Universität zu Köln

Master Thesis

Astrophysics Institut Cologne

"Modelling turbulent gases with Finite Volume and Discontinuous Galerkin methods "

---

submitted by

Johannes Markert

Köln - June 7, 2017

Contents

1 Introduction **1**

2 Theory **1**

2.1 Governing equations 1

2.1.1 Compressible Euler Equations 2

2.1.2 Weak Formulation 4

2.2 Turbulence Statistics 5

2.2.1 Energy Cascade & Powerspectrum 7

2.2.2 Probability Distribution Functions (PDF) 10

2.3 Supersonic Shocks 10

2.4 Finite Element Schemes 12

2.4.1 Discontinuous Galerkin Method 13

2.4.2 Flux Functions 16

2.4.3 Time Integration 16

1 Introduction

Astrophysical turbulence simulations of interstellar gases play vital role in the understanding and modelling of star formation. A multitude of numerical schemes and computational fluid dynamics software has been developed each with their own merits and drawbacks.

Finding the best numerical solver for a specific problem domain is a vibrant field of research as of today even though the first pioneers began to conduct simple physics simulations many decades ago. A lot has changed since these days. The present explosion in computing power paralleled by a striking inflation of operational costs passes on a mighty tool to the astrophysics community which opens the door to numerical experiments unfeasible ten years ago.

Alongside the advent of new technical possibilities modern numerical schemes appear which are better suited for utilizing lates CPU architectures to gain better performance. Especially, the aeronautics and car industries push forward the development of new simulation software. They are interested in modelling flight characteristics for new airplane designs, the airodynamic drag of cars or want to increase the efficiency of combustion engines. Consequently, innovative numerical schemes should be highly flexible with regards to meshing complex geometries, very accurate, massively parallelizable and blazingly fast. The demands for underlying physical model are moderate. The Navier-Stokes equations suffice on most occasions. One of the latest specimen of their kind is the high-order accurate CFD software FLEXI developed by a team lead by Prof. Claus-Dieter Munz in the Aeronautics Institut in Stuttgard, Germany. The basic idea is to introduce piecewise polynomial functions of order higher order. The scheme is called Discontinuous Galerkin method.

In contrast to aeronautical models the demands for turbulence simulations of astrophysical gases are quite the opposite. In free space there are no complex geometries at least not in proximity of massive objects like black holes. The physical model become extremely complex when electro-magnetic fields, radiation, chemistry and gravity are introduced to the governing equations. Furthermore a huge range of scales in time, space, energy, density and pressure must be covered, too. As if this is not enough high Mach numbers in the gas give rise to strong shock conditions and discontinuties which challenge the stability and accuracy of every numerical solver. Up to now the astrophysics community considered finite-volume schemes of first/second order to be the only viable method for conducting their simulations. FLASH originally developed by researchers at the University of Chicago is the name for a well established nuclear and astrophysics simulation software with finite-volumes as their foundational scheme.

This thesis is a first attempt to introduce higher order Galerkin methods to astrophysical turbulence simulations. The statistical evaluation of simple turbulence setups run by FLASH and FLEXI are expected to give an insight into the potential Galerkin schemes might offer to the astrophysics community.

2 Theory

2.1 Governing equations

In astrophysics, the *interstellar medium* (ISM) is the matter that exists in the space between the star systems in a galaxy. It contains gas in ionic, atomic, and molecular form, as well as dust and cosmic rays. ISM fills interstellar space and blends smoothly into the surrounding

intergalactic space. The medium is composed primarily of atomic hydrogen followed by helium with traces of carbon, oxygen, and nitrogen. Magnetic fields and turbulent motions also provide pressure in the ISM, and are typically more important dynamically than the thermal pressure.

Many theoretical models for ISM are basesd on the ideal *Magneto-Hydrodynamic equations* which are a blend of the *compressible Euler equations* and of the *Maxwell equations*, describing the hydrodynamics and (electro-)magnetodynamics, respectively.

In this thesis we solely focus on the hydrodynamics.

2.1.1 Compressible Euler Equations

In 1757 Leonhard Euler (1707-1783) published a set of equations for inviscid flow, known as the Euler equations. They are hyperbolic conservation equations which model perfect fluids without any interaction of their constituents. Hence, we assume no heat conduction ($\mathbb{T}^{i0} = \mathbb{T}^{0i} = 0$), no viscosity ($\mathbb{T}^{ij} = p\,\mathbb{I}$, $\mu_d = 0$) and no gravity $g = 0$. Within the comoving frame the *stress-energy tensor* $\mathbb{T}$ reads:

$$\mathbb{T}^{\alpha\beta} = \mathrm{diag}(\rho c^2, p, p, p) = \left(\rho + \frac{p}{c^2}\right) u^\alpha u^\beta + p\,\mathbb{G}^{\alpha\beta}. \tag{1}$$

In flat spacetime the *metric tensor* is set to $\mathbb{G} = \mathrm{diag}(-1, 1, 1, 1)$. The total energy and the number of particles are conserved.

$$\partial_\nu \mathbb{T}^{\mu\nu} = 0 \tag{2}$$

$$\partial_\mu (n\,u^\mu) = 0 \tag{3}$$

Taking the non-relativistic limit, we arrive at the conservative form of the Euler equations.

$$\partial_t \rho + \nabla \cdot (\rho\,\underline{u}) = 0 \qquad \textit{mass conservation} \tag{4}$$

$$\partial_t (\rho\,\underline{u}) + \nabla \cdot (\rho \underline{u}\underline{u}^T) + \nabla p = \underline{F} \qquad \textit{momentum conservation} \tag{5}$$

$$\partial_t E + \nabla \cdot (\underline{u}\,(E + p)) = 0, \qquad \textit{energy conservation} \tag{6}$$

where the total energy E is composed of the internal energy $\mathcal{I}$ and the kinetic energy $\mathcal{K}$.

$$E = \mathcal{I} + \mathcal{K} = \frac{p}{\gamma - 1} + \frac{\rho}{2} u^2, \tag{7}$$

with γ being the *adiabatic constant*.

The source term $\underline{F}$ (F for forcing) allows us to perpetually inject a force field which gets important in the discussion of driven turbulence later on. See section **??**.

Equation of State If not stated otherwise all simulations follow the *ideal gas law*.

$$p = \frac{c^2}{\gamma}\,\rho = R\,T\,\rho = \frac{R}{c_v}\mathcal{I} = (\gamma - 1)\mathcal{I}, \tag{8}$$

where R is the specific ideal gas constant, T is the gas temperature and $c_v = \frac{\gamma - 1}{R}$ is the specific heat capacity at constant volume.

The γ is set to

$$\gamma = \frac{c_p}{c_v} := \frac{5}{3}, \tag{9}$$

which represents a mono-atomic gas without interacting forces.

The speed of sound c is a direct consequence of the ideal gas equation.

$$c^2 = \gamma \frac{p}{\rho} := C_P = \text{const.} \tag{10}$$

During the numerical simulation the equation of state is enforced via the *polytropic process* (also called *polytropic cooling*) at every timestep.

$$p = C_P \, \rho^{\Gamma}, \tag{11}$$

where the *polytropic exponent* is set to $\Gamma := 1$ which is equivalent to an isothermal process. A thorough derivation can be found in [**?**], p.2-7.

