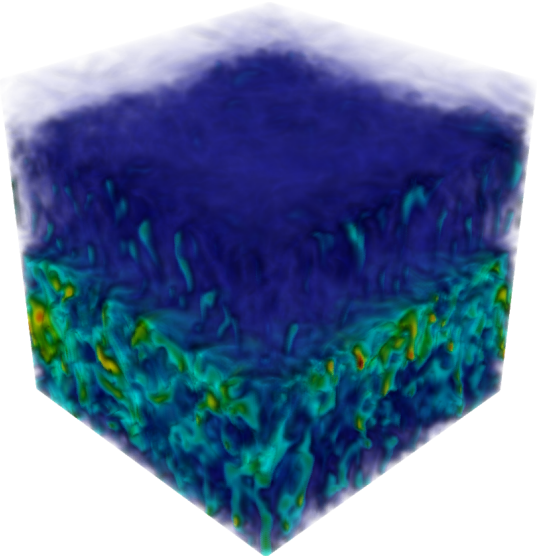
Coupling of Nuclear Reaction Networks and Hydrodynamics for Application in Stellar Astrophysics



Dissertation

Philipp V. F. Edelmann

TECHNISCHE UNIVERSITÄT MÜNCHEN

MAX-PLANCK-INSTITUT FÜR ASTROPHYSIK

Coupling of Nuclear Reaction Networks and Hydrodynamics for Application in Stellar Astrophysics

Philipp Valentin Ferdinand Edelmann

Vollständiger Abdruck der von der Fakultät für Physik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzende(r):		Univ.-Prof. Dr. L. Oberauer
Prüfer der Dissertation:		
	1.	Univ.-Prof. Dr. F. K. Röpke, Julius-Maximilians-Universität Würzburg
	2.	Univ.-Prof. Dr. A. Ibarra

Die Dissertation wurde am 19.12.2013 bei der Technischen Universität München eingereicht und durch die Fakultät für Physik am 19.02.2014 angenommen.

Abstract

Multidimensional hydrodynamics simulations are becoming an increasingly useful tool for understanding processes in stellar astrophysics. Nuclear reactions are important for many of these processes. This work extends a hydrodynamics code with a nuclear reaction network and investigates different methods of coupling. Additionally, a well-balanced method for the treatment of gravity is introduced to prevent numerical problems connected to many low Mach number schemes. Convective mixing at the core of a Population III star is simulated as a first application.

Zusammenfassung

Mehrdimensionale Hydrodynamiksimulationen sind für das Verständnis von Prozessen in der stellaren Astrophysik von zunehmendem Nutzen. Nukleare Reaktionen sind für viele dieser Prozesse wichtig. Diese Arbeit erweitert einen Hydrodynamik-Code durch ein nukleares Reaktionsnetzwerk und untersucht verschiedene Methoden der Kopplung. Außerdem wird eine ausbalancierte Methode zur Behandlung der Gravitation eingeführt, um numerische Probleme zu verhindern, die bei vielen Verfahren für kleine Machzahlen auftreten. Als erste Anwendung wird das konvektive Mischen im Kern eines Sterns der Population III simuliert.

Contents

1 Introduction

Stars in various stages of their life are basically the sole producer of all elements heavier than lithium in the Universe. Through nuclear reactions enabled by the high densities and temperatures in their interior, they convert the primordial elements H, He, and Li to heavier elements. Even though this conversion happens inside stars, it is the driving mechanism of the evolution of all baryonic matter in the Universe. This is because stars feed a part of the elements they produce back to the interstellar medium, from which new generations of stars and planetary systems are formed.

Stars go through different so-called *burning phases* in which the conditions for fusion of a fuel element are met. For example, during the H-burning phase they turn hydrogen into helium, in the He-burning phase they produce carbon from helium. Nuclear burning occurs at the core or in radial shells around the core. Different phases of shell and core burning can overlap. The last of these phases is Si-burning, which ultimately forms iron. At this point the core of the star will collapse due to effects discussed below.

Stars of low and intermediate mass ($\lesssim 8 M_\odot$) never reach the last burning stages. In a very simplified picture this can be understood by the following argument. When a star has depleted the fuel for a burning stage at its core, nuclear energy generation stops and the core starts to contract. If the critical density and temperature for the ignition of the next burning phase can be reached, it will commence releasing energy through this burning process. If it cannot be reached, the core will stay inert, while the previous burning stages still occur in shells around the core. The final outcome of this system is a *white dwarf star*, which is the remaining degenerate core, surrounded by a *planetary nebula* formed by the outer layers of the star that were gravitationally unbound. The material in the planetary nebula is mixed into the interstellar medium.

Many important nucleosynthetic processes occur in supernova explosions. Supernovae are classified into two main categories according to the physical explosion mechanism. *Thermonuclear supernovae*, also known as Type Ia supernovae due to their spectral classification, are believed to be thermonuclear explosions in white dwarf stars in binary systems (e.g. Hillebrandt & Niemeyer, 2000). The other kind of supernova, called *Core-collapse supernovae*, are triggered when the cores of massive stars ($M_{\rm star} \gtrsim 8\,M_\odot$) become gravitationally unstable due nuclear changes in the composition. For stars of about $9\,M_\odot$ this

is expected to happen before they reach hydrostatic Ne-burning. The electrons in the core become degenerate and the low reaction threshold of Ne and Mg together with the increased Fermi energy of the electrons cause a large rate of electron captures. This reduces the pressure in the core and causes it to collapse. More massive stars form a Fe-core through hydrostatic Ne-, O-, and Si-burning. When this core reaches about 10^{10} K, photodissociation of Fe to α-particles and free nucleons occurs and the subsequent electron captures make the core gravitationally unstable. In the ensuing formation of a neutron star or black hole at the core the outer layers of the star are ejected by an explosion whose exact mechanism is still not perfectly understood. It probably involves hydrodynamic instabilities and the effect of neutrinos created in neutron star formation (Janka, 2012). The ejecta of both types of supernova enrich the interstellar medium with heavy elements.

A detailed knowledge of the conditions in stars is essential for an understanding of nucleosynthesis in the Universe. Mixing processes inside stars can have a great influence on the structure of a star and the elements produced through nuclear burning. Even for supernova explosions the structure of the progenitor star is of importance. In this introduction we give a brief overview of models of stellar evolution and discuss how multidimensional hydrodynamics could provide an improvement to some of the deficiencies of these models.

1.1 Stellar Evolution

The theory of stellar structure and evolution has been a cornerstone of astrophysics since its inception; stars even gave their name to this branch of physics. In order to put the methods developed and simulations performed in this thesis into context, we give a brief overview of the basic theory of stellar evolution and the numerical methods used for research in this field. This section loosely follows the detailed introduction given in Kippenhahn et al. (2013).

1.1.1 Basic Equations

The evolution of a star under the assumption of spherical symmetry and ignoring mass loss at the surface, rotation, and magnetic fields is modeled by the following set of partial differential equations (Kippenhahn et al., 2013)

$$\frac{\partial r}{\partial m} = \frac{1}{4\pi r^2 \rho}, \tag{1.1}$$

$$\frac{\partial p}{\partial m} = -\frac{Gm}{4\pi r^2}, \tag{1.2}$$

$$\frac{\partial l}{\partial m} = \epsilon_\mathrm{n} - \epsilon_\nu - c_p \frac{\partial T}{\partial t} + \frac{\delta}{\rho}\frac{\partial p}{\partial t}, \tag{1.3}$$

$$\frac{\partial T}{\partial m} = -\frac{GmT}{4\pi r^4 p}\nabla, \tag{1.4}$$

$$\frac{\partial X_i}{\partial t} = \frac{m_i}{\rho}\left(\sum_j r_{ji} - \sum_k r_{ik}\right). \tag{1.5}$$

These equations are expressed in Lagrangian coordinates using the mass variable m. The first equation defines the radial coordinate r with the density profile ρ. The second ensures that acceleration due to the pressure gradient and gravity cancel and gives rise to a state called *hydrostatic equilibrium* (see Section 2.3.1 for more information). The constant G is from Newton's law of gravitation. The next equation is derived from energy conservation. The net energy that passes through a sphere at radius r is denoted by the luminosity l. It is determined by the nuclear energy generation rate ϵ_n and the energy loss ϵ_ν through neutrinos that are generated in nuclear interactions and can freely stream out of the star. The last two terms track the change in energy due to mechanical work exerted on the neighboring mass shells. This uses the pressure p, the temperature T, the specific heat c_p, and $\delta = -\left(\frac{\partial \ln \rho}{\partial \ln T}\right)$, a derivative of the equation of state. The fourth equation describes energy transport by radiation as well as turbulent flow driven by buoyancy, called convection. The quantity ∇ is chosen according to the locally applicable mode of transport. In convectively stable regions that transport energy through radiation[1] it is given by

$$\nabla_\mathrm{rad} = \frac{3}{16\pi a c_\mathrm{light} G}\frac{\kappa l p}{m T^4}, \tag{1.6}$$

[1] This also includes thermal conductivity because it follows an equation of the same form and the radiative and conductive opacities can just be reciprocally added.

with the opacity κ of stellar matter at the local thermodynamic state and composition, the radiation constant a and the speed of light c_{light}. In regions that are convectively unstable ∇ is set to another value. In the deep interior of the star it is

$$\nabla_{\mathrm{ad}} = \left(\frac{\partial \ln T}{\partial \ln p} \right)_s , \tag{1.7}$$

which is a derivative of the equation of state at constant entropy. For convection in the envelope it is taken from some description of convection, most commonly *mixing-length theory* (see Section 1.1.2). The last equation deals with the change of composition via nuclear reactions. The composition is defined by the mass fraction X_i, i.e. the fraction of the total mass of the shell that is made up of species i. The reactions that create species i are denoted by r_{ji}; those that consume it by r_{ik}.

Equations (1.1) to (1.5) are the basic equations that are solved to calculate the evolution of a star. They are complemented with prescriptions that capture phenomena occurring on time and length scales that cannot be easily captured in the framework of the basic equations. One of these is convection, which is inherently multidimensional and cannot be expressed by a simple one-dimensional average velocity. If a region is deemed to be convectively unstable due to the Schwarzschild or Ledoux criteria (see Section 2.3.2), the composition in this region is instantaneously set to the average value, usually causing discontinuities at the edges. In the case of extremely rapid burning, a diffusive approximation for convective mixing is used instead because convection and nuclear burning occur on similar timescales. Two other effects that are added to basic equations, if necessary, are various diffusive effects (diffusion of composition, temperature, and pressure) and mass loss due stellar winds at the surface. These two are not discussed further as this thesis is concerned with hydrodynamic processes, which are mostly related to convection.

No analytic solution of this system is possible due to the complexity of the material functions (equation of state, opacities, nuclear reaction rates). That is why, for many years, numerical solutions have been computed, starting with the work of Kippenhahn et al. (1967). In the simplest case of a star in complete equilibrium, i.e. $\ddot{r} = \dot{p} = \dot{T} = 0$, the four coupled ordinary differential equations Eqs. (1.1) to (1.4) are solved for the variables r, p, l, and T for given composition and boundary conditions. A very successful method for accomplishing this numerically is the *Henyey method*, a generalized Newton–Raphson method (Henyey et al., 1964). The solution is then advanced for a small time step Δt, by computing the change in composition due to nuclear burning using Eq. (1.5), and a new solution to stellar structure equations Eqs. (1.1) to (1.4) is sought for the modified composition. This approach is

called *operator splitting*. If the model is just in hydrostatic equilibrium, i.e. the acceleration $\ddot{r} = 0$, but T and p are changing, their rate of change can be computed by taking the difference to the previous time step. The results are then used as the time derivatives on the right side of Eq. (1.3).

In evolutionary stages that are not in perfect hydrostatic equilibrium, the acceleration term $\ddot{r} = \partial v/\partial t$ cannot be neglected anymore. The momentum equation, Eq. (1.2), has to be extended with $-(\partial v/\partial t)/4\pi r^2$ on the right side, which turns the set of equations into *hydrodynamical* equations for radial motions with velocity v. While this extension is useful for dynamic phases of the life of a star, e.g. pulsations or even supernova explosions, it is still far from a complete treatment of hydrodynamical effects in a star. This is because spherical symmetry is still enforced in this one-dimensional theory. *Convection*, being the most important hydrodynamical phenomenon from the viewpoint of stellar structure, cannot be described in this picture. Convection is a turbulent flow of the stellar plasma driven by a stratification of the atmosphere that is unstable to buoyancy. Turbulence and buoyancy are inherently multidimensional effects that can only be included in the one-dimensional equations by using physically motivated prescriptions, often involving free parameters.

There are at least four main sources of errors in stellar evolution calculations. The first is due to uncertainties in the microphysical properties of the stellar plasma, especially in the opacities and reaction rates. The second is in the treatment of mass loss through stellar winds, for which mainly phenomenological models are used at the moment. The third is the treatment of rotation, especially the effect of large scale circulations and shear instabilities. The fourth, and possibly most crucial, is the treatment of convection. The most common prescription for this since the beginning of stellar evolution calculations is *mixing-length theory*. Although there have been many attempts to provide a more sophisticated theory, none of them could provide results that match the observations significantly better. Most stellar evolution codes are still using this theory today because of its simplicity and single free parameter.

1.1.2 Mixing-Length Theory

One of the most widely used versions of mixing-length theory in stellar astrophysics is that by Böhm-Vitense (1958), which we briefly introduce here based on the presentation in Kippenhahn et al. (2013) in order to put the multidimensional simulations of convection in this thesis into context.

Mixing-length theory considers the motion of convective elements surrounded by a background atmosphere. It is assumed that this element is always in pressure equilibrium with its surroundings, which is reasonable as long as its velocity is small compared to the sound speed. The central concept of mixing-

length theory is the *mixing-length* ℓ_m, which is the length an element has to travel before it is absorbed into the surroundings. Considering all elements passing through a sphere at radius r the average element has traveled a length of $\ell_\mathrm{m}/2$ when reaching r. The temperature difference from the background can be approximated linearly with

$$\frac{DT}{T} = \frac{1}{T}\frac{\partial DT}{\partial r}\frac{\ell_\mathrm{m}}{2} = (\nabla - \nabla_\mathrm{e})\frac{\ell_\mathrm{m}}{2}\frac{1}{H_p}. \tag{1.8}$$

This uses the temperature gradient of the background $\nabla = \partial \ln T / \partial \ln P$. The index "e" indicates that the quantity is computed for the convective element. The quantity $H_p = \partial r / \partial \ln P$ is the pressure scale height. The details of this notation are discussed in Section 2.3.2. The density contrast is given by

$$\frac{D\rho}{\rho} = -\delta\frac{DT}{T}, \tag{1.9}$$

with the derivative of the equation of state δ as defined for Eq. (1.3). From this the buoyancy force can be computed

$$k_r = -g\frac{D\rho}{\rho}, \tag{1.10}$$

with the local gravitational acceleration g. As k_r increases linearly with the traveled distance, on average half of the force present at radius r acted on the element during the time before. Using Eq. (1.8) this yields the total work

$$\frac{1}{2}k_r\frac{\ell_\mathrm{m}}{2} = g\delta(\nabla - \nabla_\mathrm{e})\frac{{\ell_\mathrm{m}}^2}{8H_p}. \tag{1.11}$$

Assuming further that half of this work is transformed into kinetic energy of the element while the rest is transferred to the surroundings, we can deduce a velocity of the element

$$v^2 = g\delta(\nabla - \nabla_\mathrm{e})\frac{{\ell_\mathrm{m}}^2}{8H_p}. \tag{1.12}$$

The expression for energy flux due to convective motions is

$$F_\mathrm{con} = \rho v c_p DT. \tag{1.13}$$

This can be easily understood as it is just the product of the rate of mass flux ρu and the change in internal energy due to the different temperature of the element $c_p DT$. Inserting the derived expressions for v and DT into this equations yields

$$F_\mathrm{con} = \rho c_p T\sqrt{g\delta}\frac{{\ell_\mathrm{m}}^2}{4\sqrt{2}}H_p^{-3/2}(\nabla - \nabla_\mathrm{e})^{3/2}. \tag{1.14}$$

The piece that is still missing is ∇_e. The element can change its temperature through adiabatic expansion or contraction and through radiative exchange with the surroundings. Assuming a certain surface area to volume ratio of the element the resulting expression for ∇_e is (Kippenhahn et al., 2013, Sec. 7.1)

$$\frac{\nabla_\mathrm{e} - \nabla_\mathrm{ad}}{\nabla - \nabla_\mathrm{e}} = \frac{6acT^3}{\kappa\rho^2 c_p \ell_\mathrm{m} v}. \tag{1.15}$$

We now have two equations for the three unknowns F_con, ∇, and ∇_e. To close this system, we look at expressions for the energy fluxes. From a known value of the luminosity l at a certain radius we can compute the temperature gradient necessary so that all of this energy would be transported by radiation. This gradient is

$$\nabla_\mathrm{rad} = \frac{3}{16\pi acG}\frac{\kappa l p}{m T^4}. \tag{1.16}$$

This can be used in an expression for the sum radiative flux F_rad and convective flux F_con

$$F_\mathrm{rad} + F_\mathrm{con} = \frac{4acG}{3}\frac{T^4 m}{\kappa p r^2}\nabla_\mathrm{rad}. \tag{1.17}$$

Now the actual radiative flux depends on the real temperature gradient ∇ and is given by

$$F_\mathrm{rad} = \frac{4acG}{3}\frac{T^4 m}{\kappa p r^2}\nabla. \tag{1.18}$$

Thus the convective flux is

$$F_\mathrm{con} = \frac{4acG}{3}\frac{T^4 m}{\kappa p r^2}(\nabla_\mathrm{rad} - \nabla), \tag{1.19}$$

which is the additional equation we need to close the system.

Using some algebra and educated substitutions (Kippenhahn et al., 2013, Section 7.2) this system can be reduced to a single cubic equation that has only one real (as in not complex) solution. This solution determines the temperature gradient ∇ in presence of convection.

The perhaps most obvious flaw of mixing-length theory the mixing length ℓ_m itself because it is a free parameter. It is typically determined by comparing stellar models with observations of stars, especially the Sun for which the best data are available. It is of the order of the pressure scale height H_p. This assumes that ℓ_m is universal, i.e. it has the same value for all kinds of convection in different types of stars. Another strong assumption is that all convective elements have the same size and the same shape. This is important for the amount of heat they exchange with the surroundings. From the theory

of turbulence, numerical simulations, and even laboratory experiments we know that convective elements are certainly not uniform. There is a full spectrum of sizes and shapes instead. A theory taking this into account has been developed by Canuto & Mazzitelli (1991). While it gives results that compare significantly better to observations, it is still dependent on a mixing-length parameter.

Another problem is that mixing of chemical species at convective boundaries in presence of fast nuclear reactions is usually modeled as a diffusive process with a diffusion constant derived from the convective velocity given by mixing-length theory. This is certainly not the correct picture as the simulations by Stancliffe et al. (2011) and also our own simulations in Chapter 3 indicate. Species can be advected further into non-convective regions than diffusion would permit.

The inadequate model of convection does not just influence the temperature stratification of a star, it also has an effect on other phenomena that are modeled using prescriptions in one-dimensional codes. An important one is the treatment of convective overshooting, where convective fluid elements are entering the neighboring radiative zone, causing additional mixing of chemical species. This can have an important impact on the evolution of a star. A typical treatment of convective overshooting is to increase the region in which the species are mixed by a fraction of the pressure scale height into the radiative zone.

