Implementation and Performance Analyses of a Highly Efficient Algorithm for Pressure-Velocity Coupling

Implementierung und Untersuchung einer hoch effizienten Methode zur Druck-Geschwindigkeits-Kopplung

Master-Thesis von Fabian Gabel
Tag der Einreichung:

1. Gutachten: Prof. Dr. rer. nat. Michael Schäfer
2. Gutachten: Dipl.-Ing Ulrich Falk

TECHNISCHE UNIVERSITÄT DARMSTADT

Studienbereich CE
FNB

Implementation and Performance Analyses of a Highly Efficient Algorithm for Pressure-Velocity Coupling
Implementierung und Untersuchung einer hoch effizienten Methode zur Druck-Geschwindigkeits-Kopplung

Vorgelegte Master-Thesis von Fabian Gabel

1. Gutachten: Prof. Dr. rer. nat. Michael Schäfer
2. Gutachten: Dipl.-Ing Ulrich Falk

Tag der Einreichung:

Erklärung zur Master-Thesis

Hiermit versichere ich, die vorliegende Master-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 22. Januar 2015

(Fabian Gabel)

Contents

List of Figures

List of Tables

List of Algorithms

1 Introduction

This thesis is about.

2 Fundamentals of Continuum Physics for Thermo-Hydrodynamical Problems

This section covers the set of fundamental equations for thermo-hydrodynamical problems which the numerical solution techniques of the following chapters are aiming to solve. Furthermore the notation regarding the physical quantities to be used throughout this thesis is introduced. The following paragraphs are based on (Kundu, Spurk, Ferziger, Anderson). For a thorough derivation of the matter to be presented the reader may consult the mentioned sources. Since the present thesis focusses on the application of finite-volume methods the focus lays on stating the integral forms of the relevant conservation laws. However in the process of deriving the final set of equations the use of differential formulations of the stated laws are required. Einstein's convention for taking sums over repeated indices is used to simplify certain expressions. For the remainder of this thesis non-moving inertial frames in a Cartesian coordinate system with the coordinates x_i are used. This approach is also known as *Eulerian approach*.

2.1 Conservation of Mass – Continuity Equation

The conservation law of mass embraces the physical concept that, neglecting relativistic and nuclear reactions, mass cannot be created or destroyed. Using the notion of a mathematical control volume, which is used to denote a constant domain of integration, one can state the integral mass balance of a control volume V with control surface S with surface normal unit vector $\mathbf{n} = (n_i)_{i=1,\dots,3}$ using Gauss' theorem as

$$\iiint_V \frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i)\,\mathrm{d}V = \iiint_V \frac{\partial \rho}{\partial t}\mathrm{d}V + \iint_S \rho u_i n_i \mathrm{d}S = 0,$$

where ρ denotes the material density, t denotes the independent variable of time and $\mathbf{u} = (u_i)_{i=1,\dots,3}$ is the velocity vector field. Since this equation remains valid for arbitrary control volumes the equality has to hold for the integrands as well. In this sense the differential form of the conservation law of mass can be formulated as

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) = 0. \tag{1}$$

2.2 Conservation of Momentum – Cauchy-Equations

The conservation law of momentum, also known as Newton's Second Law, axiomatically demands the balance of the temporal change of momentum and the sum of all attacking forces on a body. Those forces can be divided into body forces and surface forces. Let $\mathbf{k} = (k_i)_{i=1,\dots,3}$ denote a mass specific force and $\mathbf{t} = (t_i)_{i=1,\dots,3}$ the stress vector. A first form of the integral momentum balance in the direction of x_i can be formulated as

$$\iiint_V \frac{\partial(\rho u_i)}{\partial t}\mathrm{d}V + \iint_S \rho u_i \left(u_j n_j\right)\mathrm{d}S = \iiint_V \rho k_i \mathrm{d}V + \iint_S t_i \mathrm{d}S. \tag{2}$$

In general the stress vector $\mathbf{t}$ is a function not only of the location $\mathbf{x} = (x_i)_{i=1,\dots,3}$ and of the time t but also of the surface normal unit vector $\mathbf{n}$. A central simplification can be introduced, namely Cauchy's stress theorem, which states that the stress vector is the image of the normal vector under a linear mapping $\mathbf{T}$. With respect to the Cartesian canonical basis $(\mathbf{e}_i)_{i=1,\dots,3}$ the mapping $\mathbf{T}$ is represented by the coefficient matrix $\left(\tau_{ji}\right)_{i,j=1,\dots,3}$ and Cauchy's stress theorem reads

$$\mathbf{t}(\mathbf{x}, t, \mathbf{n}) = \mathbf{T}(\mathbf{x}, t, \mathbf{n}) = \left(\tau_{ji} n_j\right)_{i=1,\dots,3}.$$

Assuming the validity of Cauchy's stress theorem one can derive Cauchy's first law of motion, which in differential form can be formulated as

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_i u_j\right) = \rho k_i + \frac{\partial \tau_{ji}}{\partial x_j} \tag{3}$$

and represents the starting point for the modelling of fluid mechanical problems. One should note, that Cauchy's first law of motion does not take any assumptions regarding material properties, which is why the set of equations (1,2) is not closed in the sense that there exists a independent equation for each of the dependent variables.

2.3 Closing the System of Equations – Newtonian Fluids

As result of Cauchy's theorem the stress vector **t** can be specified once the nine components τ_{ji} of the coefficient matrix are known. As is shown in (Spurk usw.) by formulating the conservation law of angular momentum the coefficient matrix is symmetric,

$$\tau_{ji} = \tau_{ij}, \tag{4}$$

hence the number of unknown coefficients may be reduced to six unknown components. In a first step it is assumed that the coefficient matrix can be decomposed into fluid-static and fluid-dynamic contributions,

$$\tau_{ij} = -p\delta_{ij} + \sigma_{ij},$$

where p is the thermodynamic pressure, δ_{ij} is the *Kronecker*-Delta and σ_{ij} is the so called *deviatoric stress tensor*.

For the fluids the studies that the present thesis performs it is sufficient to consider viscous fluids for which there exists a linear relation between the components of the deviatoric stress tensor and the symmetric part of the transpose of the jacobian of the velocity field $\left(S_{ij}\right)_{i,j=1,\ldots,3}$,

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right).$$

If one now imposes material-isotropy and the mentioned stress-symmetry (4) restriction it can be shown (Aries) that the constitutive equation for the deviatoric stress tensor reads

$$\sigma_{ij} = 2\mu S_{ij} + \lambda S_{mm}\delta_{ij},$$

where λ and μ denominate scalars which depend on the local thermodynamical state. Taking everything into account (3) can be formulated as the differential conservation law of momentum for newtonian fluids, better known as the *Navier-Stokes equations* in differential form:

$$\frac{\partial\left(\rho u_i\right)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_i u_j\right) = \rho k_i - \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right) + \frac{\partial}{\partial x_i}\left(\lambda\frac{\partial u_m}{\partial x_m}\right) \tag{5}$$

2.4 Conservation Law for Scalar Quantities

The modelling of the transport of scalar quantities, convection, by a flow field **u** is necessary if the fluid mechanical problem to be analyzed includes for example heat transfer. Other scenarios that involve the necessity to model scalar transport surge, when turbulent flows are to be modeled by two-equation models like the k-ε-model (REFERENCE,Pope).

Since this thesis focusses on the transport of the scalar temperature T this section introduces the conservation law for energy in differential form,

$$\frac{\partial\left(\rho T\right)}{\partial t} + \frac{\partial}{x_j}\left(\rho u_j - \kappa\frac{\partial T}{\partial x_j}\right) = q_T, \tag{6}$$

where κ denotes the thermal conductivity of the modelled material and q_T is a scalar field representing sources and sinks of heat throughout the domain of the problem.

2.5 Necessary Simplification of Equations

Negligible viscous dissipation and pressure work source terms in the enery equation (vakilipour)

The purpose of this section is to motivate and introduce further common simplifications of the previously presented set of constitutive equations.

2.5.1 Incompressible Flows and Hydrostatic Pressure

A common simplification when modelling low Mach number flows ($Ma < 0.3$), is the assumption of *incompressibility*, or the assumption of an *isochoric* flow. If one furthermore assumes homogeneous density ρ in space and time, a restrictive assumption that will be partially alleviated in the following section the continuity equation in differential form (1) can be simplified to

$$\frac{\partial u_i}{\partial x_i} = 0. \tag{7}$$

In other words: In order for a velocity vector field $\mathbf{u}$ to be valid for an incompressible flow it has to be free of divergence, or *solenoidal* (Aries).