Dimensionless Euler Equations We want to show that the Euler equations are invariant to changes of units. This discussion is useful since most numerical frameworks do not support physical units and rescaled phyiscal quantities avoid truncation errors due to the limits of floating point operations. For this, we choose a characteristic length l_r, a characteristic velocity u_r and a characterstic density ρ_r. Multiplying suitable combinations of these constants with the Euler equations yields

$$[\partial_t \rho + \nabla \cdot (\rho \, \underline{u})] \cdot \frac{l_r}{\rho_r \, u_r} = 0 \tag{12}$$

$$\left[\partial_t (\rho \, \underline{u}) + \nabla \cdot (\rho \underline{u} \underline{u}^T) + \nabla p - \underline{F}\right] \cdot \frac{l_r}{\rho_r \, u_r^2} = 0 \tag{13}$$

$$[\partial_t E + \nabla \cdot (\underline{u} \, (E + p))] \cdot \frac{l_r}{\rho_r \, u_r^3} = 0 \tag{14}$$

We simplify and get

$$\partial_{\tilde{t}} \tilde{\rho} + \tilde{\nabla} \cdot (\tilde{\rho} \, \underline{\tilde{u}}) = 0 \tag{15}$$

$$\partial_{\tilde{t}} (\tilde{\rho} \, \underline{\tilde{u}}) + \tilde{\nabla} \cdot (\tilde{\rho} \underline{\tilde{u}} \underline{\tilde{u}}^T) + \tilde{\nabla} \tilde{p} - \underline{\tilde{F}} = 0 \tag{16}$$

$$\partial_{\tilde{t}} \tilde{E} + \tilde{\nabla} \cdot (\underline{\tilde{u}} \, (\tilde{E} + \tilde{p})) = 0, \tag{17}$$

where $t_r = \frac{l_r}{u_r}$ (characteristic time) and

$$\tilde{t} = \frac{t}{t_r}, \; \tilde{\rho} = \frac{\rho}{\rho_r}, \; \underline{\tilde{u}} = \frac{\underline{u}}{u_r}, \; \tilde{\nabla} = l_r \, \nabla, \; \tilde{E} = \frac{E}{\rho_r \, u_r^2}, \; \tilde{p} = \frac{p}{\rho_r \, u_r^2}, \; \underline{\tilde{F}} = \underline{F} \, \frac{l_r}{\rho_r \, u_r^2}. \tag{18}$$

Conseqently, the dimensionless Euler equations do not change under unit transformation. If not stated otherwise we drop the tilde sign ($\tilde{\cdot}$) and assume always dimensionless quantites from now on.

Choice of parameters One consequence of dimensionless units is the free choice of parameters. We want to use this feature in choose a sensible set of parameters. Considering the Euler equations in conservative form, eqn. (4), their functions of space and time

$$\rho = \rho(t,x,y,z), \quad (\rho\underline{u}) = (\rho\underline{u})(t,x,y,z), \quad E = E(t,x,y,z) \tag{19}$$

are completed with

$$\gamma := 5/3, \quad R := 1, \langle\rho\rangle := 1, \langle c\rangle := 1, \tag{20}$$

From that we derive

$$C_P = \frac{c_0^2}{\gamma} = 3/5 = 0.6, \quad \langle p\rangle = C_P \cdot \langle\rho\rangle = 0.6, \quad \langle E\rangle = \frac{\langle p\rangle}{\gamma - 1} = 0.9, \quad \langle T\rangle = \frac{\langle c\rangle^2}{\gamma\, R} = 0.6, \tag{21}$$

where $\langle\cdot\rangle$ is the *volume-weighted average* or *mean value* over the domain Ω

$$\langle q\rangle = \frac{\int_\Omega q\, d\Omega}{\int_\Omega d\Omega} \tag{22}$$

If not state otherwise, these set of parameters define the global state at all times.

2.1.2 Weak Formulation

A natural way to define a generalized solution of the Euler equations that does not require differentiability is, to go back to the integral form of the conservation law,

The basic idea is to take the PDE, multiply it by a smooth *test function*, integrate one or more times over some domain, and then use integration by parts to move derivatives off the function q and onto the smooth test function. The result is an equation involving fewer derivatives on q, and hence requiring less smoothness.

In this section we want to derive the *weak formulation* of the governing equations. This establishes the basis for the polynomial formulation which is the core idea of all DG methods. First, the Euler equations get split up into terms resembling the independent one temporal and three spatial dimensions with respect to the linear differential operator.

$$\partial_t\underline{U} + \partial_x\underline{F}(\underline{U}) + \partial_y\underline{G}(\underline{U}) + \partial_z\underline{H}(\underline{U}) + \underline{S} = 0, \tag{23}$$

where

$$\underline{U} = (\rho, \rho u_1, \rho u_2, \rho u_3, E)^T \tag{24}$$

$$\underline{F}(\underline{U}) = (\rho u_1, \rho u_1^2 + p, \rho u_1 u_2, \rho u_1 u_3, u_1(E+p))^T \tag{25}$$

$$\underline{G}(\underline{U}) = (\rho u_2, \rho u_2 u_1, \rho u_2^2 + p, \rho u_2 u_3, u_2(E+p))^T \tag{26}$$

$$\underline{H}(\underline{U}) = (\rho u_3, \rho u_3 u_1, \rho u_3 u_2, \rho u_3^2 + p, u_3(E+p))^T \tag{27}$$

$$\underline{S} = (0, -f_1, -f_2, -f_z, 0)^T \tag{28}$$

Defining a vector-valued test function $\underline{\phi} = (0, .., 0, \phi_i, 0, ..., 0)^T$ $(i \in 1, ..., 5)$, multiplying componentwise with above equation and integrating over the domain Ω we get

$$\int_\Omega \left(\partial_t U_i\, \phi^i + \partial_x F_i(\underline{U})\, \phi^i + \partial_y G_i(\underline{U})\, \phi^i + \partial_z H_i(\underline{U})\, \phi^i + S_i\phi^i\right) d^3x = 0 \tag{29}$$

Integration-by-parts rearranges the integral into a *source term*, *volume term* and *surface term*.

$$\int_\Omega \partial_t U_i \, \psi(x,y,z)^i d^3x + \int_\Omega S_i \psi(x,y,z)^i d^3x = \tag{30}$$

$$\int_{\partial\Omega} \Big(F_i(\underline{U}) \, \psi(x,y,z)^i \, n_x + G_i(\underline{U}) \, \psi(x,y,z)^i \, n_z + H_i(\underline{U}) \, \psi(x,y,z)^i \, n_z \Big) \, d^2x,$$

$$- \int_\Omega \Big(F_i(\underline{U}) \, \partial_x \psi(x,y,z)^i + G_i(\underline{U}) \, \partial_y \psi(x,y,z)^i + H_i(\underline{U}) \, \partial_z \psi(x,y,z)^i \Big) \, d^3x \tag{31}$$

where $\underline{n} = (n_x, n_y, n_z)^T$ is the outward surface normal to $\partial\Omega$.