1.2 Hydrodynamical Simulations of Stars

The presentation of the deficiencies of stellar evolution calculations due to the spherically symmetric, hydrostatic approximation leads to the natural conclusion that the models should be improved by relaxing these assumptions and moving to a more general set of equations to solve. The flow of the stellar plasma can be quite accurately described using the Navier–Stokes equations, which are derived from the principles of conservation of mass, momentum, and energy and discussed in detail in Section 2.1.1. The purpose of this section is to examine the possibilities of this approach, to give an overview of previous work on this topic by other authors, and to present what is new in the methods used in this thesis.

1.2.1 A Matter of Timescales

For estimating the feasibility of multidimensional hydrodynamical simulations of a star it is instructive to look at the timescales at which the relevant processes take place (e.g. Kippenhahn et al., 2013). The first one is the *free-fall timescale*

$$\tau_{\mathrm{ff}} = \sqrt{\frac{R}{g}} = \sqrt{\frac{R^3}{GM}}, \tag{1.20}$$

with the radius R and enclosed mass M at the point under consideration. In the second equality, Newton's law of gravitation with the gravitational constant G was used. It is the time it would take a particle at radius R to fall to the center of the star if gravitational acceleration remained constant at its initial value and pressure would not balance it. This timescale gives an estimate of how fast the star returns to hydrostatic equilibrium when the pressure profile changes. It is close to the time it takes a sound wave to cross the star. In the case of the Sun $\tau_{\mathrm{ff}} \approx 27\,\mathrm{min}$.

The next timescale of importance is the *Kelvin–Helmholtz timescale*

$$\tau_{\mathrm{KH}} = \frac{|E_{\mathrm{grav}}|}{L} \approx \frac{3}{5}\frac{GM^2}{RL}, \tag{1.21}$$

with the total potential energy of the star in its own gravitational field

$$E_{\mathrm{grav}} = -G\int_0^R \frac{m(r)4\pi r^2}{r}\rho dr, \tag{1.22}$$

and the total luminosity of the star L. The approximation lies in assuming that density is constant in the star, which is clearly not true but suffices for the purpose of estimating a timescale. The Kelvin–Helmholtz timescale of the Sun is about 1.9×10^7 years. It is the time for which the Sun could radiate energy by just using its gravitational energy. It gives an estimate for the time it takes the thermal structure to adjust when the rate of energy generation changes. This means even if we neglect nuclear energy release in a stellar model, its structure will not significantly change as long as the simulated time is well below τ_{KH}.

The last relevant timescale is the *nuclear timescale*. It is estimating how long a star can support its luminosity L by thermonuclear fusion. It is given by

$$\tau_{\mathrm{nuc}} = \frac{E_{\mathrm{nuc}}}{L}. \tag{1.23}$$

The nuclear energy release from the conversion of ^{1}H to ^{4}He is $Q = 6.3 \times 10^{18}\,\mathrm{erg/g}$, which yields $\tau_{\mathrm{nuc}} = QM_\odot/L_\odot = 10^{11}\,\mathrm{years}$. This is the

timescale at which the chemical composition of the Sun evolves and determines the lifetime of the star[2].

Comparing these three timescales for the Sun yields

$$\tau_{\mathrm{nuc}} = 5 \times 10^3 \, \tau_{\mathrm{KH}} = 2 \times 10^{15} \, \tau_{\mathrm{ff}}. \tag{1.24}$$

This shows that the different processes that influence a star act on vastly different timescales. Especially the free-fall timescale, which is connected to hydrodynamic phenomena, is so far from the other two that fully dynamic, numerical modeling of the whole lifetime of a star is firmly out of reach, save for any revolutionary advances in computer science. Even removing the necessity to track sound waves with implicit time-stepping (see Section 2.1.3) only allows us to gain a factor of the inverse Mach number at most, which is roughly 10^{-2} for convective zones. To undermine this argument with some real world numbers we look at the model of a $250\,M_\odot$ star used in Chapter 3. Its luminosity is about $2 \times 10^6\,M_\odot$ giving a nuclear timescale of 12×10^6 years.[3] The free-fall timescale is about one day. At the time we started the hydrodynamic simulation the star had an age of 2×10^6 years. The three-dimensional simulations covered a physical time of about 4 days. This means our hydrodynamical simulation covered about only *one billionth* of the lifetime of the star.

Yet the situation is not hopeless. The longest part of their life stars spend in hydrostatic hydrogen burning, which even though it involves convective regions and large scale hydrodynamic circulation, can be described very well as a steady state. The particular detailed outcome of hydrodynamic simulations is not very useful anyway because of the chaotic nature of turbulence, for which only statistical averages are meaningful. An approach that could lead to progress in stellar models is to perform hydrodynamic simulations of phenomena like convection, convective overshoot, or shear instabilities until they reach a steady state and extract better prescriptions for use in the classical stellar evolution codes.

Another area where multidimensional hydrodynamics calculations could provide substantial insight is stages of stellar evolution in which the nuclear and hydrodynamic timescale are not so extremely different. This is the case for the very late burning phases of massive stars. For example the silicon burning phase lasts only several days. This could even be covered in its entirety with

[2] The actual lifetime of stars is much shorter because events that occur in later burning stages interrupt the complete conversion of elements and eject large amounts of unburnt material. Lower mass stars eject their envelope as planetary nebulae, while higher mass stars explode as core-collapse supernovae.

[3] This uses the luminosity during core helium burning instead of core hydrogen burning as was done in the case of the Sun. Still it enough for getting an order of magnitude for the lifetime of the star.

hydrodynamics simulations. Two-dimensional simulations have shown that the star departs significantly from spherical symmetry during this phase (Arnett & Meakin, 2011) but three-dimensional studies are yet to be done.

1.2.2 Previous Work

Here we give an overview of different hydrodynamics codes that were applied to stellar evolution, focusing on their distinguishing features and the problems they were applied to.

The ANTARES code by Muthsam et al. (2010) employs high-order WENO methods to model the Euler equations. In recent work (Happenhofer et al., 2013) its capability to simulate low Mach number flows was improved using the method by Kwatra et al. (2009). It allows to split the advective and acoustic parts of the equation, handling the former using explicit methods and the latter implicitly. It also gained support for curvilinear grids to avoid the problems with small cells and grid singularities in spherical grids (Grimm-Strele et al., 2013). One of its most prominent applications is the simulation of Cepheid variables (Mundprecht et al., 2013).

The Djehuty code aspires to be a three-dimensional stellar evolution code (Bazán et al., 2003). It solves the Euler equations using radiation in the diffusion limit, nuclear reactions and neutrino losses, an equation of state for stellar matter. The gravitational potential is computed using the monopole approximation. To be able to simulate the core of a star the code employs a small Cartesian grid at the center that is connected to a spherical grid in the outer regions. Its hydrodynamic method is called *arbitrary Lagrange-Eulerian* (ALE). Lagrangian means that the grid moves along with the fluid flow where possible. As purely Lagrangian methods cause grid tangling in more than one dimension, the grid needs to be relaxed in regions where shear develops, making the method partially Eulerian there. The time-stepping method is purely explicit, which means that the code is limited to time steps that allow a sound wave to cross only one grid cell per time step. This is a severe limitation in the case of low Mach number flows, which are predominant in the stellar core in many phases of stellar evolution. Also, to our knowledge, no special precautions have been taken to avoid the huge numerical dissipation in low Mach number flows (see Section 2.2). Stancliffe et al. (2011) performed 3D simulations of proton ingestion into a convective shell in an asymptotic giant branch star. They observe greatly enhanced convective velocities and mixing compared to the 1D predictions from mixing-length theory.

A series of simulations of convection in stellar interiors and of the late burning phases of massive stars has been performed using the PROMPI code, a parallelized version of the explicit hydrodynamics code PROMETHEUS

(Fryxell et al., 1989). PROMETHEUS combines the Piecewise-Parabolic Method (PPM) of Colella & Woodward (1984) with a nuclear reaction network. They simulated carbon and oxygen shell burning in two and three spatial dimensions and analyzed the behavior of turbulent convection in these (Meakin & Arnett, 2006, 2007b). Arnett & Meakin (2011) used the same code for a two-dimensional simulation of the silicon burning phase. The main limitation of these simulations is that they only covered rather short timescales due to the limitations of explicit time-stepping. Meakin & Arnett (2007a) compare their compressible simulations of oxygen shell burning using PROMPI to anelastic simulations by Kuhlen et al. (2003) which used the numerical methods of Glatzmaier (1984). They found significant deviations from the horizontal averages that were absent in the anelastic simulations due to the physics not included in this formulation.

Another descendant of the PROMETHEUS code is the HERAKLES code by Kifonidis et al. (2003), which includes some improvements to PPM, the HLLE Riemann solver, and the Consistent Multifluid Advection scheme (Plewa & Müller, 1999). It was used in a series of papers on the core helium flash (Mocák et al., 2008, 2009, 2010). The main result of these studies was that the behavior of the flash is not explosive, the deviation from mixing-length theory is not huge in general but especially the boundaries of convective zone behave differently which influences mixing of chemical species in the models. Also here the use of an explicit time-stepping scheme limited the physical time that could be covered in the simulations.

Herwig et al. (2011) also use an enhanced version of PPM for the simulation of the ingestion of ^{1}H into ^{12}C-enriched convective shells. A distinguishing property of these simulations is the use of the *Piecewise-Parabolic Boltzmann* (PPB) scheme by Woodward et al. (2008), which greatly improves the accuracy of species advection. In recent work the group investigates the numerical accuracy of their simulations and find that they need at least 1024^3 grid cells to achieve convergence, which they verified in 1536^3 simulations (Woodward et al., 2013).

In the work by Viallet et al. (2011, 2013a) a hydrodynamics code with implicit time-stepping called MUSIC was developed. The implicit nature of this code allows it to cover longer physical times than explicit codes. It uses a staggered grid, where the values of certain quantities are stored at the cell interfaces instead of the cell centers. This improves the codes behavior at low Mach numbers. It was applied to the envelopes of red giants (Viallet et al., 2011), for which also the properties of convection were studied (Viallet et al., 2013b). The code includes radiation in the diffusion limit but lacks a reaction network for the treatment of nuclear reactions so far.

Freytag et al. (2012) developed the CO5BOLD code which simulates the equations of hydrodynamics or magnetohydrodynamics coupled with non-local

radiative transfer. The group uses the code in *box-in-a-star* simulations that take out a small part of the outer atmosphere of a star in plane-parallel approximation, as well as in *star-in-a-box* simulations, where a whole star is put on a Cartesian grid. The main focus of their investigations are the outer regions of stars that are becoming optically thin. This is where non-local radiative transfer becomes important and explicit time-stepping is efficient due to the relatively high Mach numbers. This is a very different regime from the stellar interiors that are the target of the work in this thesis.

A similar code also used in modeling the outer regions of stars including radiative transfer and magnetohydrodynamics is the STAGGER code by Nordlund & Galsgaard (1995). A recent application of this code was the construction of a grid of three-dimensional stellar atmosphere models by Magic et al. (2013).

The MAESTRO code (Almgren et al., 2007; Nonaka et al., 2010) follows an approach different from all other codes mentioned above. It does not solve the compressible Euler equations but a modified set of equations that was derived to filter out sound waves while keeping the correct asymptotic behavior in the low Mach number case. This limits to code strictly to application in the low Mach number regime. It can cover long timescales because it does not resolve sound waves. Its primary astrophysical application so far was the simulation of convection in the simmering phase of white dwarfs prior to the ignition of a Type Ia supernova (e.g. Nonaka et al., 2012). For this reason the code also includes a nuclear reaction network. It does not yet include radiative diffusion. Recently, Gilet et al. (2013) simulated core convection in a $15\,M_\odot$ star using MAESTRO. Their results are qualitatively similar to the results of Kuhlen et al. (2003) and Meakin & Arnett (2007b) but they argue that the latter results were influenced by the small size of the computational domain.

1.2.3 The Hydrodynamics Code SLH

Having seen that hydrodynamics simulations of stellar interiors are a valuable contribution for stellar astrophysics and that the inclusion of nuclear reactions would improve the range of applicability, we need to chose the right basis for this work. The *Seven-League Hydro* (SLH) code offers many of the features we need. It was originally developed by Miczek (2013) during his PhD thesis.[4] It is a finite-volume hydrodynamics code that simulates the Euler equations in one, two, or three spatial dimensions. One of its most distinguishing features is a special *preconditioned Roe solver* for the computation of the fluxes that allows it to simulate flows at low Mach numbers ($\lesssim 10^{-3}$) while still retaining all

[4] In Miczek (2013) the code is called by its original name *Low Mach number Hydro Code* (LHC). Due to fact that there is a particle collider by the same name, it was renamed to SLH.

terms of the Euler equations and thereby the ability to have regions of higher Mach number on the same grid (see Section 2.2). Another advantage over many other multidimensional hydrodynamics codes is its use of *implicit time stepping*. This allows it to use time steps that are much larger than the ones in explicit schemes, which are bound to the so-called CFL criterion for numerical stability (see Section 2.1.3). Apart from these two major points the code has a very flexible and modular design and the components are implemented in the most general way still practicable. For example the reconstruction method for higher order spatial accuracy can be chosen independent of numerical flux function. In fact a whole suite of flux functions that are well suited for low Mach numbers are implemented in the code.

For the implicit time-stepping a non-linear system of equations is solved using the Newton–Raphson method. At every step of the Newton iteration the solution of a linear system needs to be found. This is done using *Krylov subspace methods*. The computational efficiency of these methods strongly depends on the properties of the matrix being inverted and the type of parallel compute cluster being used. This is why SLH employs its own linear solver framework with a host of different methods and preconditioning techniques to ensure efficient computation after some initial tuning.

The code takes a very general approach to geometry as well. Instead of the typically used Cartesian or spherical grids it uses a general *curvilinear grid* that can be adjusted to almost any geometry without being affected by coordinate singularities (see Section 2.1.2). This can be achieved without sacrificing much of the simplicity of Cartesian grids as only metric terms have to be inserted at a few locations.

An *arbitrary equation of state* can be used, provided certain derivatives can be computed, either analytically or numerically. As the equation of state generally also depends on the local composition, SLH implements a treatment of the chemical species as *active scalars* in addition to the operator-split method used in most other hydrodynamics codes. The code also includes radiation in the diffusion limit and thermal conduction, which is mathematically equivalent.

One issue raised by Miczek (2013) is the occurrence of an instability in hydrostatic atmospheres that are convectively stable (see Section 2.4.1). This thesis discusses a possible numerical cause of the instability in Section 2.4.2 and provides a fix in Sections 2.4.3 and 2.4.4 that works in the cases we investigated so far but definitely needs further study in future work.

1.3 Outline of the Thesis

Section 2.1 introduces the basic equations of hydrodynamics and gives an overview of the numerical scheme used in the SLH code. This is expanded in Section 2.2 by a discussion of the principal problems of numerical hydrodynamics as low Mach numbers and how they are solved in SLH. Section 2.2.3 shows new tests of the numerical viscosity of the code. Section 2.3 discusses hydrostatic equilibrium and the convective stability of atmospheres, which is crucial to stellar models. In Section 2.4 we give an analysis of an instability that causes convectively stable atmospheres to become unstable as soon as a low Mach number method is used. Furthermore, a possible cause and solution is presented. Section 2.5 introduces the differential equations for the nuclear reaction network, discusses their solution with and without coupling to hydrodynamics and compares the result of using different coupling methods. A first application of the methods described is presented in Chapter 3, where a profile of a massive primordial star is mapped to a three-dimensional simulation grid to investigate the interplay of convective overshooting and enhanced reaction rates due to the mixing of ^{12}C and ^{1}H.

2 Physical Concepts and Computational Methods

2.1 Multidimensional Hydrodynamics Simulations

As noted in the last chapter the two major caveats of the traditional formulation of stellar evolution are the reduction to spherical symmetry and the removal of all time-dependent terms from the Navier–Stokes equations. The natural solution to this problem is to extend the set of model equations to three spatial dimensions and to retain all time-dependent terms in the equations. This section will discuss the properties of the full set of hydrodynamic equations and necessary extensions for their use in modeling stellar interiors.

2.1.1 The Three-Dimensional Navier–Stokes Equations

For their treatment in the context of finite-volume hydrodynamic codes it is convenient to express the Navier–Stokes equations in terms of the conservative variables $\boldsymbol{U}$ given by

$$\boldsymbol{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \\ \rho X_i \end{pmatrix}. \tag{2.1}$$

Here, ρ is density, ρu, ρv, ρw are the components of momentum in x-, y-, and z-direction, and ρE is the total energy, i.e. kinetic and internal energy, per volume. X_i is the mass fraction of species i; ρX_i is the density of that species. This term is used as a placeholder for all involved species. In practice, it is replaced by as many components as there are species in the model. This choice of variables is especially useful for numerical considerations because the equations take a flux form. From this, it can be deduced that the integral of each component of $\boldsymbol{U}$ over the whole domain is conserved except for fluxes through the boundaries and source terms.