If furthermore, one assumes also constant dynamic viscosity μ, which can be suitable in the case of isothermal flow or if the temperature differences within the flow are small, the Navier-Stokes equations in differential form can be reduced to

$$\frac{\partial (\rho u_i)}{\partial t} + \rho \frac{\partial}{\partial x_j}\left(u_i u_j\right) = \rho k_i - \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right) \tag{8a}$$

$$= \rho k_i - \frac{\partial p}{\partial x_i} + \mu \frac{\partial}{\partial x_j}\left(\frac{\partial u_i}{\partial x_j}\right) \tag{8b}$$

by using *Schwartz*'s lemma to interchange the order of differentiation. A common simplification to further simplify the set of equations is the assumption of a volume specific force $\rho\mathbf{k}$ that can be modelled by a potential, such that is can be represented as the gradient of a scalar field $\Phi_{\mathbf{k}}$ as

$$-\rho k_i = \frac{\partial \Phi_{\mathbf{k}}}{\partial x_i}.$$

In the case of this thesis this assumption is valid since the mass specific force is the mass specific gravitational force $\mathbf{g} = (g_i)_{i=1,\dots,3}$ and the density is assumed to be constant, so the potential can be modelled as

$$\Phi_g = -\rho g_j x_j.$$

This term can be interpreted as the hydrostatic pressure p_{hyd} and can be added to the thermodynamical pressure p to simplify calculations

$$\begin{aligned}
\rho g_i - \frac{\partial p}{\partial x_i} &= \frac{\partial}{\partial x_i}\left(\rho g_j x_j\right) - \frac{\partial p}{\partial x_i} \\
&= \frac{\partial}{\partial x_i}\left(\rho g_j x_j\right) - \frac{\partial}{\partial x_i}\left(\hat{p} + p_{hyd}\right) \\
&= -\frac{\partial \hat{p}}{\partial x_i}.
\end{aligned} \tag{9}$$

Since in incompressible fluids only pressure differences matter, this has no effect on the solution. After finishing the calculations p_{hyd} can be calculated and added to the resulting pressure $\hat{p}$.

2.5.2 Variation of Fluid Properties – The Boussinesq Approximation

If modelling of an incompressible flow involves heat transfer fluid properties like the density change with varying temperature. If the variation of temperature is small one can still assume a constant density to maintain the structure of the advection and diffusion terms in (5) and only consider the changes of the density in the gravitational term. If linear variation of density with respect to temperature is assumed this approximation is called *Boussinesq*-approximation. In this case the Navier-Stokes equations are formulated using a reference pressure ρ_0 at the reference temperature T_0 and the now temperature dependent density ρ, with

$$\rho(T) = \rho_0\left(1 - \beta\left(T - T_0\right)\right). \tag{10}$$

Here β denotes the coefficient of thermal expansion. Under the use of the Boussinesq-approximation the incompressible Navier-Stokes equations in differential form can be formulated as

$$\begin{aligned}
\rho_0 \frac{\partial (u_i)}{\partial t} + \rho_0 \frac{\partial}{\partial x_j} \left(u_i u_j\right) =& \rho_0 g_i + (\rho - \rho_0)\, g_i - \frac{\partial p}{\partial x_i} + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \\
=& \frac{\partial}{x_i} \left(\rho_0 g_j x_j\right) + (\rho - \rho_0)\, g_i - \frac{\partial p}{\partial x_i} + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \\
=& - \frac{\partial \hat{p}}{\partial x_i} + (\rho - \rho_0)\, g_i + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \\
=& - \frac{\partial \hat{p}}{\partial x_i} - \rho_0 \beta \,(T - T_0) + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)
\end{aligned}$$

using $\rho \mathbf{g}$ as the mass specific force.

- Talk about natural and forced convection. Differences for the solver algorithm. (s.a.) Peric P447
- Talk about flows with variation in fluid properties -> mms has to map this behaviour (Buoyancy force driven, i.e. naturally convected fluid), mixed Convection
- Also talk about non-dimensional values like Prandtl number, Rayleigh and Reynolds
- Talk about the validity of this approximation

2.6 Final Form of the Set of Equations

In the previous subsections different simplifications have been introduced which will be used throughout the thesis. The final form of the set of equations to be used is thereby presented. As further simplification the modified pressure $\hat{p}$ will be treated as p and since the use of the Boussinesq-approximation replaces the variable ρ by a linear function of the temperature T the reference pressure ρ_0 for the remainder of this thesis will be referred to as ρ. Note that incompressibility has been taken into account:

$$\frac{\partial u_i}{\partial x_i} = 0. \tag{11a}$$

$$\rho \frac{\partial (u_i)}{\partial t} + \rho \frac{\partial}{\partial x_j} \left(u_i u_j\right) = - \frac{\partial p}{\partial x_i} - \rho \beta \,(T - T_0) + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \tag{11b}$$

$$\frac{\partial (\rho T)}{\partial t} + \frac{\partial}{x_j} \left(\rho u_j T - \kappa \frac{\partial T}{\partial x_j} \right) = q_T. \tag{11c}$$

4 Implicit Finite Volume Method for Incompressible Flows – Segregated Approach

The purpose of this section is to present the discretization applied to the set of equations (11). Since the system of partial differential equations to be solved always exhibits a coupling at least between the dependent variables pressure and velocity a first solution algorithm, namely the *SIMPLE* algorithm addressed to resolve the pressure velocity coupling is introduced. Furthermore an under-relaxation factor independent method of calculating mass fluxes by interpolation is introduced and the detailed derivation of all coefficients that result from the discretization process is presented. Finally the boundary conditions, that are relevant for the present thesis will be presented.

4.1 Discretization of the Mass Balance

Integration of equation (11a) over the integration domain of a single control volume P yields after the application of Gauss' integration theorem and the additivity of the Riemann integral

$$\iint\limits_{S_f} u_i n_i \mathrm{d}S = \sum_{f\in\{w,s,b,t,n,e\}} \iint\limits_{S_f} u_i n_i \mathrm{d}S = 0.$$

In the present work the mass balance is discretized using the midpoint rule for the surface integrals and linear interpolation of the velocity to to center of mass of the surface. This leads to the following form of the mass balance:

$$\sum_{f\in\{w,s,b,t,n,e\}} u_{i,f} n_{f_i} S_f = 0, \tag{16}$$

where no interpolation to attain the values of u_i at the face S_f is performed yet, since the straightforward linear interpolation will lead to undesired oscillations in the solution fields. An interpolation method to circumvent this so called *checker boarding* effect is presented in subsection 4.2.

4.2 A Pressure-Weighted Interpolation Method for Velocities

The advantages of using a cell-centered variable arrangement are evident: The treatment of non-orthogonality is simplified and the conservation property of finite volume methods is retained [5, 20, 22, 34]. A major drawback with cell centered variable arrangements is that pressure field may delink, which will then lead to unphysical oscillations in both the pressure and the velocity results. If the oscillations are severe enough the solution algorithm might even get unstable and diverge. The described decoupling occurs, when the pressure gradient in the momentum balances and the mass fluxes in the continuity equation are discretized using central differences.

A common practice to eliminate this behaviour is the use of a momentum interpolation technique, also known as *Rhie-Chow Interpolation* [26]. The original interpolation scheme however doesn't guarantee a unique solution, independent of the amount of under-relaxation. The performance of one of the algorithms that are used in the present thesis heavily relies on the under-relaxation of variables to accomplish stability. Furthermore the original method as proposed by [26] does not account for large body forces which also may lead to unphysical results. This issues will be addressed in this subsection which at the end will present an interpolation method that assures an under-relaxation independent solution, the *pressure-weighted interpolation method* [22].

The starting point of the pressure-weighted interpolation method is formed by the discretized momentum balances at node P and an arbitrary neighbouring node Q. The discretization for finite volume methods and details including the incorporation of under-relaxation factors will be handled in subsection 4.5. The semi-discrete implicit momentum balances, if one solves for the velocity at node P or Q, read

$$u_{i,P}^{(n)} = -\frac{\alpha_{\mathbf{u}_P}}{a_{P,u_i}} \left(\sum_{F\in NB(P)} a_{F,u_i} u_{i,F}^{(n)} + b_{P,u_i}^{(n-1)} - V_P \left(\frac{\partial p}{\partial x_i} \right)_P^{(n-1)} \right) + (1-\alpha_{\mathbf{u}}) u_{i,P}^{(n-1)} \tag{17a}$$

and $$u_{i,Q}^{(n)} = -\frac{\alpha_{\mathbf{u}_Q}}{a_{Q,u_i}} \left(\sum_{F\in NB(Q)} a_{F,u_i} u_{i,F}^{(n)} + b_{Q,u_i}^{(n-1)} - V_Q \left(\frac{\partial p}{\partial x_i} \right)_Q^{(n-1)} \right) + (1-\alpha_{\mathbf{u}}) u_{i,Q}^{(n-1)} \quad , \tag{17b}$$

where the superscript $(n-1)$ denotes the previous outer iteration number. The reader should note, that the pressure gradient has not been discretized yet. This has the advantage that the selective interpolation technique [28] can be applied, which is crucial for the elimination of the mentioned oscillations. In almost the same manner a semi-discrete implicit momentum balance can be formulated for a virtual control volume located between nodes P and Q. Image 4.2 gives an interpretation of the virtual control volume.