Unfortunately, weak solutions are not unique, and so an additional problem is to identify which weak solution is the physically correct vanishing-viscosity solution. Again, one would like to avoid working with the viscous equation directly, but it turns out that there are other conditions one can impose on weak solutions that are easier to check and will also pick out the correct solution. These are called *entropy conditions* by analogy with the gas dynamics case, where a discontinuity is physically realistic only if the entropy of the gas increases as it crosses the shock.

2.2 Turbulence Statistics

Turbulences are a common phenomena in nature. It names a flow regime in fluid dynamics characterized by chaotic changes in pressure and flow velocity and contrasts *laminar* flow, which occurs when a fluid flows in parallel layers, with no disruption between them.

In astrophysics turbulence is suspected to play a major role in star formation within interstellar media (ISM). Hence, a theoretical understanding of the underlying mechanics is crucial in order correctly model turbulences in numerical simulations. While the problem for incompressible media has been thoroughly studied in the past (*Kolmogorov scaling*), the additional dynamics introduced by compressibility are still an active field of research.

It is important not to forget that real ISM turbulence is neither isothermal, nor polytropic. The real ISM has a local temperature that is not a simple function of density but results from the evolution of the thermal energy. Gravity takes over control over the most dense regions. Magnetic fields lead to effects like frozen filaments called *Alfvén's frozen in theorem*.

Reynolds Number An important dimensionless quantity in fluid mechanics used to help predict flow patterns in different fluid flow situations is the *Reynolds number Re*. The Irish-British physicist Osborne Reynolds (1842-1912) discovered this fundemental relationship during his famous flow tube experiments in the second half of the 19th century. It represents the ratio between *inertial* and *viscous* forces and is calculated by

$$Re = \frac{\text{inertial force}}{\text{viscous force}} = \frac{\rho_0 \, u_0 \, L}{\mu_{vis}}, \tag{32}$$

where ρ_0, u_0, L, μ_{visc} are the fluid density, fluid velocity, characteristic length and dynamic viscosity, respectively. Since the governing equations are defined to be the inviscid Euler equations (cf. section 2.1), in theory, there is no viscosity anywhere present.

$$\mu_{visc} \equiv 0 \implies Re = \infty \tag{33}$$

This has far-reaching consequences because the Reynolds number predicts the transition from *laminar* to *turbulent* flow. Laminar flow becomes unstable when the inertial forces dominate over the viscous forces: $Re \gg 1$. Consequently, the Euler equations lead to chaotic motion over all time and length scales.

Reynolds Decomposition Turbulences is characterized by a random fluctuation of flow variables in time meassured at a fixed point in space. Their evolution can be described with the *Reynolds decomposition*.

$$q(t) = \langle q \rangle_t + \tilde{q}(t), \quad \langle \tilde{q} \rangle \equiv 0 \tag{34}$$

where

$$\langle q \rangle_t = \frac{\int_{t_0}^{t1} q(t) dt}{t_1 - t_0} \tag{35}$$

is the time-average of the flow property $q(t)$. Turbulent flows are globally described in terms of their mean values of properties like density, velocity and pressure. The velocity fluctuations are independent of the axis of reference, i.e. invariant to translation, rotation and reflection; they are *isotropic*. *Isotropic turbulence* is by its definition always homogeneous. In such a situation, the gradient of the mean velocity does not exists, the mean velocity is either zero or constant throughout.

As the Reynolds decomposition suggests it does not make sense to talk about a turbulence at one specific point in time or space. Thus, we define important average properties or *mean values* which help to quantify the current state of the turbulent system as a whole.

Root-mean-square Velocity is the mass-weighted average squared velocity of the system.

$$u_{\text{rms}} = \sqrt{\frac{\int_\Omega \rho \, \underline{u}^2 \, d\Omega}{\int_\Omega \rho \, d\Omega}} \tag{36}$$

We utilize mass-weighting since the conducted turbulence simulations are highly compressible.

Sonic Mach Number is directly related to u_{rms} via

$$\mathcal{M} = \frac{u_{\text{rms}}}{\langle c \rangle}. \tag{37}$$

From $\langle c \rangle = 1$, according to eqn. (20), follows that Mach number and root-mean-square velocity are equivalent and, if not stated otherwise, used interchangeably.

Turning time represents a time span characterizing large scale turbulent motions. Under section 2.1 a characteristic time scale $t_r = \frac{l_r}{u_r}$ was introduced exhibiting the invariance of the Euler equations under unit transformations. Now it gets a practical meaning since we define a *turning time* that tells how long it takes for a turbulence vortex to traverse the length L of the physical domain.

$$T_{turn} = \frac{L}{u_{rms}} \tag{38}$$

Equivalent terms for the turning time are *crossing time* or *dynamic time scale*.

Dynamic time is the scaled *physical* time t.

$$t_d = \frac{t}{T_{turn}} \tag{39}$$

All time evolution plots presented in this thesis are scaled to dynamic time which sets the time axis in relation to the analyzed turbulence.

Bulk Motion is the integrated momentum over the whole domain.

$$\underline{\mathcal{P}} = \int_\Omega \rho\, \underline{u}\, d\Omega \tag{40}$$

For turbulences in closed systems, say a periodic box (three-dimensional torus), bulk motion distorts the measurement of mean values like the sonic Mach number. In the worst case scenario the whole fluid would coherently move in one direction pretending to be a turbulent flow of a certain Mach number. Correction of bulk motion is a crucial subject in turbulence simulations.

Total Kinetic Energy is important in the discussion of energy accounting, especially in closed systems where the total energy $E = \mathcal{I} + \mathcal{K} = \text{const}$ is conserved.

$$\mathcal{K} = \int_\Omega \frac{\rho}{2}\, \underline{u}^2\, d\Omega \tag{41}$$

Total Kinetic Energy Dissipation Rate is the negated time derivative of the total kinetic energy.

$$\varepsilon = -\frac{d\mathcal{K}}{dt} \tag{42}$$

It quantifies how fast kinetic energy gets transformed into internal energy at a certain point in time.

$$\frac{dE}{dt} = \frac{d\mathcal{I}}{dt} + \frac{d\mathcal{K}}{dt} = 0 \iff \frac{d\mathcal{I}}{dt} = -\frac{d\mathcal{K}}{dt} \tag{43}$$

2.2.1 Energy Cascade & Powerspectrum

Real turbulences always have a three-dimensional character and they lead to rotational flow structures, called turbulent eddies, with a wide range of length and energy scales.

Fluid particles which were initially separated by a large distance can be brought close together by eddying motions. Consequently, mass, heat and momentum are very effectively exchanged. It is an established fact that this property has a profound influence in birth of stars, solar systems and cosmic structures.

