FDY Annual Report 2014

Notes on the implementation of the viscous terms in CNS: Code Verification

Stephan Krämer-Eis

Abstract: The implementation of the viscous terms in *BoSSS* is presented. Code verification with different steady and unsteady Manufactured Solutions (MMS) in 1D and 2D is done. Additionally, the viscous fluxes are verified with an exact similarity solution of the compressible Navier-Stokes equations (CNS), which involves supersonic, transonic and subsonic regions. Furthermore, this solution consists of a non-linear temperature-depending variation of the viscosity. With a Galilean coordinate transformation the similarity solution is extended to an unsteady test case. Convergence rates of order $\mathrm{p_{DG}}+1$ for all tests shows the correct implementation of the viscous fluxes in *BoSSS*.

1 Dimensionless compressible Navier-Stokes equation

Equation (1) gives the compressible Navier-Stokes equations (CNS) in conservative form with dimensional quantities. The derivation of these equations can be found in many text books, cf. Anderson (2011), Feistauer et al. (2003) or Chung (2002). They read as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_i^c(\mathbf{U})}{\partial x_i} - \frac{\partial \mathbf{F}_i^v(\mathbf{U}, \nabla \mathbf{U}))}{\partial x_i} = \mathbf{B}, \tag{1}$$

where $\mathbf{U}$ are the conserved flow variables, $\mathbf{F}_x^c$ and $\mathbf{F}_x^v$ are the convective and viscous fluxes and $\mathbf{B}$ are source terms, respectively

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho v_j \\ \rho E \end{pmatrix}, \quad \mathbf{F}_i^c = \begin{pmatrix} \rho v_i \\ \rho v_i v_j + p\delta_{ij} \\ v_i\,(\rho E + p) \end{pmatrix}, \quad \mathbf{F}_i^v = \begin{pmatrix} 0 \\ \tau_{ij} \\ \tau_{ij} v_j + q_i \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ \rho F_j \\ \rho F_j v_j + Q_i \end{pmatrix}. \tag{2}$$

ρ is the density per unit mass, v_i are the components of the velocity vector, p is the pressure, F_j are body forces, Q_i are heat sources and ρE the total energy per volume. The total energy is the sum of the inner energy ρe and the kinetic energy of the motion of the fluid. The viscous stress tensor τ_{ij} is given by

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

with the dynamic viscosity μ. The heat flux q_i is modeled using Fourier's Law

$$q_i = k \frac{\partial T}{\partial x_i}, \tag{4}$$

where k is the thermal conductivity coefficient and T the temperature. To close the system of equations (1) a suitable Equation of State (EOS) is needed, which links the inner Energy ρe to the pressure p. In the following, the ideal gas law is used:

$$p = (\gamma - 1)\rho e. \tag{5}$$

The specific heat ratio is set to $\gamma = 1.4$.
In computational fluid dynamics (CFD) it is convenient to use non dimensional equations, e.g to transfer results from experiments or simulation on small scale to flows on larger scales.

1.1 Reference quantities and operators

In order to derive the dimensionless form of the CNS, some *reference quantities* need to be introduced: A reference length L_∞, a reference velocity V_∞, a reference density ρ_∞, a reference volume force (e.g gravitational constant) g_∞, a reference viscosity μ_∞, a reference thermal conductivity coefficient k_∞ and the gas constant R. All other reference quantities can be derived with these basic ones: Time t_∞ with L/V_∞, $\rho_\infty V_\infty^2$ for both pressure p and energy ρE, V_∞^3/L for heat sources Q and V_∞^2/R for the temperature T. Dimensionless quantities are denoted with an asterix $()^*$:

$$\begin{aligned}
&t^* = \frac{V_\infty}{L_\infty} \cdot t, \quad x_i^* = \frac{1}{L} \cdot x_i, \quad v_i^* = \frac{1}{V_\infty} \cdot v_i, \quad \rho^* = \frac{1}{\rho_\infty} \cdot \rho, \quad p^* = \frac{1}{\rho_\infty V_\infty^2} p \\
&\mu = \frac{1}{\mu_\infty} \cdot \mu, \quad k^* = \frac{1}{k_\infty} \cdot k, \quad T^* = \frac{R}{V_\infty^2} \cdot T, \quad F_j^* = \frac{1}{g_\infty} \cdot F_j, \quad \rho E^* = \frac{1}{\rho_\infty V_\infty^2} \cdot \rho E, \\
&Q_i^* = \frac{L}{V_\infty^3} \cdot Q.
\end{aligned} \tag{6}$$

The dimensionless operators can be written as

$$\begin{aligned}
\frac{\partial}{\partial t} &= \frac{\partial t^*}{\partial t} \frac{\partial}{\partial t^*} = \frac{V_\infty}{L} \frac{\partial}{\partial t^*} \\
\frac{\partial}{\partial x_i} &= \frac{\partial x_i^*}{\partial x_i} \frac{\partial}{\partial x_i^*} = \frac{1}{L} \frac{\partial}{\partial t^*} \\
\nabla &= \frac{1}{L} \nabla^*.
\end{aligned} \tag{7}$$

1.2 Dimensionless equations

Inserting the dimensionless quantities (6) and operators (7) into the equations gives the dimensionless equations.

1.2.1 Continuity equation

The continuity equation remains the same with the dimensionless quantities, i.e

$$\begin{aligned}
& \frac{\partial \rho}{\partial t} + \frac{\partial \rho v_i}{\partial x_i} = 0 \\
\Rightarrow \quad & \frac{V_\infty \rho_\infty}{L} \frac{\partial \rho^*}{\partial t^*} + \frac{V_\infty \rho_\infty}{L} \frac{\partial \rho^* v_i^*}{\partial x_i} = 0 \\
\Rightarrow \quad & \frac{\partial \rho}{\partial t^*} + \frac{\partial \rho v_i^*}{\partial x_i^+} = 0
\end{aligned} \tag{8}$$

1.2.2 Momentum equation

The dimensional momentum equation is given as

$$\frac{\partial \rho v_i}{\partial t} + \frac{\partial \rho v_i v_j}{\partial x_i} + \frac{\partial p}{\partial x_j} - \frac{\partial \tau_{ij}}{\partial x_i} = \rho F_j. \tag{9}$$

The viscous stress tensor in dimensionless quantities reads as

$$\begin{aligned} \tau_{ij} &= \mu \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) - \frac{2}{3} \mu \frac{\partial v_k}{\partial x_k} \delta_{ij} \\ \Rightarrow \quad \widetilde{\tau}_{ij} &= \frac{\mu_\infty V_\infty}{L} \underbrace{\left(\mu^* \left[\left(\frac{\partial v_i^*}{\partial x_j^*} + \frac{\partial v_j^*}{\partial x_i^*} \right) - \frac{2}{3} \frac{\partial v_k^*}{\partial x_k^*} \delta_{ij} \right] \right)}_{\tau_{ij}^*} \end{aligned} \tag{10}$$

Plugging the dimensionless quantities (6) and equation (10) into the momentum equation (9) leads to

$$\begin{aligned} \frac{\rho_\infty V_\infty^2}{L} \frac{\partial \rho^* v_i^*}{\partial t^*} + \frac{\rho_\infty V_\infty^2}{L} \frac{\partial \rho^* v_i^* v_j^*}{\partial x_i^*} + \frac{\rho_\infty V_\infty^2}{L} \frac{\partial p^*}{\partial x_i^*} - \frac{\mu_\infty V_\infty}{L^2} \frac{\partial \tau_{ij}^*}{\partial x_i^*} &= \rho_\infty g \rho^* F_j^*, \\ \Rightarrow \quad \frac{\partial \rho^* v_i^*}{\partial t^*} + \frac{\partial \rho^* v_i^* v_j^*}{\partial x_i^*} + \frac{\partial p^*}{\partial x_i^*} - \frac{1}{\mathrm{Re}} \frac{\partial \tau_{ij}^*}{\partial x_i^*} &= \frac{1}{\mathrm{Fr}^2} \rho^* F_j^*, \end{aligned} \tag{11}$$

with the Reynolds number

$$\mathrm{Re} = \frac{\rho_\infty V_\infty L}{\mu_\infty} \propto \frac{\text{inertia forces}}{\text{viscous forces}} \tag{12}$$

and the Froude Number

$$\mathrm{Fr} = \frac{V_\infty}{\sqrt{gL}} \propto \frac{\text{body inertia}}{\text{gravitational forces}}. \tag{13}$$

1.2.3 Energy equation

The energy equation is given as

$$\frac{\partial \rho E}{\partial t} + \frac{\partial \rho E v_i}{\partial x_i} + \frac{\partial p v_i}{\partial x_i} = \frac{\partial \tau_{ij} v_j}{\partial x_i} + \frac{\partial}{\partial x_i} \left(k \frac{\partial T}{\partial x_i} \right) + \rho F_j v_i + Q_j. \tag{14}$$

Again, plugging in the dimensionless quantities (6) and using equation (10) gives the dimensionless energy equation:

$$\begin{aligned} \frac{\rho_\infty V_\infty^3}{L} \frac{\partial \rho^* E^*}{\partial t^*} + \frac{\rho_\infty V_\infty^3}{L} \frac{\partial \rho^* E^* v_i^*}{\partial x_i^*} + \frac{\rho_\infty V_\infty^3}{L} \frac{\partial p^* v_i^*}{\partial x_i^*} &= \frac{\mu_\infty V_\infty^2}{L^2} \frac{\partial \tau_{ij}^* v_i^*}{\partial x_i^*} + \frac{k_\infty V_\infty^2}{L^2 R} \left(\frac{\partial}{\partial x_i^*} k^* \frac{\partial T^*}{\partial x_i^*} \right) \\ &\quad + \rho_\infty V_\infty g \rho^* F_j^* + \frac{\rho_\infty V_\infty^3}{L} Q_i^* \\ \Rightarrow \quad \frac{\partial \rho^* E^*}{\partial t^*} + \frac{\partial \rho^* E^* v_i^*}{\partial x_i^*} + \frac{\partial p^* v_i^*}{\partial x_i^*} &= \frac{1}{\mathrm{Re}} \frac{\partial \tau_{ij}^* v_i^*}{\partial x_i^*} + \frac{\gamma}{\mathrm{RePr}(\gamma - 1)} \left(\frac{\partial}{\partial x_i^*} k^* \frac{\partial T^*}{\partial x_i^*} \right) \\ &\quad + \frac{1}{\mathrm{Fr}^2} \rho^* F_j^* + Q_i^*, \end{aligned} \tag{15}$$

with the Prandtl number

$$\mathrm{Pr} = \frac{\mu_\infty c_p}{k_\infty} \propto \frac{\text{viscous diffuison rate}}{\text{thermal diffusion rate}}, \tag{16}$$

where the following relations for heat capacity at constant volume c_v and heat capacity at constant pressure c_p, which are constant, hold:

$$\gamma = \frac{c_p}{c_v} \quad \text{and} \quad R = c_p - c_v, \text{ i.e. } c_p = \frac{\gamma R}{\gamma - 1}. \tag{17}$$

1.2.4 Ideal gas law

To close the system of equations (continuity, momentum and energy) an EOS is needed. As mentioned before, the ideal gas law is used in the following. The non-dimensional ideal gas law reads as

$$\begin{aligned} p &= \rho R T \\ aufderBhne \rho_\infty V_\infty p^* &= \rho_\infty \rho^* \frac{V_\infty}{R} T^* \\ aufderBhne p^* &= \rho^* T^* \end{aligned} \tag{18}$$

with the caloric equation of state

$$\begin{aligned} e &= c_v T \\ aufderBhne V_\infty^2 e^* &= c_v \frac{R}{V_\infty^2} T^* \\ aufderBhne e^* &= \frac{1}{\gamma - 1} T^*, \end{aligned} \tag{19}$$

which yields

$$\begin{aligned} p^* &= \rho^* (\gamma - 1) e^* \\ &= (\gamma - 1) \left(\rho E^* - \frac{1}{2} \rho^* \mathbf{v}^{*2} \right) \end{aligned} \tag{20}$$