Following Toro (2009) the Navier–Stokes equations including all relevant source terms can be conveniently expressed in the form

$$\partial_t \boldsymbol{U} + \partial_x \boldsymbol{F}(\boldsymbol{U}) + \partial_y \boldsymbol{G}(\boldsymbol{U}) + \partial_z \boldsymbol{H}(\boldsymbol{U}) = \partial_x \boldsymbol{F}^d + \partial_y \boldsymbol{G}^d + \partial_z \boldsymbol{H}^d + \boldsymbol{S} \quad (2.2)$$

with the hydrodynamical terms

$$\boldsymbol{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(\rho E + p) \\ \rho u X_i \end{pmatrix} \quad \boldsymbol{G} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(\rho E + p) \\ \rho v X_i \end{pmatrix}, \quad \boldsymbol{H} = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(\rho E + p) \\ \rho w X_i \end{pmatrix}, \quad (2.3)$$

the diffusive terms

$$\boldsymbol{F}^d = (0, \tau^{xx}, \tau^{xy}, \tau^{xz}, u\tau^{xx} + v\tau^{xy} + w\tau^{xz} + K\partial_x T, 0)^T, \quad (2.4)$$

$$\boldsymbol{G}^d = (0, \tau^{yx}, \tau^{yy}, \tau^{yz}, u\tau^{yx} + v\tau^{yy} + w\tau^{yz} + K\partial_y T, 0)^T, \quad (2.5)$$

$$\boldsymbol{H}^d = (0, \tau^{zx}, \tau^{zy}, \tau^{zz}, u\tau^{zx} + v\tau^{yy} + w\tau^{yz} + K\partial_z T, 0)^T, \quad (2.6)$$

and the gravity and nuclear burning source terms

$$\boldsymbol{S} = \begin{pmatrix} 0 \\ \rho g_x \\ \rho g_y \\ \rho g_z \\ \rho \boldsymbol{u} \cdot \boldsymbol{g} + \rho \sum_i e_{\mathrm{nuc},i} \dot{X}_i \\ \rho \dot{X}_i \end{pmatrix}. \quad (2.7)$$

Here, K is the thermal conductivity, which also includes radiation in the diffusion limit as the two phenomena are mathematically equivalent. The nuclear energy release $e_{\mathrm{nuc},i} = N_\mathrm{A} m_i {c_\mathrm{light}}^2 / A_i$ is determined by the rest mass per nucleon m_i of species i and the speed of light c_light. The atomic mass number A_i and Avogadro's constant N_A are used to convert from mass fraction to number of nuclei. The vector $\boldsymbol{g}$ is gravitational acceleration. In principle, it depends on the mass distribution inside and outside of the considered domain. In the scope of this thesis we only consider problems for which $\boldsymbol{g}$ is solely determined by an external mass distribution, i.e. the core of the star, which is outside of the grid. So gravity is assumed to be constant in time. If, in the future, problems that include the stellar core are investigated, this assumption is not valid anymore. The gravitational potential would then have to be

computed and $\boldsymbol{g}$ be updated accordingly. The components $\tau^{xx}, \ldots$ define the *viscous stress tensor*. They are given by

$$\tau^{xx} = \frac{4}{3}\eta\partial_x u - \frac{2}{3}\eta(\partial_y v + \partial_z w) + \eta_\mathrm{b}(\partial_x u + \partial_y v + \partial_z w), \tag{2.8}$$

$$\tau^{yy} = \frac{4}{3}\eta\partial_y v - \frac{2}{3}\eta(\partial_z w + \partial_x u) + \eta_\mathrm{b}(\partial_x u + \partial_y v + \partial_z w), \tag{2.9}$$

$$\tau^{zz} = \frac{4}{3}\eta\partial_z w - \frac{2}{3}\eta(\partial_x u + \partial_y v) + \eta_\mathrm{b}(\partial_x u + \partial_y v + \partial_z w), \tag{2.10}$$

$$\tau^{xy} = \tau^{yx} = \eta(\partial_y u + \partial_x v), \tag{2.11}$$

$$\tau^{yz} = \tau^{zy} = \eta(\partial_z v + \partial_y w), \tag{2.12}$$

$$\tau^{zx} = \tau^{xz} = \eta(\partial_x w + \partial_z u). \tag{2.13}$$

This introduces the *shear viscosity* η and *bulk viscosity* η_b. In a stellar environment η has a value of the order of $1\,\mathrm{g\,(cm\,s)^{-1}}$ (Kippenhahn et al., 2013, Sec. 45.1). For a fully ionized, non-relativistic gas η_b is 0 (Landau & Lifshitz, 1981, §8). This information makes an estimate of the importance of viscous terms possible. Usually one introduces the *Reynolds number* Re to estimate ratio of inertial to viscous forces. It is defined by

$$\mathrm{Re} = \frac{\rho u L}{\eta}. \tag{2.14}$$

To give a rough estimate of Re we can use values of a Population III star model studied in Chapter 3. Using $\rho = 10\,\mathrm{g\,cm^{-3}}$, $u = 8 \times 10^5\,\mathrm{cm\,s^{-1}}$, and $L = 10^{11}\,\mathrm{cm}$ we obtain $\mathrm{Re} = 8 \times 10^{17}$. If the Sun is used to estimate these parameters instead, we arrive at a less extreme value of $\mathrm{Re} = 3 \times 10^{12}$ (with values for the solar radiative zone: $\rho = 0.2\,\mathrm{g\,cm^{-3}}$, $M = 10^{-5}$, $T = 2 \times 10^6\,\mathrm{K}$, $L = 0.5\,R_\odot = 6.5 \times 10^{10}\,\mathrm{cm}$, $\mu = 0.5$, $\gamma = 5/3$, $u = M\sqrt{\gamma\mathcal{R}T/\mu}$).

The Euler Equations

Even the more conservative estimate of Re is in a regime that is very far from what can be reached using direct numerical simulations. The reason is that the discretization introduces numerical viscosity that is orders of magnitude higher than the physical viscosity. Simulations at the current limit of computational feasibility reach a numerical Reynolds number of less than 10^4 (cf. Section 2.2.3). This means that inclusion of physical viscosity will not benefit simulations of stellar systems. This allows us to leave out the stress tensor from the Navier–Stokes equations. One should bear in mind that all simulations of stellar environments have an unphysically high numerical viscosity, the full effect of which cannot be easily estimated. In the future, a comprehensive study

of the properties of turbulence in simulations using the SLH code could help to improve our understanding of the errors introduced by physical viscosity. A suitable subgrid-scale model could possibly be constructed to improve the situation.

These simplifications result in the *Euler equations* with thermal conduction, gravity, and nuclear burning. They can be expressed in the form

$$\partial_t \boldsymbol{U} + \partial_x \boldsymbol{F} + \partial_y \boldsymbol{G} + \partial_z \boldsymbol{H} = \boldsymbol{S}, \tag{2.15}$$

$$\boldsymbol{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(\rho E + p) - K\partial_x T \\ \rho u X_i \end{pmatrix}, \qquad \boldsymbol{G} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(\rho E + p) - K\partial_y T \\ \rho v X_i \end{pmatrix},$$
$$\boldsymbol{H} = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(\rho E + p) - K\partial_z T \\ \rho w X_i \end{pmatrix}, \qquad \boldsymbol{S} = \begin{pmatrix} 0 \\ \rho g_x \\ \rho g_y \\ \rho g_z \\ \rho \boldsymbol{u} \cdot \boldsymbol{g} + \rho \sum_i e_{\mathrm{nuc},i} \dot{X}_i \\ \rho \dot{X}_i \end{pmatrix}. \tag{2.16}$$

This constitutes the full system of partial differential equations that needs to be solved for modeling stellar interiors. It is valid as long as the medium is optically thick (high opacity). At lower opacities radiation can no longer be treated as a purely diffusive process and must be addressed using a more elaborate method. This is only necessary for the outer regions of a star, which are not the topic of this work.

Equation of State

The Euler equations, Eqs. (2.15) and (2.16), need to be complemented by a few microphysical quantities. One is the change in composition $\dot{X}_i$, which is determined by nuclear network calculations. These are detailed in Section 2.5. Other quantities to be determined are pressure p and temperature T. These are given by the *equation of state* (EoS). The precise choice of EoS depends on the state of the system being modeled. The simplest choice is an *ideal gas*. Its pressure and temperature are given by

$$p_{\mathrm{ideal}} = \rho \mathcal{R} \frac{T}{\mu}, \quad e_{\mathrm{ideal}} = \frac{1}{\gamma - 1} \mathcal{R} \frac{T}{\mu}. \tag{2.17}$$

This introduces $\mathcal{R} = 8.314\,462\,1 \times 10^7\,\mathrm{erg\,(mol\,K)^{-1}}$, the universal gas constant. The adiabatic constant γ takes a value of $5/3$ for a monoatomic gas. The monoatomic gas is appropriate here because the plasma in stellar interiors is fully ionized due to the high temperature and therefore cannot form molecules.[1] Full ionization is also important for computing the mean molecular weight μ as it is largely dominated by the contribution from electrons. It is given by

$$\mu = \frac{1}{\sum_i X_i \frac{1+Z_i}{\mu_i}}, \tag{2.18}$$

with the mean molecular weight μ_i and charge number Z_i of species i. This formula shows that μ only covers a small range of values in the case of a fully ionized gas. Pure hydrogen has a μ of $1/2$, for ^{4}He $\mu = 4/3$. The value approaches the ratio of nucleon number A_i to charge number Z_i, which is roughly 2 for most low to intermediate mass elements except hydrogen.

In high temperature environments a significant fraction of energy is stored in the radiation field. Its pressure and energy are given by

$$p_\mathrm{rad} = \frac{4\sigma T^4}{3c_\mathrm{light}}, \quad e_\mathrm{rad} = \frac{3p_\mathrm{rad}}{\rho}. \tag{2.19}$$

This uses the Stefan–Boltzmann constant σ. This description is valid as long as radiation is in thermal equilibrium with matter. This is true in essentially the same regions in which the diffusion limit of radiation is applicable.

In high-density regions, such as stellar cores in late stages of evolution, *electron degeneracy* can also play a role. The detailed expressions for its pressure and energy using Fermi integrals can be found in the work of Timmes & Arnett (1999). These contributions are also implemented in the simulation code used in this work but are switched off because their influence is negligible for the problems addressed here.

For use in the SLH code the *Helmholtz equation of state* by Timmes & Swesty (2000) was adopted. In this approach all thermodynamic quantities are computed as derivatives of the Helmholtz free energy. This ensures thermodynamic consistency even when tabulated values are used as in the case for the degenerate electron term.

All the components of the equation of state given here have the form $p(\rho, T, X_i)$ and $e(\rho, T, X_i)$. This is because they are in thermal equilibrium, i.e. they exchanged energy until they have reached a common temperature. A problem with this is that in hydrodynamical calculations only the total energy

[1] The assumption of full ionization starts to break down at temperatures below 10^7 K. (e.g Paxton et al., 2011, Fig. 1) A more appropriate EoS including partial ionization should be used in these regions, e.g. the one by Rogers & Nayfonov (2002).

is provided as a component of the conservative variables $\boldsymbol{U}$. It is not easily possible to give the individual contributions to p and T explicitly as a function of total energy because the distribution of energy among them is not a priori clear. So it is not enough to be able to invert each component separately but the sum has to be inverted as a whole. In numerical calculations this problem is solved by using the Newton–Raphson method. To ensure convergence it is sufficient to use the ideal gas law for the initial guess.

2.1.2 Finite-Volume Schemes

The special form of the Euler equations, Eq. (2.15), becomes evident when they are integrated over a volume $\Omega_{i,j,k}$, defined by the boundaries $[x_{i-1/2}, x_{i+1/2}]$, $[y_{j-1/2}, y_{j+1/2}]$, and $[z_{k-1/2}, z_{k+1/2}]$. This introduces a common notation for indices on a computational grid. Integer-valued indices refer to cell centers while half-valued indices refer to the cell faces. The resulting equation is

$$\partial_t \boldsymbol{U}_{i.j.k} + {V_{i,j,k}}^{-1} \left(\int_{\Omega_{i,j,k}} \partial_x \boldsymbol{F} + \int_{\Omega_{i,j,k}} \partial_y \boldsymbol{G} + \int_{\Omega_{i,j,k}} \partial_z \boldsymbol{H} \right) = \boldsymbol{S}_{i,j,k}. \quad (2.20)$$

This equation was divided by the cell volume $V_{i,j,k}$ and the integral and derivative operator were exchanged in the first term. The latter is valid because we start from the strong form of the differential equation which implies the differentiability of $\boldsymbol{U}$. The notation $\boldsymbol{U}_{i,j,k}$ denotes the average of $\boldsymbol{U}$ over the volume $\Omega_{i.j.k}$. Using the divergence theorem the volume integral over the derivatives of the flux functions can be replaced by an integral over the respective surface of $\Omega_{i,j,k}$. This yields

$$\begin{aligned} \partial_t \boldsymbol{U}_{i.j.k} + {V_{i,j,k}}^{-1} (& \boldsymbol{F}_{i+1/2,j,k} - \boldsymbol{F}_{i-1/2,j,k} \\ & + \boldsymbol{G}_{i,j+1/2,k} - \boldsymbol{G}_{i,j-1/2,k} \\ & + \boldsymbol{H}_{i,j,k+1/2} - \boldsymbol{H}_{i,j,k-1/2}) = \boldsymbol{S}_{i,j,k}. \end{aligned} \quad (2.21)$$

This directly shows the conservative nature of this kind of discretization. The conservative variables $\boldsymbol{U}$ only change through fluxes through the interfaces of a grid cell and through the cell-averaged source terms $\boldsymbol{S}_{i,j,k}$. Thus the sum of all $\boldsymbol{U}_{i,j,k}$ is conserved except for fluxes through the boundaries and the effect of the source terms. This approach is called *finite-volume discretization*.

The precise method of determining the flux at the interface, e.g. $\boldsymbol{F}_{i,j,k}$, is a major distinguishing factor of numerical hydro schemes. A widespread class of schemes uses the solution to the *Riemann problem* to compute the flux at the interface. The Riemann problem is the initial value problem for a conservation

law, in this case the one-dimensional Euler equations, with initial conditions (e.g. Toro, 2009, Sec. 4.1)

$$\boldsymbol{U}(x,0) = \begin{cases} \boldsymbol{U}_\mathrm{L} & \text{for } x < 0, \\ \boldsymbol{U}_\mathrm{R} & \text{for } x > 0. \end{cases} \tag{2.22}$$

The quantities $\boldsymbol{U}_\mathrm{L}$ and $\boldsymbol{U}_\mathrm{R}$ are the conservative variables left and right of the interface for which the flux is to be computed. The exact solution $\boldsymbol{U}_{1/2}$ to the problem can be used to construct the well-known *Godunov method*. Here, the flux at the cell interface is given by $\boldsymbol{F}(\boldsymbol{U}_{1/2}(0))$.

It is also possible to use approximate solutions to the Riemann problem instead of using the exact solution. For the work in this thesis the approximate Riemann solver due to Roe (1981) is of great importance because it can be modified to treat flows at low Mach number very accurately. These modifications are discussed in Section 2.2. In the Roe scheme the flux is given by

$$\boldsymbol{F}_{1/2} = \frac{1}{2}\left(\boldsymbol{F}(\boldsymbol{U}_L) + \boldsymbol{F}(\boldsymbol{U}_R) - |A_\mathrm{roe}|(\boldsymbol{U}_R - \boldsymbol{U}_L)\right). \tag{2.23}$$

This uses the Jacobian matrix A_roe of the flux function $\boldsymbol{F}$. The subscript "roe" means that it is evaluated at a specific average state of $\boldsymbol{U}_\mathrm{L}$ and $\boldsymbol{U}_\mathrm{R}$. The absolute value of matrix means that the absolute values of each eigenvalue are taken. A full presentation of the scheme can be found, for example, in Toro (2009, Chapter 11).

A very different approach is pursued in the *advection upstream splitting method* (AUSM) of Liou & Steffen (1993). Here, the flux is split into a pressure part, comprising just the pressure term in the momentum component, and an advective part with all the remaining terms of the flux function.

$$\boldsymbol{F}_{1/2} = \dot{m}_{1/2}\boldsymbol{\psi}_{L/R} + \boldsymbol{P}_{1/2}, \quad \boldsymbol{\psi} = (1, u, H)^T, \tag{2.24}$$

introducing the total specific enthalpy $H = E + p/\rho$. Depending on the sign of the mass flux $\dot{m}_{1/2}$, $\boldsymbol{\psi}$ is chosen based on values from the right or left. The challenge in this scheme lies in the proper choice of $\dot{m}_{1/2}$ and pressure flux $\boldsymbol{P}_{1/2}$. These choices are described in the original publication (Liou & Steffen, 1993) and the following ones about improvements to the scheme called AUSM$^+$ (Liou, 1996) and AUSM$^+$-up (Liou, 2006). The last of these schemes shows the proper behavior for low Mach number flows, as discussed in Section 2.2.

Grid Geometry

It should be noted that the discussion in this section assumes a Cartesian computational grid. This is enough to illustrate the basic principles and also

well suited for the astrophysical problems addressed in this thesis. Other grid geometries are beneficial for other types of problems, e.g. those involving spherical symmetry. The simulation code SLH has a very general implementation of geometries. The underlying idea is that there is a map between a uniformly spaced, Cartesian grid, called the *computational grid*, and the physical grid. This allows us to use arbitrary curvilinear grids. The only limitation is that the grids need to be structured, i.e. each grid cell needs to have exactly two neighbors per dimension. This offers greater flexibility than traditional spherical coordinate systems because the grid can really be adjusted to the physical object. For example, the grid can be aligned along the isobars of a rotating star, which are not perfectly spherical. Another advantage is that coordinate singularities at the poles and the origin can be avoided by choosing a so-called *cubed sphere* grid. One suggestion for such a grid by Calhoun et al. (2008) is depicted in Fig. 2.1. This approach to geometry has the advantage that most of the implementation of the hydrodynamical schemes is identical to that in Cartesian codes. The properties of geometry mainly show up in the computation of the metric terms. The details of this approach are described by Kifonidis & Müller (2012) and Miczek (2013, Sec. 3.3).

2.1.3 Time-Stepping

After performing the spatial discretization as described in Section 2.1.2, the Euler equations can be rewritten as an ordinary differential equation of the form

$$\frac{\partial \boldsymbol{U}_{i,j,k}}{\partial t} + \boldsymbol{R}_{i,j,k}(\boldsymbol{U}) = 0. \tag{2.25}$$

The function $\boldsymbol{R}_{i,j,k}$ is called the spatial residual. The separation of the spatial and temporal derivatives is known as the *method of lines*. It allows us to flexibly choose an appropriate time-stepping method without changing anything in the discretization of the hydrodynamical fluxes. This approach is different from the one taken in another wide-spread method for fluid dynamics, the Piecewise Parabolic Method (PPM) of Colella & Woodward (1984). Here, the higher order accuracy in time is achieved by averaging over the reconstructed values of the fluxes. This has the problem that an extension to even higher order accuracy is not straight-forward and cell-centered source terms (like nuclear burning) are only integrated first-order accurate. Also, it is difficult to extend this method to implicit time stepping because the averaging would include several cells which makes the scheme unstable. Fryxell et al. (1986) report that it is possible to construct a stable, implicit PPM by introducing an intermediate time step. This is conceptually similar to the Runge–Kutta methods employed in the SLH code. In the design of SLH, the method of lines

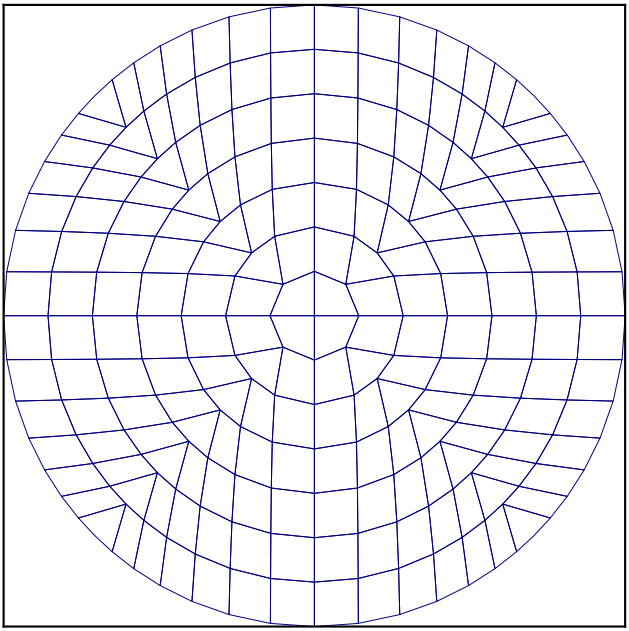


Figure 2.1: A curvilinear grid that is adapted to spherical geometry while avoiding the singularity at the center. It was suggested by Calhoun et al. (2008). The cell centers are aligned to be at the same radius, except for the diagonal cells which are located at a marginally larger radius.

approach was chosen because of its flexibility and conceptual simplicity. The details of time stepping are described by Miczek (2013). Here, we give a quick overview of explicit and implicit methods and discuss them with respect to the time integration of source terms.

The astrophysical phenomena in stellar evolution whose modeling we try to improve using multidimensional simulations typically occur at low Mach numbers (10^{-2} to 10^{-4}). This is problematic if traditional hydrodynamics codes are used that perform the time integration explicitly. The reason is the Courant–Friedrichs–Lewy (CFL) condition for stability of solution schemes for partial differential equations (Courant et al., 1928). It states that the physical domain of dependence of a grid point at the new time step must not be larger than its numerical domain of dependence. It is a *necessary* condition for stability but not sufficient. The specific time-stepping method being used can be subject to even tighter constraints on the time step.