Andrey Kolmogorov (1903-1987) was a pioneering mathematician who provided a statistical treatment of turbulent flows, the *Kolmogorov picture*, with methods from *dimensional analysis*. The spatial wavenumber k for an eddy of diameter λ is defined as

$$k = \frac{2\pi}{\lambda}. \tag{44}$$

Based on k the eddies in a turbulent flow are arranged into a hierarchy or more precisely, a *spectrum*. The largest eddies interact and transfer energy from the mean flow. We call it the

range of *large scales*. Since large eddies are of the same order of the characteristic length L and velocity scale u_{rms} the flow is inviscid and their dynamics are dominated by inertial effects. At the *inertial range* kinetic energy gets transferred down to smaller eddies via *vortex stretching*. An *energy cascade* from larger to smaller scales is emerges. The so called *Kolmogorov length scales*, denoting the smallest scales in the spectrum, form the *viscous* or *dissipative sublayer range* where the energy input from nonlinear interactions and the energy drain from viscous dissipation are in exact balance. The Reynolds number is 1 (eqn. (32)) and the laminar flow dominates.

$$\eta = \left(\frac{\mu_{\text{visc}}^3}{\rho^3\,\varepsilon}\right)^{1/4} \qquad \text{length scale} \tag{45}$$

$$\tau = \left(\frac{\mu_{\text{visc}}}{\rho\,\varepsilon}\right)^{1/2} \qquad \text{time scale} \tag{46}$$

$$\upsilon = \left(\frac{\mu_{\text{visc}}\,\varepsilon}{\rho}\right)^{1/4} \qquad \text{velocity scale,} \tag{47}$$

where ρ is the characteristic density and ε is the kinetic energy dissipation rate defined in eqn. (**??**). Fig. 1 gives an illustration of the energy spectrum of a turbulent flow as just described.

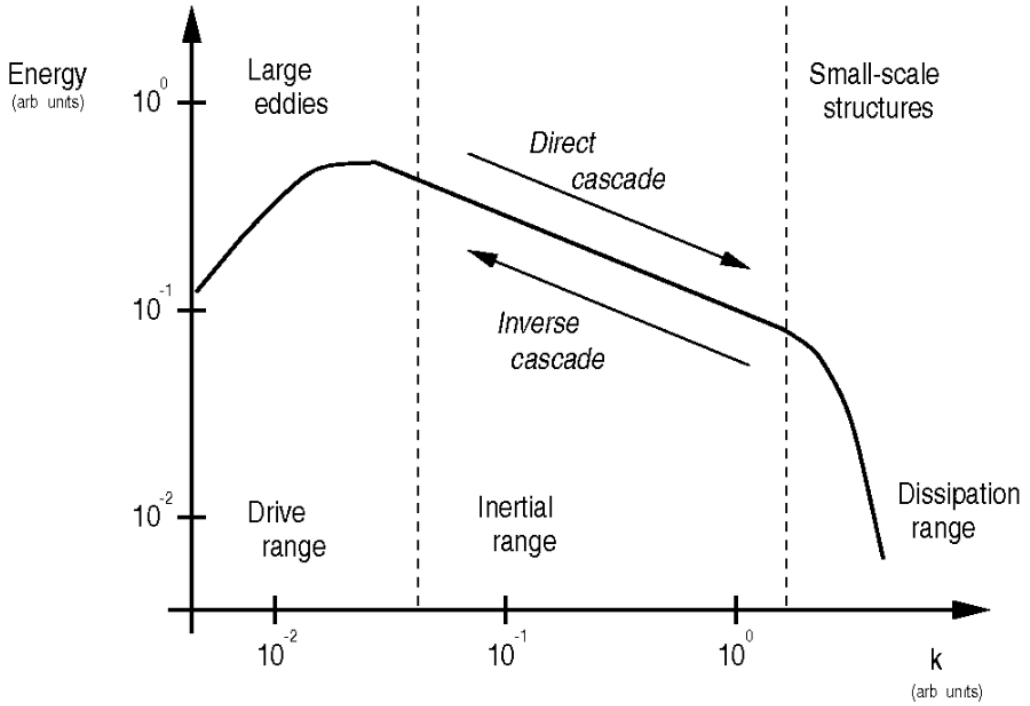

Figure 1: Schematic Source: [**?**], p6 of the Kolmogorov picture of turbulence showing the spatial energy powerspectrum over k. The mathematical definition is given by eqn. (50). The physical meaning of the marked ranges are detailed in the text.

Numerical Viscosity By the very definition of the Euler equations there is no viscosity: $\mu_{\text{visc}} \equiv 0$. This would mean turbulent flow all the way down. The dissipation range is non existent. It is rather evident that no physical simulation, running on a machine with computing power and memory limits, is able to model this flow. The profound consequence is that ISM simulations are, no matter what, always underresolved.

Every numerical scheme introduces a so called *numerical* viscosity $\mu_{\text{n-visc}}$ that marks the very limit of length, time and velocity scale this scheme is still capabable to resolve. The exact amount of $\mu_{\text{n-visc}}$ is in general unknown but should be unique for every numerical method. For grid based approaches it is possible to estimate a lower limit for the length scale η. The *Nyquist–Shannon sampling theorem* states that:

> If a function $x(t)$ contains no frequencies higher than B Hertz, it is completely determined by giving its ordinates at a series of points spaced $1/(2B)$ seconds apart.

Applying this theorem to a uniform grid with N nodes, the smallest length scale would then become $\eta = 2/N$. With eqn. (45) we arrive at a relation between kinetic energy dissipation and numeric viscosity.

$$\mu_{\text{n-visc}} = (\eta^4\ \varepsilon)^{1/3} = 2^{4/3}\ \left(\frac{\varepsilon}{N^4}\right)^{1/3} \propto \varepsilon^{1/3} \tag{48}$$

When the grid resolution N is equal among different numerical schemes then their capability of resolving smallest scales can be compared by looking at the kinetic energy dissipation rate.

Kinetic Energy Powerspectrum Fig. 1 shows the schematic of the shell-averaged kinetic energy spectrum $\overline{P_{\mathcal{K}}}$ of a fully developed three-dimensional turbulence. The powerspectrum $P_{\mathcal{K}}$ is defined as

$$P_{\mathcal{K}} = \hat{\mathcal{K}}(\underline{k}) \cdot \hat{\mathcal{K}}^{\dagger}(\underline{k}), \tag{49}$$

where $\underline{k} = (k_1, k_2, k_3)^T$ is the spatial wave vector analog to eqn. (44), $\hat{\mathcal{K}}(\underline{k})$ is the Fourier transformed kinetic energy field $\mathcal{K}(\underline{x})$ and $\hat{\mathcal{K}}^{\dagger}(\underline{k})$ its complex conjugate. Taking the three-dimensional shell-average over $P_{\mathcal{K}}(\underline{k})$ yields

$$\overline{P_{\mathcal{K}}}\ dk = 4\pi\, k^2\ \hat{\mathcal{K}}(\underline{k}) \cdot \hat{\mathcal{K}}^{\dagger}(\underline{k})\ dk = 4\pi\, k^2\ \hat{\mathcal{K}} \cdot \hat{\mathcal{K}}^{\dagger}\ dk. \tag{50}$$

The pre-factor $4\pi\, k^2$ is a contribution from the differential volume of a thin sphere at radius $k = |\underline{k}| \geq 0$. The area under $\overline{P_{\mathcal{K}}}$

$$A_{\mathcal{K}^2} = \frac{1}{V_\Omega}\ \int_0^\infty 4\pi\, k^2\ \hat{\mathcal{K}} \cdot \hat{\mathcal{K}}^{\dagger}\ dk \tag{51}$$

is equal to the total squared kinetic energy $\mathcal{K}^2$ of the system in accordance with Parseval's theorem

$$\int_{-\infty}^{\infty} |Y(x)|^2\, dx = \frac{1}{2\pi}\ \int_{-\infty}^{\infty} |\hat{Y}(k)|^2\, dk. \tag{52}$$

Velocity Powerspectrum It has been shown that the decline of the energy cascade among turbulences is universal. By determining the slope of log-log scale velocity powerspectra one can ensure that a turbulence simulation is valid.