1.2.5 Complete system

In the following only dimensionless quantities are used, hence the asterix is dropped for simplicity. The complete system is then given as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_i^c(\mathbf{U})}{\partial x_i} - \frac{\partial \mathbf{F}_i^v(\mathbf{U}, \nabla \mathbf{U}))}{\partial x_i} = \mathbf{B}, \tag{21}$$

with the corresponding dimensionless fluxes

$$\mathbf{F}_i^c = \begin{pmatrix} \rho v_i \\ \rho v_i v_j + p \delta_{ij} \\ v_i (\rho E + p) \end{pmatrix}, \quad \mathbf{F}_i^v = \frac{1}{\mathrm{Re}} \begin{pmatrix} 0 \\ \tau_{ij} \\ \tau_{ij} v_j + \frac{\gamma}{\mathrm{Pr}(\gamma - 1)} q_i \end{pmatrix}, \quad \mathbf{B} = \frac{1}{\mathrm{Fr}^2} \begin{pmatrix} 0 \\ \rho F_j \\ \rho F_j v_j \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ Q_j \end{pmatrix}. \tag{22}$$

2 DG discretization of the viscous terms

In this section the discretization of the viscous terms for the compressible Navier-Stokes is given. The formulation is based on the derivation by Hartmann (2008). For programming purpose the symbolic and index notation is presented. Here, the viscous fluxes are rewritten with the homogeneity tensor $G(\mathbf{u})$. This is possible because viscous fluxes are linear in $\nabla\mathbf{u}$, i.e $\mathbf{F}_k^v(\mathbf{u},\nabla\mathbf{u}) = G(\mathbf{u})_{kl}\partial\mathbf{u}/\partial x_l$:

$$\begin{aligned} -\nabla\cdot(G(\mathbf{u})\nabla\mathbf{u}) &= 0 \quad \text{in} \quad \Omega \\ -\frac{\partial}{\partial x_k}\left(\left(G\left(\mathbf{u}\right)_{kl}\right)_{ij}\frac{\partial u_j}{\partial x_l}\right) &= 0. \end{aligned} \tag{23}$$

Rewriting as first order system

$$\begin{aligned} \underline{\sigma} &= G(\mathbf{u})\nabla\mathbf{u}, & -\nabla\cdot\underline{\sigma} &= 0 \\ \sigma_{ik} &= \left(G\left(\mathbf{u}\right)_{kl}\right)_{ij}\frac{\partial u_j}{\partial x_l}, & -\frac{\partial}{\partial x_k}\sigma_{ik} &= 0. \end{aligned} \tag{24}$$

Multiplying both equations by test functions $\underline{\tau}$ and $\mathbf{v}$, integrating on an element $\kappa \in \mathcal{T}_h$ and integrating by parts

$$\begin{aligned} \int_\kappa \underline{\sigma} : \underline{\tau}\,\mathrm{d}\mathbf{x} &= \int_\kappa G(\mathbf{u})\nabla\mathbf{u} : \underline{\tau}\,\mathrm{d}\mathbf{x} \\ &= -\int_\kappa \mathbf{u}\nabla\cdot\left(G^\mathsf{T}\left(\mathbf{u}\right)\underline{\tau}\right)\,\mathrm{d}\mathbf{x} + \int_{\partial\kappa}\mathbf{u}\left(G^\mathsf{T}\left(\mathbf{u}\right)\underline{\tau}\right)\cdot\mathbf{n}\,\mathrm{d}s \\ \int_\kappa \sigma_{ik}\tau_{ik}\,\mathrm{d}\mathbf{x} &= \int_\kappa \left(G\left(\mathbf{u}\right)_{kl}\right)_{ij}\frac{\partial u_j}{\partial x_l}\tau_{jk}\,\mathrm{d}\mathbf{x} = \int_\kappa \frac{\partial u_j}{\partial x_l}\left(G\left(\mathbf{u}\right)_{kl}\right)_{ij}\tau_{jk}\,\mathrm{d}\mathbf{x} \\ &= -\int_\kappa u_j\frac{\partial}{\partial x_l}\left(G\left(\mathbf{u}\right)_{kl}\right)_{ij}\tau_{jk}\,\mathrm{d}\mathbf{x} + \int_{\partial\kappa}u_j\left(G\left(\mathbf{u}\right)_{kl}\right)_{ij}\tau_{jk}n_l\,\mathrm{d}s \end{aligned} \tag{25}$$

and

$$\begin{aligned} \int_\kappa -\nabla\cdot\underline{\sigma} : \mathbf{v}\,\mathrm{d}\mathbf{x} = 0 \quad \Rightarrow \quad \int_\kappa \underline{\sigma} : \nabla\mathbf{v}\,\mathrm{d}\mathbf{x} &= \int_{\partial\kappa}\underline{\sigma}\cdot\mathbf{n}\mathbf{v}\,\mathrm{d}s \\ \int_\kappa \sigma_{ik}\frac{\partial v_i}{\partial x_k}\mathrm{d}\mathbf{x} &= \int_{\partial\kappa}\sigma_{ik}n_k v_i\,\mathrm{d}s. \end{aligned} \tag{26}$$

Summing over all elements and replacing $\mathbf{u},\mathbf{v},\underline{\sigma}$ and $\underline{\tau}$ by discrete functions, the so-called *flux formulation* is obtained:

$$\begin{aligned} \int_\Omega \underline{\sigma}^h : \underline{\tau}^h\,\mathrm{d}\mathbf{x} &= -\int_\Omega \mathbf{u}^h\nabla^h\cdot\left(G^\mathsf{T}\left(\mathbf{u}^h\right)\underline{\tau}^h\right)\,\mathrm{d}\mathbf{x} + \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\hat{\mathbf{u}}^h\left(G^\mathsf{T}\left(\mathbf{u}^h\right)\underline{\tau}^h\right)\cdot\mathbf{n}\,\mathrm{d}s \\ &= -\int_\Omega u_j^h\frac{\partial}{\partial x_l}\left(G\left(\mathbf{u}^h\right)_{kl}\right)_{ij}\tau_{jk}^h\,\mathrm{d}\mathbf{x} + \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\hat{u}_j^h\left(G\left(\mathbf{u}^h\right)_{kl}\right)_{ij}\tau_{jk}^h n_l\,\mathrm{d}s \end{aligned} \tag{27}$$

and

$$\begin{aligned} \int_\Omega \underline{\sigma}^h : \nabla^h\mathbf{v}^h\,\mathrm{d}\mathbf{x} &= \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\hat{\underline{\sigma}}^h\cdot\mathbf{n}\mathbf{v}^h\,\mathrm{d}s \\ \int_\Omega \sigma_{ik}^h\frac{\partial v_i^h}{\partial x_k}\,\mathrm{d}\mathbf{x} &= \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\hat{\sigma}_{ik}^h n_k v_i^h\,\mathrm{d}s. \end{aligned} \tag{28}$$

Here, the *numerical fluxes* $\hat{\mathbf{u}}^h$ and $\hat{\underline{\sigma}}^h$ are approximations to $\mathbf{u}$ and $\underline{\sigma} = \nabla\mathbf{u}$. The stability and accuracy of the DG scheme depends on the particular choice of the numerical fluxes. The *flux formulation* represents a discretization of a first order system with the unknowns $\mathbf{u}^h \in \mathbf{V}_p^{h,d}$ (vector-valued discrete function space) and $\underline{\sigma}^h \in \underline{\Sigma}_p^{h,d}$ (tensor-valued discrete function space). By rewriting the viscous fluxes in a first order system the problem size increased by a factor of $(d+1)$. To reduce this, the auxiliary variable $\underline{\sigma}^h$ is eliminated in (27) and (28), which gives the *primal formulation* involving only the primal variable $\mathbf{u}^h$. It is realized by a second integration by parts of (27) and setting $\tau^h = \nabla^h\mathbf{v}^h$:

$$\int_\Omega \underline{\sigma}^h : \nabla^h\mathbf{v}^h \,\mathrm{d}\mathbf{x} = \int_\Omega G\left(\mathbf{u}^h\right)\nabla^h\mathbf{u}^h : \nabla^h\mathbf{v}^h\mathrm{d}\mathbf{x} + \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\left(\hat{\mathbf{u}}^h - \mathbf{u}^h\right)\left(G^\mathsf{T}\left(\mathbf{u}^h\right)\nabla^h\mathbf{v}^h\right)\cdot\mathbf{n}\,\mathrm{d}s$$

$$\int_\Omega \sigma_{ik}^h\frac{\partial v_i^h}{\partial x_k} = \int_\Omega \frac{\partial u_j^h}{\partial x_l}\left(G\left(\mathbf{u}^h\right)_{kl}\right)_{ij}\frac{\partial v_i^h}{\partial x_k} + \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\left(\hat{u}_j^h - u_j^h\right)\left(G\left(\mathbf{u}^h\right)_{kl}\right)_{ij}\frac{\partial v_i^h}{\partial x_k}n_l\,\mathrm{d}s \tag{29}$$

Substituting (29) into (28) yields to the following problem: Find $\mathbf{u}^h \in \mathbf{V}_p^{h,d}$ such that

$$\hat{N}^h(\mathbf{u}^h, \mathbf{v}^h) = 0 \quad \forall\mathbf{v}^h \in \mathbf{V}_p^{h,d}, \tag{30}$$

where $\hat{N}^h(\cdot,\cdot)\colon \left[H^1\left(\mathcal{T}_h\right)\right]^m \times \left[H^1\left(\mathcal{T}_h\right)\right]^m \to \mathbb{R}$ is the semilinear form. It is nonlinear in its first and linear in its second argument and defined as:

$$\begin{aligned}\hat{N}^h(\cdot,\cdot) &= \int_\Omega G\left(\mathbf{u}^h\right)\nabla^h\mathbf{u}^h : \nabla^h\mathbf{v}^h\mathrm{d}\mathbf{x} - \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\hat{\underline{\sigma}}^h\cdot\mathbf{n}\mathbf{v}^h\,\mathrm{d}s \\ &\quad + \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\left(\hat{\mathbf{u}}^h - \mathbf{u}^h\right)\left(G^\mathsf{T}\left(\mathbf{u}^h\right)\nabla^h\mathbf{v}^h\right)\cdot\mathbf{n}\,\mathrm{d}s \\ &= \int_\Omega \frac{\partial u_j^h}{\partial x_l}\left(G\left(\mathbf{u}^h\right)_{kl}\right)_{ij}\frac{\partial v_i^h}{\partial x_k} - \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\hat{\sigma}_{ik}^h n_k v_i^h\,\mathrm{d}s \\ &\quad + \sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}\left(\hat{u}_j^h - u_j^h\right)\left(G\left(\mathbf{u}^h\right)_{kl}\right)_{ij}\frac{\partial v_i^h}{\partial x_k}n_l\,\mathrm{d}s\end{aligned} \tag{31}$$

Equation (30) is called the *primal formulation* of the problem and (31) is the *primal form*. So far, $\hat{N}^h(\cdot,\cdot)$ is written in *cell-based* formulation, where each interior face occurs twice in the summation over all elements ($\sum_{\kappa\in\mathcal{T}_h}\int_{\partial\kappa}$). In the following, this cell-based primal form is transfered into a *face-based* form, where each interior face is only treated once. For that reason, the definition of a *mean value* ($\{\!\{\cdot\}\!\}$) and *jump operator* ($[\![\cdot]\!]$) are introduced.