The original CFL condition depends on the largest signal speed in the system. Its determination in case of the Euler equations is complicated as it depends on the local solutions to the Riemann problem. Typically, a simplified criterion is used instead. It states that the time step Δt must fulfill

$$\Delta t \leq \mathrm{CFL}_{uc} \min_{q \in \{u,v,w\}} \frac{\Delta x}{|q| + c}. \tag{2.26}$$

The constant CFL_{uc} depends on the specific temporal and spatial discretization being used. It is typically of the order of one. Equation (2.26) shows that the CFL limited time step is constant when reducing the velocity of the flow to a low Mach number M because

$$|q| + c = c(|M| + 1) \approx c. \tag{2.27}$$

This is undesirable because the physical processes of interest occur at the timescale of the fluid velocity. Resolving sound waves is not important as further explained in Section 2.2. The following sections discuss time-stepping methods that avoid this restriction.

Explicit Time-Stepping

The stability constraint given by the CFL condition is valid for all *explicit* time-stepping methods. The difference between explicit and implicit methods

can be understood in the context of a general s-stage Runge–Kutta method and its corresponding Butcher tableau.

$$\begin{aligned} \boldsymbol{U}_{i,j,k}^{(q)} &= \Delta t\, \boldsymbol{R}\left(\boldsymbol{U}^n + \sum_{l=1}^{s} a_{ql}\boldsymbol{U}^{(l)}, t^n + c_q \Delta t\right), \\ \boldsymbol{U}_{i,j,k}^{n+1} &= \sum_{l=1}^{s} b_l \boldsymbol{U}_{i,j,k}^{(l)}, \end{aligned} \tag{2.28}$$

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array} . \tag{2.29}$$

Here, $\boldsymbol{R}(\boldsymbol{U}, t)$ is the spatial residual from Eq. (2.25) including the hydrodynamic fluxes and the other source terms. In general, it also has an explicit time dependence, although this is not the case for the reactive Euler equations. Note that the subscript of $\boldsymbol{U}_{i,j,k}$ denotes the position on the spatial grid. No subscript means all grid cells at the specified time. The values $\boldsymbol{U}^{(q)}$ are intermediate evaluations of the conservative variables between the old time step $\boldsymbol{U}^n$ and the new time step $\boldsymbol{U}^{n+1}$. In the general case, each $\boldsymbol{U}^{(q)}$ depends on all $\boldsymbol{U}^{(i)}$ including $\boldsymbol{U}^{(q)}$ itself. This is a coupled system of s non-linear, implicit equations for $\boldsymbol{U}^{(i)}$. A method is called explicit if every $\boldsymbol{U}_{i,j,k}^{(q)}$ can be computed by

$$\boldsymbol{U}_{i,j,k}^{(1)} = \Delta t\, \boldsymbol{R}\left(\boldsymbol{U}^n, t^n + c_1 \Delta t\right), \tag{2.30}$$

$$\boldsymbol{U}_{i,j,k}^{(q)} = \Delta t\, \boldsymbol{R}\left(\boldsymbol{U}^n + \sum_{l=1}^{q-1} a_{ql}\boldsymbol{U}^{(l)}, t^n + c_q \Delta t\right) \quad \text{for } q > 1. \tag{2.31}$$

In the picture of the Butcher tableau this means that the values a_{ql} form a strictly lower triangular matrix, i.e. the values on the diagonal and in the upper right part are zero. This system can be solved just by s successive evaluations of $\boldsymbol{R}$ without solving a single non-linear equation. This is very efficient in terms of computation (no non-linear system needs to be solved using iterative methods) and memory (only s times the size of $\boldsymbol{U}$ is needed instead of a large Jacobian matrix for the non-linear solver). The caveat is that all explicit methods are subject to the CFL condition as the value of $\boldsymbol{U}_{i,j,k}^{n+1}$ only depends on the values of the cells connected by the stencil of the

spatial discretization. The size of this stencil is two cells in every direction at most for the usual reconstruction methods, i.e. constant, linear, and parabolic reconstruction. The two explicit methods implemented in SLH are the forward Euler method and RK3 (Shu & Osher, 1988). While forward Euler is only first-order accurate and therefore used only for testing, RK3 is third-order accurate and also has the Total Variation Diminishing (TVD) property (e.g. Toro, 2009, Chapter 13). RK3 is well suited for explicit calculations but limited to the CFL time step.

Implicit Time-Stepping

A way to avoid to the time step restriction of explicit methods is the use of *implicit* methods. A method is called implicit if has at least one value on the diagonal or in the upper right part of the Butcher tableau. This implies that at least one non-linear equation has to be solved for a time step. This enlarges the numerical domain of dependence to the whole computational domain, thereby removing the restrictions posed by the CFL condition. The reason for this is that the value in one cell at the new time step depends on the new values of its neighbors connected by the stencil. These values depend on their neighbors in turn. This indirectly connects the whole domain. In the most general form of an implicit Runge–Kutta method the non-linear system has not only to be solved for one set of conservative variables but for s sets, s being the number of stages. This would enlarge the Jacobian matrix in the non-linear solution by a factor of s, which is very undesirable because memory is already a limiting factor in the size of 3D simulations. Fortunately, there is a class of implicit Runge–Kutta methods that only requires the solution of a non-linear equation for one set of conservative variables per stage, the so-called *Diagonally Implicit Runge–Kutta* (DIRK) methods (Alexander, 1977). Their Butcher tableau has the form

$$\begin{array}{c|cccc} c_1 & a_{11} & 0 & \cdots & 0 \\ c_2 & a_{21} & a_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array} . \tag{2.32}$$

Here, the individual stages can be evaluated successively; at each stage the number of unknowns is only a single set of conservative variables per grid cell. A subclass of the DIRK methods are the *Singly Diagonally Implicit Runge–Kutta* (SDIRK) methods, which are constructed so that $a_{11} = a_{22} = \cdots = a_{ss}$. This would enable us to reuse the LU decomposition for the solution of the non-linear system at every stage and only compute it once. Unfortunately, this

advantage cannot be used in the context of large hydrodynamics simulations as the LU decomposition of the Jacobian matrix is not sparse and cannot be stored in memory.

A subclass of the DIRK methods that saves computational resources are the Explicit, Singly Diagonally Implicit Runge–Kutta (ESDIRK) methods. These are like the SDIRK methods but their first diagonal value is zero. The advantage here is that the first stage can be computed without solving a non-linear system; it is explicit. The method as a whole is still implicit, so it is not subject to the CFL condition.

The ESDIRK methods that are implemented in SLH are ESDIRK23 (Hosea & Shampine, 1996), ESDIRK34 , ESDIRK46 , and ESDIRK58 (all from Kennedy & Carpenter, 2001). The first number is the formal order of the method, the second is the number of Runge–Kutta stages, e.g. ESDIRK58 is an eight stage Runge–Kutta method, whose error converges with Δt^5, for sufficiently small Δt. The Butcher tableaux for these methods are summarized in Miczek (2013, Annex B).

An advantage of the ESDIRK methods is that they have an integrated error estimator. It works by constructing a second set of coefficients $\tilde{b}_1, \ldots, \tilde{b}_s$ for combining the intermediate solutions $\boldsymbol{U}^{(q)}_{i,j,k}$ that gives a result accurate to an order one below the original scheme. The difference between the two results serves as an error estimate for the lower order method. As the higher order method is more accurate we use higher order solution with the error estimate from the lower order. This idea goes back to Erwin Fehlberg. A simple example is the method given by the Butcher tableau:[2]

$$\begin{array}{c|ccccc}
0 & & & & & \\
\frac{1}{2} & \frac{1}{2} & & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & & \\
1 & 0 & 0 & 1 & & \\
1 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} & \\
\hline
b & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} & 0 \\
\hline
\tilde{b} & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & 0 & \frac{1}{6}
\end{array} \tag{2.33}$$

Missing values have a zero value. If the set of coefficients b is used to compute the result the method is of fourth order. The coefficients $\tilde{b}$ give a method of third order. The difference between the two results serves as an error estimate.

As a final note on the topic of time-stepping methods it should be mentioned that not all methods can be understood in the framework of Runge–Kutta methods. There is also a class of methods that achieve higher order not

[2]Example taken from the lecture notes on numerics of ordinary differential equations by Prof. Bernd Simeon (TU München, 2006).

by dividing one time step into smaller sub-steps but instead by using the information from previous steps. They are called *Linear Multistep Methods*. The most common implicit multistep methods are the Adams–Moulton method and the backward differentiation formulas (BDF) (e.g. Hairer et al., 1993, Chapter III). The method BDF1 is actually a single step method, better known under the name *backward Euler*. One problem with multistep methods in general is that information from the previous steps is unavailable for the initial time steps. The usual solution is to start with a first order method and then increase the order at every step until the desired value is reached. Another problem is that allowing variable time step size makes the formulas and implementation far more complicated. Therefore, the SLH code only implements single step methods that can be formulated as Runge–Kutta schemes, even though multistep might be more efficient in certain cases.

Treatment of Composition Variables

In general, the equation of state has a dependence on the composition, i.e. through the mean molecular weight μ in Eq. (2.17). For implicit time-stepping this means we need values for the composition variables X_i in order to compute the pressure that enters the momentum and energy equations. The most rigorous way of doing this is by extending the set of equations to be solved simultaneously in one implicit time step with the equation for the advection of each species, i.e. the last line in Eqs. (2.15) and (2.16). This approach is treating the composition as so-called *active scalars*. The issue here is that this enlarges the dense blocks in the sparse Jacobian matrix, which needs to be inverted at every timestep. Even a moderate number of only 5 chemical species would already double the storage size of the matrix and make operations on it significantly more expensive. Therefore a common approach is to treat the species as *passive scalars*. This means the composition is frozen at the beginning of a time step and the Euler equations are solved assuming no advection takes place. After that the newly determined mass flux is used to compute the advection of the species. This is a typical operator splitting approach. It introduces an error that scales with the time step. Fortunately, even in the case of large implicit time steps the error is usually quite small. This is because the dependence of the equation of state on the composition is often rather weak as argued in Section 2.1.1. However, the active scalar is important as a benchmark and a requirement for the unsplit coupling of a nuclear reaction network to hydrodynamics as discussed in Section 2.5.3.

Choice of Time Step

Implicit methods are not subject to the CFL condition for stability. Still, for reasons of accuracy, the time step cannot be arbitrarily large. A good choice of time step must be found that is large enough to make the simulation of the slow phenomena we are modeling feasible but also small enough to yield the appropriate accuracy. There are two principal ideas for choosing the time step. One is the *advective CFL criterion* given by

$$\Delta t \leq \mathrm{CFL}_u \min_{q \in \{u,v,w\}} \frac{\Delta x}{|q|}. \tag{2.34}$$

This is simply the normal CFL criterion, Eq. (2.26), with the sound speed in the denominator removed. With $\mathrm{CFL}_u \approx 1$ this means that the fluid moves roughly one cell per time step. Miczek (2013, Section 5.6.1) performed tests of the propagation of sound and advection waves with varying time step and implicit time-stepping. The result was that for $\mathrm{CFL}_{uc} \leq 5$ sound waves are well resolved; above this value the wave is strongly dampened. The same is found for advective waves but using the advective CFL number CFL_u instead. This means that for flows at low Mach number, where CFL_{uc} is considerably larger than CFL_u, time steps with $\mathrm{CFL}_u \approx 1$ will damp the sound waves but resolve advective waves very well. Thus, it is a practical criterion for simulation of low Mach flows, in which the effect of sound waves is unimportant.

So far, only the Euler equations without the additional terms in Eqs. (2.15) and (2.16) have been discussed. The terms for thermal conduction, gravity, and nuclear burning impose additional accuracy restrictions on the time step that are not reflected at all in the advective CFL criterion. For the gravity source term, Miczek (2013) suggested a CFL-like criterion that includes the free-fall signal propagation speed. This criterion, called CFL_{ug}, is actually overly restrictive because stellar atmospheres are very close to hydrostatic equilibrium (see Section 2.3.1) and the gravitational acceleration is therefore nearly perfectly balanced by the pressure gradient. However, it has the advantage that it also gives a reasonable time step for zero-velocity initial conditions, for which the CFL_u time step is infinite.

For thermal conduction and nuclear burning such CFL-like criteria for the time step are harder to find as these do not directly involve the fluid velocity. Instead of trying to construct such a criterion, we evaluate the error estimator of the ESDIRK schemes. As mentioned before, these schemes can estimate the error of the temporal discretization by computing the result with a high- and low-order method simultaneously. Miczek (2013, Section 5.3.2) explains how this is used to adapt the time step to maintain a constant accuracy. There is one issue when using the ESDIRK error estimator for low Mach flows. If sound

waves are excited for any reason, the time step will be reduced far enough to resolve them. Miczek (2013) suggests the implementation of a stiff error estimator to avoid this effect. As such a stiff error estimator has not been implemented in the SLH code to date, a minimum time step, e.g. $\mathrm{CFL}_u > 0.1$, is used as a simple, yet effective workaround. This is sufficient for pure hydrodynamics simulations but it does not ensure accuracy when including thermal conduction and nuclear burning. The applicability of the adaptive time stepper in this case is evaluated in Section 2.5.4 with a one-dimensional deflagration flame as a test problem.

2.2 Low Mach Number Hydrodynamics

Riemann solver based finite-volume methods are known to exhibit excessive numerical dissipation at low Mach numbers. This section briefly explains the causes of this effect and discusses modifications to the spatial discretizations in SLH to deal with this problem. At the end we show a test problem that gives an estimate of the numerical viscosity of the code.

2.2.1 Problems with Compressible Schemes

The problem of low Mach flows and numerical schemes for the compressible Euler equations was theoretically analyzed by Guillard & Viozat (1999). The authors start from the non-dimensional Euler equations. Here, all physical quantities are replaced by the product of the non-dimensional quantity and a global reference value for the quantity, e.g. $\rho = \hat{\rho}\rho_\mathrm{r}, p = \hat{p}p_\mathrm{r}, \ldots$. The hat denotes the non-dimensional quantity; the subscript "r" the reference quantity. After canceling all possible reference quantities, the non-dimensional Euler equations, without any additional terms, read

$$\partial_{\hat{t}} \begin{pmatrix} \hat{\rho} \\ \hat{\rho}\hat{u} \\ \hat{\rho}\hat{v} \\ \hat{\rho}\hat{w} \\ \hat{\rho}\hat{E} \end{pmatrix} + \partial_{\hat{x}} \begin{pmatrix} \hat{\rho}\hat{u} \\ \hat{\rho}\hat{u}^2 + \frac{1}{{M_\mathrm{r}}^2}\hat{p} \\ \hat{\rho}\hat{u}\hat{v} \\ \hat{\rho}\hat{u}\hat{w} \\ \hat{u}(\hat{\rho}\hat{E} + \hat{p}) \end{pmatrix} + \partial_{\hat{y}} \begin{pmatrix} \hat{\rho}\hat{v} \\ \hat{\rho}\hat{u}\hat{v} \\ \hat{\rho}\hat{v}^2 + \frac{1}{{M_\mathrm{r}}^2}\hat{p} \\ \hat{\rho}\hat{v}\hat{w} \\ \hat{v}(\hat{\rho}\hat{E} + \hat{p}) \end{pmatrix} + \partial_{\hat{z}} \begin{pmatrix} \hat{\rho}\hat{w} \\ \hat{\rho}\hat{u}\hat{w} \\ \hat{\rho}\hat{v}\hat{w} \\ \hat{\rho}\hat{w}^2 + \frac{1}{{M_\mathrm{r}}^2}\hat{p} \\ \hat{w}(\hat{\rho}\hat{E} + \hat{p}) \end{pmatrix} = 0. \quad (2.35)$$

The important point is that the only remaining reference quantity is the reference Mach number $M_\mathrm{r} = u_\mathrm{r}/c_\mathrm{r}$, with the common reference velocity u_r and the reference sound speed c_r. The next step in the analysis is an asymptotic expansion for $M_\mathrm{r} \to 0$ in powers of M_r. This means that all quantities are replaced with terms of the form

$$\hat{\rho} = \hat{\rho}^{(0)} + M_\mathrm{r}\hat{\rho}^{(1)} + {M_\mathrm{r}}^2\hat{\rho}^{(2)} + \cdots . \tag{2.36}$$

The terms with the same power of M_r are then collected, which yields conditions for the low Mach number limit. These conditions are (with the hat dropped for convenience):

- order $1/{M_\mathrm{r}}^2$

$$\nabla p^{(0)} = 0, \tag{2.37}$$

- order $1/M_\mathrm{r}$

$$\nabla p^{(1)} = 0, \tag{2.38}$$

- order 1

$$\frac{\partial \rho^{(0)}}{\partial t} + \nabla \cdot \left(\rho^{(0)}\boldsymbol{V}^{(0)}\right) = 0, \tag{2.39}$$

$$\frac{\partial \rho^{(0)}\boldsymbol{V}^{(0)}}{\partial t} + \nabla \cdot \left(\rho^{(0)}\boldsymbol{V}^{(0)} \otimes \boldsymbol{V}^{(0)}\right) + \nabla p^{(2)} = 0, \tag{2.40}$$

$$\frac{\partial \rho^{(0)} e^{(0)}}{\partial t} + \nabla \cdot \left(\rho^{(0)} e^{(0)}\boldsymbol{V}^{(0)} + p^{(0)}\boldsymbol{V}^{(0)}\right) = 0, \tag{2.41}$$

$$p^{(0)} = (\gamma - 1)\rho^{(0)} e^{(0)} \quad \text{(EoS)}. \tag{2.42}$$

Here, $\boldsymbol{V} = (u, v, w)^T$ is the velocity vector. It can be seen from Equations (2.37) and (2.38) that pressure is constant in space up to terms with ${M_\mathrm{r}}^2$, i.e. it takes the form

$$p(\boldsymbol{x}, t) = p^{(0)}(t) + {M_\mathrm{r}}^2 p^{(2)}(\boldsymbol{x}, t). \tag{2.43}$$

If the pressure at the boundaries is fixed, it is also constant in time. Using this information with Eq. (2.42), we see that

$$\frac{\partial \rho^{(0)} e^{(0)}}{\partial t} = \nabla \left(\rho^{(0)} e^{(0)}\right) = 0. \tag{2.44}$$

Inserting this into Eq. (2.41) yields

$$\nabla \cdot \boldsymbol{V}^{(0)} = 0, \tag{2.45}$$

which is the condition for incompressibility of the zeroth component of the asymptotic expansion.

Guillard & Viozat (1999) then continue by performing a similar asymptotic analysis for the discretized equations. For simplicity, they chose Roe's approximate Riemann solver (see Section 2.1.2) for the spatial discretization. However, their result is valid for a more general class of upwind methods.[3] The main result of this analysis is that pressure has a component that scales linearly with the Mach number in the discrete case,

$$p(\boldsymbol{x}, t) = p^{(0)}(t) + M_{\mathrm{r}} p^{(1)}(\boldsymbol{x}, t). \tag{2.46}$$

This is clearly in violation to the analytic result in Eq. (2.43). It means that the discrete solution shows significantly larger pressure fluctuations at low Mach numbers.