For each velocity component we take the Fourier transform of the velocity field $\underline{u} = (u_1, u_2, u_3)^T$. We denote these Fourier transforms as $\underline{\hat{u}} = (\hat{u}_1, \hat{u}_2, \hat{u}_3)^T$. Analog to above we define the power-spectrum as

$$P_u = \frac{1}{2}\ \underline{\hat{u}} \cdot \underline{\hat{u}}^{\dagger}. \tag{53}$$

Taking the three-dimensional shell-average we get

$$\overline{P_u}\, dk = 4\pi\, k^2\ \ \frac{1}{2}\ \underline{\hat{u}} \cdot \underline{\hat{u}}^{\dagger}\, dk, \tag{54}$$

which we call the *volume-weighted* velocity powerspectrum. By mass-weighting the velocity beforehand, $\rho^{1/2}\,\underline{u}$, we get the *mass-weighted* velocity powerspectrum.

$$\overline{P_{mu}}\,dk = 4\pi\,k^2\;\;\frac{1}{2}\,\widehat{(\rho^{1/2}\,\underline{u})}\cdot\widehat{(\rho^{1/2}\,\underline{u})}^{\,\dagger}\,dk \tag{55}$$

The slopes for the volume-weighted and mass-weighted powerspectra should amount to -19/9 and -5/3, respectively.

2.2.2 Probability Distribution Functions (PDF)

Another tool to verify the properness of a turbulence simulation are density and velocity distributions. According to various studies ... fully developed turbulence models yield a nearly log-normal density distribution.

$$P_s(s) = \frac{1}{\sqrt{2\pi}\sigma_s}\exp\left[-\frac{(s-s_0)^2}{2\sigma_s^2}\right], \tag{56}$$

where $s = \ln(\rho/\rho_0)$. The standard deviation σ_s of the distribution is related to the sonic Mach number $\mathcal{M}$ of the turbulence via

$$(2\,\sigma_s)^2 = \ln(1 + b^2\,\mathcal{M}^2) \tag{57}$$

Reording yields an explicit expression for the Mach number

$$\mathcal{M}_{PDF} = \frac{\sqrt{\exp((2\,\sigma_s)^2) - 1}}{b} \tag{58}$$

The proportionality constant b dependents on the ratio of compressible to solenoidal forcing $\zeta \in [0,1]$, see ..., and the number of spatial dimensions $D = 1, 2, 3$.

$$b = 1 + (D^{-1} - 1)\,\zeta \tag{59}$$

2.3 Supersonic Shocks

Astrophysical flows often involve shock waves. Shocks, or in more technical terms singular compression waves, are escalating highly localized spikes in density and pressure due to nonlinear dynamics inherent to the Euler equation. When a shock wave is emerging the velocity behind the wave front is higher than in front of it. The media gets highly compressed untill an unphysical state is reached. The velocity characteristica begin to cross. Nature solves this dilemma by introducing additional physics like extreme heat radiation, explosions, bangs or detachement of media in case of surface waves. Either way, it involves an increase of entropy according to the second law of thermodynamics. A numerical solver has to capture this kind of physics in order to prevent unphysical solutions.

Method of Characteristics The prototype for differential equations of second order is the Burger's equation.

$$u_t + u\,u_x = \epsilon\,q_{xx}, \tag{60}$$

where $u(x,t)$ represents the velocity at position x at time t. Eqn. (60) becomes inviscid in the limit of $\epsilon \to 0$. The characteristic equations are

$$\frac{dx}{dt} = u \quad \text{and} \quad \frac{du}{dt} = 0 \tag{61}$$

with their obvious solutions $x(t) = u\,t + C_1$ and $u(t) = C_2$. Since C_2 must be a function of C_1 we derive a general solution for eqn. (60) in the inviscid limit.

$$u = C_2(C_1) \quad \Longleftrightarrow \quad u(x,t) = C_2(x - u\,t) \tag{62}$$

Now if we set the initial condition to $u(x,0) = 1 - \cos(x)$ and plot (fig. 2) the solution of characteristics in eqn. (61) we get intersecting lines which is unphysical since there the velocity-time profile in fig. 3 becomes a multi-valued solution.

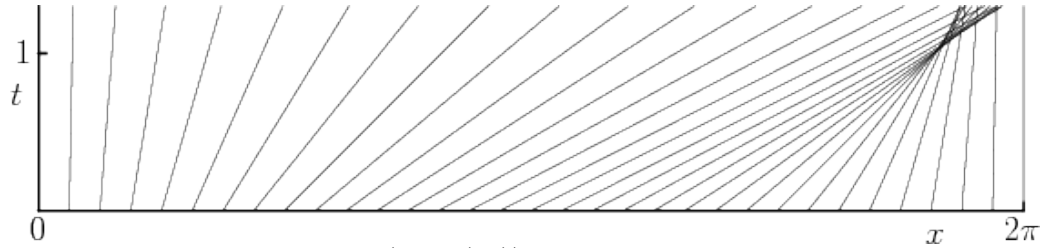

Figure 2: Intersecting characteristics (eqn. (61)) of the inviscid Burger's equation in eqn. (60) are symptoms of a shock wave. The left area where lines diverge from each other is called *rarefaction* fan. Converging lines are part of the *compression* fan. The region of intersection marks the shock discontinuity where the solution (cf. fig. 3) gets multi-valued.

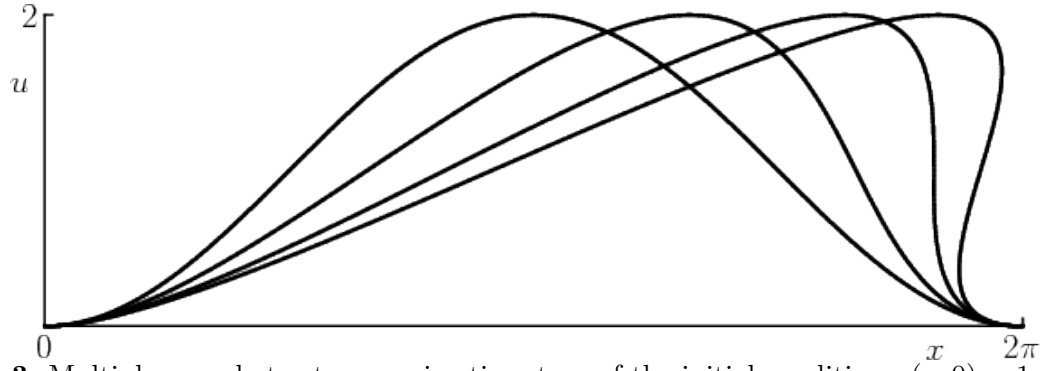

Figure 3: Multiple snapshots at successive timesteps of the initial condition $u(x,0) = 1 - \cos(x)$ evolving under the inviscid Burger's equation (eqn. (60)). It eventually develops a shock and becomes multi-valued which is unphysical.