Mean value and jump operator

Let $e \in \Gamma_\mathcal{I}$ be an interior edge with the unit outward normal vectors $\mathbf{n}^+, \mathbf{n}^- \in \mathbb{R}^d$ and on the boundary $e \in \Gamma$ with $\mathbf{n}^+ \in \mathbb{R}^d$. Let $\mathbf{v} \in \left[T(\mathcal{T}_h)\right]^d$ and $\underline{\tau} \in \left[T(\mathcal{T}_h)\right]^{d\times m}$ be the traces of a vector valued and tensor valued function, respectively. The mean value and jump operator defined, as follows

$$\begin{aligned}\{\!\{\mathbf{v}\}\!\} &= \frac{1}{2}(\mathbf{v}^+ + \mathbf{v}^-) && \mathit{on}\ \Gamma_\mathcal{I}, \quad \{\!\{\mathbf{v}\}\!\} = \mathbf{v}^+ && \mathit{on}\ \Gamma \\ \{\!\{\underline{\tau}\}\!\} &= \frac{1}{2}\left(\underline{\tau}^+ + \underline{\tau}^-\right) && \mathit{on}\ \Gamma_\mathcal{I}, \quad \{\!\{\underline{\tau}\}\!\} = \underline{\tau}^+ && \mathit{on}\ \Gamma \\ [\![\underline{\tau}]\!] &= \underline{\tau}^+\mathbf{n}^+ + \underline{\tau}^-\mathbf{n}^- && \mathit{on}\ \Gamma_\mathcal{I}, \quad [\![\underline{\tau}]\!] = \underline{\tau}^+\mathbf{n}^+ && \mathit{on}\ \Gamma \\ \underline{[\![\mathbf{v}]\!]} &= \underline{\mathbf{v}}^+ \otimes \mathbf{n}^+ + \underline{\mathbf{v}}^- \otimes \mathbf{n}^- && \mathit{on}\ \Gamma_\mathcal{I}, \quad \underline{[\![\mathbf{v}]\!]} = \underline{\mathbf{v}}^+ \otimes \mathbf{n}^+ && \mathit{on}\ \Gamma.\end{aligned} \tag{32}$$

For transferring between cell-based and face-based integrals, the following equations holds:

$$\begin{aligned} \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa} \left(\underline{\tau}^{+}\mathbf{n}^{+}\right) \cdot \mathbf{v}^{+} ds &= \int_{\Gamma_{\mathcal{I}}} \{\!\{\underline{\tau}\}\!\} : \underline{\llbracket \mathbf{v} \rrbracket} ds + \int_{\Gamma_{\mathcal{I}}} \llbracket \underline{\tau} \rrbracket \cdot \{\!\{\mathbf{v}\}\!\} ds \\ \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa} \tau_{ik}^{x} n_k^{+} v_i^{+} ds &= \int_{\Gamma_{\mathcal{I}}} \{\!\{\tau_{ik}\}\!\} \left(v_i^{+} n_k^{+} + v_i^{-} n_k^{-}\right) ds + \int_{\Gamma_{\mathcal{I}}} \left(\tau_{ik}^{+} n_k^{+} + \tau_{ik}^{-} n_k^{-}\right) \{\!\{v_i\}\!\} ds \end{aligned} \tag{33}$$

Now the primal form (31) can be rewritten in the face based form by using equation (33) twice (for $\underline{\tau} = \hat{\underline{\sigma}}^h$ and $\mathbf{v} = \mathbf{v}^h$, and for $\underline{\tau} = G^{\mathsf{T}}(\mathbf{u}^h)\nabla^h \mathbf{v}^h$, i.e. $\tau_{jl} = \left(G\left(u\right)_{kl}\right)_{ij} \partial_{x_k} v_i$ and $\mathbf{v} = \hat{\mathbf{u}}^h - \mathbf{u}^h$)

$$\begin{aligned} \hat{N}^h(\cdot,\cdot) = &\int_{\Omega} G(\mathbf{u}^h)\nabla^h \mathbf{u}^h : \nabla^h \mathbf{v}^h d\mathbf{x} \\ &+ \int_{\Gamma_{\mathcal{I}} \cup \Gamma} \underline{\llbracket \hat{\mathbf{u}}^h - \mathbf{u}^h \rrbracket} : \{\!\{G^{\mathsf{T}}(\mathbf{u}^h)\nabla^h \mathbf{v}^h\}\!\} - \{\!\{\hat{\underline{\sigma}}^h\}\!\} : \underline{\llbracket \mathbf{v}^h \rrbracket}\, ds \\ &+ \int_{\Gamma_{\mathcal{I}}} \{\!\{\hat{\mathbf{u}}^h - \mathbf{u}^h\}\!\} \cdot \llbracket G^{\mathsf{T}}(\mathbf{u}^h)\nabla^h \mathbf{v}^h \rrbracket - \llbracket \hat{\underline{\sigma}}^h \rrbracket \cdot \{\!\{\mathbf{v}^h\}\!\}\, ds \\ = &\int_{\Omega} \frac{\partial u_j^h}{\partial x_l} \left(G(\mathbf{u}^h)_{kl}\right)_{ij} \frac{\partial v_i^h}{\partial x_k} \\ &+ \int_{\Gamma_{\mathcal{I}} \cup \Gamma} \left([\hat{u}_j^{h+} - u_j^{h+}]n_l^{+} + [\hat{u}_j^{h-} - u_j^{h-}]n_l^{-}\right) \{\!\{\left(G(\mathbf{u}^h)_{kl}\right)_{ij} \frac{\partial v_i^h}{\partial x_k}\}\!\} \\ &\qquad - \{\!\{\hat{\sigma}_{ik}^h\}\!\} \left(v_i^{h+} n_k^{+} + v_i^{h-} n_k^{-}\right)\, ds \\ &+ \int_{\Gamma} \{\!\{\hat{u}_j^h - u_j^h\}\!\} \left((G(\mathbf{u}^h)_{kl}^{+})_{ij} \frac{\partial v_i^{h+}}{\partial x_k} n_l^{+} + (G(\mathbf{u}^h)_{kl}^{-})_{ij} \frac{\partial v_i^{h-}}{\partial x_k} n_l^{-} \right) \\ &\qquad - \left(\hat{\sigma}_{ik}^{+} n_k^{+} + \hat{\sigma}_{ik}^{-} n_k^{-}\right) \{\!\{v_i^h\}\!\}\, ds. \end{aligned} \tag{34}$$

Until now, there are no requirements to the numerical fluxes $\hat{\mathbf{u}}^h$ and $\hat{\underline{\sigma}}^h$. Arnold et al. (2002) show that the *SIPG* flux formulation is conservative and consistent. The SIPG formulation uses the following numerical fluxes

$$\hat{\mathbf{u}}^h = \{\!\{\mathbf{u}^h\}\!\}, \qquad \hat{\underline{\sigma}}^h = \{\!\{G(\mathbf{u}^h)\nabla^h \mathbf{u}^h\}\!\} - \delta(\mathbf{u}^h), \tag{35}$$

where the *penalization term* $\delta(\mathbf{u}^h)$ is given by Hartmann & Houston (2008)

$$\delta(\mathbf{u}^h) = \delta^{\text{ip}}(\mathbf{u}^h) = C_{\text{IP}} \frac{p^2}{h_e} \{\!\{G(\mathbf{u}^h)\}\!\} \underline{\llbracket \mathbf{u}^{\mathbf{h}} \rrbracket}. \tag{36}$$

Inserting the SIPG fluxes (35) into (34) and because of

$$\begin{aligned} \underline{\llbracket \hat{\mathbf{u}} \rrbracket}^h &= \llbracket \underline{\{\!\{\mathbf{u}^h\}\!\}} \rrbracket = 0, \\ \{\!\{\hat{\mathbf{u}}^h\}\!\} &= \{\!\{\{\!\{\mathbf{u}^h\}\!\}\}\!\} = \{\!\{\mathbf{u}^h\}\!\}, \\ \{\!\{\hat{\underline{\sigma}}^h\}\!\} &= \{\!\{\{\!\{G(\mathbf{u}^h)\nabla^h \mathbf{u}^h\}\!\} - \delta(\mathbf{u}^h)\}\!\} = \{\!\{G(\mathbf{u}^h)\nabla^h \mathbf{u}^h\}\!\} - \delta(\mathbf{u}^h), \\ \llbracket \hat{\underline{\sigma}}^h \rrbracket &= \llbracket \{\!\{G(\mathbf{u}^h)\nabla^h \mathbf{u}^h\}\!\} \rrbracket - \llbracket \delta(\mathbf{u}^h) \rrbracket = 0, \end{aligned} \tag{37}$$

the face based formulation (34) reduces to

$$\begin{aligned}
N^h(\cdot,\cdot) &= \int_\Omega G(\mathbf{u}^h)\nabla^h\mathbf{u}^h : \nabla^h\mathbf{v}^h d\mathbf{x} \\
&\quad - \int_{\Gamma_\mathcal{I}} \underline{[\![\mathbf{u}^h]\!]} : \{\!\{G^\mathsf{T}(\mathbf{u}^h)\nabla^h\mathbf{v}^h\}\!\}\, ds \\
&\quad - \int_{\Gamma_\mathcal{I}} \{\!\{G(\mathbf{u}^h)\nabla^h\mathbf{u}^h\}\!\} : \underline{[\![\mathbf{v}^h]\!]}\, ds \\
&\quad + \int_{\Gamma_\mathcal{I}} \delta(\mathbf{u}^h) : \underline{[\![\mathbf{v}^h]\!]} ds + \hat{N}^h_\Gamma(\mathbf{u}^h,\mathbf{v}^h) \\
&= \int_\Omega \frac{\partial u^h_j}{\partial x_l}\left(G(\mathbf{u}^h)_{kl}\right)_{ij}\frac{\partial v^h_i}{\partial x_k} \\
&\quad - \int_{\Gamma_\mathcal{I}} \left(u^{h+}_j n^+_l + u^{h-}_j n^-_l\right)\{\!\{\left(G(\mathbf{u}^h)_{kl}\right)_{ij}\frac{\partial v^h_i}{\partial x_k}\}\!\} \\
&\quad - \int_{\Gamma_\mathcal{I}} \{\!\{(G(\mathbf{u}^h)_{kl})_{ij}\frac{\partial u_j}{\partial x_l}\}\!\}\,(v^+_i n^+_k + v^-_i n^-_k) \\
&\quad + \int_{\Gamma_\mathcal{I}} C_\mathrm{IP}\frac{p^2}{h_e}\{\!\{(G(\mathbf{u}^h)_{kl})_{ij}\}\!\}\left(u^+_j n^+_l + u^-_j n^-_l\right)(v^+_i n^+_k + v^-_i n^-_k) + \hat{N}^h_\Gamma(\mathbf{u}^h,\mathbf{v}^h),
\end{aligned} \tag{38}$$

where $\hat{N}^h_\Gamma(\mathbf{u}^h,\mathbf{v}^h)$ are the boundary terms. A detailed specification for the different boundary conditions can be found in Hartmann (2008).

2.1 Diffusion Tensor $\mathbf{G}(\mathbf{u})$

For the SIPG discretization the diffusive tensor $G(\mathbf{u})$ is needed.

$$\mathbf{F}^v(\mathbf{u},\nabla\mathbf{u}) = G(\mathbf{u})\frac{\partial\mathbf{u}}{\partial x_l} \quad \text{with} \quad G_{ikjl} = \frac{\mathbf{F}^v_{ik}(\mathbf{u},\nabla\mathbf{u})}{\partial\left(\frac{\partial u_j}{\partial x_l}\right)} \in \mathbb{R}^{5\times3\times5\times3}. \tag{39}$$

The components of the diffusive tensor $G(\mathbf{u})$ can be grouped into 9 matrices:

$$\begin{matrix}
G_{11} := G_{i1j1} & \cdots & G_{13} := G_{i1j3} \\
\vdots & \ddots & \vdots \\
G_{31} := G_{i3j1} & \cdots & G_{33} := G_{i3j3},
\end{matrix} \tag{40}$$

such that

$$\mathbf{F}^v_{i1} = G_{11}\cdot\frac{\partial\mathbf{u}}{\partial x_1} + G_{12}\cdot\frac{\partial\mathbf{u}}{\partial x_2} + G_{13}\cdot\frac{\partial\mathbf{u}}{\partial x_3}, \tag{41}$$

and analog for $\mathbf{F}^v_{i2}$, $\mathbf{F}^v_{i3}$.
The 9 matrices $\mathbf{G}_{kl}$ reads as (again $\mathbf{v} = (v_1, v_2, v_3)^\mathsf{T}$ is the velocity vector):

$$G_{11} = \frac{\mu}{\rho\mathrm{Re}}\begin{pmatrix}
0 & 0 & 0 & 0 & 0 \\
-\frac{3}{4}v_1 & \frac{3}{4} & 0 & 0 & 0 \\
-v_2 & 0 & 1 & 0 & 0 \\
-v_3 & 0 & 0 & 1 & 0 \\
-\left(\frac{4}{3}v_1^2 + v_2^2 + v_3^2 + \frac{\gamma}{\mathrm{Pr}}\left(\mathbf{E}-\mathbf{v}^2\right)\right) & \left(\frac{4}{3}-\frac{\gamma}{\mathrm{Pr}}\right)v_1 & \left(1-\frac{\gamma}{\mathrm{Pr}}\right)v_2 & \left(1-\frac{\gamma}{\mathrm{Pr}}\right)v_3 & \frac{\gamma}{\mathrm{Pr}}
\end{pmatrix},$$

