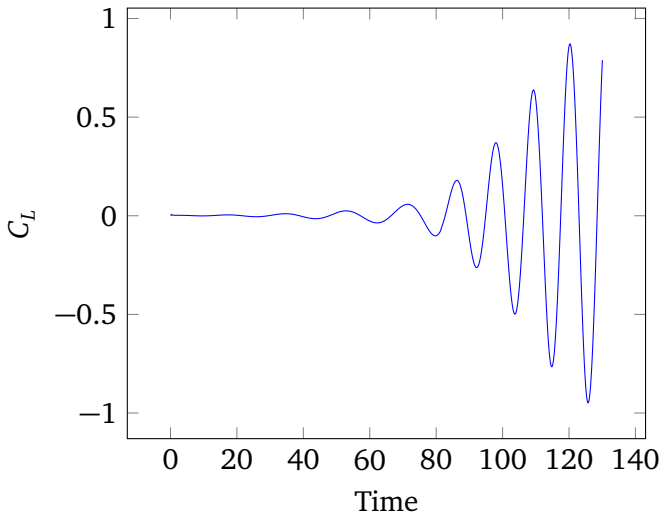


Figure 6.6.: Lift coefficient over time for Re = 100

Re = 100	Source	2D/3D	St	C_D	C_L
Numerical - Incompressible	Dennis et al [1970]	2D	0.94	2.05	
	Forberg [1980]	2D	0.91	2.00	
	Linnick et al. [2005]	2D	0.93	2.06	
Experimental	Coutanceau et al. [1978]	-	0.93	-	
	Tritton [1959]	-	-	2.09	
Numerical Compressible	Brehm et al. [2015] (Ma = 0.1)	3D	0.96	2.02	
	Ayers [2015]	2D	0.975	2.06	
	Present Results:	2D	d	4	

Table 6.7.: Comparison of Results for W^* and C_D, taken from [2], modified

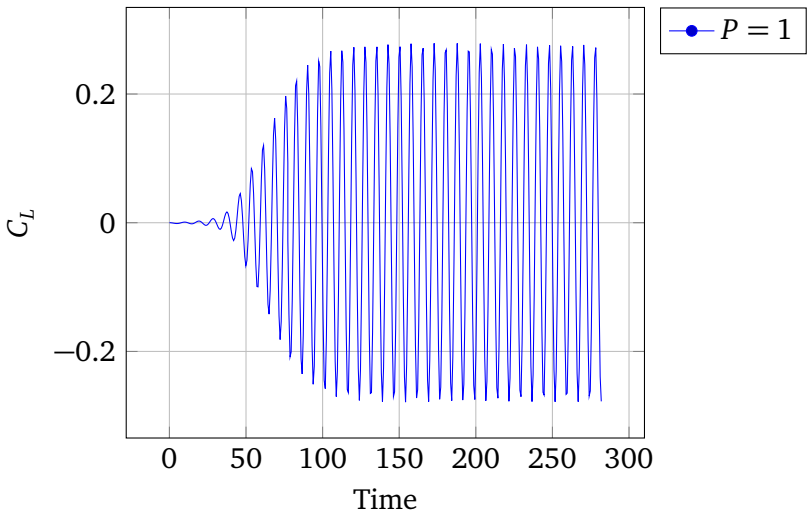


Figure 6.7.: Lift coefficient over time for Re=200

7 Conclusion

8 Bibliography

[1] Müller, B., "Methods for higher order numerical simulations of complex inviscid fluids with immersed boundaries," 2014.

[2] Ayers, L. F., "Validation of a discontinuous galerkin based compressible cfd solver," Bachelor thesis, Technische Universität Darmstadt, Dec. 2015.

[3] Krämer-Eis, Stephan, *Notes on the implementation of the viscous terms in cns: Code verification, fdy annual report 2014*, 2014.

[4] Cockburn, B., "Advanced numerical approximation of nonlinear hyperbolic equations: Lectures given at the 2nd session of the centro internazionale matematico estivo (c.i.m.e.) held in cetraro, italy, june 23–28, 1997," in, Quarteroni, A., Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, ch. An introduction to the Discontinuous Galerkin method for convection-dominated problems, pp. 151–268, isbn: 978-3-540-49804-9. doi: 10.1007/BFb0096353.

[5] Müller, B., Krämer-Eis, S., Kummer, F., *et al.*, "A high-order discontinuous galerkin method for compressible flows with immersed boundaries," *INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING*, vol. 295, pp. 475–504, 2016.

[6] Williamson, C. H., "Vortex dynamics in the cylinder wake," *Annual review of fluid mechanics*, vol. 28, no. 1, pp. 477–539, 1996.

[7] Persillon, H. and Braza, M., "Physical analysis of the transition to turbulence in the wake of a circular cylinder by three-dimensional navier–stokes simulation," *Journal of Fluid Mechanics*, vol. 365, pp. 23–88, Jun. 1998, issn: 1469-7645. doi: 10.1017/S0022112098001116.

[8] Canuto, D. and Taira, K., "Two-dimensional compressible viscous flow around a circular cylinder," *Journal of Fluid Mechanics*, vol. 785, pp. 349–371, 2015.

[9] Brehm, C., Hader, C., and Fasel, H. F., "A locally stabilized immersed boundary method for the compressible navier–stokes equations," *Journal of Computational Physics*, vol. 295, pp. 475–504, 2015.

[10] Hartmann, R., "Numerical analysis of higher order discontinuous Galerkin finite element methods," in *VKI LS 2008-08: CFD - ADIGMA course on very high order discretization methods, Oct. 13-17, 2008*, Deconinck, H., Ed., Von Karman Institute for Fluid Dynamics, Rhode Saint Genèse, Belgium, 2008.

List of Figures

List of Tables

A Appendix

@TODO: richtige Bilder einfügen

A.1 Mesh properties

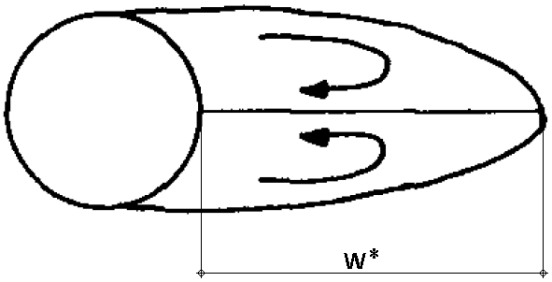


Figure A.1.: Mesh for CpD40

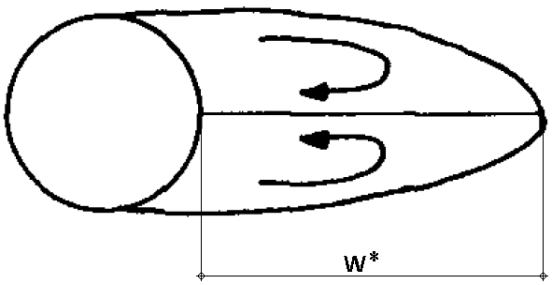


Figure A.2.: Mesh for CpD80

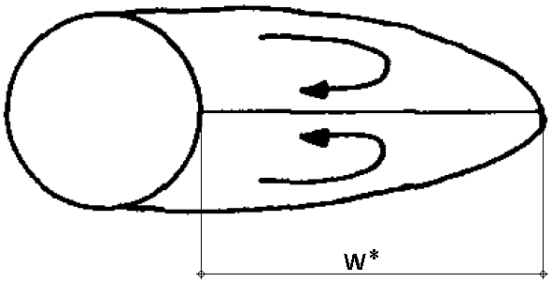


Figure A.3.: Mesh for CpD160