Figure 4: Comparison of vertex oriented and cell center oriented variable arrangement

$$u_{i,f}^{(n)} = -\frac{\alpha_{\mathbf{u}_f}}{a_{f,u_i}} \left(\sum_{F \in NB(f)} a_{F,u_i} u_{i,F}^{(n)} + b_{f,u_i}^{(n-1)} - V_f \left(\frac{\partial p}{\partial x_i} \right)_f^{(n-1)} \right) + (1-\alpha_{\mathbf{u}}) u_{i,f}^{(n-1)}. \tag{18}$$

To guarantee convergence of this expression for $u_{i,f}$, under-relaxation is necessary [20]. To eliminate the undefined artifacts surging form the virtualization of a control volume the following assumptions have to be made to derive a closed expression for the velocity on the boundary face S_f

$$\frac{\alpha_{\mathbf{u}_f}}{a_{f,u_i}} \left(\sum_{F \in NB(f)} a_{F,u_i} u_{i,F}^{(n)} \right) \approx (1-\gamma_f) \frac{\alpha_{\mathbf{u}_P}}{a_{P,u_i}} \left(\sum_{F \in NB(P)} a_{F,u_i} u_{i,F}^{(n)} \right) + \gamma_f \frac{\alpha_{\mathbf{u}_Q}}{a_{Q,u_i}} \left(\sum_{F \in NB(Q)} a_{F,u_i} u_{i,F}^{(n)} \right) \tag{19a}$$

$$\text{and} \quad \frac{\alpha_{\mathbf{u}_f}}{a_{f,u_i}} \approx (1-\gamma_f) \frac{\alpha_{\mathbf{u}_P}}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}_Q}}{a_{Q,u_i}}, \tag{19b}$$

where γ_f is a geometric interpolation factor.

Using the assumptions made in equation (19) the expression in equation (18) can be closed in a way that it only depends on the variable values in node P and Q

$$\begin{aligned}
u_{i,f}^{(n)} &\approx (1-\gamma_f) \left(-\frac{\alpha_{\mathbf{u}}}{a_{P,u_i}} \sum_{F \in NB(P)} a_{F,u_i} u_{i,F}^{(n)} \right) + \gamma_f \left(-\frac{\alpha_{\mathbf{u}}}{a_{Q,u_i}} \sum_{F \in NB(Q)} a_{F,u_i} u_{i,F}^{(n)} \right) \\
&\quad + \frac{\alpha_{\mathbf{u}}}{a_{f,u_i}} b_{f,u_i}^{(n-1)} - \frac{\alpha_{\mathbf{u}_f}}{a_{f,u_i}} V_f \left(\frac{\partial p}{\partial x_i} \right)_f^{(n-1)} + (1-\alpha_{\mathbf{u}}) u_{i,f}^{(n-1)} \\
&= (1-\gamma_f) u_{i,P}^{(n)} - (1-\gamma_f) \left(b_{Q,u_i}^{(n-1)} - V_Q \left(\frac{\partial p}{\partial x_i} \right)_Q^{(n-1)} \right) + \gamma_f u_{i,Q}^{(n)} - \gamma_f \left(b_{Q,u_i}^{(n-1)} - V_Q \left(\frac{\partial p}{\partial x_i} \right)_Q^{(n-1)} \right) \\
&\quad + \frac{\alpha_{\mathbf{u}_f}}{a_{f,u_i}} b_{f,u_i}^{(n-1)} - \frac{\alpha_{\mathbf{u}_f}}{a_{f,u_i}} V_f \left(\frac{\partial p^{(n-1)}}{\partial x_i} \right)_f + (1-\alpha_{\mathbf{u}}) u_{i,f}^{(n-1)} \\
&= \left[(1-\gamma_f) u_{i,P}^{(n)} + \gamma_f u_{i,Q}^{(n)} \right] \\
&\quad - \left[\left((1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \right) \left(\frac{\partial p}{\partial x_i} \right)_f^{(n-1)} - (1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} \left(\frac{\partial p}{\partial x_i} \right)_P^{(n-1)} - \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \left(\frac{\partial p}{\partial x_i} \right)_Q^{(n-1)} \right] \\
&\quad + (1-\alpha) \left[u_{i,f}^{(n-1)} - (1-\gamma_f) u_{i,P}^{(n-1)} - \gamma_f u_{i,Q}^{(n-1)} \right] \\
&\approx \left[(1-\gamma_f) u_{i,P}^{(n)} + \gamma_f u_{i,Q}^{(n)} \right] \\
&\quad - \left((1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \right) \left[\underline{ \left(\frac{\partial p}{\partial x_i} \right)_f^{(n-1)} - (1-\gamma_f) \left(\frac{\partial p}{\partial x_i} \right)_P^{(n-1)} - \gamma_f \left(\frac{\partial p}{\partial x_i} \right)_Q^{(n-1)} } \right] \\
&\quad + (1-\alpha_{\mathbf{u}}) \left[u_{i,f}^{(n-1)} - (1-\gamma_f) u_{i,P}^{(n-1)} - \gamma_f u_{i,Q}^{(n-1)} \right].
\end{aligned} \tag{20}$$

It should be noted that the argumentation that led to the last expression, is that the task of the underlined pressure gradient corrector in equation (20) is to suppress oscillations in the pressure field. If there are no oscillations this part should not become active. As long as the behaviour of this corrector remains consistent, i.e. that there are no oscillations in the pressure field, it can be multiplied with arbitrary constants [11]. This is however true on equidistant grids, where $\gamma_f = 1/2$ and central differences are used to calculate the gradients. On arbitrary orthogonal grids another modification has to be performed which is based on a special case of the mean value theorem of differential calculus and the following

Proposition. *Let $x_1, x_2 \in \mathbb{R}$ with $x_1 \neq x_2$ and $p(x) = a_0 + a_1 x + a_2 x^2$ a real polynomial function. Then*

$$\frac{dp}{dx}\left(\frac{x_1 + x_2}{2}\right) = \frac{p(x_2) - p(x_1)}{x_2 - x_1},$$

i.e. the slope of the secant equals the value of the first derivative of p exactly half the way between x_1 and x_2.

Proof. Evaluation of the derivative yields

$$\frac{dp}{dx}\left(\frac{x_1 + x_2}{2}\right) = a_1 + 2a_2 \frac{x_1 + x_2}{2} = a_1 + a_2(x_1 + x_2).$$

On the other hand the slope of the secant, using the third binomial rule can be expressed as

$$\begin{aligned}
\frac{p(x_2) - p(x_1)}{x_2 - x_1} &= \frac{a_0 + a_1 x_2 + a_2 x_2^2 - \left(a_0 + a_1 x_1 + a_2 x_1^2\right)}{x_2 - x_1} \\
&= \frac{a_1(x_2 - x_1) + a_2\left(x_2^2 - x_1^2\right)}{x_2 - x_1} \\
&= a_1 + a_2(x_2 + x_1).
\end{aligned}$$

The comparison of both expressions completes the proof. □

It is desirable for the pressure corrector to vanish independent of the grid spacing if the profile of the pressure is quadratic and hence does not exhibit oscillations. According to the preceding proposition this can be accomplished by modifying equation (20) to average the pressure gradients from node P and Q instead of interpolating linearly

$$\begin{aligned}
u_{i,f}^{(n)} = &\left[(1-\gamma_f)u_{i,P}^{(n)} + \gamma_f u_{i,Q}^{(n)}\right] \\
&- \left((1-\gamma_f)\frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}}\right)\left[\left(\frac{\partial p}{\partial x_i}\right)_f^{(n-1)} - \frac{1}{2}\left(\left(\frac{\partial p}{\partial x_i}\right)_P^{(n-1)} + \left(\frac{\partial p}{\partial x_i}\right)_Q^{(n-1)}\right)\right] \\
&+ \underline{(1-\alpha_{\mathbf{u}})\left[u_{i,f}^{(n-1)} - (1-\gamma_f)u_{i,P}^{(n-1)} - \gamma_f u_{i,Q}^{(n-1)}\right]}.
\end{aligned} \tag{21}$$

Comparing this final expression with the standard interpolation scheme it is evident that the underlined term is not taken into consideration normally [11]. However section 7.5 shows that neglecting this term creates under-relaxation factor dependent results indeed. This section concludes with a final

Proposition. *The pressure weighted momentum interpolation scheme (21) guarantees the converged solution for $u_{i,f}$ to be independent of the velocity under-relaxation $\alpha_{\mathbf{u}}$.*

Proof. An equivalent formulation of (21) is given by

$$\begin{aligned}
\alpha_{\mathbf{u}} u_{i,f}^{(n-1)} + u_{i,f}^{(n-1)} - u_{i,f}^{(n)} = \alpha_{\mathbf{u}} &\left[(1-\gamma_f)u_{i,P}^{(n-1)} + \gamma_f u_{i,Q}^{(n-1)}\right] \\
&+ \left[(1-\gamma_f)\left(u_{i,P}^{(n)} - u_{i,P}^{(n-1)}\right) + \gamma_f\left(u_{i,Q}^{(n)} - u_{i,Q}^{(n-1)}\right)\right] \\
&- \alpha_{\mathbf{u}}\left((1-\gamma_f)\frac{V_P}{a_{P,u_i}} + \gamma_f \frac{V_Q}{a_{Q,u_i}}\right)\left[\left(\frac{\partial p}{\partial x_i}\right)_f^{(n-1)} - \frac{1}{2}\left(\left(\frac{\partial p}{\partial x_i}\right)_P^{(n-1)} + \left(\frac{\partial p}{\partial x_i}\right)_Q^{(n-1)}\right)\right].
\end{aligned}$$