The problem becomes even more evident in the analysis by Miczek (2013). Here, the asymptotic behavior of the single components of the Jacobian matrix of the flux function is analyzed. Instead of the three Cartesian fluxes $\boldsymbol{F}$, $\boldsymbol{G}$, and $\boldsymbol{H}$ in Eq. (2.16), the general result is expressed using the normal vector $\boldsymbol{n} = (n_x, n_y, n_z)^T$ on the respective cell interface and $q_n = \boldsymbol{V} \cdot \boldsymbol{n}$, the velocity component perpendicular to this interface. The expressions take an easier form when they are transformed to primitive variables given by $\boldsymbol{W} = (\rho, u, v, w, p)^T$. In these variables the flux Jacobian takes the form

$$\begin{pmatrix} q_n & \rho n_x & \rho n_y & \rho n_z & 0 & 0 \\ 0 & q_n & 0 & 0 & \frac{n_x}{{M_{\mathrm{r}}}^2} & 0 \\ 0 & 0 & q_n & 0 & \frac{n_y}{{M_{\mathrm{r}}}^2} & 0 \\ 0 & 0 & 0 & q_n & \frac{n_z}{{M_{\mathrm{r}}}^2} & 0 \\ 0 & \rho c^2 n_x & \rho c^2 n_y & \rho c^2 n_z & q_n & 0 \\ 0 & 0 & 0 & 0 & 0 & q_n \end{pmatrix}. \tag{2.47}$$

The same analysis can be applied to the Roe scheme, Eq. (2.23). This leads to the Jacobian matrix with the asymptotic scaling

$$\begin{pmatrix} \mathcal{O}(1) & \mathcal{O}(M_{\mathrm{r}}) & \mathcal{O}(M_{\mathrm{r}}) & \mathcal{O}(M_{\mathrm{r}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & 0 \\ 0 & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & 0 \\ 0 & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & 0 \\ 0 & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & 0 \\ 0 & \mathcal{O}(M_{\mathrm{r}}) & \mathcal{O}(M_{\mathrm{r}}) & \mathcal{O}(M_{\mathrm{r}}) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}}) & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathcal{O}(1) \end{pmatrix}. \tag{2.48}$$

---

[3] It is valid for any scheme in which the dissipation term has the form $|A|(\boldsymbol{U}_{\mathrm{L}} - \boldsymbol{U}_{\mathrm{R}})$. A is the Jacobian, but not necessarily evaluated at the Roe average state.

This uses the Landau $\mathcal{O}$-notation.[4] It follows from this result that for low Mach numbers some of the physical flux terms, i.e. the ones in the function F, are completely overwhelmed by the terms stemming from the *upwind* term $|A_{\text{roe}}|(\boldsymbol{U}_R - \boldsymbol{U}_L)$. Upwind means that this term ensures the numerical domain of dependence matches the physical one. For example, a supersonic shock does not depend on the state of the downwind material but only on the upwind material. If this dependence is violated by the numerical scheme, it becomes unstable (for details, see Toro, 2009). The upwind term can be viewed as a *numerical dissipation* term that is important for stability. If such a term dominates the physical flux, it causes excessive dissipation that is of a purely numerical nature. Miczek (2013) showed this using the *Gresho problem* (in the form given by Liska & Wendroff, 2003), a rotating vortex that is a stationary solution to the Euler equations. It was found that Godunov-like methods as well as flux vector splitting methods are subject to strong numerical viscosity as the Mach number is lowered.

2.2.2 Modified Schemes for Use with Low Mach Number Flows

Several of the commonly used hydrodynamics schemes can be modified to reduce the effect of rising numerical dissipation as the Mach number is lowered. Ideally, the dissipation of a scheme does not depend on the Mach number at all. In this section we review two methods in the form in which they have been suggested by Miczek (2013) for use in simulations of astrophysical flows and implemented in SLH.

Preconditioned Roe Solver

A scheme is called *preconditioned Roe scheme* if can be expressed in the form

$$\boldsymbol{F}_{1/2} = \frac{1}{2}\left(\boldsymbol{F}(\boldsymbol{U}_L) + \boldsymbol{F}(\boldsymbol{U}_R) - (P^{-1}|PA|)_{\text{roe}}(\boldsymbol{U}_R - \boldsymbol{U}_L)\right), \tag{2.49}$$

with a non-singular matrix P that depends on the local fluid state $\boldsymbol{U}$. This method was suggested for time-dependent flows by Turkel (1999). It is called *flux preconditioning* to avoid confusion with preconditioners used in solvers for systems of linear equations.

[4] $f \in \mathcal{O}(g)$ if $\limsup_{M_\text{r}\to 0} \frac{f(M_\text{r})}{g(M_\text{r})} < \infty$.

The first preconditioning matrix we consider is given in primitive variables V by

$$P_{\boldsymbol{V}} = \begin{pmatrix} 1 & 0 & 0 & 0 & \frac{\delta^2-1}{c^2} & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \delta^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{2.50}$$

with

$$\delta = \min(1, \max(M, M_{\mathrm{cut}})). \tag{2.51}$$

This form of P was introduced by Weiss & Smith (1995). It depends on the local Mach number M, which is limited by a cut-off Mach number M_{cut} to prevent the matrix from becoming singular at zero velocity. Additionally, the value is capped at $M = 1$, so that P reduces to the identity matrix for supersonic flow. The complete scheme with this preconditioner is known as the *Roe–Turkel scheme.* Miczek (2013) found the asymptotic scaling of the upwind term in this scheme to be

$$\begin{pmatrix} \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}{}^2}) & 0 \\ 0 & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}{}^2}) & 0 \\ 0 & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}{}^2}) & 0 \\ 0 & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}{}^2}) & 0 \\ 0 & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(\frac{1}{M_{\mathrm{r}}{}^2}) & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathcal{O}(1) \end{pmatrix}. \tag{2.52}$$

This is far more consistent to the scaling of the physical flux in Eq. (2.47) than that of the original Roe upwind term in Eq. (2.48). The only discrepancy is in the first and fifth row of the fifth column. This means it overestimates the change of density and pressure flux with respect to the difference in pressure. While this is inconsequential for many of the low Mach number flows considered in engineering, which have very low pressure fluctuations, it is a problem for astrophysical applications. The reason is that stellar atmospheres have a considerable pressure stratification, which is balanced by gravity (see Section 2.3.1). This causes fluxes in the vertical direction even if the atmosphere is in perfect hydrostatic equilibrium and prevents the code from resolving very low Mach numbers.

To improve the preconditioned Roe scheme Miczek (2013) suggested a new preconditioning matrix

$$P_{\boldsymbol{V}} = \begin{pmatrix} 1 & n_x \frac{\rho\delta M_\mathrm{r}}{c} & n_y \frac{\rho\delta M_\mathrm{r}}{c} & n_z \frac{\rho\delta M_\mathrm{r}}{c} & 0 & 0 \\ 0 & 1 & 0 & 0 & -n_x \frac{\delta}{\rho c M_\mathrm{r}} & 0 \\ 0 & 0 & 1 & 0 & -n_y \frac{\delta}{\rho c M_\mathrm{r}} & 0 \\ 0 & 0 & 0 & 1 & -n_z \frac{\delta}{\rho c M_\mathrm{r}} & 0 \\ 0 & n_x \rho c \delta M_\mathrm{r} & n_y \rho c \delta M_\mathrm{r} & n_z \rho c \delta M_\mathrm{r} & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{2.53}$$

with

$$\delta = \frac{1}{\min(1, \max(M, M_\mathrm{cut}))} - 1. \tag{2.54}$$

It should be noted that the definition of δ is different from Eq. (2.50). The asymptotic scaling of this preconditioner is

$$\begin{pmatrix} \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & 0 \\ 0 & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(\frac{1}{{M_\mathrm{r}}^2}) & 0 \\ 0 & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(\frac{1}{{M_\mathrm{r}}^2}) & 0 \\ 0 & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(\frac{1}{{M_\mathrm{r}}^2}) & 0 \\ 0 & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & \mathcal{O}(1) & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathcal{O}(1) \end{pmatrix}. \tag{2.55}$$

Comparison with Eq. (2.47) reveals that this preconditioning matrix makes the scaling of the upwind term fully consistent with the physical flux. Test calculations with the Gresho vortex (see Section 2.2.1) verified that this scheme features numerical dissipation that is independent of Mach number. While it has not been proven that the resulting upwind term is sufficient to stabilize the scheme, numerical tests indicate its stability. (Miczek, 2013)

AUSM$^+$-up

Another scheme for the compressible Euler equations that also features accurate behavior in the low Mach number limit is AUSM$^+$-up by Liou (2006). To introduce the scheme we start from the general from of AUSM schemes, Eq. (2.24). The mass flux is given by

$$\dot{m}_{1/2} = c_{1/2} M_{1/2} \begin{cases} \rho_\mathrm{L} & \text{if } M_{1/2} > 0, \\ \rho_\mathrm{R} & \text{otherwise.} \end{cases} \tag{2.56}$$

This uses

$$M_{1/2} = \mathcal{M}^+_{(4)}(M_\mathrm{L}) + \mathcal{M}^-_{(4)}(M_\mathrm{R}) - \frac{K_p}{f_a} \max(1 - \sigma \bar{M}^2) \frac{p_\mathrm{R} - p_\mathrm{L}}{\rho_{1/2} c^2_{1/2}} \tag{2.57}$$

with

$$f_a = M_\mathrm{o}(2 - M_\mathrm{o}), \quad M_\mathrm{o}^2 = \min(1, \max(\bar{M}^2, M_\mathrm{cut})), \tag{2.58}$$

$$\bar{M}^2 = \frac{u_\mathrm{L}^2 + u_\mathrm{R}^2}{2c_{1/2}^2}, \quad \rho_{1/2} = (\rho_\mathrm{L} + \rho_\mathrm{R})/2, \quad c_{1/2} = (c_\mathrm{L} + c_\mathrm{R}). \tag{2.59}$$

The pressure flux is

$$\begin{aligned} p_{1/2} = {} & \mathcal{P}_{(5)}^{+}(M_\mathrm{L})p_\mathrm{L} + \mathcal{P}_{(5)}^{-}(M_\mathrm{R})p_\mathrm{R} \\ & - K_u \mathcal{P}_{(5)}^{+}(M_\mathrm{L})\mathcal{P}_{(5)}^{-}(M_\mathrm{R})(\rho_\mathrm{L} + \rho_\mathrm{R})(f_a c_{1/2})(u_\mathrm{R} - u_\mathrm{L}). \end{aligned} \tag{2.60}$$

The functions $\mathcal{M}_{(n)}^{\pm}(M)$ and $\mathcal{P}_{(n)}^{\pm}(M)$ are polynomial functions of Mach number given in Liou (2006). The constants K_u, K_p, and σ can be chosen between 0 and 1. The difference between this scheme and the original AUSM$^+$ scheme (Liou, 1996) are the u-term in $p_{1/2}$, denoted by the constant K_u, and the p-term in $M_{1/2}$, denoted by K_p. The u-term improves the behavior for flows at low Mach numbers. The p-term was introduced "to enhance calculations of low Mach number or multi-phase flow" (Liou, 2006). The origin of this term is explained by Edwards & Liou (1998). It was chosen to match the pressure diffusion term that arises in another member of the class of AUSM schemes, the AUSMDV scheme (Wada & Liou, 1994). The reason is to ensure pressure–velocity coupling at low speeds. We discuss the importance of this term in Section 2.4.3.

Miczek (2013) noted that the p-term in the mass flux causes problems with nearly hydrostatic flows similar to the Roe–Turkel scheme. It scales with $1/M$ and the local pressure gradient. That means it causes a vertical mass flux in a perfectly hydrostatic atmosphere. To remove this problem, the factor f_a is set to unity just for the computation of the p-term. The resulting scheme is called AUSM$^+$-Lowmach. Its ability to resolve low Mach number flows, tested using the Gresho vortex, and to maintain hydrostatic stability, tested in one-dimensional stratified atmospheres, is competitive with the preconditioned Roe scheme described in the section before.

2.2.3 Tests of Numerical Viscosity

To conclude this section about dissipation in low Mach number flows we present the *Taylor–Green vortex* (TGV), a test problem that can be used to estimate the numerical Reynolds number of the schemes implemented in SLH. It is a three-dimensional vortex, first suggested by Taylor & Green (1937), which starts with large scale eddies that gradually decay to smaller ones until the

kinetic energy is dissipated by viscous effects, either physical or numerical. We use the initial conditions given by Drikakis et al. (2007)

$$\begin{aligned}
\rho(t=0) &= \rho_0 = 1.178 \times 10^{-3}\,\mathrm{g/cm^3},\\
u(t=0) &= u_0 \sin kx \cos ky \cos kz,\\
v(t=0) &= -u_0 \cos kx \sin ky \cos kz,\\
w(t=0) &= 0,\\
u_0 &= 10^4\,\mathrm{cm/s},\\
k &= 10^{-2}\,\mathrm{cm^{-1}}\\
p(t=0) &= p_0 + \left[{u_0}^2 \rho/16\right] \left[2 + \cos\frac{2z}{100\,\mathrm{cm}}\right]\\
&\quad \cdot \left[\cos\frac{2x}{100\,\mathrm{cm}} + \cos\frac{2y}{100\,\mathrm{cm}}\right],\\
p_0 &= 10^6\,\mathrm{Ba}.
\end{aligned} \tag{2.61}$$

The formulas were adjusted for CGS units because the original publication used SI units. The physical domain is a box with an edge length of $2\pi \times 100\,\mathrm{cm}$ with periodic boundaries on all sides. The equation of state is that of air, i.e. an ideal gas with $\gamma = \frac{7}{5} = 1.4$. The maximum Mach number of this vortex is $u_0/\sqrt{\gamma p_0/\rho_0} = 0.29$. The initial conditions are visualized in Fig. 2.2.

The *turbulent kinetic energy* K of the vortex is given by

$$K = \frac{1}{2}\left\langle \left(\boldsymbol{V} - \langle \boldsymbol{V} \rangle\right)^2 \right\rangle, \tag{2.62}$$

with the velocity vector $\boldsymbol{V}$. This uses the volumetric average, denoted by $\langle\cdot\rangle$. For the analysis of turbulence the *kinetic energy dissipation rate* dK/dt is required. It is obtained from numerical simulations using the finite difference approximation for the time step t_i

$$\left.\frac{dK}{dt}\right|_{t=t_i} \approx \frac{K(t_i) - K(t_{i-1})}{t_i - t_{i-1}}. \tag{2.63}$$

Another property of the turbulent flow is the *mean enstrophy* Ω defined by

$$\begin{aligned}
\Omega = \frac{1}{2}\left\langle |\nabla \times \boldsymbol{V}|^2 \right\rangle =\\
\frac{1}{2}\left\langle \left(\frac{\partial v}{\partial z} - \frac{\partial w}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial x} - \frac{\partial u}{\partial z}\right)^2 + \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right)^2 \right\rangle.
\end{aligned} \tag{2.64}$$

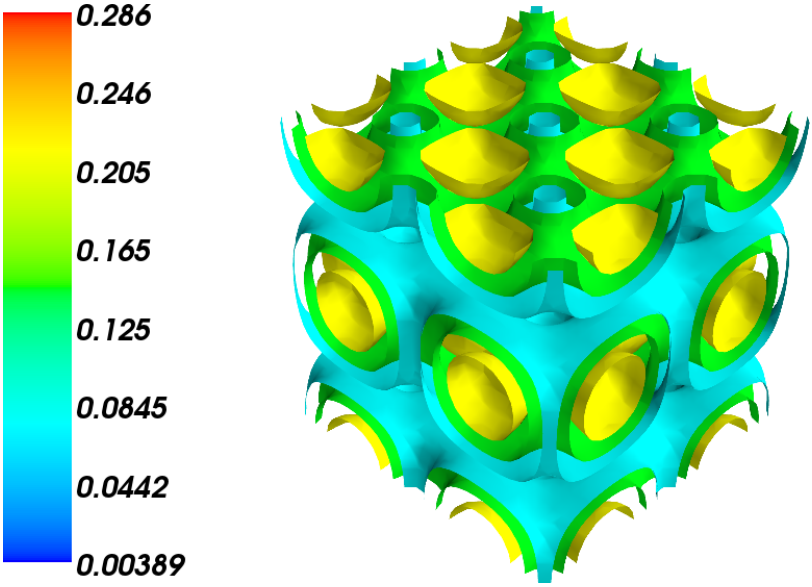


Figure 2.2: Contour plot of the initial Mach number of the Taylor–Green vortex.

Again, this is determined using finite differences. For easier comparability with different initial conditions, all quantities are non-dimensionalized. This is denoted by *, in accordance with Drikakis et al. (2007). The transformation relations are

$$t^* = ku_0 t, \quad K^* = K/{u_0}^2, \quad \Omega^* = \Omega/(ku_0)^2. \tag{2.65}$$

To estimate the numerical Reynolds number, we use an expression for the viscous energy dissipation rate that is valid for the incompressible Navier–Stokes equations (Taylor & Green, 1937),

$$\frac{dK}{dt} = -\eta\Omega. \tag{2.66}$$

Using the definition of the Reynolds number in Eq. (2.14) and switching to non-dimensional variables, results in

$$\frac{dK^*}{dt^*} = -\frac{\Omega^*}{\mathrm{Re}}. \tag{2.67}$$

By independently measuring dK^*/dt and Ω^* we can determine the numerical Reynolds number of the scheme. This means, even though we are simulating a flow without viscosity, it behaves as if it had a viscosity corresponding to Re.

To visualize the decay of the large vortex to smaller scales we use a criterion suggested by Jeong & Hussain (1995). They suggest to use the value of the second largest eigenvalue of the matrix

$$S^2 + \Omega^2, \tag{2.68}$$

with

$$S = \frac{1}{2}\left((\nabla \boldsymbol{V}) + (\nabla \boldsymbol{V})^T\right) \text{ and } \Omega = \frac{1}{2}\left((\nabla \boldsymbol{V}) - (\nabla \boldsymbol{V})^T\right). \tag{2.69}$$

The tensors S and Ω are the symmetric and anti-symmetric components of the velocity gradient tensor $\nabla \boldsymbol{V}$. The combination $S^2 + \Omega^2$ is consequently symmetric. Connected regions with a negative second largest eigenvalue define a vortex core. According to Jeong & Hussain (1995) this criterion captures the location of vortices more accurately than other commonly used criteria, e.g. vorticity or local pressure minima. Figure 2.3 shows the decay of vortices to smaller scales for a simulation of the TGV with 256^3 grid cells.

Figure 2.4 shows the evolution of kinetic energy and enstrophy of the TGV at different resolutions. The numerical Reynolds number is calculated for all times. It also shows the viscosity reducing effect of the flux preconditioner described in Section 2.2. The plots show that the position and height of the peak of the kinetic energy dissipation rate approach constant values. The same phenomenon was observed by Drikakis et al. (2007). Moreover, they compared inviscid simulations at moderate resolution with direct numerical simulations (DNS), i.e. high-resolution simulations that resolve the flow down to the viscous scale, including an explicit viscosity term. For DNS the energy dissipation curve converges at $\mathrm{Re} \gtrsim 3000$. The limit is very close to the result of the inviscid simulations, which are dominated by numerical viscosity. It is expected that such a viscosity independent limit of the energy dissipation rate exists for large values of Re (e.g. Frisch, 1995, Chapter 5).