Riemann Problem The correct modelling of shocks is summarized under the term *Riemann problem*. It provides the theoretical basis for the correct treatment of discontinuities in solutions of nonlinear PDEs. Furthermore, the Riemann problem is an integral part of finite element schemes which approximate the physical solution by piecewise constant or polynomial functions as Finite-Volume and discontinuous Galerkin methods do. Confer section 2.4.2.

Gibb's Phenomenon An inherent downside of schemes involving polynomes is the GIBB'S phenomenon. It states that polyonomes of higher order trying to approximate discontinuties yield spurious oscillations. See fig. 4. Their are many approaches to get the ringing near shocks

under control. Two of them, *Artificial Viscosity* and *FV-DG mode switching*, are presented in section **??**.

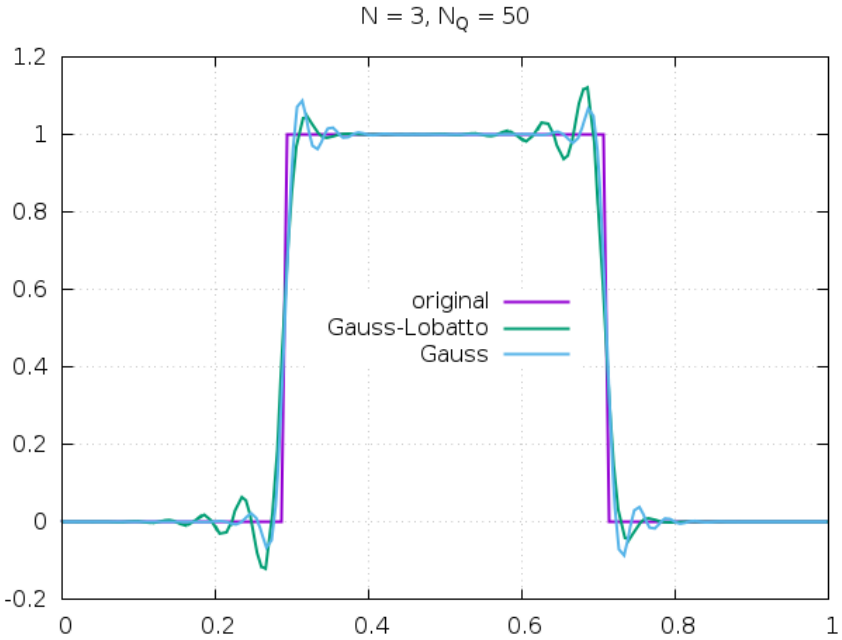

Figure 4: Examples of spurious oscillation of the discontinuous Galerkin method solving the one-dimensional advection equations with a discontinuous initial condition. The domain is divided into 50 elements each containing a third-order polymome. Gauss and Gauss-Lobatto are two types of distributing interpolation nodes within each element (cf. section **??**).

2.4 Finite Element Schemes

Under the motto "Divide and Conquer" the physical domain is cut into smaller subdomains or finite elements. All finite elements methods (FEM) contain multiple stages of operation and are explained briefly.

Meshing Divide the problem domain into adjunct self-contained sub-domains, called elements or cells. Depending on the scheme and requirements this step can happen periodically. Via *Adaptive Mesh Refinement* (AMR) small scale phenomena within the simulation can be resolved where needed without degrading the overall performance disproportionately. AMR is not part of any simulation conducted in this thesis.

Reconstruction Approximate the exact solution in every element by a piecewise constant function (finite volume scheme) or polynome of order N_p (Galerkin scheme): section 2.4.1

Evolution Based on the current set of variables the governing equations (compressible Euler equations) yield a new state which gets evolved one timestep into the future: section 2.4.2.

Averaging/Propagation Flux functions solve the Riemann problem and communicate the lately acquired state across boundaries and propagate the new information throughout the element: section 2.4.3.

2.4.1 Discontinuous Galerkin Method

By refining the mesh the error in the numerical approximation of the physical solution decays algebraically, that is, introducing more elements. Traditionally, numerical steps in space or time are labeled with the letter "h". Hence, mesh refinement is also known as *h-refinement*.

Since the author of the thesis cannot formulate the advanates of DG methods better than following praise on page ... in ..., he takes the freedom to quote:

> An alternative approach is to keep the number of subdomains fixed and increase the order of the interpolating polynomials. This is called *p-refinement*. For infinitely smooth solutions p-refinement usually leads to an exponential decay of the numerical error.
>
> Owing to their finite element nature, the DG methods have the following main advantages over classical finite volume methods [...]:
>
> - The actual order of accuracy of DG methods solely depends on the exact solution; DG methods of arbitrarily high formal order of accuracy can be obtained by suitably choosing the degree of the approximating polynomials.
> - DG methods are highly parallelizable. Since the elements are discontinuous, the mass matrix [eqn. (70)] is block diagonal and since the size of the blocks is equal to the number of degrees of freedom inside the corresponding elements, the blocks can be inverted [beforehand].
> - DG methods are very well suited to handling complicated geometries and require an extremely simple treatment of the boundary conditions in order to achieve uniformly high-order accuracy.
> - DG methods can easily handle adaptivity strategies since refinement or unrefinement of the grid can be achieved without taking into account the continuity restrictions typical of conforming finite element methods. Moreover, the degree of the approximating polynomial can be easily changed from one element to the other. Adaptivity is of particular importance in hyperbolic problems given the complexity of the structure of the discontinuities.
>
> They provide fast convergence, small diffusion and dispersion errors, better data volume-over-surface ratio for efficient parallel processing and better input/output handling due to the smaller volume of data.

The section is concluded now with a brief derivation of the DG method.

At first the physical domain Ω is divided into a *mesh* of adjunct self-contained sub-domains Ω_l ($l \in \mathbb{N}$) with well-defined boundaries. For the rest of this text we omit the element index l. So Ω denotes now the domain within an element. The unknown solution $\underline{U}$ is replaced by polynomes of order N_p constructed from linear combinations of orthogonal basis functions $\underline{\Psi}^j$.

$$U_{\underline{i}}(t,x,y,z) \approx p_{\underline{i}}(t,x,y,z) = \sum_{\underline{j}=0}^{N_p} U_{\underline{i}}^{\underline{j}}(t)\, \Psi^{\underline{i}}(x,y,z), \tag{63}$$

where $\underline{i}$ and $\underline{j}$ are three-dimensional *multi-indices*.

$$\underline{i} = (i,j,k)^T, \quad i,j,k \in \mathbb{N}_0 \tag{64}$$

Remark Usally, the polynomes of every element are transformed to a reference space $\hat{\Omega} = [-1,1]^3$ where the actual interpolation takes place. This meassure massively increases efficiency since the basis functions are equal among all elements. For the sake of simplicity this step is ommitted here.