Upon convergence $u_{i,P}^{(n)} = u_{i,P}^{(n-1)}$ and $u_{i,f}^{(n)} = u_{i,f}^{(n-1)}$. This leads to

$$\begin{aligned}
\alpha_{\mathbf{u}} u_{i,f}^{(n-1)} = \alpha_{\mathbf{u}} &\left[(1-\gamma_f)u_{i,P}^{(n-1)} + \gamma_f u_{i,Q}^{(n-1)}\right] \\
&- \alpha_{\mathbf{u}}\left((1-\gamma_f)\frac{V_P}{a_{P,u_i}} + \gamma_f \frac{V_Q}{a_{Q,u_i}}\right)\left[\left(\frac{\partial p}{\partial x_i}\right)_f^{(n-1)} - \frac{1}{2}\left(\left(\frac{\partial p}{\partial x_i}\right)_P^{(n-1)} + \left(\frac{\partial p}{\partial x_i}\right)_Q^{(n-1)}\right)\right],
\end{aligned}$$

which shows, after division by $\alpha_{\mathbf{u}} > 0$, that $u_{i,f}$ is independent of the under-relaxation factor. □

4.3 Implicit Pressure Correction and the SIMPLE Algorithm

The goal of finite volume methods is to deduce a system of linear algebraic equations from a partial differential equation. In the case of the momentum balances the general structure of this linear equations is

$$u_{i,P}^{(n)} = -\frac{\alpha_{\mathbf{u}P}}{a_{P,u_i}} \left(\sum_{F\in NB(P)} a_{F,u_i} u_{i,F}^{(n)} + b_{P,u_i}^{(n-1)} - V_P \left(\frac{\partial p}{\partial x_i}\right)_P^{(n-1)} \right) + (1-\alpha_{\mathbf{u}}) u_{i,P}^{(n-1)} \tag{22}$$

where the pressure gradient has been discretized only symbolically and b_{P,u_i} denotes the source term. At this stage the equations are still coupled and non-linear. As described in section 3.4 the Picard iteration process can be used to linearize the equations. Every momentum balance equation then only depends on the one dominant variable u_i. Furthermore the coupling of the momentum balances through the convective term $(u_i u_j)$ is resolved in the process of linearization. The decoupled momentum balances can then be solved sequentially for the dominant variable u_i. All coefficients $a_{\{P,F\},u_i}$, the source term and the pressure gradient will be evaluated explicitly by using results of the preceding outer iteration $(n-1)$. For the pressure gradient this is to be interpreted as taking the pressure of the antecedent iteration outer iteration as a first guess for the following iteration that has to be corrected until all the non-linear equations are fulfilled up to a certain tolerance. Section (6.4.2) presents a suitable convergence criterion and its implementation. This linearization process in conjunction with the pressure guess leads to the following linear equation

$$u_{i,P}^{(n*)} = -\frac{\alpha_{\mathbf{u}P}}{a_{P,u_i}} \left(\sum_{F\in NB(P)} a_{F,u_i} u_{i,F}^{(n*)} + b_{P,u_i}^{(n-1)} - V_P \left(\frac{\partial p}{\partial x_i}\right)_P^{(n-1)} \right) + (1-\alpha_{\mathbf{u}}) u_{i,P}^{(n-1)} \tag{23}$$

Here $(*)$ indicates that the solution of this equation still needs to be corrected to also fulfill the discretized mass balance

$$\sum_{F\in NB(P)} (u_i)_f^{(n)} n_i S_f = 0. \tag{24}$$

Applying the same procedure as in section 4.2 to equation (23) results in the following expression for the face velocities after solving the discretized momentum balances using a pressure guess

$$\begin{aligned} u_{i,f}^{(n*)} &= \left[(1-\gamma_f) u_{i,P}^{(n*)} + \gamma_f u_{i,Q}^{(n*)} \right] \\ &\quad - \left((1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \right) \left[\left(\frac{\partial p}{\partial x_i}\right)_f^{(n-1)} - (1-\gamma_f)\left(\frac{\partial p}{\partial x_i}\right)_P^{(n-1)} - \gamma_f \left(\frac{\partial p}{\partial x_i}\right)_Q^{(n-1)} \right] \\ &\quad + (1-\alpha_{\mathbf{u}}) \left[u_{i,f}^{(n-1)} - (1-\gamma_f) u_{i,P}^{(n-1)} - \gamma_f u_{i,Q}^{(n-1)} \right]. \end{aligned} \tag{25}$$

The lack of an equation with the pressure as dominant variable leads to the necessity to alter the mass balance as the only equation left. Methods of this type are called projection methods. A common class of algorithms of this family of methods uses an equation for the additive pressure correction p' instead of the pressure itself and enforces continuity by correcting the velocities with an additive corrector u_i' as in

$$u_{i,P}^{(n)} = u_{i,P}^{(n*)} + u_{i,P}', \quad u_{i,f}^{(n)} = u_{i,f}^{(n*)} + u_{i,f}' \quad \text{and} \quad p_P^{(n)} = p_P^{(n-1)} + p_P'.$$

It is now possible to formulate the discretized momentum balance for the corrected velocities and the corrected pressure as

$$u_{i,P}^{(n)} = -\frac{\alpha_{\mathbf{u}P}}{a_{P,u_i}} \left(\sum_{F\in NB(P)} a_{F,u_i} u_{i,F}^{(n)} + b_{P,u_i}^{(n-1)} - V_P \left(\frac{\partial p}{\partial x_i}\right)_P^{(n)} \right) + (1-\alpha_{\mathbf{u}}) u_{i,P}^{(n-1)}. \tag{26}$$

It should be noted that the only difference to the equation which will be solved in the next outer iteration is that the source term b_{P,u_i} has not been updated yet. The same applies for the equation for the face velocity $u_{i,f}$

$$\begin{aligned} u_{i,f}^{(n)} &= \left[(1-\gamma_f) u_{i,P}^{(n)} + \gamma_f u_{i,Q}^{(n)} \right] \\ &\quad - \left((1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \right) \left[\left(\frac{\partial p}{\partial x_i}\right)_f^{(n)} - (1-\gamma_f)\left(\frac{\partial p}{\partial x_i}\right)_P^{(n)} - \gamma_f \left(\frac{\partial p}{\partial x_i}\right)_Q^{(n)} \right] \\ &\quad + (1-\alpha_{\mathbf{u}}) \left[u_{i,f}^{(n-1)} - (1-\gamma_f) u_{i,P}^{(n-1)} - \gamma_f u_{i,Q}^{(n-1)} \right]. \end{aligned} \tag{27}$$

To couple velocity and pressure correctors one can subtract equations (23) from (26) and (25) from (27) to get

$$u'_{i,P} = -\frac{\alpha_{\mathbf{u}_P}}{a_{P,u_i}} \left(\underline{\sum_{F \in NB(P)} a_{F,u_i} u'_{i,F}} - V_P \left(\frac{\partial p'}{\partial x_i} \right)_P^{(n)} \right) \quad \text{and} \tag{28}$$

$$u'_{i,f} = \left[(1-\gamma_f) u'_{i,P} + \gamma_f u'_{i,Q} \right] - \left((1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \right) \left[\left(\frac{\partial p}{\partial x_i} \right)'_f - (1-\gamma_f) \left(\frac{\partial p}{\partial x_i} \right)'_P - \gamma_f \left(\frac{\partial p}{\partial x_i} \right)'_Q \right]. \tag{29}$$

The majority of the class of pressure correction algorithms has this equations as a common basis. Each algorithm then introduces special distinguishable approximations of the velocity corrections that are, at the moment of solving the pressure equation, still unknown. The method used in the present work is the SIMPLE Algorithm (Semi-Implicit Method for Pressure-Linked Equations [23]). The approximation this algorithm performs is severe since the term containing the unknown velocity corrections is dropped entirely. The respective term has been underlined in equation (28). Since the global purpose of the presented method is to enforce continuity by implicitly calculating a pressure correction, the velocity correction has to be expressed solely in terms of the pressure correction. This can be accomplished by inserting equation (28) in to equation (29). This gives an update formula

$$u'_{i,f} = -\left((1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \right) \left(\frac{\partial p}{\partial x_i} \right)'_f , \tag{30}$$

which is then, together with (25), inserted into the discretized continuity equation (24) to obtain

$$\sum_{F \in NB(P)} \left((1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_F}{a_{F,u_i}} \right) \left(\frac{\partial p}{\partial x_i} \right)'_f n_i S_f = b_{P,p} \quad , \tag{31}$$

where the right hand side $b_{P,p}$ is defined as

$$b_{P,p} := \sum_{F \in NB(P)} u_{i,f}^{(n*)} n_i S_f . \tag{32}$$

The complete discretization with central differences as approximation for the gradient of the pressure correction is straightforward and will be presented in subsection 4.4.