To understand the importance of such a result, we introduce the concept of *large eddy simulations* (LES). In LES only the largest eddies are resolved on the grid; the viscous scale is far below the grid resolution. Yet it is still possible to ensure that the properties of the flow are the same as if the turbulent flow was resolved down to the viscous scale. There are several approaches to achieve this. One is by using an explicit sub-grid scale (SGS) model. Here, viscous terms are added to the Euler equations that are a function of the resolved properties of the flow. These terms model the transport of kinetic energy from the resolved scale to the sub-grid scale. The first model of this kind, which is still widely used, was developed by Smagorinsky (1963). A different idea is taking into account that simulation codes have an inherent numerical

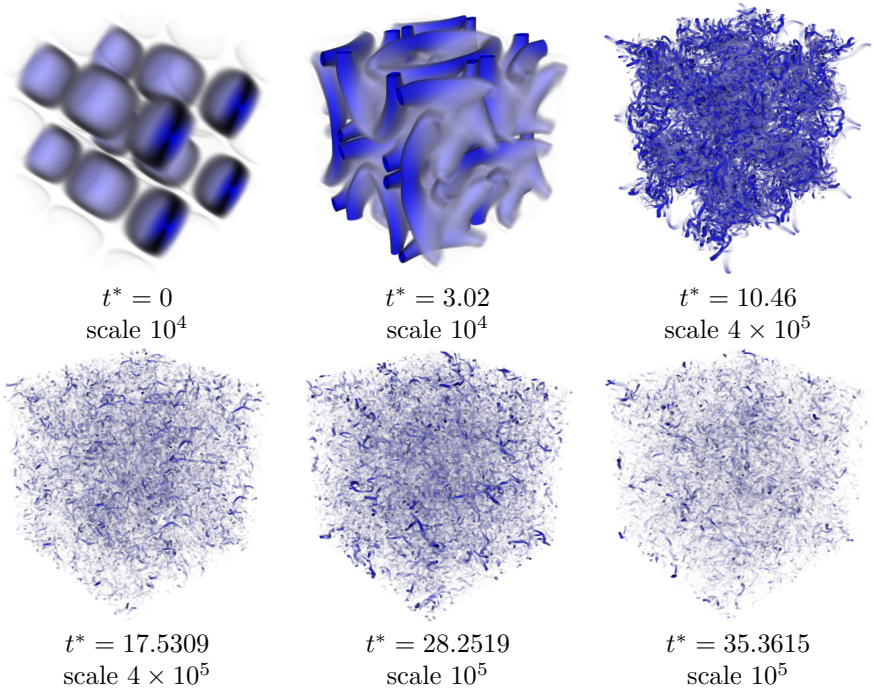


Figure 2.3: Temporal evolution of vortex cores in the Taylor–Green vortex visualized using the second largest eigenvalue λ_2 of Eq. (2.68). Blue regions have negative λ_2. Regions with $\lambda_2 > 0$ are transparent. The magnitude of the color scale was adjusted as stated below each panel.

viscosity. If it shows the correct behavior, no additional effort for modeling the sub-grid scales is required. This approach is called *implicit large eddy simulation* (ILES).

Drikakis et al. (2007) compared simulations of the TGV using an explicit SGS model, ILES, and DNS. They found that ILES is equally capable of capturing the effects of turbulence as the explicit SGS model, at least for the discretization schemes they considered. In the SLH code we follow the ILES approach. The fact that the resolution independent energy dissipation curve is reproduced fairly well is a promising indicator that this is justified. Still, final conclusions can only be drawn after a detailed analysis of the other properties of turbulent flows. Drikakis et al. (2007) also verify the existence and correct

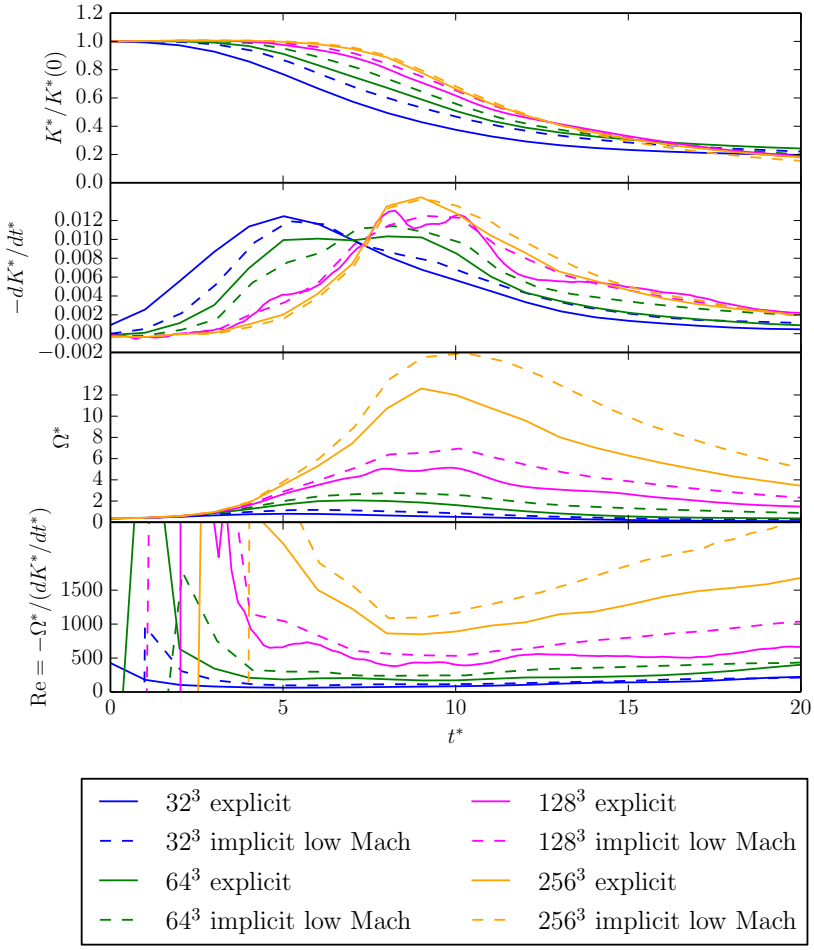


Figure 2.4: This shows kinetic energy, kinetic energy dissipation rate, enstrophy, and numerical Reynolds number for the original Taylor–Green vortex from Eq. (2.61) with $M = 0.29$. The numerical Reynolds numbers were computed using Eq. (2.67). The colors indicate different resolutions. The dashed lines are computed with the low Mach number flux preconditioner.

slope of the inertial range of the kinetic energy spectrum. Such tests have not yet been performed in detail with SLH and are subject of future work.

The original form of the TGV in Eq. (2.61) has a fairly high Mach number of 0.29. We repeat the simulations shown in Fig. 2.4 with the velocity scaled down to $M = 10^{-2}$. The results are shown in Fig. 2.5. In this range the positive effect of the low Mach number preconditioner becomes even more apparent. The preconditioned scheme reaches about the same numerical Reynolds number as in the simulations with $M = 0.29$. In the unpreconditioned case, however, Re is drastically reduced. Also, the regime of resolution independent dissipation of kinetic energy is reached much later. This result clearly emphasizes the importance of using a suitable spatial discretization when simulating turbulent flows at low Mach number.

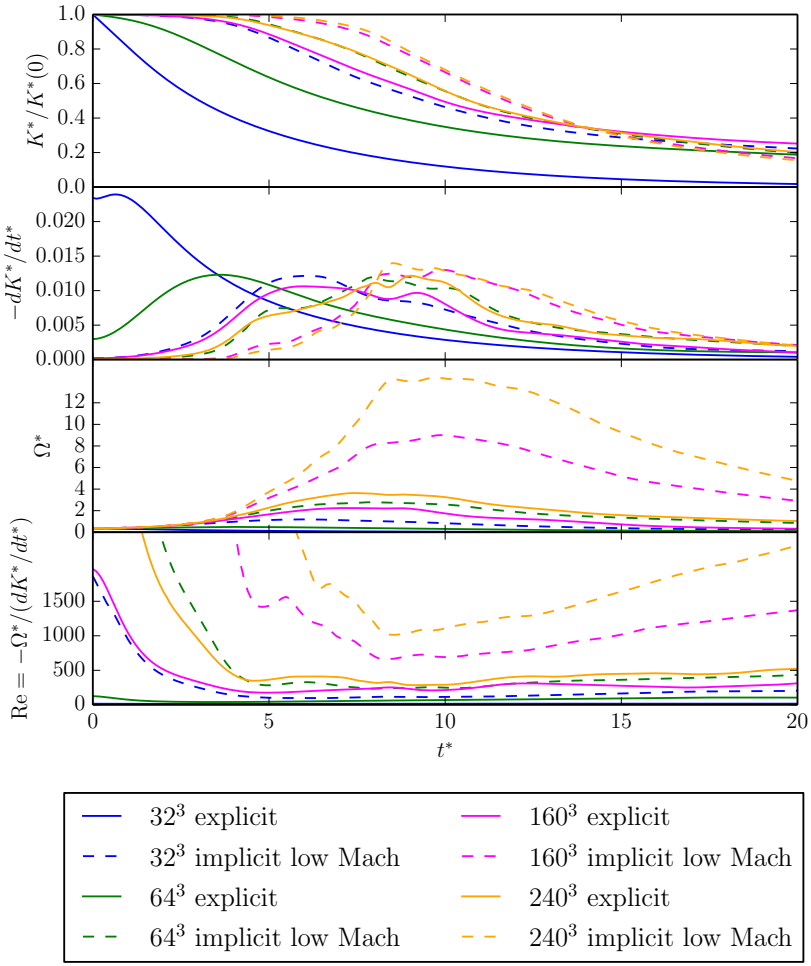


Figure 2.5: This figure shows the same quantities as Fig. 2.4 but with the Mach number reduced to 10^{-2}. The grid size of the high-resolution simulation is slightly different from Fig. 2.4 due to the size of the nodes of the cluster that was used for these computations.

2.3 Hydrostatic Atmospheres

2.3.1 Hydrostatic Equilibrium

Except for the dynamic phases near the end of their lives (e.g. pulsations, explosive burning, gravitational collapse), stars spend nearly their entire lifespan in a state close to perfect *hydrostatic equilibrium*. Resolving this equilibrium and its related instabilities to high accuracy is of major importance to the astrophysical systems that this work is applied to.

A hydrostatic atmosphere is a temporally constant solution to the Navier-Stokes equations, Eq. (2.2), including gravity. Gravity is assumed to be temporally constant and independent of the mass in the domain of interest. This is a reasonable simplification because only small sections of a stellar atmosphere are considered where the gravitational force is entirely dominated by mass outside the domain. First a class of solutions with vanishing velocity is investigated. To derive these all velocities and temporal derivatives are set to 0. The resulting set of equations is

$$\begin{pmatrix} 0 \\ \partial_x p \\ \partial_y p \\ \partial_z p \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ \rho g_x \\ \rho g_y \\ \rho g_z \\ 0 \\ 0 \end{pmatrix}, \tag{2.70}$$

or $\nabla p = \rho \boldsymbol{g}$ in vector notation. A thermodynamic state is completely determined by two of the thermodynamic variables and the composition. That means ρ can be written as a function of pressure p, composition X_i and one other variable, e.g. temperature T, specific internal energy e, or specific entropy s. Choosing T without loss of generality, the condition for hydrostatic equilibrium is

$$\nabla p = \rho(p, X_i, T)\boldsymbol{g}. \tag{2.71}$$

Furthermore, gravity can be assumed to be parallel to one of the coordinate axes because the coordinate system can easily be adapted in a suitable way

(e.g. by rotation or by introducing spherical coordinates).[5] The y-axis is chosen in this discussion. This simplifies Eq. (2.70) to

$$\begin{pmatrix} 0 \\ \partial_x p \\ \partial_y p \\ \partial_z p \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \rho(p, X_i, T) g_y \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.72}$$

This implies that p must be constant on a surface perpendicular to $\boldsymbol{g}$. The pressure profile on the vertical axis is governed by an ordinary differential equation

$$p'(y) = \rho(p(y), X_i(y), T(y)) g_y(y). \tag{2.73}$$

This can be solved for p given an initial value for p and the profiles the abundances X_i and temperature T.

2.3.2 Stability of Hydrostatic Atmospheres

The fact the time derivatives for an atmosphere in perfect hydrostatic equilibrium vanish is not enough ensure that such a state would actually stay constant in time. For that it must also be stable, i.e. slight perturbations should revert back to the original stratification. Following (Kippenhahn et al., 2013, Sec. 6.1) a brief discussion of stability criteria is given in this section.

We consider a small element of the atmosphere that rises from its original position at radius r to a new position at $r+\Delta r$. To compare physical properties of the element to its surroundings we introduce the operator D defined by

$$DA = A_\mathrm{e} - A_\mathrm{s}, \tag{2.74}$$

where A is an arbitrary physical quantity and the subscripts e and s denote the element and the surroundings, respectively. For the question of stability it is crucial how the balance of the force caused by the pressure gradient and the gravitational force is affected. Since we consider elements that rise substantially slower than the speed of sound and pressure equilibrates by sound waves, it is justified to assume that the element is in pressure equilibrium with its surroundings, i.e. $DP = 0$. Although it seems to be obviously fulfilled, it

[5] If the gravitational force depended on the density distribution inside the domain, this could cause coordinate singularities. An example would be a whole-star simulation with a coordinate singularity at the center. It might still be possible to find simple solutions to hydrostatic equilibrium if symmetries can be exploited, e.g. spherical symmetry of stars.

is important to note that this condition is necessary for any of the stability criteria discussed below to work. This is especially important in the light of low Mach number discretizations for which the properties of sound waves might be affected.

With the pressure gradient being fixed by the surroundings the important term in determining the motion of the element is the gravitational force. It is enough to compare it with the gravitational force on the surroundings. This is called the buoyancy force F_b. With the absolute value of gravitational acceleration g it is

$$F_\mathrm{b} = (-g\rho_e) - (-g\rho_s) = -gD\rho. \tag{2.75}$$

This means that an element that is lighter than its surroundings after being lifted up will continue to rise, resulting in an unstable behavior. An element that is heavier than its surroundings will move back to its original position. This is called a stable stratification. In differential terms the condition for stability is

$$D\rho = \rho_\mathrm{e} - \rho_\mathrm{s} = \left[\left(\frac{d\rho}{dr}\right)_\mathrm{e} - \left(\frac{d\rho}{dr}\right)_\mathrm{s}\right]\Delta r > 0. \tag{2.76}$$

To predict the change of ρ under a change in P more assumptions about the thermodynamics of the system have to be made. A straight-forward assumption for this is an *adiabatic* process, i.e. no energy exchange with the surroundings. This is reasonable as long as thermal and radiative conductivity is small enough. To relate $D\rho$ with other quantities we start with a general equation of state $\rho = \rho(P, T, \mu)$ and its differential form

$$\frac{d\rho}{\rho} = \alpha\frac{dP}{P} - \delta\frac{dT}{T} + \varphi\frac{d\mu}{\mu}, \tag{2.77}$$

with the partial derivatives

$$\alpha = \left(\frac{\partial \ln\rho}{\partial \ln P}\right), \quad \delta = -\left(\frac{\partial \ln\rho}{\partial \ln T}\right), \quad \varphi = \left(\frac{\partial \ln\rho}{\partial \ln\mu}\right). \tag{2.78}$$

The peculiar sign in the definition of δ was chosen so that $\alpha = \delta = \varphi = 1$ in the case of an ideal gas.

Rewriting Eq. (2.76) using Eq. (2.77) we arrive at

$$\left(\frac{\alpha}{P}\frac{dP}{dr}\right)_\mathrm{e} - \left(\frac{\delta}{T}\frac{dT}{dr}\right)_\mathrm{e} + \left(\frac{\varphi}{\mu}\frac{d\mu}{dr}\right)_\mathrm{e} - \left(\frac{\alpha}{P}\frac{dP}{dr}\right)_\mathrm{s} + \left(\frac{\delta}{T}\frac{dT}{dr}\right)_\mathrm{s} - \left(\frac{\varphi}{\mu}\frac{d\mu}{dr}\right)_\mathrm{s} > 0. \tag{2.79}$$

Because pressure equilibrium was assumed above, the pressure derivatives are identical and cancel each other. The composition of an element is advected with it so $d\mu/dr$ vanishes for the element but not for the surroundings, which can have a gradient in composition. Multiplying the equation with the *pressure scale height*,

$$H_P = -\frac{dr}{d\ln P} = -P\frac{dr}{dP}, \tag{2.80}$$

and rearranging yields

$$\left(\frac{d\ln T}{d\ln P}\right)_{\rm s} < \left(\frac{d\ln T}{d\ln P}\right)_{\rm e} + \frac{\varphi}{\delta}\left(\frac{d\ln \mu}{d\ln P}\right)_{\rm s}. \tag{2.81}$$

Usually the following quantities are defined in stellar astrophysics

$$\nabla = \left(\frac{d\ln T}{d\ln P}\right)_{\rm s}, \quad \nabla_{\rm e} = \left(\frac{d\ln T}{d\ln P}\right)_{\rm e}, \quad \nabla_\mu = \left(\frac{d\ln \mu}{d\ln P}\right)_{\rm s}. \tag{2.82}$$

Under the assumption that the element changes its state adiabatically $\nabla_{\rm e}$ can be replaced by $\nabla_{\rm ad}$, which is $d\ln T/d\ln P$ for an adiabatic process. The criterion for stability is then

$$\nabla < \nabla_{\rm ad} + \frac{\varphi}{\delta}\nabla_\mu. \tag{2.83}$$

This is known as the *Ledoux criterion* for convective stability. If the composition of the atmosphere is homogeneous, the criterion reduces to the *Schwarzschild criterion*

$$\nabla < \nabla_{\rm ad}. \tag{2.84}$$

2.4 Treatment of Gravity

It was noted by Miczek (2013) that atmospheres that are supposed to be convectively stable showed a checkerboard-like instability as soon as low Mach preconditioning was switched on. In further investigations he found that this instability is not tied to a specific type of low Mach number discretization. It appeared with every low Mach number scheme that was tried. Also the result could be reproduced independently in tests with the "Fire Dynamics Simulator", a code developed to simulate fires in buildings. This makes it highly unlikely that the cause is a mere programming error. The following sections introduce the problem in detail, explain the possible numerical cause, and suggest possible solutions.

2.4.1 Details of the Spurious Instability

A typical test problem to illustrate the above mentioned instability is that of a hydrostatic atmosphere that has a convectively stable temperature stratification. The composition is homogeneous which means that the Schwarzschild and Ledoux criteria are equivalent. All quantities are given in non-dimensional form, denoted by the hat (e.g. $\hat{y}$). The equation of state is that of an ideal gas with a mean molecular weight of 1. Gravity is directed in the negative y-direction and is spatially constant. The temperature profile is given by

$$T = 1 + \Delta T \tanh \frac{\hat{y}}{w}. \tag{2.85}$$

The corresponding pressure profile that fulfills hydrostatic equilibrium is then

$$p = \exp\left(-\frac{\hat{y} - \Delta T w \log(\cosh(\hat{y}/w) + \Delta T \sinh(\hat{y}/w))}{1 - \Delta T^2}\right). \tag{2.86}$$

The following test calculations use the parameters $\Delta T = 0.1$ and $w = 0.02$. For these values the square of the Brunt–Väisälä frequency is positive on the whole domain, making the atmosphere convectively stable as seen in Fig. 2.6.