$$G_{22} = \frac{\mu}{\rho \mathrm{Re}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -v_1 & 1 & 0 & 0 & 0 \\ -\frac{3}{4} v_2 & 0 & \frac{3}{4} & 0 & 0 \\ -v_3 & 0 & 0 & 1 & 0 \\ -\left(v_1^2 + \frac{4}{3} v_2^2 + v_3^2 + \frac{\gamma}{\mathrm{Pr}} \left(\mathbf{E} - \mathbf{v}^2\right)\right) & \left(1 - \frac{\gamma}{\mathrm{Pr}}\right) v_1 & \left(\frac{4}{3} - \frac{\gamma}{\mathrm{Pr}}\right) v_2 & \left(1 - \frac{\gamma}{\mathrm{Pr}}\right) v_3 & \frac{\gamma}{\mathrm{Pr}} \end{pmatrix},$$

$$G_{33} = \frac{\mu}{\rho \mathrm{Re}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -v_1 & 1 & 0 & 0 & 0 \\ -v_2 & 0 & 1 & 0 & 0 \\ -\frac{3}{4} v_3 & 0 & 0 & \frac{3}{4} & 0 \\ -\left(v_1^2 + v_2^2 + \frac{4}{3} v_3^2 + \frac{\gamma}{\mathrm{Pr}} \left(\mathbf{E} - \mathbf{v}^2\right)\right) & \left(1 - \frac{\gamma}{\mathrm{Pr}}\right) v_1 & \left(1 - \frac{\gamma}{\mathrm{Pr}}\right) v_2 & \left(\frac{4}{3} - \frac{\gamma}{\mathrm{Pr}}\right) v_3 & \frac{\gamma}{\mathrm{Pr}} \end{pmatrix},$$

$$G_{12} = \frac{\mu}{\rho \mathrm{Re}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3} v_2 & 0 & -\frac{2}{3} & 0 & 0 \\ -v_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{3} v_1 v_2 & v_2 & -\frac{2}{3} v_1 & 0 & 0 \end{pmatrix}, \quad G_{21} = \frac{\mu}{\rho \mathrm{Re}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -v_2 & 0 & 1 & 0 & 0 \\ \frac{2}{3} v_1 & -\frac{2}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{3} v_1 v_2 & -\frac{2}{3} v_2 & v_1 & 0 & 0 \end{pmatrix},$$

$$G_{13} = \frac{\mu}{\rho \mathrm{Re}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3} v_3 & 0 & 0 & -\frac{2}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -v_1 & 1 & 0 & 0 & 0 \\ -\frac{1}{3} v_1 v_3 & v_3 & 0 & -\frac{2}{3} v_1 & 0 \end{pmatrix}, \quad G_{31} = \frac{\mu}{\rho \mathrm{Re}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -v_3 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3} v_1 & -\frac{2}{3} & 0 & 0 & 0 \\ -\frac{1}{3} v_1 v_3 & -\frac{2}{3} v_3 & 0 & v_1 & 0 \end{pmatrix},$$

$$G_{23} = \frac{\mu}{\rho \mathrm{Re}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3} v_3 & 0 & 0 & -\frac{2}{3} & 0 \\ -v_2 & 0 & 1 & 0 & 0 \\ -\frac{1}{3} v_2 v_3 & 0 & v_3 & -\frac{2}{3} v_2 & 0 \end{pmatrix}, \quad G_{32} = \frac{\mu}{\rho \mathrm{Re}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -v_3 & 0 & 0 & 1 & 0 \\ \frac{2}{3} v_2 & 0 & -\frac{2}{3} & 0 & 0 \\ -\frac{1}{3} v_2 v_3 & 0 & -\frac{2}{3} v_3 & v_2 & 0 \end{pmatrix}.$$

3 Method of Manufactured Solution (MMS)

The source of errors in numerical simulations can be categorized into two categories: physical modeling errors (validation) and mathematical errors (verification). The physical errors arise through shortcomings in the chosen model or usage of a specific model outside of its intended range. Verification deals with the correctness of the numerical solution within a given model. Verification can be thought of as solving the equations right and validation as choosing the right equations and models in the first place. Roache (2002) distinguishes between the verification of calculation, which involves error *estimation* and verification of a code, which involves error *evaluation*. The verification of calculations needs to be done in every simulation and takes into account problem specific errors, e.g. grid dependency study. In many cases this error can only be estimated. The correctness of the code can only be done by a systematic discretization convergence study using a known solution or benchmark results. Typically, an analytical solution is used for that purpose. But it is not sufficient that the analytical solution is exact, it is also necessary that the solution is general enough such that all terms in the governing equations are executed. For many systems of equations, e.g compressible Navier-Stokes, a general analytical solution does not exist. In this case the Method of Manufactured Solutions (MMS) gives a procedure for generating such an analytical solution for code verification. The idea of the procedure is simply to manufacture an analytical solution without concerning about the physical realism. In fact, code verification is a purely mathematical process and can be exactly evaluated. Hence, the lack of realism does not bother at all. Once a non trivial exact solution is generated, it is plugged into the system of partial differential equations (PDEs). Normally this solution does not satisfy the equations and a residual is generated. This residual is then used as an artificial source term **S** for the original PDEs. The analytical solution can now be used for code verification by performing systematic discretization convergence tests. Salari & Knupp (2000) give some guidelines for constructing a MMS:

- MMS should be composed of smooth analytic functions, e.g polynomials, trigonometric or exponential functions. Solution smoothness is essential to ensure the theoretical order of accuracy.

- MMS should be general enough to exercise all terms, e.g x and y dependency for cross derivatives such as $\partial u^2/\partial x \partial y$

- MMS should have a sufficient number of non trivial derivatives, e.g at least second order in space for CNS.

- MMS derivatives should be bound by a small constant. It ensures that the solution is not a strongly varying function of space and/or time. It also means that the MMS should not contain singularities within the domain.

- MMS should still be physically consistent, e.g. if the code assumes a positive density, then the MMS should also have $\rho > 0$.

The MMS itself can help to debug the code by turning off certain solution terms, e.g viscous terms. However, sometimes it is useful to neglect the second guideline: By constructing a MMS which exercise only specific terms, it can help to further debug the code. Later, a MMS will be used which only takes the temperature gradient into a account.

3.1 1D MMS for viscous Fluxes

In a first test series only the newly implemented viscous fluxes are tested. The equations read

$$-\frac{\partial \mathbf{F}_x^v(\mathbf{U}, \nabla \mathbf{U}))}{\partial x} = \mathbf{S}, \tag{42}$$

where **S** is the source term coming from the corresponding MMS.
Two different MMS are constructed:

- A solution without velocity gradient which tests the correct implementation of the temperature gradient term, given by Gassner et al. (2008).

- A solution which activates all viscous terms, i.e temperature, pressure and velocity gradients, given by Malaya et al. (2013).

3.1.1 Gassner et al. (2008)

This MMS is constructed in such a way that the velocity is constant, i.e only the temperature gradient fluxes have an influence on the solution. The solution is given as

$$\mathbf{U} = \begin{pmatrix} \sin(2\pi x) + 2 \\ \sin(2\pi x) + 2 \\ (\sin(2\pi x) + 2)^2 \end{pmatrix}. \tag{43}$$

By inserting the solution (43) into equation (42) the source term **S** with the components

$$\begin{aligned} s_1 &= 0 \\ s_2 &= 0 \\ s_3 &= \sin(2\pi x)\left(\frac{(2\pi)^2 \mu\gamma}{\text{RePr}}\right) \end{aligned} \tag{44}$$

is obtained.
A convergence study is carried out with the following coefficients

$$\text{Re} = 1.0 \,\text{and}\, \text{Re} = 1000.0, \quad \text{Pr} = 0.72, \quad \gamma = 1.4, \quad \mu = 1.0. \tag{45}$$

The computational domain is set to $[0;1]$ with uniform cell sizes. The boundary condition are set to the Dirichlet values obtained from the solution (43). The simulations are carried out until a steady state is reached, i.e. $\int_\Omega |\rho E^h_{n+1} - \rho E^h_n| d\Omega < 1.0 \times 10^{-12}$.

The convergence results for the energy ρE are shown in figure 1. It can be seen, that the expected convergence rate of $\text{p}_{\text{DG}}+1$ can be achieved, where p_{DG} is the polynomial order of the DG test functions. For density and momentum the same results are obtained and hence, are not shown here. Furthermore, the same rates are achieved for the different Reynolds numbers.

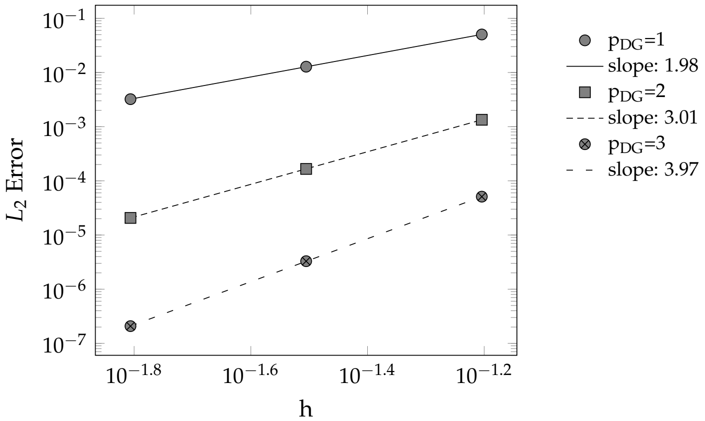


Figure 1: MMS (Gassner et al. 2008) for viscous fluxes: h-p convergence for ρE.

3.1.2 MASA Library

This MMS is given by the Manufactured Analytical Solution Abstraction (MASA) library (Malaya et al. 2013). It offers various MMS and their source terms. The main advantage is that it is a C++ library which can be integrated into many codes and it is well tested. Thus, a coding mistake in a part of the MMS can be excluded. A disadvantage is that the solutions

are designed for dimensional equations. BoSSS uses non-dimensional equations, as described in section 1. Hence, some coefficients in the source terms needs to be adapted.
For this test case the solution is given in the primary variables ρ, u and p

$$\begin{aligned} \rho &= \sin(2\pi x) + 2, \\ u &= \sin(2\pi x) + 2, \\ p &= \cos(2\pi x) + 2, \end{aligned} \tag{46}$$

which lead to the conserved variables

$$\mathbf{U} = \begin{pmatrix} \sin(2\pi x) + 2 \\ (\sin(2\pi x) + 2)^2 \\ \frac{\cos(2\pi x)+2}{\gamma-1} + \frac{1}{2}(\sin(2\pi x) + 2)^3 \end{pmatrix}. \tag{47}$$

In contrast to the previous Ansatz (43), this MMS is sufficiently general. Thus, it exercises all terms in the viscous fluxes, i.e pressure and velocity gradients exist. Consequently, the source term $\mathbf{S}$ is more complex:

$$\begin{aligned} s_1 &= 0 \\ s_2 &= \frac{4}{3}\frac{\mu}{\text{Re}} 4\pi^2 \sin(2\pi x) \\ s_3 &= -\frac{\gamma}{\gamma-1}\frac{\mu}{\text{RePr}}\frac{8\pi^2(\cos(2\pi x)+2)}{(\sin(2\pi x)+2)^3}\cos(2\pi x)^2 \\ &\quad - \frac{\gamma}{\gamma-1}\frac{\mu}{\text{RePr}}\frac{8\pi^2}{(\sin(2\pi x)+2)^2}\cos(2\pi x)\sin(2\pi x) \\ &\quad - \frac{\gamma}{\gamma-1}\frac{\mu}{\text{RePr}}\frac{4\pi^2(\cos(2\pi x)+2)}{(\sin(2\pi x)+2)^2}\sin(2\pi x) \\ &\quad + \frac{\gamma}{\gamma-1}\frac{\mu}{\text{RePr}}\frac{4\pi^2}{\sin(2\pi x)+2}\cos(2\pi x) \\ &\quad - \frac{4}{3}\frac{\mu}{\text{Re}} 16\pi^2 \cos(2\pi x)^2 \\ &\quad + \frac{4}{3}\frac{\mu}{\text{Re}} 16\pi^2 (\sin(2\pi x)+2)\sin(2\pi x). \end{aligned} \tag{48}$$

Similar to the previous convergence study a uniform mesh in the domain [0;1] and the same coefficients are chosen. On the left and right boundary Dirichlet values are used. Fig. 2 shows the result of the convergence study for the energy. Here again, the desired convergence rate of $\text{p}_{\text{DG}} + 1$ can be obtained. Both test cases indicates that for the implementation of the SIPG fluxes of the viscous terms in 1D the theoretical order of convergence is attained.