The approximation performed in the SIMPLE algorithm affects convergence in a way that the pressure correction has to be under-relaxed with a parameter $\alpha_p \in [0,1]$

$$p_P^{(n)} = p_P^{(n-1)} + \alpha_{\mathbf{p}} p'_P . \tag{33}$$

As shown in section 4.2 the behaviour of the pressure weighted interpolation method on non-equidistant grids can be improved by replacing the linear interpolation of pressure gradients with a simple average in equation (25). This leads to the following equation for calculating mass fluxes

$$\begin{aligned} u_{i,f}^{(n*)} = & \left[(1-\gamma_f) u_{i,P}^{(n*)} + \gamma_f u_{i,Q}^{(n*)} \right] \\ & - \left((1-\gamma_f) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \right) \left[\left(\frac{\partial p}{\partial x_i} \right)_f^{(n-1)} - \frac{1}{2} \left(\left(\frac{\partial p}{\partial x_i} \right)_P^{(n-1)} + \left(\frac{\partial p}{\partial x_i} \right)_Q^{(n-1)} \right) \right] \\ & + (1-\alpha_{\mathbf{u}}) \left[u_{i,f}^{(n-1)} - (1-\gamma_f) u_{i,P}^{(n-1)} - \gamma_f u_{i,Q}^{(n-1)} \right]. \end{aligned} \tag{34}$$

Generally the SIMPLE algorithm can be represented by the following iterative procedure as in Algorithm 4.3.

4.4 Discretization of the Mass Fluxes and the Pressure Correction Equation

Subsections 4.2 and 4.3 introduced the concept of pressure weighted interpolation to avoid oscillating results and an algorithm to calculate a velocity field that obeys continuity. The derived equations have not been discretized completely, furthermore the approach has not been generalized to non-orthogonal grids.

Algorithm 1 SIMPLE Algorithm

INITIALIZE variables
while (convergence criterion not accomplished) **do**
  SOLVE linearized momentum balances, equation **(23)**
  CALCULATE mass fluxes using **(27)** or **(34)**
  SOLVE pressure correction equation to assure continuity, equation **(37)**
  UPDATE pressure using **(33)**
  UPDATE velocities and mass fluxes using **(28)**
  if (Coupled scalar equation) **then**
    SOLVE scalar equation as described in **(4.6)**
  end if
end while

The discretized mass balance (16) only depends on the normal velocities $u_{i,f} n_{i,f}$. By analogy with equation (34) an interpolated normal face velocity and thus the mass flux can be calculated as

$$\begin{aligned} u_{i,f}^{(n*)} n_{i,f} &= \left[\left(1-\gamma_f\right) u_{i,P}^{(n*)} + \gamma_f u_{i,Q}^{(n*)} \right] n_{i,f} \\ &- \left(\left(1-\gamma_f\right) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_Q}{a_{Q,u_i}} \right) \left[\left(\frac{\partial p}{\partial n} \right)_f^{(n-1)} - \frac{1}{2} \left(\left(\frac{\partial p}{\partial n} \right)_P^{(n-1)} + \left(\frac{\partial p}{\partial n} \right)_Q^{(n-1)} \right) \right] \\ &+ \left(1-\alpha_{\mathbf{u}}\right) \left[u_{i,f}^{(n-1)} - \left(1-\gamma_f\right) u_{i,P}^{(n-1)} - \gamma_f u_{i,Q}^{(n-1)} \right] n_{i,f}, \end{aligned} \tag{35}$$

where the scalar product of pressure gradients and the normal vector has been replaced by a directional derivative in the direction of the face normal vector. In the present work pressure gradients in (35) and pressure correction gradients in equation (37) and will be discretized by central differences

$$\left(\frac{\partial p}{\partial n} \right)_f \approx \frac{p_P - p_Q}{\left(\mathbf{x}_P - \mathbf{x}_Q\right) \cdot \mathbf{n}_f} \quad \text{and} \quad \left(\frac{\partial p'}{\partial n} \right)_f \approx \frac{p'_P - p'_Q}{\left(\mathbf{x}_P - \mathbf{x}_Q\right) \cdot \mathbf{n}_f}. \tag{36}$$

This discretization can then be inserted into the semi-discretized pressure correction equation (37)

$$\sum_{F \in NB(P)} \left(\left(1-\gamma_f\right) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_F}{a_{F,u_i}} \right) \frac{p'_P - p'_F}{\left(\mathbf{x}_P - \mathbf{x}_F\right) \cdot \mathbf{n}_f} S_f = b_{P,p}. \tag{37}$$

The resulting coefficients for the pressure correction equation

$$a_{P,p'} p'_P + \sum_{F \in NB(P)} a_{F,p'} p'_F = b_{P,p'},$$

can be calculated as

$$a_{F,p'} = -\left(\left(1-\gamma_f\right) \frac{\alpha_{\mathbf{u}} V_P}{a_{P,u_i}} + \gamma_f \frac{\alpha_{\mathbf{u}} V_F}{a_{F,u_i}} \right) \frac{S_f}{\left(\mathbf{x}_P - \mathbf{x}_F\right) \cdot \mathbf{n}_f} \quad \text{and} \quad a_{P,p'} = \sum_{F \in NB(P)} -a_{F,p'}.$$

The right hand side can be calculated as in equation (32), if the presented discretization is applied.

4.5 Discretization of the Momentum Balance

The stationary momentum balance integrated over a single control volume P reads as

$$\underbrace{\iint_S (\rho u_i u_j) n_j \mathrm{d}S}_{\text{convective term}} - \underbrace{\iint_S \left(\mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right) n_j \mathrm{d}S}_{\text{diffusive term}} = - \underbrace{\iiint_V \frac{\partial p}{\partial x_i} \mathrm{d}V}_{\text{sourceterm pressure}} - \underbrace{\iiint_V \rho \beta \left(T - T_0\right) \mathrm{d}V}_{\text{sourceterm temperature}}, \tag{38}$$

where the different terms to be addressed individually in the following sections are indicated. The reader should note that the form of this equation has been modified by using Gauss' integration theorem. The terms residing on the left will be treated in an implicit and due to deferred corrections in an explicit manner whereas the terms on the right will be treated exclusively in an explicit manner.

4.5.1 Linearization and Discretization of the Convective Term

The convective term $\left(\rho u_i u_j\right)$ of the Navier-Stokes equations is the reason for the non-linearity of the equations. In order to deduce a set of linear algebraic equations from the Navier-Stokes equations this term has to be linearized. As introduced in section (3.4), the non linearity will be dealt with by means of an iterative process, the Picard iteration. The part dependent on the non dominant dependent variable therefore will be approximated by its value from the previous iteration as $\rho u_i^{(n)} u_j^{(n)} \approx \rho u_i^{(n)} u_j^{(n-1)}$. However this linearization will not be directly visible because it will be covered by the mass flux $\dot{m}_f = \iint\limits_{S_f} \rho u_j^{(n-1)} n_j \mathrm{d}S$. Using the additivity of the Riemann integral the first step is to decompose the surface integral into individual contributions from each boundary face of the control volume P

$$\iint\limits_{S} \rho u_i u_j n_j \mathrm{d}S = \sum_{f \in \{w,s,b,t,n,e\}} \iint\limits_{S_f} \rho u_i u_j n_j \mathrm{d}S = \sum_{f \in \{w,s,b,t,n,e\}} F_{i,f}^c,$$

where $F_{i,f}^c := \iint\limits_{S_f} \rho u_i^{(n)} u_j^{(n-1)} n_j \mathrm{d}S$ is the convective flux of the velocity u_i through the boundary face S_f.

To improve diagonal dominance of the resulting linear system while maintaining the smaller discretization error of a higher order discretization, a blended discretization scheme is applied and combined with a deferred correction. Since due to the non-linearity of the equations to be solved an iterative solution process is needed by all means, the overall convergence doesn't degrade noticeably when using a deferred correction [11]. Blending and deferred correction result in a decomposition of the convective flux into a lower order approximation that is treated implicitly and the explicit difference between the higher and lower order approximation for the same convective flux. Since for coarse grid resolutions the use of higher order approximations may lead to oscillations of the solution which in turn may degrade or even impede convergence, the schemes can be blended by a control factor $\eta \in [0,1]$. To show the generality of this approach all further derivations are presented for the generic boundary face S_f that separates control volume P from its neighbour $F \in NB(P)$. This decomposition then leads to

$$F_{i,f}^c \approx \underbrace{F_{i,f}^{c,l}}_{\text{implicit}} + \eta \underbrace{\left[F_{i,f}^{c,h} - F_{i,f}^{c,l}\right]^{(n-1)}}_{\text{explicit}}.$$

It should be noted that the convective fluxes carrying an l for *lower* or an h for *higher* as exponent, already have been linearized and discretized. The discretization applied to the convective flux in the present work is using the midpoint integration rule and blends the upwind interpolation scheme with a linear interpolation scheme. Applied to above decomposition one can derive the following approximations

$$\begin{aligned} F_{i,f}^{c,l} &= u_{i,F} \min(\dot{m}_f, 0) + u_{i,P} \max(0, \dot{m}_f) \\ F_{i,f}^{c,h} &= u_{i,F}\, \gamma_f + u_{i,P}\, (1 - \gamma_f), \end{aligned}$$

where the variable values have to be taken from the previous iteration step $(n-1)$ as necessary and the mass flux $\dot{m}_f$ has been used as result of the linearization process. The results can now be summarized by presenting the convective contribution to the matrix coefficients a_{F,u_i} and a_{P,u_i} and the right hand side b_{P,u_i} which are calculated as

$$a_{F,u_i}^c = \min(\dot{m}_f, 0), \qquad a_{P,u_i}^c = \sum_{F \in NB(P)} \max(0, \dot{m}_f) \tag{39a}$$

$$\begin{aligned} b_{P,u_i}^c = \sum_{F \in NB(P)} & \eta \left(u_{i,F}^{(n-1)} \left(\min(\dot{m}_f, 0) - \gamma_f \right) \right) \\ & + \eta \left(u_{i,P}^{(n-1)} \left(\max(0, \dot{m}_f) - \left(1 - \gamma_f\right) \right) \right). \end{aligned} \tag{39b}$$

4.5.2 Discretization of the Diffusive Term

The diffusive term contains the first partial derivatives of the velocity as a result of the material constitutive equation that characterizes the behaviour of Newtonian fluids. As pointed out in section 3.3 directional derivatives can be discretized using central differences on orthogonal grids or in the more general case of non-orthogonal grids using central differences implicitly and an explicit deferred correction comprising the non-orthogonality of the grid. As seen in equation (8) the diffusive term of the Navier-Stokes equations can be simplified using the mass balance in the case of an incompressible flow with constant viscosity μ. To sustain the generality of the presented approach this simplification will be omitted.