The following treatment is analog for all five conservative variables of the Euler equation hence we ignore the index i. Remembering the general weak formulation, eqn. (30), of the solution integral,

$$\int_\Omega \left(\sum_{j=0}^{N_p} (\partial_t U^j(t))\, \Psi^j(\underline{x}) \right) \phi(\underline{x}) d^3x + \int_\Omega S(t)\, \phi(\underline{x}) d^3x =$$

$$\int_{\partial\Omega} \left[F(t)\, \phi(\underline{x})\, n_x + G(t)\, \phi(\underline{x})\, n_z + H(t)\, \phi(\underline{x})\, n_z \right] d^2x, \tag{65}$$

$$- \int_\Omega \left[\left(\sum_{j=0}^{N_p} F^j(t)\, \Psi^j(\underline{x}) \right) \partial_x \phi(\underline{x}) + \left(\sum_{j=0}^{N_p} G^j(t)\, \Psi^j(\underline{x}) \right) \partial_y \phi(\underline{x}) + \left(\sum_{j=0}^{N_p} H^j(t)\, \Psi^j(\underline{x}) \right) \partial_z \phi(\underline{x}) \right] d^3x,$$

we put the time variable into the coefficients S, F, G, H and the space variables into the basis functions Ψ and test functions ϕ.

Lagrange Polynome If we associate the basis functions $\Psi^{\underline{J}} := L^{\underline{J}}$ and the test functions $\phi := L^{\underline{I}}$ with LAGRANGE polyonmes of equal order N_p, we can formulate an interpolation and integration scheme (*collocation*) over the domain Ω. The polynome in one dimension reads as follows

$$l_j(x) = \prod_{k=0, k\neq j}^{p} \frac{x - x_k}{x_j - x_k}, \quad j = 0, \ldots, p, \tag{66}$$

with the KRONECKER property $l_j(x_i) = \delta_{ij}$.

For illustration purposes we begin with the simplest case: the one-dimensional LAGRANGE interpolation.

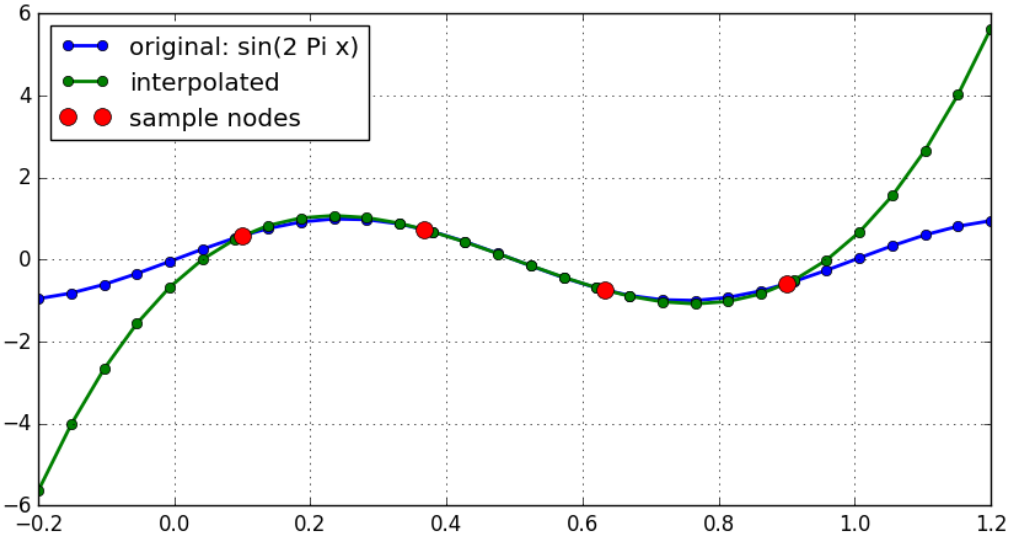

Figure 5: One-dimensinal Lagrange interpolation with four sample nodes.

The Lagrange polynome of third order needs four anchor nodes with their associated values in order to continuously interpolate all values in between. Unfortunately, Lagrange polynomes go to infinity going further away from the outer anchor nodes. This effect occurs in the Gauss-Quadrature where the outermost anchor nodes do not lie on the interval boundaries.

One gets to three-dimensional formulation via the *Tensor Product Ansatz*.

$$L_{\underline{i}}(\underline{x}) = L_{ijk}(x, y, z) = l_i(x) \cdot l_j(y) \cdot l_k(z) \tag{67}$$

We write the thee-dimensional polynome as:

$$P^{\underline{i}}(t, \underline{x}) = \sum_{\underline{i}=0}^{N_p} F_{\underline{i}}(t) L_{\underline{i}}(\underline{x}) = \sum_{i,j,k=0}^{N_p} f_{ijk}(t) \cdot l_i(x) \cdot l_j(y) \cdot l_k(z) \tag{68}$$

Note, that the time varying part of the polynome lives exclusively in the coefficients.

Galerkin Polyonome Replacing $\Psi^{\underline{j}}$ and ϕ accordingly, we get an explicit ansatz, called *Galerkin* method.

$$\begin{aligned}
&\int_\Omega \left(\sum_{\underline{j}=0}^{N_p} \dot{U}_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}) \right) L_{\underline{i}}(\underline{x}) d^3x + \int_\Omega S(t, \underline{x})\, L_{\underline{i}}(\underline{x}) d^3x = \\
&\int_{\partial\Omega} \left[F(t)\, n_x(\underline{x}) + G(t)\, n_y(\underline{x}) + H(t)\, n_z(\underline{x}) \right] L_{\underline{i}}(\underline{x})\, d^2x \\
&- \int_\Omega \left[\left(\sum_{\underline{j}=0}^{N_p} F_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}) \right) \partial_x L_{\underline{i}}(\underline{x}) + \left(\sum_{\underline{j}=0}^{N_p} H_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}) \right) \partial_y L_{\underline{i}}(\underline{x}) + \left(\sum_{\underline{j}=0}^{N_p} G_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}) \right) \partial_z L_{\underline{i}}(\underline{x}) \right] d^3x
\end{aligned} \tag{69}$$

Evaluating above formula at discrete nodes $\underline{x_{\underline{i}}}$ and introducing *integration weigths*

$$\omega_{\underline{i}} = \int_\Omega L_{\underline{i}}(\underline{x}) d^3x \tag{70}$$

we can discretize the continuous integrals.