When setting up this atmosphere with zero initial velocity and simulating its dynamical evolution, it is expected that the Mach number initially rises due to discrepancies between the analytic and discretized hydrostatic equilibrium. After a transient phase the Mach number settles at a value that is compatible with the discretization (see Miczek (2013) for a detailed study of the capability to resolve hydrostatic equilibria of different discretizations). This behavior can be observed in Fig. 2.7 for the case of the unpreconditioned Roe scheme. When switching on low Mach number preconditioning for the Roe solver, it can be

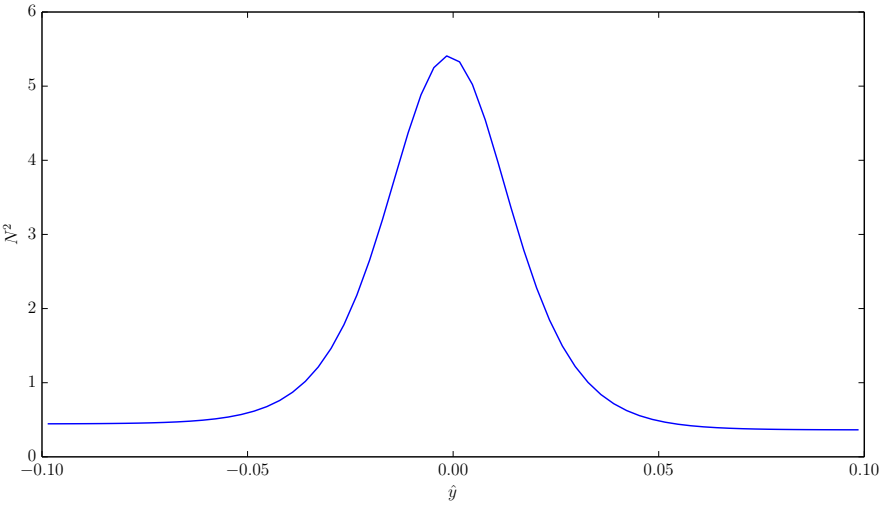


Figure 2.6: Vertical profile of the Brunt–Väisälä frequency in the convectively stable test problem. Quantities are given in non-dimensional units.

noticed that the flow reaches a significantly lower Mach number. This effect is attributed to the lower diffusivity of this method (Miczek, 2013). However, after a phase of exponential decrease the Mach number does not stabilize at this level but instead rises again exponentially to saturate at a value that is even higher than that of the unpreconditioned case. This does not occur in the one-dimensional case (cf. Miczek, 2013, Fig. 6.2).

A two-dimensional view on this issue is given in Fig. 2.8. It shows the Mach number and the horizontal density and pressure fluctuations. All three show a checkerboard-like pattern that is nearly identical except in amplitude. Only one component of the Mach number is shown to reveal its checkerboard structure, which would not be visible if the absolute value were shown. This pattern increases exponentially in amplitude until reaches saturation ($\hat{t} \approx 8.5$ in Fig. 2.7). At this point it is distorted by the large velocities present on the grid. As seen in Fig. 2.9 it closely resembles convective motions that would appear in convectively unstable atmospheres.

The fact that this effect appears exactly at the grid scale suggests that the cause of the instability is numerical. In the case of a physical instability, the size of the cells should stay constant in physical space, that means the number of grid cells per instability cell should change with resolution. Of course, an unresolved physical instability could also appear at the grid scale.

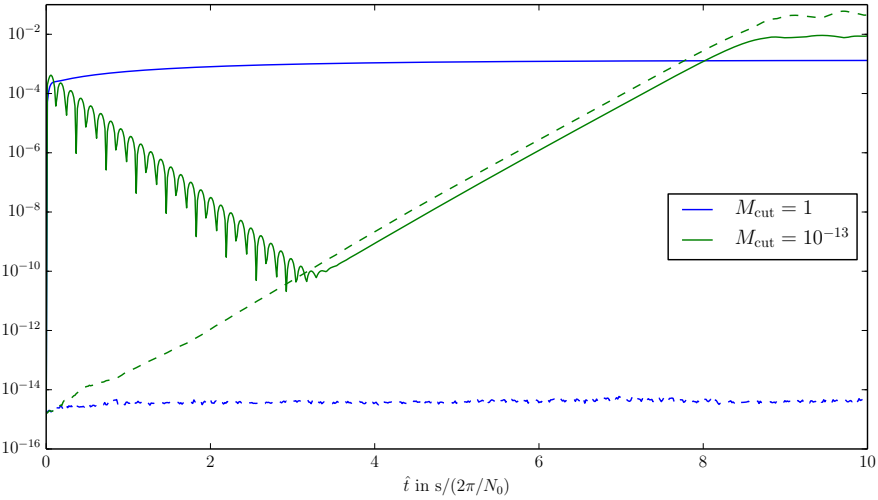


Figure 2.7: Temporal evolution of the maximum Mach number (solid lines) on the grid for a convectively stable atmosphere computed with the unpreconditioned Roe scheme. The dashed lines show the maximum deviation of density from its horizontal average. Time is given in non-dimensional units defined by the reference Brunt–Väisälä frequency N_0.

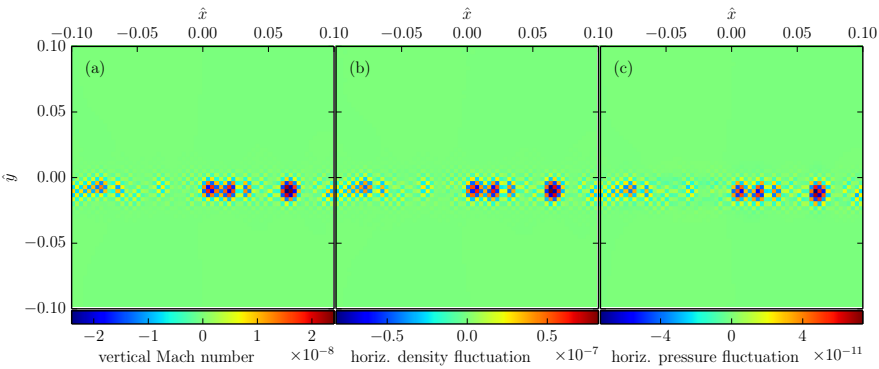


Figure 2.8: Vertical slice through the atmosphere at $\hat{t} = 5$. Panel (a) shows the vertical Mach number. Panel (b) shows the deviation of density from its horizontal mean. Panel (c) shows the same for pressure.

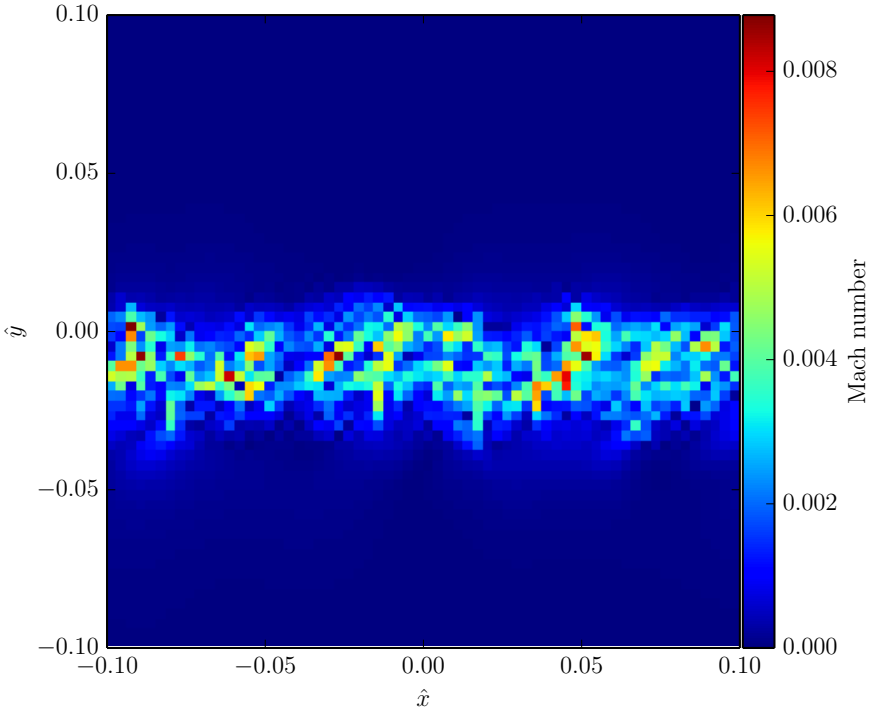


Figure 2.9: Mach number in the atmosphere at $\hat{t} = 10$, i.e. after the instability reaches its saturation.

It has to be considered that the simulation code models the Euler equations, neglecting physical viscosity (see Section 2.1.1). Since the Euler equations can be written in a completely non-dimensional form as in Eq. (2.35), the length scales used in the code are only meaningful when they are related to the microphysical properties of the system. In the Navier-Stokes equations this relation is given by the *Reynolds number*,

$$\mathrm{Re} = \frac{\rho_\mathrm{r} u_\mathrm{r} L_\mathrm{r}}{\mu_\mathrm{r}}, \tag{2.87}$$

where ρ_r, u_r, L_r, and μ_r are typical (reference) values for density, velocity, length, and dynamic viscosity. Miczek (2013, Sec. 6.3.5) explored the effect of adding artificial viscosity to the system in order to stabilize it while still keeping the low Mach number discretization. The amount of viscosity needed for this corresponds to $\mathrm{Re} = 0.07$, which is a completely unrealistic regime. It is not only very far from the stellar values of $\mathrm{Re} \approx 10^{10}$ but even far from the regime of turbulent motions at $\mathrm{Re} \gtrsim 1000$. Such a simulation would not capture the essential properties of convection at all.

2.4.2 A Possible Cause of the Instability

The discussed instability looks remarkably similar to the development of convection in convectively unstable atmospheres. One obvious approach to the problem is therefore to understand why the physical mechanism that stabilizes the atmosphere, the Schwarzschild criterion (cf. Section 2.3.2), does not work for all the low Mach number discretizations that were investigated so far.

Figure 2.8 shows one feature that disagrees with the assumptions made for the derivation of the Schwarzschild criterion. It was assumed that there are no horizontal fluctuations in pressure, because sound waves are supposed to even these out on the timescale of the sound crossing time. In the simulations from the previous section the time it takes a sound wave to cross the whole domain is roughly 0.15 (in non-dimensional units). The time to cross the distance between two neighboring cells is just 1/64th of that. Physically we would expect the pressure fluctuation to decay and not to build up exponentially. There are two possibilities to explain this phenomenon. Either the mechanism that is building up the pressure fluctuation is working on a faster timescale than expected or the decay of a pressure fluctuation by sound waves is slower than expected.

Rieper (2011) claim that the checkerboard modes in pressure decaying too slowly is typical of many low Mach number discretizations. We repeat their argument here. It is known from the asymptotic analysis of the analytic Euler

equations in powers of reference Mach number that pressure is constant up to terms of the order ${M_\mathrm{r}}^2$. In particular we have Eq. (2.38),

$$\nabla p^{(1)} = 0.$$

The analysis of the Roe scheme showed terms in pressure that scale linearly with M_r, Eq. (2.46),

$$p(\boldsymbol{x}, t) = p^{(0)}(t) + M_\mathrm{r} p^{(1)}(\boldsymbol{x}, t).$$

This scaling behavior is the cause of the increased dissipation. In fact, this is not a problem just of the Roe solver but of many compressible hydrodynamics schemes. A common approach to make a scheme usable for low Mach numbers is by removing the $p^{(1)}$ term. On a discrete level Rieper (2011) find that for their low Mach number Roe scheme this results in

$$p^{(1)}_{i+1,j} - p^{(1)}_{i-1,j} = 0, \quad p^{(1)}_{i,j+1} - p^{(1)}_{i,j-1} = 0. \tag{2.88}$$

Li & Gu (2008) see the same for the preconditioned Roe scheme they developed. This type of solution is called a *four-field* solution or *checkerboard mode* due to the pattern of values it allows as illustrated in Fig. 2.10. The existence of such a solution is not a problem per se as it still allows $p^{(1)}$ to be spatially constant and flows in the low Mach number regime are not expected to excite checkerboard modes. The problem becomes visible when these modes are excited, for example by errors introduced at the boundary or by source terms like gravity or energy release through nuclear reactions. The question is then how fast the checkerboard modes are damped away, specifically whether they are damped more quickly than they are built up by the source term. In the literature this damping property is also known as *pressure–velocity coupling*.

A	C	A	C	A
B	D	B	D	D
A	C	A	C	A
B	D	B	D	D
A	C	A	C	A

Figure 2.10: Structure of the four-field solution allowed by Eq. (2.88).

Dellacherie (2009) analyzed the dampening of checkerboard modes for the linear wave equation and found that the energy in the checkerboard mode behaves according to

$$u_0 \exp\left(-\frac{8\nu_u t}{\Delta x^2}\right), \tag{2.89}$$

where u_0 is the initial energy, t the simulation time, and Δx the mesh size. They computed the numerical diffusion factor ν_u for the standard Roe scheme and for a Roe scheme where central differences are used to discretize the momentum flux. Rieper (2011) also compute ν_u for their modified low Mach number scheme. The results are

$$\nu_u = \begin{cases} \frac{c\Delta x}{2M_r} & \text{unmodified Roe solver,} \\ \frac{c\Delta x}{2} & \text{LMRoe (Rieper, 2011),} \\ 0 & \text{pressure centering.} \end{cases} \tag{2.90}$$

At low Mach numbers the unmodified Roe scheme dampens the checkerboard modes very fast, which is in accordance with the observation that the instability does not occur in the unpreconditioned case. Also the scheme by Rieper (2011) has some dampening properties, which is enough to prevent the appearance of the modes in their test cases. As these did not include the gravity source term, it still has to be tested whether it is enough to prevent the checkerboard modes in our problems. The scheme with pressure centering does not include any dampening.

This analysis has not yet been performed for the preconditioned Roe scheme used in SLH from Section 2.2.2. However, the results of Li & Gu (2008) for their preconditioned low Mach Roe scheme lets us suspect strongly that missing pressure–velocity coupling is the cause for the spurious, checkerboard-like instability that causes convectively stable atmospheres to become unstable.

2.4.3 Ensuring Pressure–Velocity Coupling in Low Mach Number Schemes

It seems that the key to removing checkerboard modes is ensuring pressure–velocity coupling in the flow. Li & Gu (2008) discuss several options applicable for their preconditioned Roe solver. The method they call *momentum interpolation* works by replacing the central term $\frac{1}{2}(\boldsymbol{F}(\boldsymbol{U}_L) + \boldsymbol{F}(\boldsymbol{U}_R))$ of the Roe flux, Eq. (2.23), with

$$U_c \boldsymbol{Q} + \boldsymbol{P}, \tag{2.91}$$

where

$$U_\mathrm{c} = (u_\mathrm{L} + u_\mathrm{R})/2 - \frac{c_2}{\rho_\mathrm{r} u_\mathrm{r}}(p_\mathrm{R} - p_\mathrm{L}), \tag{2.92}$$

$$\boldsymbol{Q} = (\rho, \rho u, \rho v, \rho w, \rho E + p)^T, \tag{2.93}$$

$$\boldsymbol{P} = (0, p, 0, 0, 0)^T. \tag{2.94}$$

They suggest to choose the constant c_2 as small as possible for accuracy but greater than 0.04 for stability. They note that momentum interpolation leads to a non-zero velocity divergence, which is in violation of the analytic low Mach number limit. As an alternative that is not subject to this flaw, they mention the use of staggered grids, where pressure is stored using its value at the cell interface instead of the cell center. In their application they used the former approach due to its simplicity. Some work has been done on implementing the momentum interpolation method for the preconditioned Roe scheme in SLH but it is not finished yet. Instead we focused on another approach.

Edwards & Liou (1998) present another method for pressure–velocity coupling which they introduced for the AUSM$^+$ scheme. They introduce an additional term in the mass flux, which is called the p-term in the AUSM$^+$-up scheme. It is the term proportional to K_p in Eq. (2.57). It introduces an additional flux proportional to the local pressure gradient. As it was mentioned in Section 2.2.2 this is problematic in the case of atmospheres with a pressure stratification as it then causes fluxes in the vertical direction. This in term prevents the scheme from reaching very low Mach numbers in hydrostatic equilibrium. That is why Miczek (2013) reduced the p-term by removing the inverse scaling with Mach number. While the scheme can keep hydrostatic equilibrium really well in one-dimensional test cases, this very modification also makes it susceptible to the checkerboard-like instability.

Keeping hydrostatic atmospheres stable down to very low Mach numbers is of great importance for the use of a scheme in stellar astrophysics. So we clearly cannot use the unmodified AUSM$^+$-up scheme. Still the p-term is important to prevent checkerboard modes from growing and making the atmosphere unstable. One possibility to retain the p-term and preserve hydrostatic equilibrium is to "remove" the vertical pressure gradient by using a special well-balancing method.

2.4.4 Well-Balancing

One way of counteracting the problems of the gravity source term in low Mach number hydrodynamics is *well-balancing*. This was already suggested by Miczek (2013). A numerical scheme that is able to maintain a hydrostatic

initial state exactly is called well-balanced. Most hydrodynamics schemes do not have this property because the $\rho\boldsymbol{g}$ term is discretized in a completely different way than the ∇p term. That means that $\nabla p \neq \rho\boldsymbol{g}$ on a discrete level even if the initial condition was in perfect hydrostatic equilibrium analytically. There are methods that achieve well-balancing by removing the hydrostatic pressure gradient. This enables us to keep the pressure diffusion term from AUSM$^+$-up without the modifications introduced with AUSM$^+$-Lowmach (see Section 2.2.2).

A well-balancing method that was found to be particularly useful is the one developed by Cargo & Le Roux (1994) for the one-dimensional Euler equations. We give a full explanation of the one-dimensional scheme here and then develop a way to extend it to multiple dimensions.

The One-Dimensional Cargo–Le Roux Method

We start from the one-dimensional Euler equations with gravity,

$$\partial_t \begin{pmatrix} \rho \\ \rho u \\ \rho E \end{pmatrix} + \partial_x \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(\rho E + p) \end{pmatrix} = \begin{pmatrix} 0 \\ \rho g \\ u\rho g \end{pmatrix}. \tag{2.95}$$

Cargo & Le Roux (1994) suggest the introduction of a potential q with the properties

$$\partial_x q = \rho g \tag{2.96}$$

and

$$\partial_t q = -\rho g u. \tag{2.97}$$

This potential is treated like a composition variable, i.e. it is advected with the fluid flow according to

$$\partial_t(\rho q) + \partial_x(\rho q u) = 0. \tag{2.98}$$

At this stage q has no influence on the equation of state and is therefore a truly passive scalar (see Section 2.1.3).

Equation (2.98) can be simplified using the product rule and the law of mass conservation from the Euler equations,

$$q\partial_t\rho + \rho\partial_t q + q\partial_x(\rho u) + \rho u\partial_x q = 0, \tag{2.99}$$

$$\underbrace{q\partial_t\rho + q\partial_x(\rho u)}_{=0} + \rho\partial_t q + \rho u\partial_x q = 0, \tag{2.100}$$

$$\partial_t q + u\partial_x q = 0. \tag{2.101}$$

This means that if Eq. (2.96) is fulfilled initially and the temporal evolution of q is determined by Eq. (2.98), Eq. (2.97) is automatically satisfied. Eq. (2.98) ensures that the potential keeps its properties.

Equations (2.96) and (2.97) can be inserted into Eq. (2.95), which yields

$$\partial_t \begin{pmatrix} \rho \\ \rho u \\ \rho E \end{pmatrix} + \partial_x \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(\rho E + p) \end{pmatrix} = \begin{pmatrix} 0 \\ \partial_x q \\ -\partial_t q \end{pmatrix}. \tag{2.102}$$

Collecting derivatives with respect to the same variable and inserting $0 = q - q$ results in

$$\partial_t \begin{pmatrix} \rho \\ \rho u \\ \rho E + q \end{pmatrix} + \partial_x \begin{pmatrix} \rho u \\ \rho u^2 + p - q \\ u(\rho E + q + p - q) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.103}$$

The next step is the introduction of a modified equation of state. It is related to the original equation of state via the newly introduced modified pressure

$$\phi = p - q, \tag{2.104}$$

and modified total energy per volume

$$F = \rho E + q. \tag{2.105}$$

In this equation of state q is an active scalar.