As before, by using the additivity and furthermore linearity of the Riemann integral, the integration of the diffusive term will be divided into integration over individual boundary faces S_f

$$\iint_S \left(\mu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right) n_j \mathrm{d}S\mathrm{d}S = \sum_{f\in\{w,s,b,t,n,e\}}\left[\underline{\iint_{S_f}\mu\frac{\partial u_i}{\partial x_j}n_j\mathrm{d}S}+\iint_{S_f}\mu\frac{\partial u_j}{\partial x_i}n_j\mathrm{d}S\right]=\sum_{f\in\{w,s,b,t,n,e\}}F^d_{i,f},$$

where $F^d_{i,f}$ denotes the diffusive flux through an individual boundary face. Section 3.3 only covered the non-orthogonal corrector for directional derivatives. Since the velocity is a vector field and not a scalar field, the results of section 3.3 may only be applied to the underlined term. The other term will be treated explicitly since it is considerably smaller than the underlined term and does not cause oscillations and thus will not derogate convergence [11]. To begin with all present integrals will be approximated using the midpoint rule of integration. The diffusive flux $F^d_{i,f}$ for a generic face S_f then reads

$$F^d_{i,f}\approx \underline{\mu\left(\frac{\partial u_i}{\partial x_j}\right)_f n_j S_f} + \mu\left(\frac{\partial u_j}{\partial x_i}\right)_f n_j S_f .$$

Using central differences for the implicit discretization of the directional derivative and furthermore using the *orthogonal correction* approach from 3.3.2 the approximation can be derived as

$$\begin{aligned}F^d_{i,f} &\approx \underline{\mu\left(||\mathbf{\Delta}_f||_2\frac{u_{P_i}-u_{F_i}}{||\mathbf{x}_P-\mathbf{x}_F||_2}-(\nabla u_i)_f^{(n-1)}\cdot(\mathbf{\Delta}_f-\mathbf{S}_f)\right)}+\mu\left(\frac{\partial u_j}{\partial x_i}\right)_f^{(n-1)} n_{f_i}\\
&= \underline{\mu\left(S_f\frac{u_{P_i}-u_{F_i}}{||\mathbf{x}_P-\mathbf{x}_F||_2}-\left(\frac{\partial u_i}{\partial x_j}\right)_f^{(n-1)}(\xi_{f_i}-n_{f_i})S_f\right)}+\mu\left(\frac{\partial u_j}{\partial x_i}\right)_f^{(n-1)} n_{f_i},\end{aligned}$$

where the unit vector pointing in direction of the straight line connecting control volume P and control volume F is denoted as

$$\xi_f=\frac{\mathbf{x}_P-\mathbf{x}_F}{||\mathbf{x}_p-\mathbf{x}_F||_2}.$$

The interpolation of the cell center gradients to the boundary faces is performed as in (13). Now the contribution of the diffusive part to the matrix coefficients and the right hand side can be calculated as

$$a^d_{F,u_i}=-\frac{\mu S_f}{||\mathbf{x}_P-\mathbf{x}_F||_2},\qquad a^d_{P,u_i}=\sum_{F\in NB(P)}\frac{\mu S_f}{||\mathbf{x}_P-\mathbf{x}_F||} \tag{40a}$$

$$\begin{aligned}b^d_{F,u_i}&=\sum_{F\in NB(P)}\left(\frac{\partial u_i}{\partial x_j}\right)_f^{(n-1)}(\xi_{f_i}-n_{f_i})S_f-\mu\left(\frac{\partial u_j}{\partial x_i}\right)_f^{(n-1)} n_{f_i}S_f\\
&=\sum_{F\in NB(P)}\left(\frac{\partial u_i}{\partial x_j}\right)_f^{(n-1)}\xi_{f_i}S_f-\mu\left(\left(\frac{\partial u_i}{\partial x_j}\right)_f^{(n-1)}-\left(\frac{\partial u_j}{\partial x_i}\right)_f^{(n-1)}\right)n_{f_i}S_f.\end{aligned} \tag{40b}$$

4.5.3 Discretization of the Source Terms

Since in the segregated solution approach in every equation all other variables but the dominant one are treated as constants and furthermore the source terms in equation (38) do not depend on the dominant variable the discretization is straightforward. The source terms of the momentum balance are discretized using the midpoint rule of integration, which leads to the source term

$$-\iiint_V\frac{\partial p}{\partial x_i}\mathrm{d}V-\iiint_V\rho\beta(T-T_0)\mathrm{d}V\approx-\left(\frac{\partial p}{\partial x_i}\right)_P^{(n-1)}V_P-\rho\beta\left(T_P^{(n-1)}-T_0\right)V_P=b^{sc}_{P,u_i} \tag{41}$$

4.6 Discretization of the Temperature Equation

The discretization of the temperature equation is performed by the same means as for the momentum balance. The only difference is a simpler diffusion term. The integral form of the temperature equation after applying the Gauss' theorem of integration is

$$\underbrace{\iint_S \rho u_j T n_j \mathrm{d}S}_{\text{advective term}} - \underbrace{\iint_S \kappa \frac{\partial T}{\partial x_j} n_j \mathrm{d}V}_{\text{diffusive term}} = \underbrace{\iiint_V q_T \mathrm{d}V}_{\text{source term}}.$$

Proceeding as in the previous subsections one can now discretize the advective, the diffusive term and the source term. Since this process does not provide further insight, just the final results will be presented. The discretization yields the matrix coefficients as

$$a_{F,T} = \min(\dot{m}_f, 0) + \frac{\kappa S_f}{||\mathbf{x}_P - \mathbf{x}_F||_2} \tag{42a}$$

$$a_{P,T} = \sum_{F \in NB(P)} \max(0, \dot{m}_f) - \frac{\kappa S_f}{||\mathbf{x}_P - \mathbf{x}_F||_2} \tag{42b}$$

$$\begin{aligned} b_{P,T} = & \sum_{F \in NB(P)} \eta \left(T_F^{(n-1)} \left(\min(\dot{m}_f, 0) - \gamma_f \right) \right) \\ & \qquad + \eta \left(T_P^{(n-1)} \left(\max(0, \dot{m}_f) - (1 - \gamma_f) \right) \right) \\ & + \sum_{F \in NB(P)} \left(\frac{\partial T}{\partial x_j} \right)_f^{(n-1)} \left(\xi_{f_j} - n_{f_j} \right) S_f \\ & + q_{T_P} V_P. \end{aligned} \tag{42c}$$

Again it is possible though not always necessary, as in the case of the velocities, to under-relax the solution of the resulting linear system with a factor α_T. This can be accomplished as shown in the previous sections.

4.7 Boundary Conditions

As shown it is possible to deduce a linear algebraic equation for each control volume. The approach presented in the preceding subsection however did not cover the treatment of control volumes at the domain boundaries. This subsection introduces the boundary conditions which are relevant for the present work and furthermore deals with transition conditions at block boundaries.

4.7.1 Dirichlet Boundary Conditions

The first boundary condition is the Dirichlet boundary condition. It is characterized by specifying the value of the variable for which the equation is solved explicitly. As a result boundary fluxes can be calculated directly. Especially the mass flux $\dot{m}_f$ is known and hence has not to be calculated using the pressure weighted interpolation method. Since no special modifications have to be made as the resulting coefficient for a neighbouring control volume laying past the boundary is considered on the right hand side of the linear system, the implementation approach will be presented only for the temperature equation. Since there is no boundary condition that fixes the gradient at Dirichlet boundaries it is assumed that the partial derivatives of the respective variable is constant and can hence be extrapolated

$$\left(\frac{\partial T}{\partial x_j} \right)_f \approx \left(\frac{\partial T}{\partial x_j} \right)_P.$$

The modification to the central coefficient of the linear equation can be recursively formulated as

$$a_{P,T} = a_{P,T} + \left(\max(0, \dot{m}_f) - \frac{\kappa S_f}{||\mathbf{x}_P - \mathbf{x}_f||_2} \right),$$

whereas the contribution to the right hand side reads

$$b_{P,T} = b_{P,T} - \left(\min(\dot{m}_f, 0) - \frac{\kappa S_f}{||\mathbf{x}_P - \mathbf{x}_f||_2} \right) T_f + \left(\frac{\partial T}{\partial x_j} \right)_P^{(n-1)} \left(\xi_{f_j} - n_{f_j} \right) S_f$$

The reader should note, that instead the gradient discretization at boundaries is realized by one sided forward differencing schemes instead of a central differencing scheme. This does not affect accuracy drastically because the distance used in the differential quotient is half the distance used on an central difference inside the domain [28].