3.2 1D MMS for CNS

In this subsection the two previous solutions are extended to solve the steady compressible Navier-Stokes equations (CNS) which read

$$\frac{\partial \mathbf{F}_x^c(\mathbf{U})}{\partial x} - \frac{\partial \mathbf{F}_x^v(\mathbf{U}, \nabla \mathbf{U}))}{\partial x} = \mathbf{S}. \tag{49}$$

In the following, only the different source terms are listed, since the solutions for Gassner et al. (2008) (43) and MASA (2013) (47) remain the same. Also the computational setting is

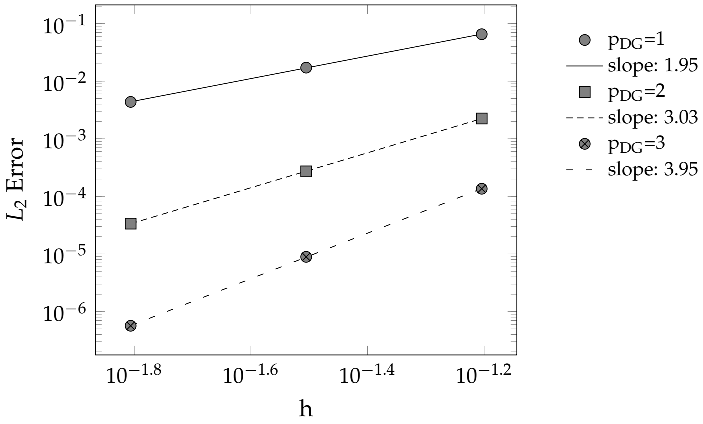


Figure 2: MMS (Malaya et al. 2013) for viscous fluxes equation: h-p convergence for ρE.

left the same, i.e. a uniform mesh in the domain [0;1], Dirichlet boundary conditions with the analytical values and the same coefficients given by (45).

3.2.1 Gassner et al. (2008)

Recalling that this MMS has no velocity gradients (43), leads also to much simpler source terms for the advection fluxes. The complete source terms for the CNS reads as

$$\begin{aligned} s_1 &= 2\pi\cos(2\pi x) \\ s_2 &= (7\gamma\pi - 5\pi)\cos(2\pi x) \\ &\quad + (\gamma - 1)\,2\pi\sin(4\pi x) \\ s_3 &= (7\gamma\pi + \pi)\cos(2\pi x) \\ &\quad + 2\gamma\pi\sin(4\pi x) \\ &\quad + \sin(2\pi x)\left(\frac{(2\pi)^2\mu\gamma}{\mathrm{RePr}}\right). \end{aligned} \tag{50}$$

In Fig. 3 the results of the h-p convergence study is shown. Here again, the expected rates are achieved.

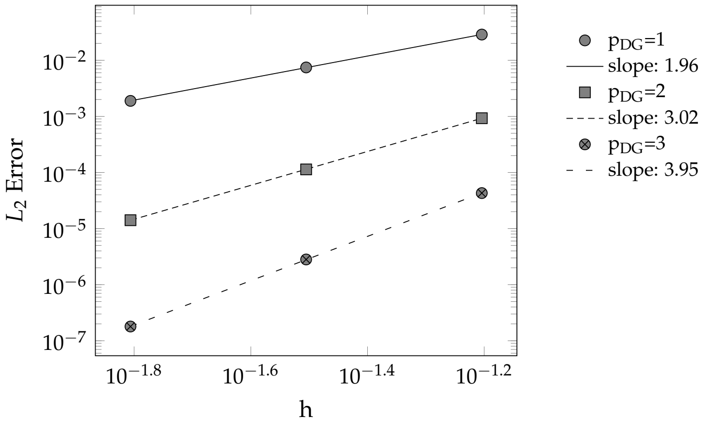


Figure 3: MMS (Gassner et al. 2008) for CNS: h-p convergence for ρE.

3.2.2 MASA Library

The source term **S** for the CNS of the MASA Ansatz (46) has the following components:

$$\begin{aligned}
s_1 &= 4\pi\left(\sin(2\pi x)+2\right)\cos(2\pi x)\\
s_2 &= 2\pi\left(\sin(2\pi x)+2\right)^2\cos(2\pi x)\\
&\quad - 2\pi\sin(2\pi x)\\
&\quad + 4\pi\left(\sin(2\pi x)+2\right)^2\cos(2\pi x)\\
&\quad + \frac{4}{3}\frac{\mu}{\mathrm{Re}}4\pi^2\sin(2\pi x)\\
s_3 &= \pi\left(\sin(2\pi x)+2\right)^3\cos(2\pi x)\\
&\quad - \frac{\gamma}{\gamma-1}2\pi\left(\sin(2\pi x)+2\right)\sin(2\pi x)\\
&\quad + \frac{\gamma}{\gamma-1}2\pi\left(\cos(2\pi x)+2\right)\cos(2\pi x)\\
&\quad + 3\pi\left(\sin(2\pi x)+2\right)^3\cos(2\pi x)\\
&\quad - \frac{\gamma}{\gamma-1}\frac{\mu}{\mathrm{RePr}}\frac{8\pi^2(\cos(2\pi x)+2)}{(\sin(2\pi x)+2)^3}\cos(2\pi x)^2\\
&\quad - \frac{\gamma}{\gamma-1}\frac{\mu}{\mathrm{RePr}}\frac{8\pi^2}{(\sin(2\pi x)+2)^2}\cos(2\pi x)\sin(2\pi x)\\
&\quad - \frac{\gamma}{\gamma-1}\frac{\mu}{\mathrm{RePr}}\frac{4\pi^2(\cos(2\pi x)+2)}{(\sin(2\pi x)+2)^2}\sin(2\pi x)\\
&\quad + \frac{\gamma}{\gamma-1}\frac{\mu}{\mathrm{RePr}}\frac{4\pi^2}{\sin(2\pi x)+2}\cos(2\pi x)\\
&\quad - \frac{4}{3}\frac{\mu}{\mathrm{Re}}16\pi^2\cos(2\pi x)^2\\
&\quad + \frac{4}{3}\frac{\mu}{\mathrm{Re}}16\pi^2(\sin(2\pi x)+2)\sin(2\pi x).
\end{aligned} \tag{51}$$

Fig. 4 shows the h-p convergence for the total energy with the corresponding convergence rates. This is the most difficult MMS test case and it can be concluded that the implementation of the SIPG flux in 1D is correct. So far, the different MMS take only into account constant viscosity (different values are successfully tested, see (45)). A non-linear variation of the vis-

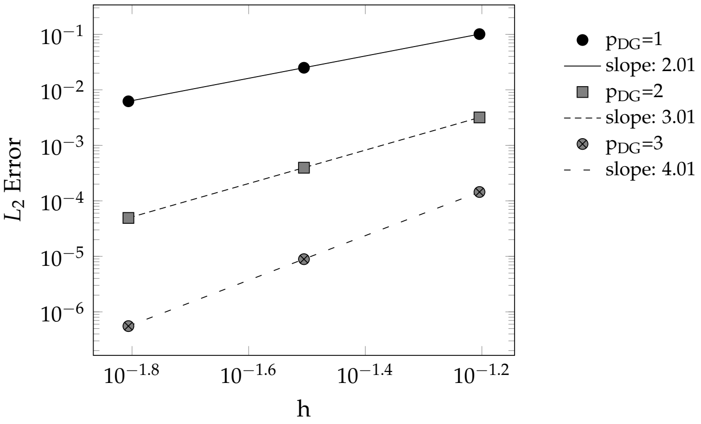


Figure 4: MMS (Malaya et al. 2013) for CNS: h-p convergence for ρE.

cosity, which is similar to Sutherlands Law or Power Law, is investigated in chapter 4.

3.3 Boundary Conditions

In this section, the implementation of farfield boundary conditions (i.e subsonic/supersonic inflow and subsonic/supersonic outflow) is verified.
Many theoretical analysis for the boundary conditions of the Euler and Navier-Stokes equations were done in the past: One technique for the hyperbolic Euler equation is to use relations based on characteristic lines, i.e the analysis of different waves crossing the boundary (Thompson 1987, Dutt 1988). Poinsot & Lele (1992) extend this analysis to the Navier-Stokes equations, the so-called Navier-Stokes characteristic boundary conditions (NSCBC). Table 1 summarizes the number of boundary conditions for a well-posed problem (suggested by theoretical analysis of Strikwerda (1977) and Dutt (1988)). Poinsot & Lele (1992) differentiate between the Euler conditions (inviscid conditions) and additional viscous conditions. They suggests that the viscous stress should be constant along the normal to the boundary, i.e $\frac{\partial \tau}{\partial n} = 0$.
In modern codes a common approach for farfield boundary conditions is to use only the inviscid boundary conditions (cf. Bassi & Rebay (1997), Hartmann & Houston (2008)). In these approaches it is assumed that the boundaries are far away from the region of interest and hence, the viscous stresses τ and temperature gradient vanishes. In this approach the necessary boundary conditions for the viscous terms are given implicitly by the interior values.

Table 1: Number of boundary conditions required for well-posedness (in 3 dimensions)

Boundary type	Euler	Navier-Stokes
Supersonic inflow	5	5
Subsonic inflow	4	5
Supersonic outflow	0	4
Subsonic Outflow	1	4

3.3.1 MMS for farfield boundary conditions

To test the implementation of the farfield boundary conditions, the manufactured solution must fulfill special requirements:

- viscous stresses and temperature gradients vanishes at the boundary. The systems reduces to the Euler equations
- no source terms at the boundaries

To meet these requirements, an exponential Ansatz function is chosen together with the idea of a constant velocity (cf. 3.1.1)

$$\mathbf{U} = \begin{pmatrix} a + b \cdot e^{(-x^2 * c)} \\ a + b \cdot e^{(-x^2 * c)} \\ \left(a + b \cdot e^{(-x^2 * c)}\right)^2 \end{pmatrix}. \tag{52}$$

a, b, c are arbitrary coefficients to adjust different boundary conditions. Far away from the origin $\mathbf{U} = (a, a, a^2)^\mathsf{T}$ and its derivatives nearly vanishes. The source term consist of the following components:

$$
\begin{aligned}
s_1 = & -2bcxe^{(-x^2c)} \\
s_2 = & -2bcxe^{(-x^2c)} \\
& -(\gamma - 1)\, 4 \left(a + be^{(-x^2c)} \right) bcxe^{(-x^2c)} \\
& +(\gamma - 1)\, bcxe^{(-x^2c)} \\
s_3 = & -4 \left(a + be^{(-x^2c)} \right) bcxe^{(-x^2c)} \\
& -(\gamma - 1)\, 4 \left(a + be^{(-x^2c)} \right) bcxe^{(-x^2c)} \\
& +(\gamma - 1)\, bcxe^{(-x^2c)} \\
& -\frac{\gamma}{\mathrm{Pr}} \left(-2bce^{(-x^2c)} + 4bc^2x^2e^{-x^2c} \right).
\end{aligned}
\tag{53}
$$

For the subsonic test case the following coefficients are chosen

$$a = 4.0, \quad b = 0.25 \quad \text{and} \quad c = 4.0. \tag{54}$$

The computational domain is set to $[-2; 2]$ to ensure that viscous stresses and temperature gradient nearly disappear at the boundaries.

Dirichlet Boundary Conditions

Dirichlet boundary conditions on both sides serves as reference solution. Here again the expected convergence rates are obtained, see Fig. 5.

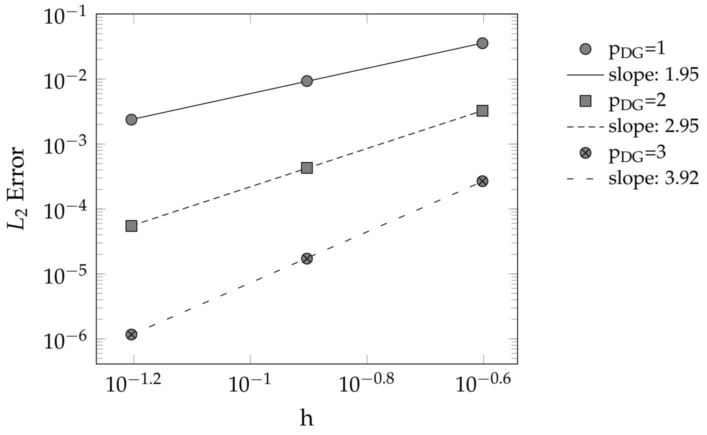


Figure 5: MMS for testing Dirichlet boundary conditions: h-p convergence for ρE.