$$\begin{aligned}
&\sum_{\underline{k}=0}^{N_p} \left(\sum_{\underline{j}=0}^{N_p} \dot{U}_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}_{\underline{k}}) \right) L_{\underline{i}}(\underline{x}_{\underline{k}})\, \omega_{\underline{k}} + \sum_{\underline{k}=0}^{N_p} S(t, \underline{x}_{\underline{k}})\, L_{\underline{i}}(\underline{x}_{\underline{k}}) \omega_{\underline{k}} = \\
&\left[F^*(t) + G^*(t) + H^*(t) \right] L_{\underline{i}}(\underline{x}) \\
&- \sum_{\underline{k}=0}^{N_p} \left[\left(\sum_{\underline{j}=0}^{N_p} F_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}_{\underline{k}}) \right) L_{\underline{i}}^{(x)}(\underline{x}_{\underline{k}}) + \left(\sum_{\underline{j}=0}^{N_p} H_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}_{\underline{k}}) \right) L_{\underline{i}}^{(y)}(\underline{x}_{\underline{k}}) + \left(\sum_{\underline{j}=0}^{N_p} G_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}_{\underline{k}}) \right) L_{\underline{i}}^{(z)}(\underline{x}_{\underline{k}}) \right] \omega_{\underline{k}}
\end{aligned} \tag{71}$$

The partial differential ∂_x turns into a discrete linear operator.

$$L_{\underline{j}}^{(x)}(\underline{x}) = \partial_x L_{\underline{j}}(\underline{x}) = (\partial_x l_{j_1}(x)) \cdot l_{j_2}(y) \cdot l_{j_3}(z) \tag{72}$$

$$D_{\underline{ij}}^{(x)} = D_{i_1 i_2 i_3 j_1 j_2 j_3}^{(x)} = l_{j_1}^{(x)}(x_{i_1}) \cdot l_{j_2}(y_{i_2}) \cdot l_{j_3}(z_{i_3}) \tag{73}$$

The operators for the y- and z-dimension are constructed in analog manner.

The surface term get replaced by flux functions (cf. section 2.4.2 where an exchange of mass, momentum and energy between element boundaries takes place. Following common tradition they get marked with a star: F^*.

Weak Formulation Finally, we arrive at the semi-discrete weak formulation of the discontinuous Galerkin method which can be directly translated to computer code.

$$\begin{aligned}
&\sum_{\underline{k}=0}^{N_p} \left(\sum_{\underline{j}=0}^{N_p} \dot{U}_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}_{\underline{k}}) \right) L_{\underline{i}}(\underline{x}_{\underline{k}})\, \omega_{\underline{k}} + \sum_{\underline{k}=0}^{N_p} S(t, \underline{x}_{\underline{k}})\, L_{\underline{i}}(\underline{x}_{\underline{k}}) \omega_{\underline{k}} = \\
&\left[F^*(t) + G^*(t) + H^*(t) \right] L_{\underline{i}}(\underline{x}) \\
&- \sum_{\underline{k}=0}^{N_p} \left[\left(\sum_{\underline{j}=0}^{N_p} F_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}_{\underline{k}}) \right) D_{\underline{ki}}^{(x)} + \left(\sum_{\underline{j}=0}^{N_p} H_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}_{\underline{k}}) \right) D_{\underline{ki}}^{(y)} + \left(\sum_{\underline{j}=0}^{N_p} G_{\underline{j}}(t)\, L_{\underline{j}}(\underline{x}_{\underline{k}}) \right) D_{\underline{ki}}^{(z)} \right] \omega_{\underline{k}}
\end{aligned} \tag{74}$$

Remark If the polynomial order is set to one $N_p = 1$ the formulation reduces to the first order FV method.

2.4.2 Flux Functions

Finite Element methods must solve the Riemann problem (cf. section 2.3) at the element boundaries in order to exchange information. The *Rankine–Hugoniot conditions* or *shock jump conditions* require that mass, momentum and energy must be conserved when crossing the boundaries. The theoretical framework of the Rieamnn problem knows a procedure that provides an exact solution to a given shock problem. Unfortunately, an exact Riemann solver is in most cases hard to construct and even harder to compute due to bad convergences rates.

Hence, a wide variety of approximate Riemann solvers or *flux functions* have been proposed that can be applied much more cheaply than the exact Riemann solver and yet give results that in many cases are equally good when used in high-resolution finite element methods.

A detailed discussion of the theory of flux functions would go beyond the scope of this thesis and has been done extensively elsewhere. The flux functions are considered an integral part of the numerical scheme (cf. section **??**).

2.4.3 Time Integration

At the end of section 2.4.1 in eqn. (74) we arrived at a semi-discrete weak formulation of the DG method. It is nothing more than an ordinary differential equation of the form

$$\frac{d}{dt} y = f(t, y). \tag{75}$$

Defining initial values $y(t_0) = y_0$ this equation can be numerically solved in the most naive way via the EULER (EU) method. Chosing an appropiate timestep Δt we can explicitly integrate from the current state y^n to the future state y^{n+1}.

$$y^{n+1} = y^n + \Delta t \cdot f(t^n, y^n) \tag{76}$$

However, if the Δt-convergence rate is worse than the p-refinement it would render the advantages of the Galerkin method useless. A widely used class of higher-order time integration schemes with good convergence properties are the Runge-Kutta methods (RK). The second-order RK, resp. *midpoint* method (MP), reads

$$y^{n+1} = y^n + \Delta t \cdot f\left(t^n + \frac{\Delta t}{2}, y^n + \frac{\Delta t}{2} \cdot f(t^n, y^n)\right), \tag{77}$$

evaluating the integrand two times per timestep. Introducing again another timestep, the integration becomes of third order (RK3):

$$y^{n+1} = y^n + \frac{1}{6}\,(k_1 + 4\,k_2 + k_3), \tag{78}$$

where

$$k_1 = \Delta t \cdot f(t^n, y^n) \tag{79}$$

$$k_2 = \Delta t \cdot f\left(t^n + \frac{\Delta t}{2}, y^n + \frac{\Delta t}{2} \cdot f(t^n, y^n)\right) \tag{80}$$

$$k_3 = \Delta t \cdot f\left(t^n + \Delta t, y^n - k_1 + 2\,k_2\right). \tag{81}$$

Courant-Friedrichs-Lewy condition To keep a numerical algorithm stable the time step has to obey the *Courant-Friedrichs-Lewy condition* (CFL condition) which states that the domain of dependence of q_i^{n+1} of the algorithm at future time time t^{n+1} should include the true domain of dependence at time $t = t^n$. Or in other words: nothing is allowed to flow more than one grid spacing within one time step. This means quantitatively

$$\Delta t \leq \frac{\Delta x}{u} \tag{82}$$

Given a CFL number: $0 < C \leq 1$

$$\Delta t = C \cdot \min_x \left(\frac{\Delta x}{u(x)}\right) \tag{83}$$

The CFL condition is a nessecary (but not sufficient) condition for the stability of any explicit differencing method. With increasing order of the Runge-Kutta scheme the stability increases though. RK3 even allows the CFL number to be greater 1.

In a three-dimensional orthogonal domain the timestep reads

$$\Delta t = C \cdot \min_{\underline{r}} \left(\frac{dx}{|v_x(\underline{r})| + c(\underline{r})}, \frac{dy}{|v_y(\underline{r})| + c(\underline{r})}, \frac{dz}{|v_z(\underline{r})| + c(\underline{r})}\right) \tag{84}$$

For supersonic regimes the sound speed c can be neglected.