4.7.2 Treatment of Wall Boundaries

A common boundary to the solution domain is given by solid walls. In viscous flows this can be interpreted as a no-slip condition, i.e. a Dirichlet boundary condition for the velocities. Convective fluxes through solid walls are thus zero by definition however the diffusive fluxes require special treatment not only for the velocities but also for the temperature. To approximate the fluid behavior on a wall boundary correctly, special modifications have to be taken into account to model the normal an shear tension. Furthermore diffusive fluxes for the temperature can be given by Neumann boundary conditions.

The derivation of the discretized diffusive flux through wall boundaries starts from the integral momentum balance (2) for the vector $\mathbf{u}$. Here only the term for surface forces is needed

$$\mathbf{F}_w = \iint_{S_w} \mathbf{t} \mathrm{d}S = \iint_{S_w} \mathbf{T}(\mathbf{n}_w) \mathrm{d}S \tag{43}$$

For the purpose of treating wall boundary conditions it is appropriate to use a local coordinate system n, t, s where n denotes the wall normal coordinate, t denotes the coordinate tangential to the wall shear force and, s is the binormal coordinate. (FIGURE). With respect to this coordinate system the wall normal vector is represented by $\mathbf{n}_w = (1, 0, 0)^T$ and the image of the wall normal vector $\mathbf{T}(\mathbf{n_w})$ is represented by

$$\mathbf{T}(\mathbf{n}_w) = \begin{bmatrix} \tau_{nn} & \tau_{nt} & \tau_{ns} \\ \tau_{nt} & \tau_{tt} & \tau_{ts} \\ \tau_{ns} & \tau_{ts} & \tau_{ss} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \tau_{nn} \\ \tau_{nt} \\ \tau_{ns} \end{bmatrix}$$

The velocity of the wall is assumed to be constant, so the directional derivative of the tangential velocity vanishes on the wall

$$\tau_{tt} = \frac{\partial u_t}{\partial x_t} = 0,$$

which in conjunction with the continuity equation in differential form leads to

$$\frac{\partial u_n}{\partial x_n} + \frac{\partial u_t}{\partial x_t} = \frac{\partial u_n}{\partial x_n} = 0,$$

what is equivalent to $\tau_{nn} = 0$ at the wall. An physical interpretation would be that the transfer of momentum at the wall occurs by shear forces exclusively. Furthermore the coordinate direction t is chosen to be parallel to the shear force which is no restriction because of the possibility to rotate the coordinate system within the plane. This leads to $\tau_{ns} = 0$. The absolute value of the surface force hence only depends on the normal derivative of the velocity tangential to the wall. After transforming the coordinates back to the system (x_1, x_2, x_3) the surface force can be calculated and the integral can be discretized using the midpoint rule by

$$\mathbf{F}_w = \iint_{S_w} \mathbf{t_w} \, \tau_{nt} \mathrm{d}S = \iint_{S_w} \mathbf{t_w} \, \mu \frac{\partial u_t}{\partial n} \mathrm{d}S = \mathbf{t_w} \, \mu \left(\frac{\partial u_t}{\partial n} \right)_w S_w \tag{44}$$

where $\mathbf{t_w}$ denotes the transformed tangential vector $(0, 1, 0)^T$ with respect to the coordinate system (x_1, x_2, x_3). In the discretization process this tangential vector will be calculated from the velocity vector as

$$\mathbf{t_w} = \frac{\mathbf{u}_t}{||\mathbf{u}_t||_2} \quad \text{, where} \quad \mathbf{u}_t = \mathbf{u} - (\mathbf{u} \cdot \mathbf{n}_w)\, \mathbf{n}.$$

According to [11] this force should not be handled explicitly for convergence reasons. On the other side, if the surface force is expressed by the velocities u_i the discretization process would lead to different central matrix coefficients, which not only affects memory efficiency but also the discretization of the pressure correction equation, because in the process of the derivation the central coefficients were assumed to be independent of the three momentum balances.

The discretization approach used in the present thesis uses a simpler implicit discretization coupled with a deferred correction that combines the explicit discretization of the surface force (44) with a central difference. At first the contained directional derivative is discretized implicitly by

$$\left(\frac{\partial u_t}{\partial n} \right) t_{wi} \approx \left(\frac{\partial u_i}{\partial \xi} \right) \approx \frac{u_{i,P} - u_{i,w}}{||\mathbf{x}_P - \mathbf{x}_w||_2}$$

and explicitly by

$$\left(\frac{\partial u_t}{\partial n}\right) t_{wi} \approx \frac{\left(u_{i,P}-u_{i,w}\right)-\left(u_{j,P}-u_{j,w}\right) n_j n_i}{(\mathbf{x}_P-\mathbf{x}_w)\cdot \mathbf{n}_w}.$$

Therefore the contributions to the central coefficient and the right hand side of the linear equation that results from the presented discretization process are

$$a_{P,u_i} = a_{P,u_i} + \mu \frac{S_f}{||\mathbf{x}_P-\mathbf{x}_w||_2} \quad \text{and}$$
$$b_{P,u_i} = a_{P,u_i} + \mu \frac{S_f}{||\mathbf{x}_P-\mathbf{x}_w||_2} u_{i,P}^{(n-1)} + \frac{\left(u_{i,P}^{(n-1)}-u_{i,w}\right)-\left(u_{j,P}^{(n-1)}-u_{j,w}\right) n_j n_i}{(\mathbf{x}_P-\mathbf{x}_w)\cdot \mathbf{n}_w}.$$

The reader should note that since the deferred correction uses a Dirichlet boundary no correction of the value of the wall velocity $u_{i,w}$ has to be accounted for.

4.7.3 Treatment of Block Boundaries

4.8 Treatment of the Singularity of the Pressure Correction Equation with Neumann Boundaries

It has to be noted that the derived pressure correction equation is a Poisson equation. As can be proven **??** the linear $N \times N$ systems surging from the presented discretization on a grid with N control volumes have a nullspace of dimension one, i.e.

$$\text{null}(A_{p'}) = \text{span}(\mathbb{1}),$$

where $\mathbb{1} = (1)_{i=1,\dots,N} \in \mathbb{R}^N$ is the vector spanning the nullspace. This singularity accounts for the property of incompressible flows, that pressure can only be determined up to a constant. To fix this constraint various possibilities exist [11]. A common method is to set the pressure correction to zero in one reference control volume and hence at one reference point in the problem domain. This can be done before solving the system by applying this Dirichlet-type condition, or it can be done afterwards when pressure is calculated from the pressure correction. This approach is not suitable for grid convergence studies since without proper interpolation it is not guaranteed that the reference pressure correction is taken at the correct location.

Since some comparisons performed in the present work rely on grid convergence studies another approach for reducing the lose constraint of the pressure correction system has been used: The reference pressure correction is taken to be the mean value of the pressure correction over the domain

$$p'_{\text{ref}} = \frac{\iiint\limits_V p' \mathrm{d}V}{\iiint\limits_V \mathrm{d}V} \approx \frac{\sum_{P=1}^N p'_P V_P}{\sum_{P=1}^N V_P}$$

such that the net pressure correction amounts to zero. This modifies equation (33) to

$$p_P^{(n)} = p_P^{(n-1)} + \alpha_{\mathbf{p}} \left(p'_P - p'_{\text{ref}}\right) \tag{45}$$

4.9 Structure of the Assembled Linear Systems

The objective of a finite volume method is to create a set of linear algebraic equations by discretizing partial differential equations. In the case of the discretized momentum balance taking all contributions together leads to the following linear algebraic equation for each control volume P

$$a_{P,u_i} u_{P_i} + \sum_{F \in NB(P)} a_F u_{F_i} = b_{P,u_i},$$

where the coefficients are composed as

$$a_{P,u_i} = a_{P,u_i}^c - a_{P,u_i}^d \tag{46}$$
$$a_{F,u_i} = a_{F,u_i}^c - a_{F,u_i}^d \tag{47}$$
$$b_{P,u_i} = b_{P,u_i}^c - b_{P,u_i}^d + b_{P,u_i}^{sc}. \tag{48}$$

Similar expressions for the pressure correction equation and the temperature equation exist.

In the case of control volumes located at boundaries some of the coefficients will be calculated in a different manner. This aspect is addressed in section 4.7. For the decoupled iterative solution process of the Navier-Stokes equations it is necessary to reduce the change of each dependent variable in each iteration. Normally this is done by an *under-relaxation* technique, a convex combination of the solution of the linear system for the present iteration (n) and from the previous iteration $(n-1)$ with the under-relaxation parameter α_{u_i}. Generally speaking this parameter can be chosen individually for each equation. Since there are no rules for choosing this parameters in a general setting the under-relaxation parameter for the velocities is chosen to be equal for all three velocities, $\alpha_{u_i} = \alpha_{\mathbf{u}}$. This has the further advantage that, in case the boundary conditions are implemented with the same intention, the linear system for each of the velocities remains unchanged except for the right hand side. This helps to increase memory efficiency.