Subsonic farfield Conditions

The characteristics analysis of the Euler equations leads to the needed boundary conditions given in table 1. A detailed analysis is given e.g by Chung (2002) or Poinsot & Lele (1992). Fig. 6 depicts the situation for the subsonic case: Two characteristics going into the domain, one is leaving the domain. At the outflow the opposite situation, two characteristics leaving the domain, one is entering the domain. For each entering characteristic a physical boundary condition is needed. In BoSSS, a subsonic inflow boundary is specified by the density ρ and

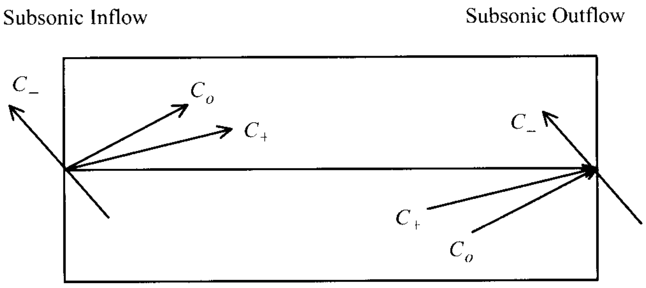


Figure 6: Characteristics at inflow and outflow for subsonic Euler equations in 1D

the velocity **v**. At a subsonic outlet only the pressure p needs to be given. All other values are extrapolated from interior points.

Subsonic Outflow

To test only the subsonic outlet condition, Dirichlet values are given at the inflow. At the outlet the pressure p is prescribed. Density ρ and velocity **v** are extrapolated. Fig. 7 shows the convergence rates for this test.

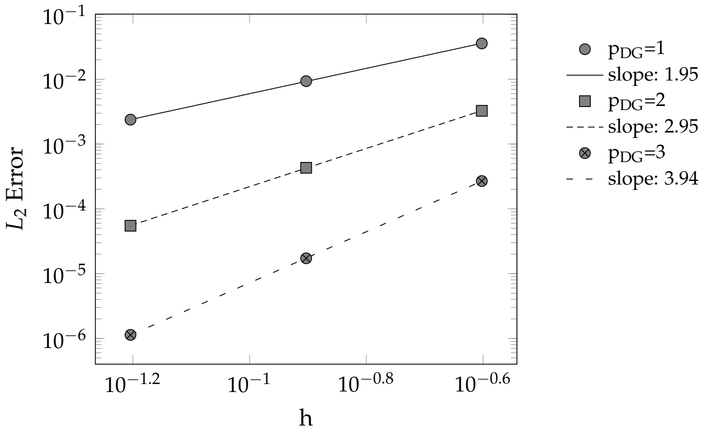


Figure 7: MMS for testing subsonic outlet boundary conditions: h-p convergence for ρE.

Subsonic Inflow

Numerical tests showed that the subsonic inlet condition is more sensitives than the outlet condition: It was not possible to run long term stable simulation for $p_{DG} > 1$ with the initial settings, due to the decreasing numerical diffusion with increasing polynomial degree p_{DG}. To remedy this, the computational domain is increased to $[-3, 3]$. With this setting long term stable simulations could be achieved and the nominal convergence rates are obtained (Fig. 8).

Subsonic Inflow/Outflow

Finally, "real" boundary conditions are used on both sides, i.e subsonic inflow on the left and subsonic outflow on the right boundary. Here again, the expected convergence rates are achieved (Fig. 9). These tests show that it possible to use Euler boundary conditions for CNS and that the assumption of using interior points for the additional viscous boundary condi-

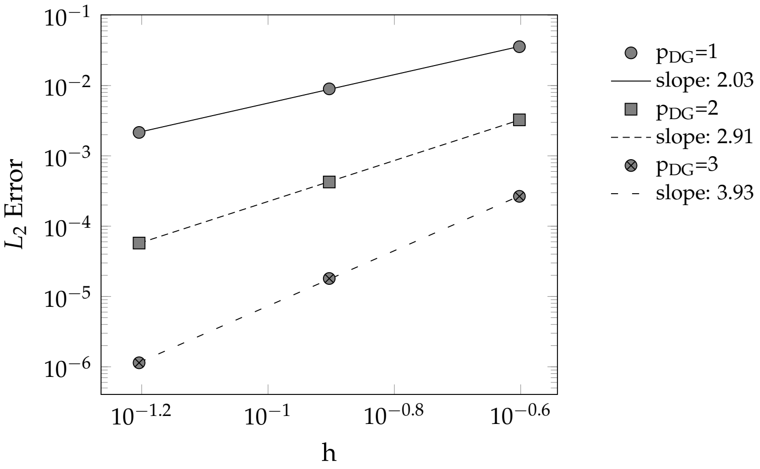


Figure 8: MMS for testing subsonic inlet boundary conditions: h-p convergence for ρE.

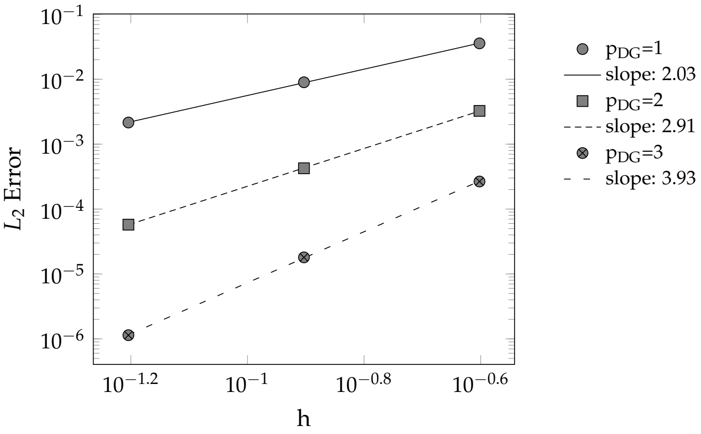


Figure 9: MMS for testing subsonic inlet and outlet boundary conditions: h-p convergence for ρE.

tions is valid. But it is important that the viscous stresses nearly vanish near the boundary, which could be seen in the tests for subsonic inflow conditions.

3.4 2D MMS for CNS

Similar to the 1D Test case 3.2.1, this MMS is chosen in such a way, that only a temperature gradient exists. Furthermore, a time dependent term is added. The solution is given as

$$\mathbf{U} = \begin{pmatrix} 2+0.5\sin(\beta) \\ 2+0.5\sin(\beta) \\ 2+0.5\sin(\beta) \\ (2+0.5\sin(\beta))^2 \end{pmatrix}, \tag{55}$$

with $\beta = 2\pi(x+y) - \omega t$. ω can be chosen arbitrarily and is related to the temporal scale. The corresponding source terms read as

$$\begin{aligned} s_1 &= 0.5\cos(\beta)(4\pi - \omega) \\ s_2 &= 0.5\cos(\beta)(-\omega + 2\pi(-1+3\gamma)) \\ &\quad + 0.5\sin(2\beta)2\pi(\gamma-1)) \\ s_3 &= 0.5\cos(\beta)(-\omega + 2\pi(-1+3\gamma)) \\ &\quad + 0.5\sin(2\beta)2\pi(\gamma-1)) \\ s_4 &= 0.5\cos(\beta)(12\gamma\pi + 4\pi - 4\omega) \\ &\quad + 0.25\sin(2\beta)(4\gamma\pi - \omega) \\ &\quad + 0.5\frac{8\pi^2\gamma\mu}{\mathrm{RePr}}\sin(\beta). \end{aligned} \tag{56}$$

The computational domain is a $[0,1] \times [0,1]$ square with a uniform Cartesian mesh.

3.4.1 Steady case

The first test is a steady simulation, hence $\omega = 0$. The other parameters are $\mathrm{Re} = 1000$, $\mathrm{Pr} = 0.72$ and $\mu = 1.0$. As boundary conditions, the analytical Dirichlet values are chosen. In Fig. 10 the results are shown. In the 2D case also the expected convergence rates can be attained.

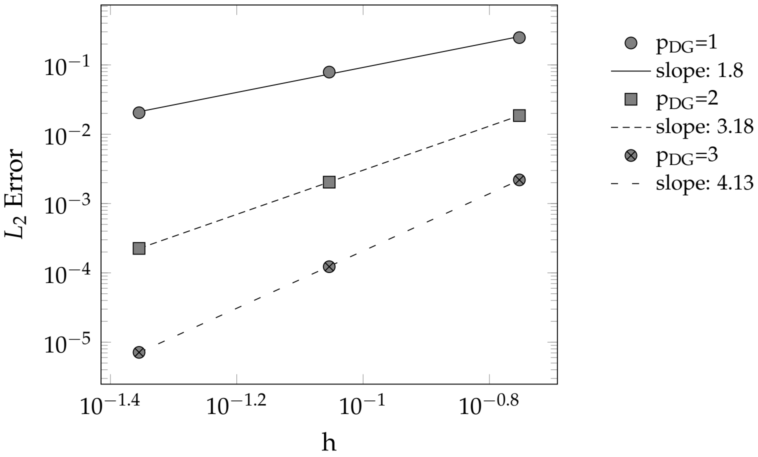


Figure 10: MMS (Gassner et al. (2008), steady) in 2D: h-p convergence for ρE.

3.4.2 Unsteady case

In the unsteady case, $\omega = 20\pi$ and a simulation time $t_{End} = 0.1$ is chosen and periodic boundary conditions for all boundaries. The time discretization scheme is a Runge-Kutta first order and a maximum stable timestep is chosen, which varies with spatial discretization and polynomial order of the DG-Ansatz (cf. Gassner (2009)). Fig. 11 shows again the expected behavior.

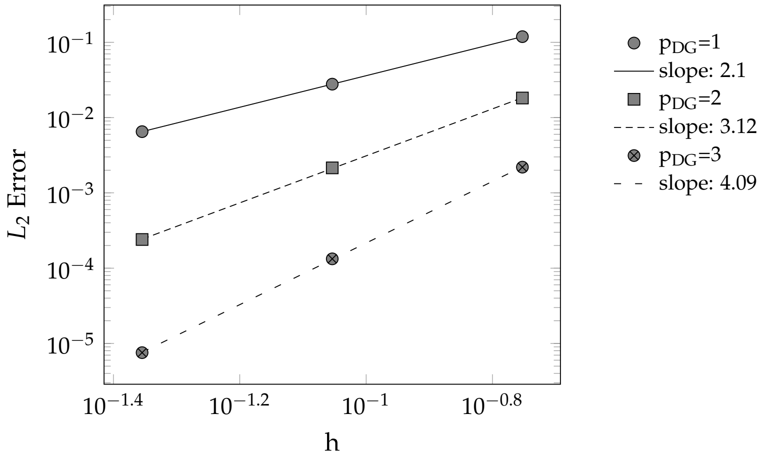


Figure 11: MMS (Gassner et al. (2008), unsteady) in 2D: h-p convergence for ρE..

4 Viscous Shock Profile (VSP)

Chapter 3 describes the importance of analytical solutions in the process of code verification. So far only solutions with constant viscosity are considered. Iannelli (2013) derives an exact similarity solution for the compressible Navier-Stokes equation, which accounts for a realistic non-linear variation of the dynamic viscosity μ. This non-trivial solution in 1D embeds supersonic, transonic and subsonic regions: It is the flow that develops in a shock wave through the rise in pressure (cf. Fig. 12). In his studies, Iannelli (2013) assumes a

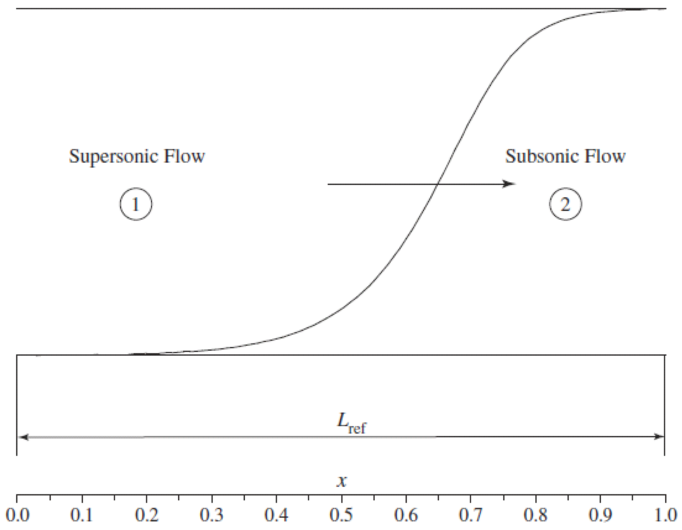


Figure 12: Shock configuration (Iannelli 2011)

perfect gas and sets the inflow conditions to 1. The upstream (denoted with $(\cdot)_1$) and downstream (denoted with $(\cdot)_2$) conditions are linked by the Rankine-Hugoniot relation and listed in Tab. 2 (Anderson 2011). Across the shock, the momentum and enthalpy is constant, i.e $\rho u \equiv \rho u_1 = \rho u_2$ and $\bar{h} \equiv \bar{h}_1 = \bar{h}_2$. The viscosity is traditionally expressed by a power law or the Sutherland law

Table 2: Upstream and downstream shock conditions

Upstream	Downstream
Ma_1	$\mathrm{Ma}_2^2 = \frac{(\gamma-1)\mathrm{Ma}_1^2+2}{2\gamma\mathrm{Ma}_1^2-(\gamma-1)}$
$\rho_1 = 1$	$\frac{\rho_2}{\rho_1} = \frac{(\gamma+1)\mathrm{Ma}_1^2}{2+(\gamma+1)\mathrm{Ma}_1^2}$
$p_1 = 1$	$\frac{p_2}{p_1} = \frac{2\gamma\mathrm{Ma}_1^2-(\gamma-1)}{(\gamma+1)}$
T_1=1	$\frac{T_2}{T_1} = \frac{p_2}{p_1}\frac{\rho_1}{\rho_2}$
$\rho u_1 = \sqrt{\gamma}\mathrm{Ma}_1$	$\rho u_2 = \rho u_1$
$u_1 = \sqrt{\gamma}\mathrm{Ma}_1$	$\frac{u_2}{u_1} = \frac{\rho_1}{\rho_2}$
$\rho E_1 = \frac{1}{\gamma-1} + \frac{\gamma}{2}\mathrm{Ma}_1^2$	$\rho E_2 = \frac{p_2}{p_1}\left(\frac{1}{\gamma-1} + \frac{\gamma}{2}\mathrm{Ma}_2^2\right)$
$\bar{h}_1 = \frac{\gamma}{\gamma-1} + \frac{\gamma}{2}\mathrm{Ma}_1^2$	$\bar{h}_2 = \bar{h}_1$

$$\mu(T) = T^{\omega} \quad \text{or} \quad \mu(T) = T^{3/2}\left(\frac{1+S/T_\infty}{T+S/T_\infty}\right) \tag{57}$$

with the viscosity exponent ω and the Sutherland constant S. In this case, a quadratic polynomial is used which best fits best both viscosity laws in (57):

$$\mu(T) = 1 + \mu_1'(T-1) + \frac{\mu''}{2}(T-1)^2. \tag{58}$$

μ_1' and μ'' are constant which can be calculated using one of the traditional viscosity laws (57). A detailed description is given by Iannelli (2011). For realistic flows $\mu_1' \simeq 1$ and $\mu'' < 0$ as well as $\mu \simeq 0$.

Iannelli (2013) gives the following analytical solution for u:

$$\begin{aligned}
&\frac{\mu_2 u_2}{(u_1-u_2)}\ln\left(\frac{u-u_2}{u_{\mathrm{in}}-u_2}\right) - \frac{\mu_1 u_1}{(u_1-u_2)}\ln\left(\frac{u_1-u}{u_1-u_{\mathrm{in}}}\right) \\
&+\mu_1'\left(\frac{\gamma-1}{2\gamma}\right)\left[\frac{u^2}{2}-\frac{u_{\mathrm{in}}^2}{2}+(u_1+u_2)(u-u_{\mathrm{in}})\right] \\
&+\frac{\mu''}{2}\left(\frac{\gamma-1}{2\gamma}\right)^2\Bigg[-\left(\frac{u^4}{4}-\frac{u_{\mathrm{in}}^4}{4}\right)-(u_1+u_2)\left(\frac{u^3}{3}-\frac{u_{\mathrm{in}}^3}{3}\right) \\
&\qquad+\left(4\bar{h}-\frac{4\gamma}{\gamma-1}+u_1u_2-(u_1+u_2)^2\right)\left(\frac{u^2}{2}-\frac{u_{\mathrm{in}}^2}{2}\right) \\
&\qquad+\left(4\bar{h}-\frac{4\gamma}{\gamma-1}-\left(u_1^2+u_2^2\right)\right)(u_1+u_2)(u-u_{\mathrm{in}})\Bigg] = -\frac{3(\gamma+1)\rho u\mathrm{Re}}{8\gamma}(x-x_{\mathrm{in}})
\end{aligned} \tag{59}$$

With this solution for u all other flow quantities, such as ρ, p, etc. and their gradients can be computed. Details can be found in Iannelli (2013).

Because of the logarithmic components, the solution gives a speed u that approaches the Rankine-Hugoniot relations asymptotically, i.e. $u \to u_{1,2}$ for $x \to \mp\infty$. Hence, u is selected as $u_1 < u_{\text{in}}$ at $x_{\text{in}} = 0$ and $u_{\text{out}} < u_2$ at $x_{\text{out}} = 1$. Convenient expressions for u_{in} and u_{out} are: $u_{\text{in}} = (1-\epsilon)u_1$ and $u_{\text{out}} = (1+\epsilon)u_2$ with $0 < \epsilon \ll 1$, e.g $\epsilon = 0.001$. The corresponding Reynolds number Re can be directly computed using the analytical solution of (59) by setting $u = u_{\text{out}}$ in the left hand side and $x = x_{\text{out}}$ in the right hand side. ϵ determines the Reynolds number and by decreasing ϵ, Re increases. By increasing the Reynolds number Re or the upstream Mach number Ma_1, it is possible to generate solutions which are more complicated to compute accurately. An increase of one of the parameters lead to a smaller transonic region, which includes a rise in the steepness of the gradients.

The solution (59) can be seen as a explicit solution for $x = x(u)$ and for each x the solutions provides a non-linear equation for u. It can be solved numerically with a Newton-Raphson method

$$u_{n+1} = u_n - \frac{f(u_n)}{f'(u_n)}, \tag{60}$$

where $f(u)$ is the left hand side of (59) and $f'(u)$ its first derivative with respect to u. It can be expressed as

$$\begin{aligned}\frac{\partial f}{\partial u} &= \frac{\mu_2 u_2}{(u_1-u_2)(u-u-2)} + \frac{\mu_1 u_1}{(u_1-u_2)(u_1-u)} \\ &\quad + \mu_1'\left(\frac{\gamma-1}{2\gamma}\right)(u+(u_1+u_2)) + \frac{\mu''}{2}\left(\frac{\gamma-1}{2\gamma}\right)^2\Bigg[-u^3-(u_1+u_2)\,u^2 \\ &\quad + \left(4\bar{h} - \frac{4\gamma}{\gamma-1} + u_1u_2 - (u_1+u_2)^2\right)u + \left(4\bar{h} - \frac{4\gamma}{\gamma-1} - \left(u_1^2+u_2^2\right)\right)(u_1+u_2))\Bigg].\end{aligned} \tag{61}$$

For the Newton-Raphson method (60) an initial estimate u_0 is needed an provided by the following expression:

$$u_0 = u_{\text{in}} + (u_{\text{out}} - u_{\text{in}})\left(\frac{1}{2} + \frac{\tanh(2.4(2x-1))}{2\tanh(2.4)}\right). \tag{62}$$

With the initial guess (62) and the derivative in (61), the analytical solution in (59) can be expressed explicitly.

The steady similarity solution (59) can be transformed to an unsteady solution by a Galilean coordinate transformation, as the Navier-Stokes equations are invariant under such transfor-

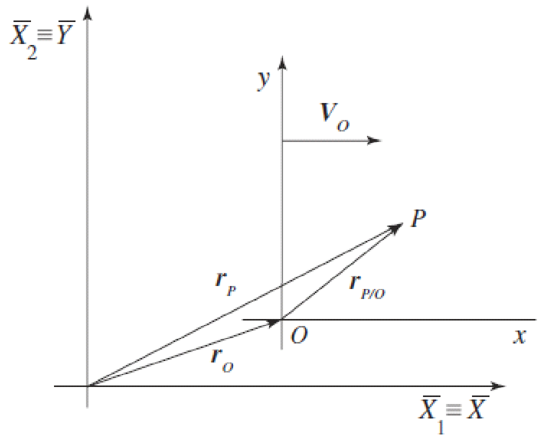


Figure 13: Galilean reference frames (Iannelli 2013)

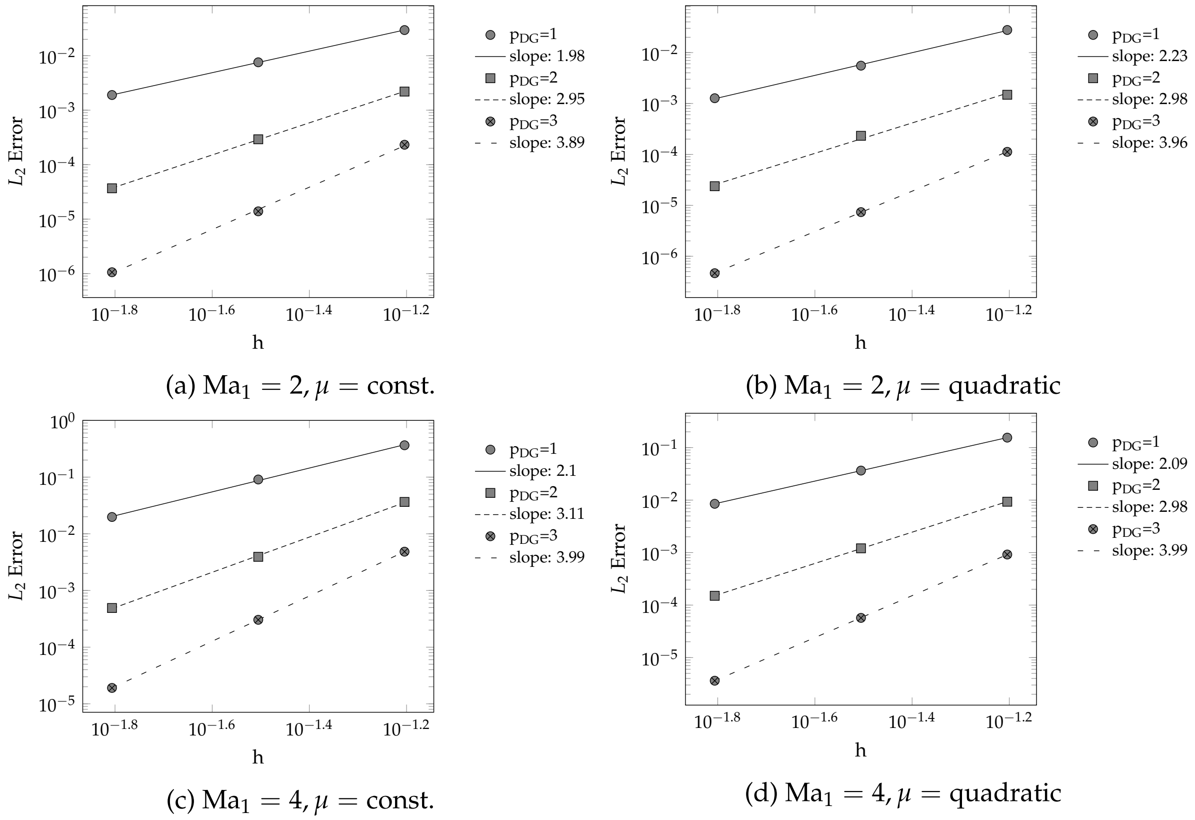
mations. The unsteady solution in the $(\overline{X_1}, \overline{X_2})$ space correspond to a steady solution in the (x, y) space that moves with respect to $(\overline{X_1}, \overline{X_2})$ with a constant velocity V_0, as illustrated in Fig. 13. It can be shown that the transformed solution is a time dependent solution of the Navier-Stokes equations, i.e

$$\rho(\overline{X}, t) = \rho(x), \quad u(\overline{X}, t) = u(x) + V_0, \quad \rho E(\overline{X}, t) = \rho E(x). \tag{63}$$

Furthermore, the analytical solution can also be extended to 2D or 3D via an Euler angle coordinate transformation. A detailed derivation is given by Iannelli (2013).