Let the solution for the linear system without under-relaxation be denoted as

$$\tilde{u}_{P_i}^{(n)} := \frac{b_{P,u_i} - \sum_{F\in NB(P)} a_F u_{F_i}}{a_{P,u_i}},$$

Which is only a formal expression for the case the underlying linear system is solved exactly. A convex combination as described yields

$$\begin{aligned} u_{P_i}^{(n)} &:= \alpha_{\mathbf{u}} \tilde{u}_{P_i}^{(n)} + (1-\alpha_{\mathbf{u}})\, u_{P_i}^{(n-1)} \\ &= \alpha_{\mathbf{u}} \frac{b_{P,u_i} - \sum_{F\in NB(P)} a_F u_{F_i}}{a_{P,u_i}} + (1-\alpha_{\mathbf{u}})\, u_{P_i}^{(n-1)}, \end{aligned}$$

an expression that can be modified to derive a linear system whose solution is the under-relaxed velocity

$$\frac{a_{P,u_i}}{\alpha_{\mathbf{u}}} u_{i,P} + \sum_{F\in NB(P)} a_{F,u_i} u_{i,F} = b_{P,u_i} + \frac{(1-\alpha_{\mathbf{u}})\, a_{P,u_i}}{\alpha_{\mathbf{u}}} u_{i,P}^{(n-1)}.$$

9 Conclusion and Outlook

Turbulence (turbulent viscosity has to be updated in each iteration), Multiphase (what about discontinuities), GPU-Accelerators, Load-Balancing, dynamic mesh refinement, Counjugate Heat Transfer with other requirements for the numerical grid, grid movement, list some papers here) Identify the optimal regimes / conditions for maximizing performance. Each solver concept has its strengths and weaknesses. Try other variants of Projection Methods like SIMPLEC, SIMPLER, PISO or PIMPLE (OpenFOAM)

References

[1] BALAY, S., ABHYANKAR, S., ADAMS, M. F., BROWN, J., BRUNE, P., BUSCHELMAN, K., EIJKHOUT, V., GROPP, W. D., KAUSHIK, D., KNEPLEY, M. G., MCINNES, L. C., RUPP, K., SMITH, B. F. AND ZHANG, H. PETSc users manual. Tech. Rep. ANL-95/11 - Revision 3.5, Argonne National Laboratory, 2014.

[2] BALAY, S., ABHYANKAR, S., ADAMS, M. F., BROWN, J., BRUNE, P., BUSCHELMAN, K., EIJKHOUT, V., GROPP, W. D., KAUSHIK, D., KNEPLEY, M. G., MCINNES, L. C., RUPP, K., SMITH, B. F. AND ZHANG, H. PETSc Web page. `http://www.mcs.anl.gov/petsc`, 2014.

[3] BALAY, S., GROPP, W. D., MCINNES, L. C. AND SMITH, B. F. Efficient management of parallelism in object oriented numerical software libraries. In *Modern Software Tools in Scientific Computing* (1997), E. Arge, A. M. Bruaset and H. P. Langtangen, Eds., Birkhäuser Press, pp. 163–202.

[4] CHEN, Z. AND PRZEKWAS, A. A coupled pressure-based computational method for incompressible/compressible flows. *Journal of Computational Physics 229*, 24 (2010), 9150 – 9165.

[5] CHOI, S. K. Note on the use of momentum interpolation method for unsteady flows. *Numerical Heat Transfer, Part A: Applications 36*, 5 (1999), 545–550.

[6] CHOI, S.-K., KIM, S.-O., LEE, C.-H. AND CHOI, H.-K. Use of the momentum interpolation method for flows with a large body force. *Numerical Heat Transfer, Part B: Fundamentals 43*, 3 (2003), 267–287.

[7] CHRISTON, M. A., GRESHO, P. M. AND SUTTON, S. B. Computational predictability of time-dependent natural convection flows in enclosures (including a benchmark solution). *International Journal for Numerical Methods in Fluids 40*, 8 (2002), 953–980.

[8] DAHMEN, W. AND REUSKEN, A. *Numerik für Ingenieure und Naturwissenschaftler*, 2. ed. Springer Verlag, Berlin, 2008.

[9] DARWISH, M., SRAJ, I. AND MOUKALLED, F. A coupled finite volume solver for the solution of incompressible flows on unstructured grids. *Journal of Computational Physics 228*, 1 (2009), 180 – 201.

[10] FALK, U. AND SCHÄFER, M. A fully coupled finite volume solver for the solution of incompressible flows on locally refined non-matching block-structured grids. In *Adaptive Modeling and Simulation 2013* (Barcelona, Spain, June 2013), J. P. M. de Almeida, P. Diez, C. Tiago and N. Parez, Eds., pp. 235–246.

[11] FERZIGER, J. H. AND PERIĆ, M. *Numerische Strömungsmechanik*. Springer Verlag, Berlin, 2002.

[12] GALPIN, P. F. AND RAITHBY, G. D. Numerical solution of problems in incompressible fluid flow: Treatment of the temperature-velocity coupling. *Numerical Heat Transfer 10*, 2 (1986), 105–129.

[13] GALPIN, P. F. AND RAITHBY, G. D. Numerical solution of problems in incompressible fluid flow: Treatment of the temperature-velocity coupling. *Numerical Heat Transfer 10*, 2 (1986), 105–129.

[14] GRESHO, P. M. AND SANI, R. L. On pressure boundary conditions for the incompressible navier-stokes equations. *International Journal for Numerical Methods in Fluids 7*, 10 (1987), 1111–1145.

[15] HAGER, G. AND WELLEIN, G. *Introduction to High Performance Computing for Scientists and Engineers*. CRC, Boca Raton, 2011.

[16] HUI, W. Exact solutions of the unsteady two-dimensional navier-stokes equations. *Journal of Applied Mathematics and Physics ZAMP 38*, 5 (1987), 689–702.

[17] JASAK, H. *Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows*. PhD thesis, Imperial College of Science, Technology and Medicine, Jun 1996.

[18] KUNDU, P. K., COHEN, I. M. AND DOWNLING, D. R. *Fluid Mechanics*, 5 ed. Elsevier, 2012.

[19] LI, W., YU, B., WANG, X.-R. AND SUN, S.-Y. Calculation of cell face velocity of non-staggered grid system. *Applied Mathematics and Mechanics 33*, 8 (2012), 991–1000.

[20] MAJUMDAR, S. Role of underrelaxation in momentum interpolation for calculation of flow with nonstaggered grids. *Numerical Heat Transfer 13*, 1 (1988), 125–132.

[21] Meister, A. *Numerik linearer Gleichungssysteme*, 4. ed. Vieweg+Teubner, Wiesbaden, 2011.

[22] Miller, T. F. and Schmidt, F. W. Use of a pressure-weighted interpolation method for the solution of the incompressible navier-stokes equations on a nonstaggered grid system. *Numerical Heat Transfer 14*, 2 (1988), 213–233.

[23] Patankar, S. and Spalding, D. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer 15*, 10 (1972), 1787 – 1806.

[24] Perić, M. Analysis of pressure-velocity coupling on nonorthogonal grids. *Numerical Heat Transfer 17* (Jan. 1990), 63–82.

[25] Ramamurti, R. and Löhner, R. A parallel implicit incompressible flow solver using unstructured meshes. *Computers & Fluids 25*, 2 (1996), 119 – 132.

[26] Rhie, C. M. and Chow, W. L. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal 21* (Nov. 1983), 1525–1532.

[27] Salari, K. and Knupp, P. Code verification by the method of manufactured solutions. Tech. Rep. SAND2000-1444, Sandia National Labs., Albuquerque, NM (US); Sandia National Labs., Livermore, CA (US), Jun 2000.

[28] Schäfer, M. *Numerik im Maschinenbau*. Springer Verlag, Berlin, 1999.

[29] Schäfer, M. and Turek, S. Recent benchmark computations of laminar flow around a cylinder, 1996.

[30] Sheu, T. W. H. and Lin, R. K. Newton linearization of the incompressible navier–stokes equations. *International Journal for Numerical Methods in Fluids 44*, 3 (2004), 297–312.

[31] Spurk, J. H. and Aksel, N. *Strömungslehre: Einführung in die Theorie der Strömungen*, 8. ed. Sringer Verlag, Berlin, 2010.

[32] Vakilipour, S. and Ormiston, S. J. A coupled pressure-based co-located finite-volume solution method for natural-convection flows. *Numerical Heat Transfer, Part B: Fundamentals 61*, 2 (2012), 91–115.

[33] Van Doormaal, J. P. and Raithby, G. D. Enhancements of the simple method for predicting incompressible fluid flows. *Numerical Heat Transfer 7*, 2 (1984), 147–163.

[34] Zhang, S., Zhao, X. and Bayyuk, S. Generalized formulations for the rhie–chow interpolation. *Journal of Computational Physics 258*, 0 (2014), 880 – 914.