4.1 Steady

At first, a steady state calculation is done where convergence is reached when $\int_\Omega |\rho E^h_{n+1} - \rho E^h_n| d\Omega < 1.0 \times 10^{-10}$. The computational domain is set to $[0, 1]$ and equidistant grids are used. Supersonic inflow conditions are used at the inlet and at the outlet subsonic outflow conditions, where the momentum ρu and the total energy ρE is prescribed. The upstream Mach number is set to $\mathrm{Ma}_1 = 2$ and $\epsilon = 0.001$. For constant viscosity $\mu = 1$, i.e $\mu'_1 = \mu'' = 0$ the Reynolds number is $\mathrm{Re} = 9.695$ and for quadratic varying viscosity (58) ($\mu_2 = 1.417$,$\mu'_1 = 0.659$ and $\mu'' = -0.153$) $\mathrm{Re} = 11.352$. In a second series of tests the upstream Mach number is set to $\mathrm{Ma}_1 = 4$, which gives for constant viscosity $\mathrm{Re} = 3.555$ and for quadratic viscosity ($\mu_2 = 2.537$,$\mu'_1 = 0.616$ and $\mu'' = -0.073$) a Reynolds number of $\mathrm{Re} = 5.561$. All four tests show the nominal convergence rates, see Fig. 14.



(a) $\mathrm{Ma}_1 = 2, \mu = \text{const.}$

(b) $\mathrm{Ma}_1 = 2, \mu = \text{quadratic}$

(c) $\mathrm{Ma}_1 = 4, \mu = \text{const.}$

(d) $\mathrm{Ma}_1 = 4, \mu = \text{quadratic}$

Figure 14: Steady VSP in 1D for: h-p convergence for ρE.

4.2 Unsteady

In a next step, the unsteady implementation is verified. For this reason the Galilean invariance is used, described above. As background velocity $V_0 = 1$ is chosen. To reduce the influence of the boundaries the computational domain is extended to $[0,4]$. The initial position of the shock is in the section $[1,2]$ and is then advected to $[2,3]$, as visualized in Fig. 15. For time integration, a Runge-Kutta first order scheme is chosen with the maximum stable timestep. The simulation time is $t_{\text{End}} = 1$. Due to the extended domain, the boundary values of the analytical solutions vary only in the magnitude of 10^{-10}. Hence, Dirichlet values are used as boundary conditions. Again the convergence studies are performed for the two different upstream Mach numbers and a constant viscosity and quadratic varying viscosity, respectively. In general, the nominal convergence rates in the case of Ma $= 2$ and Ma $= 4$ can be obtained for constant and quadratic viscosity up to $\text{p}_{\text{DG}} = 3$. Surprisingly, two test cases ($\text{Ma}_1 = 2$, p_{DG}=3, μ =quadratic and Ma $= 4$, p_{DG}=1, μ = const.) with the coarsest grid gives better results than expected. But between the intermediate and fine grid the convergence rates of $\sim$4 and $\sim$2 respectively are obtained. So far, it is not clear why the results for the coarse grid are overrated: Smaller time-step size Δt or higher order time integration schemes up to 3 does not change the results as well as a different penalty factors C_{IP}.

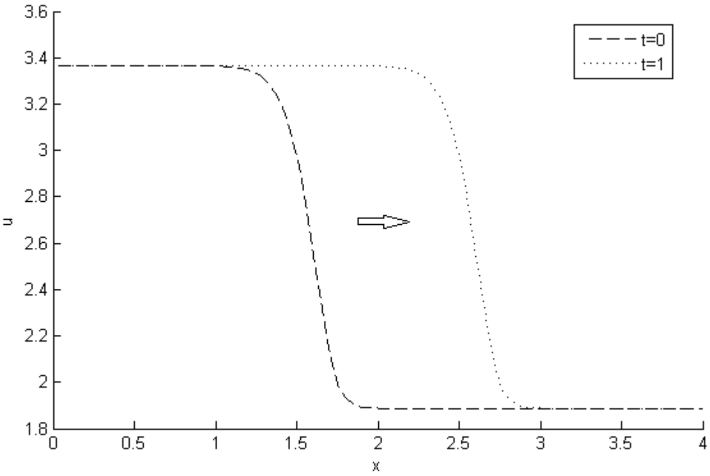
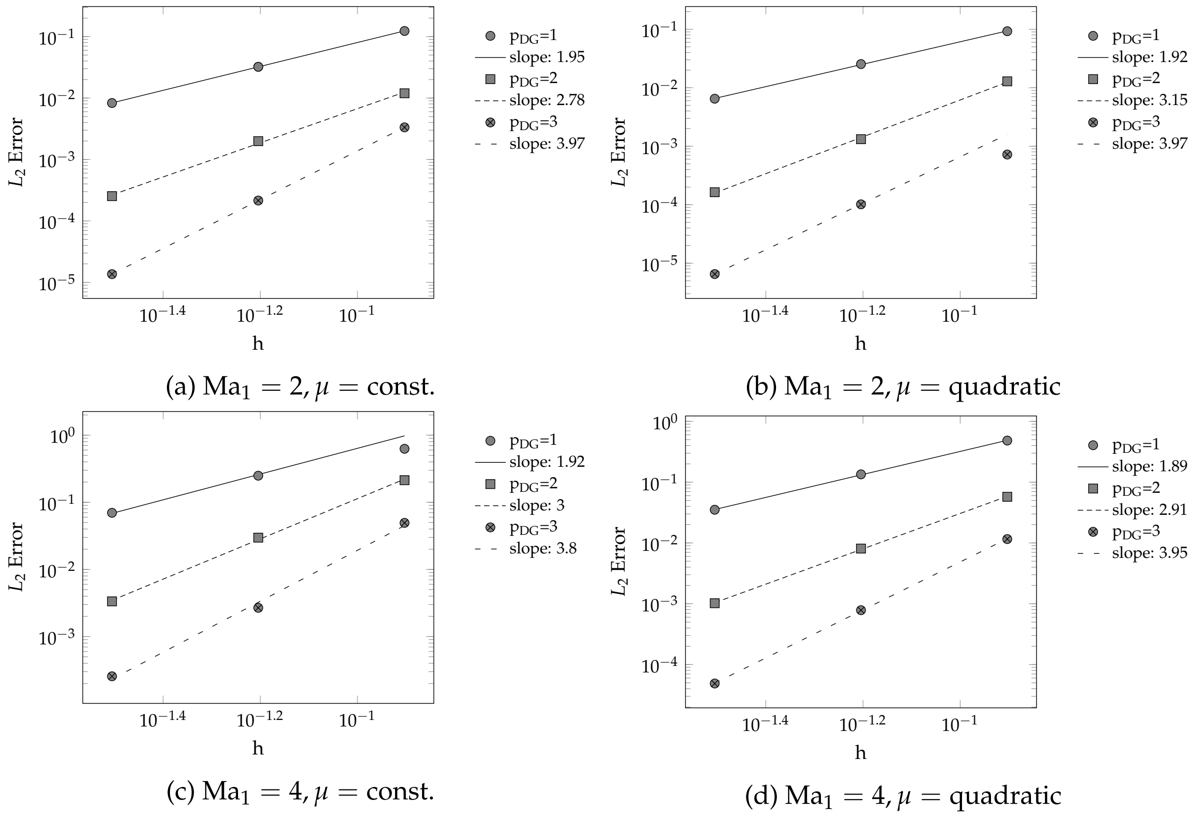


Figure 15: Unsteady configuration

5 Summary

In this Annual Report the non-dimensional compressible Navier-Stokes equations (CNS) used in BoSSS are presented. Furthermore, the derivation of the SIPG method for the viscous fluxes and their discretization details are given. The SIPG flux implementation was extensively tested in 1D with Manufactured Solutions (MMS) and an exact similarity solution for the CNS. All tests showed the nominal convergence rates in h-p convergence studies. Additionally the implementation of typical boundary conditions (e.g. subsonic inflow/outflow) were tested and their results were also satisfying. The viscous fluxes were implementated in such a way, that they can be easily extended to 2D and 3D, which was succesfully verified with a MMS in 2D. It can be assumed that the SIPG fluxes in 2D gives also the nominal convergence rates for the other test cases, e.g. 2D MMS (MASA Library) and VSP in 2D.



(a) $\mathrm{Ma}_1 = 2, \mu = \text{const.}$

(b) $\mathrm{Ma}_1 = 2, \mu = \text{quadratic}$

(c) $\mathrm{Ma}_1 = 4, \mu = \text{const.}$

(d) $\mathrm{Ma}_1 = 4, \mu = \text{quadratic}$

Figure 16: Unsteady VSP in 1D: h-p convergence for ρE.

In a next step, further benchmark tests can be computed and verified with data from other groups, e.g. flow around a cylinder or past an airfoil. For these test additional boundary conditions for the treatment of walls are needed, i.e. adiabatic or isothermal walls (cf. Hartmann (2008)).

References

ANDERSON, J. D. (2011): *Fundamentals of aerodynamics*. McGraw-Hill, New York.

ARNOLD, D. N., BREZZI, F., COCKBURN, B., MARINI, L. D. (2002): Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems. *SIAM Journal on Numerical Analysis 39*, 5, 1749–1779.

BASSI, F., REBAY, S. (1997): A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier–Stokes Equations. *Journal of Computational Physics 131*, 2, 267–279.

CHUNG, T. J. (2002): *Computational fluid dynamics*. Cambridge University Press, Cambridge ; New York.

DUTT, P. (1988): Stable Boundary Conditions and Difference Schemes for Navier–Stokes Equations. *SIAM Journal on Numerical Analysis 25*, 2, 245–267.

FEISTAUER, M., FELCMAN, J., STRASKRABA, I. (2003): *Mathematical and computational methods for compressible flow*. Numerical mathematics and scientific computation, Clarendon Press ; Oxford University Press, Oxford : New York.

GASSNER, G. (2009): *Discontinuous galerkin methods for the unsteady compressible navier-stokes equations*. Ph.D. thesis, Verl. Dr. Hut, München.

GASSNER, G., LÖRCHER, F., MUNZ, C. (2008): A Discontinuous Galerkin Scheme based on a Space-Time Expansion II. Viscous Flow Equations in Multi Dimensions. *Journal of Scientific Computing 34*, 3, 260–286.

HARTMANN, R. (2008): Numerical Analysis of Higher Order Discontinuous Galerkin Finite Element Methods.

HARTMANN, R., HOUSTON, P. (2008): An optimal order interior penalty discontinuous Galerkin discretization of the compressible Navier–Stokes equations. *Journal of Computational Physics 227*, 22, 9670–9685.

IANNELLI, J. (2011): An implicit Galerkin finite element Runge–Kutta algorithm for shock-structure investigations. *Journal of Computational Physics 230*, 1, 260–286.

IANNELLI, J. (2013): An exact non-linear Navier–Stokes compressible-flow solution for CFD code verification. *International Journal for Numerical Methods in Fluids 72*, 2, 157–176.

MALAYA, N., ESTACIO-HIROMS, K. C., STOGNER, R. H., SCHULZ, K. W., BAUMAN, P. T., CAREY, G. F. (2013): MASA : a library for verification using manufactured and analytical solutions. *Engineering with Computers 29*, 4, 487–496.

POINSOT, T. J., LELE, S. K. (1992): Boundary conditions for direct simulations of compressible viscous flows. *Journal of Computational Physics 101*, 1, 104–129.

ROACHE, P. J. (2002): Code Verification by the Method of Manufactured Solutions. *Journal of Fluids Engineering 124*, 1, 4–10.

SALARI, K., KNUPP, P. M. (2000): Code Verification by the Method of Manufactured Solutions. Tech. Rep. 1444, Sandia National Laboratories, Albuquerque, NM 87185.

STRIKWERDA, J. C. (1977): Initial boundary value problems for incompletely parabolic systems. *Communications on Pure and Applied Mathematics 30*, 6, 797–822.

THOMPSON, K. W. (1987): Time dependent boundary conditions for hyperbolic systems. *Journal of Computational Physics 68*, 1, 1–24.