Equation (2.102) now reads

$$\partial_t \begin{pmatrix} \rho \\ \rho u \\ F \end{pmatrix} + \partial_x \begin{pmatrix} \rho u \\ \rho u^2 + \phi \\ u(F + \phi) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \tag{2.106}$$

which has exactly the form of the homogeneous, one-dimensional Euler equations using the modified equation of state, Eqs. (2.104) and (2.105).

It is clear from Eq. (2.96) that the potential q is identical to the hydrostatic pressure up to an arbitrary ,constant offset. This means that the new pressure $\phi = p - q$ for a stratified hydrostatic atmosphere is spatially constant. Solving the Euler equations with gravity has been reduced to solving the Euler equations without any source term. The closer the atmosphere is to hydrostatic equilibrium the closer it is to an isobaric state within the framework of the Cargo–Le Roux method. The well-balancing property is then fulfilled for all hydrodynamics schemes used in SLH because they all preserve isobaric states

exactly, at least when density and pressure are used for the reconstruction of left and right states.[6]

It should further be noted that no part of the derivations presented here used the assumption that the state is close to hydrostatic equilibrium. In fact, is valid also for states far from equilibrium. Here, the effect of gravity just shows up in the modified pressure gradient. The method is still beneficial in these cases as it removes the need to find a suitable discretization for the source terms.

Extension to Multiple Dimensions

In the one-dimensional case the potential q can always be found by just integrating ρg,

$$q(x) = \int_{x_0}^{x} dx' \rho(x') g(x'). \tag{2.107}$$

This is no longer true in more than one dimension.

The natural, multidimensional extension of Eq. (2.96) is

$$\nabla q = \rho \boldsymbol{g}. \tag{2.108}$$

A necessary condition for the existence of such a potential is

$$\nabla \times \rho \boldsymbol{g} = 0. \tag{2.109}$$

This is a consequence of the fundamental theorem of calculus and Stokes' theorem. Using the fact that $\nabla \times \boldsymbol{g} = 0$, we find

$$\nabla \times \rho \boldsymbol{g} = \rho \nabla \times \boldsymbol{g} - \boldsymbol{g} \times \nabla \rho = -\boldsymbol{g} \times \nabla \rho. \tag{2.110}$$

This means that q can only be determined if the gradient of density is parallel or antiparallel to gravity. In other words, ρ must be constant on surfaces perpendicular to $\boldsymbol{g}$. This is not the case for general flows, where bubbles of lighter material can be in denser surroundings.

We now introduce a modified version of the Cargo–Le Roux method that works for two- and three-dimensional problems and retains some but not all of the beneficial properties of the original method.

The working hypothesis for the modified method is that a meaningful *horizontal average* can be defined, horizontal meaning perpendicular to $\boldsymbol{g}$. This

[6] Reconstruction is the method by which SLH achieves a higher spatial order. It uses information of the surrounding cells to reconstruct values at the cell interfaces that are required to compute the flux. SLH offers a choice of different sets of variables to perform this reconstruction. The details are described in Miczek (2013, Sec. 3.4)

is possible for a wide range of the applications the code SLH aims at, for example an atmosphere in plane-parallel approximation that is simulated in a box with gravity along one of the coordinate axes or a spherical stellar model in which gravity is directed in radial direction. The horizontal average is denoted with $\langle\cdot\rangle$.

We define a mean horizontal density

$$\rho_0 = \langle\rho\rangle\,. \tag{2.111}$$

By definition of the horizontal average, the gradient of ρ_0 is parallel to $\boldsymbol{g}$. Due to the above argument this allows us to find a scalar potential

$$\nabla q = \rho_0 \boldsymbol{g}. \tag{2.112}$$

In contrast to the original method this potential is not advected like a scalar composition variable but left constant.

If the equation of state is then modified identically to Eqs. (2.104) and (2.105), the three-dimensional Euler equations with gravity take the form

$$\begin{aligned}\partial_t \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ F \end{pmatrix} + \partial_x \begin{pmatrix} \rho u \\ \rho u^2 + \phi \\ \rho uv \\ \rho uw \\ u(F+\phi) \end{pmatrix} + \partial_y \begin{pmatrix} \rho u \\ \rho uv \\ \rho v^2 + \phi \\ \rho vw \\ v(F+\phi) \end{pmatrix} \\ + \partial_z \begin{pmatrix} \rho u \\ \rho uw \\ \rho vw \\ \rho w^2 + \phi \\ w(F+\phi) \end{pmatrix} = \begin{pmatrix} 0 \\ (\rho-\rho_0)(g)_x \\ (\rho-\rho_0)(g)_y \\ (\rho-\rho_0)(g)_z \\ \rho \boldsymbol{g}\cdot\boldsymbol{u} \end{pmatrix}.\end{aligned} \tag{2.113}$$

There are two fundamental differences between this expression and the one-dimensional case, Eq. (2.103). The energy equations retains the full source term $\rho\boldsymbol{g}\cdot\boldsymbol{u}$ because q is temporally constant. As long as $\boldsymbol{g}$ is not varying in time, this term can easily be removed by including the potential energy in the total energy as it is already commonly done for the Euler equations without well-balancing. The source terms in the momentum equations are a bit more problematic. They correspond to the acceleration due to buoyancy. These terms are much smaller than the full gravitational acceleration term and only present where density deviates from its horizontal average.

This new method still has the well-balancing property for states without horizontal density gradients, which is most of the cases considered in stellar

astrophysics. It also retains the feature of changing originally stratified hydrostatic atmospheres to isobaric atmospheres in the modified equation of state.

The severest issue with this method is that q does not automatically change with changing hydrostatic stratification as it does in the one-dimensional case. This means the beneficial properties of the scheme are partially lost if ρ starts to deviate further from ρ_0. One possibility to prevent this is to periodically update ρ_0 with current horizontal averages.

There are two principal methods for computing q and ρ_0. One is by taking the horizontal density average,

$$\rho_0 = \langle \rho \rangle \,, \tag{2.114}$$

and then using integration along the vertical axis (denoted by x here)

$$q(x) = \int_{x_0}^{x} dx' g(x') \rho_0(x'). \tag{2.115}$$

Gravitational acceleration is a scalar value here because the vertical direction was chosen to be the direction of $\boldsymbol{g}$.

The alternative is to determine q as the horizontal average of pressure

$$q = \langle p \rangle \,. \tag{2.116}$$

Then we choose ρ_0 so that it fulfills Eq. (2.112) by taking the derivative along the vertical axis

$$\rho_0(x) = q'(x)/g(x). \tag{2.117}$$

Both methods have their disadvantages. The integration method needs a starting point x_0. The further away from this point the larger the error becomes, so this introduces regions in the flow that are systematically better well-balanced than others. The differentiation method on the other hand just depends on local information, so it does not have these systematic effects. The challenge here is to find a proper method for numerical differentiation. Straight-forward finite differences encounter difficulties when the ρ_0 profile shows small jumps or oscillations, especially when high-order finite differences are used. One possible solution to this is using the formalism of weighted essentially non-oscillatory (WENO) schemes (Liu et al., 1994) to compute the derivative. These schemes use a linear combination of several possible stencils for the finite difference and weight them according to smoothness. This is much less susceptible to discontinuities in the profile. First tests of this method are promising. For the simulations performed for this thesis, however, the simpler approach is chosen that only initializes q once from analytic hydrostatic

equilibrium and leaves it constant for the whole simulation. As long as the profile does not change significantly this provides good well-balancing properties without introducing any of the problems connected to the update of q.

2.4.5 Stability Tests of Well-Balanced Schemes

In this section we explore the impact of Cargo–Le Roux well-balancing on the test setup from Section 2.4.1. We test the AUSM$^+$-Lowmach scheme, the original AUSM$^+$-up scheme, and the preconditioned Roe solver. All schemes are tested with and without well-balancing, except the AUSM$^+$-up scheme, which is hardly usable in a hydrostatic atmosphere without well-balancing, as discussed in Section 2.2.2.

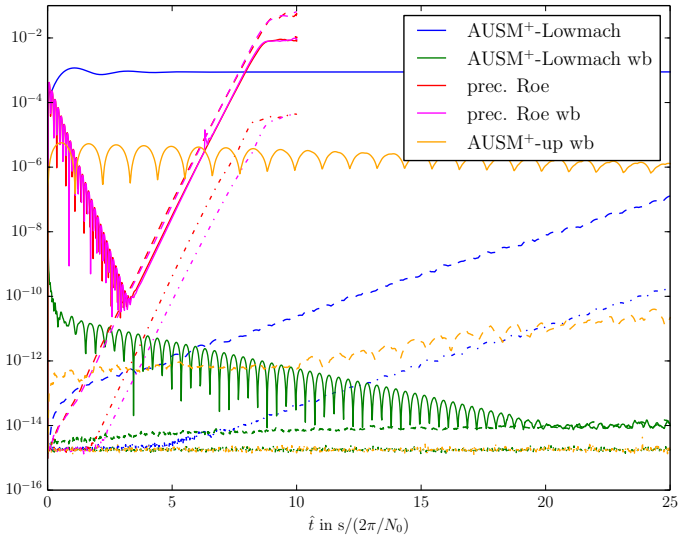


Figure 2.11: Evolution of the maximum Mach number (solid line), horizontal density fluctuation (dashed line), and horizontal pressure fluctuation (dash-dotted line). Cargo–Le Roux well-balancing is indicated by "wb". Time is measured in units corresponding to the Brunt–Väisälä frequency.

Figure 2.11 shows the evolution of the maximum Mach number and horizontal density and pressure fluctuations. The behavior of the preconditioned Roe scheme is basically unchanged by well-balancing; the instability grows quickly in both cases. The AUSM$^+$-Lowmach scheme without well-balancing does not reach Mach numbers below $\sim 10^{-3}$. This is due to the flux caused by

the p-term, which is proportional to the pressure gradient. The instability grows more slowly than in case of the Roe scheme but is still unacceptable for application. When well-balancing is used together with AUSM$^+$-Lowmach, the scheme reaches very low Mach numbers of about 10^{-14}. There are no signs of a growing instability. The well-balanced AUSM$^+$-up scheme, i.e. with the full p-term, reaches Mach numbers of about 10^{-6}. This is because multidimensional Cargo–Le Roux well-balancing does not perfectly remove the pressure gradient and the p-term causes fluxes in the vertical direction. We can also see a rise in the horizontal pressure fluctuations that is considerably slower than in the other unstable cases. To see the long term evolution, we continued the simulations with AUSM$^+$-Lowmach an AUSM$^+$-up for a longer time. The result is depicted in Fig. 2.12. AUSM$^+$-up shows a slowly growing instability that saturates at horizontal density fluctuations of $\sim 10^{-8}$. The horizontal density fluctuations in AUSM$^+$-Lowmach remain below 10^{-13}.

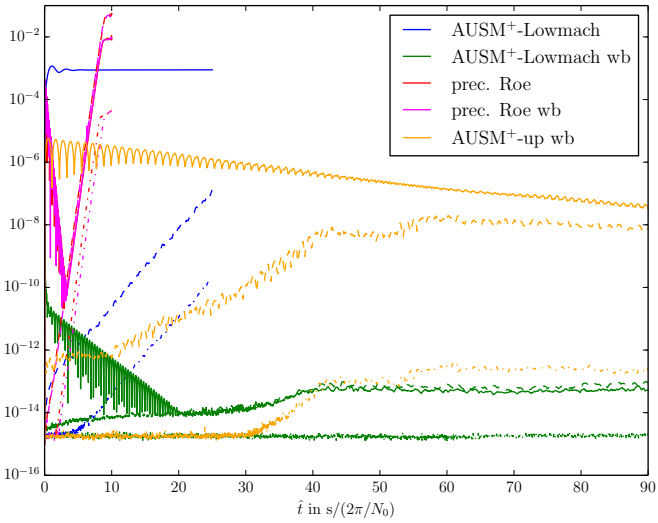


Figure 2.12: This plot is identical to Fig. 2.11 but covers a longer time scale for the well-balanced AUSM$^+$-up and AUSM$^+$-Lowmach schemes.

It is unintuitive that the scheme with the stronger pressure diffusion term, AUSM$^+$-up, experiences a growth of the instability while AUSM$^+$-Lowmach does not. We presume this is because of fluxes originating from the p-term, that appear due to non-ideal well-balancing. Further tests are necessary to verify this.

We conclude from the tests which were performed that AUSM$^+$-Lowmach combined with Cargo–Le Roux well-balancing is the preferred method in SLH to prevent the spurious instability in convectively stable atmospheres. It is also able to maintain very low Mach numbers down to $\sim 10^{-13}$. It should be kept in mind that these properties strongly depend on the effectiveness of well-balancing. The Cargo–Le Roux method has several caveats in multiple dimensions. It depends on a well-defined horizontal average, which is not available in all geometries. Furtherly, its ability to remove the hydrostatic pressure gradient is reduced when there are strong deviations from the horizontal mean density. For universal applicability a better well-balancing method must be found. Alternatively the p-term could be modified to work well with stratified atmospheres. Another viable strategy is to explore one of the methods suggested in Section 2.4.3. This could also remove the instability for the preconditioned Roe scheme.

2.5 Nuclear Reaction Networks

In order to compute the change in abundances and the energy release by nuclear reactions, the corresponding reaction rates need to be computed. Every reaction rate depends on temperature (influenced by energy release) and the abundances of all reaction partners. This produces a set of differential equations which is numerically challenging even without coupling it to the equations of hydrodynamics. The goal of this section is to introduce the relevant equations, briefly discuss solution strategies for the system without hydrodynamics and finally compare different methods for coupling nuclear reactions to hydro simulations.

2.5.1 Nuclear Reaction Rates

Nuclear reactions can be classified into categories according to the number of input nuclei involved. The simplest case is a nuclear decay in which only the decaying nucleus is involved. The two other classes are reactions with collisions of two or three input nuclei, respectively. Rates involving even more species are not relevant for the astrophysical problems considered here and are not included in the rate tables. Note that the involved nuclei may be identical, e.g. $^4\mathrm{He} + {}^4\mathrm{He} + {}^4\mathrm{He} \longrightarrow {}^{12}\mathrm{C}$. The change in number density n_i of a nucleus i due to these three categories can be cast into the equation (e.g. Hix & Meyer, 2006)

$$\left.\frac{\partial n_i}{\partial t}\right|_{\rho=\mathrm{const.}} = \sum_j \mathcal{N}_j^i r_j + \sum_{j,k} \mathcal{N}_{j,k}^i r_{j,k} + \sum_{j,k,l} \mathcal{N}_{j,k,l}^i r_{j,k,l}. \tag{2.118}$$

The reaction rates r_j, $r_{j,k}$, and $r_{j,k,l}$ are the number of reactions per volume and time with input species j, j and k, or j, k and l, respectively. The factors $\mathcal{N}$ account for the numbers of created and destroyed nuclei and include a correction for double counting identical reactants. Their definition is

$$\mathcal{N}_j^i = N_i, \tag{2.119}$$

$$\mathcal{N}_{j,k}^i = N_i / \prod_{m=1}^{n_{j,k}} (|N_m|!), \tag{2.120}$$

$$\mathcal{N}_{j,k,l}^i = N_i / \prod_{m=1}^{n_{j,k,l}} (|N_m|!). \tag{2.121}$$

N_i in the numerator is an integer indicating how many particles of species i are created or destroyed in the reaction. In the latter case the number is negative.

$n_{j,k}$ and $n_{j,k,l}$ are the numbers of *different* species that are involved in the reaction $r_{j,k}$ or $r_{j,k,l}$, respectively. N_m in the denominator is the number of reactants of the species m.

The dependence of the rates on the local physical state deserves some further discussion. The rates depend linearly on the number density of each of the input species. The reason for this can be understood by considering the different categories separately. Nuclear decays involve no interaction with other nuclei, so the rate per volume is directly proportional to the number density of the decaying species. For two-body collisions of two nuclei, the rate is found by integrating the velocity-dependent cross section over the distribution of the involved particles. Under the assumption of a Maxwell–Boltzmann velocity distribution, as valid for a plasma in thermal equilibrium, the number densities can be factored out (see e.g. Hix & Meyer, 2006). If three-body reactions are regarded as two two-body reactions in immediate succession, the resulting rate is also proportional to the number densities of the three original reactants.

The reaction network used in this work is the one described in Pakmor et al. (2012). For completeness, the constituents are briefly mentioned here. The reaction rates used in this work were taken from Rauscher & Thielemann (2000) (including the updates published by the group in 2009). In their work they combined experimentally measured reaction rates with rates from their own statistical model calculations. Additionally, weak reaction rates, i.e. rates for electron and positron capture, β^+ and β^- decay, are taken from the tables of Langanke & Martínez-Pinedo (2001). To account for the screening effect of the electron background the prescriptions of Wallace et al. (1982) with corrections from Wallace et al. (1983) are used.

For nuclear network calculations Eq. (2.118) is usually rewritten in terms of *number fractions* Y_i, defined by $Y_i = n_i/\rho N_\mathrm{A}$. The result, known as the *rate equation*, is

$$\dot{Y}_i = \sum_j \mathcal{N}^i_j \lambda_j Y_j + \sum_{j,k} \rho N_\mathrm{A} \mathcal{N}^i_{j,k} \lambda_{j,k} Y_j Y_k + \sum_{j,k,l} (\rho N_\mathrm{A})^2 \mathcal{N}^i_{j,k,l} \lambda_{j,k,l} Y_j Y_k Y_l. \tag{2.122}$$

The quantities λ_j, $\lambda_{j,k}$, and $\lambda_{j,k,l}$ are the rates with their linear dependence on the abundance of the input nuclei split off. They still depend on temperature and possibly also on density and other mass fractions due to screening effects.

Equation (2.122) constitutes a system of N_species coupled ordinary differential equations. To couple them to the Navier–Stokes equations, Eq. (2.2), the number fraction Y_i has to be transformed to the *mass fraction* X_i. The two quantities are related by $X_i = Y_i/A_i$ with the *nucleon number* A_i.

2.5.2 Solution of the Nuclear Reaction Network

Before turning to the integration of these reaction terms into the system of hydrodynamical equations, we consider solving the nuclear reaction network on its own to discuss possible difficulties. The system of ordinary differential equations, Eq. (2.122), has some properties that make its numerical solution challenging and require the use of a properly suited integration method. One of these is the fact that the rates λ for different reactions vary over many orders of magnitude. This makes it a so-called *stiff* system. A characteristic feature of these systems is that they are restricted to very small time steps if explicit methods are used for time integration. The time step is related to the fastest reaction, larger steps would make the computation unstable. Thus, implicit methods are usually used for this kind of differential equations. One method that was found to be particularly efficient was suggested by Bader & Deuflhard (1983).[7] It is the method of choice for most nuclear network computations in this thesis. The details of its numerical implementation can be found in Edelmann (2010). One major improvement that was added since is in the calculation of the Jacobian matrix, which is now computed by automated differentiation. Before, it was computed manually using only the few most important components. The new method provides a much more accurate Jacobian which allows the error-based time step estimator to chose much larger values. Overall, the additional effort taken in determining the Jacobian makes the whole method much more computationally efficient.

Another typical numerical property of nuclear networks is the structure of the Jacobian matrix of the right hand side of Eq. (2.122). Most species are only connected to the others by a few reactions, except for p, n, and ^{4}He. That means that for larger networks ($\gtrsim 100$ species) the Jacobian is very sparse. This enables the use of special methods to compute its LU decomposition, which is needed for the implicit solver. This is similar to the sparse matrix that appears in the implicit solution of the equations of hydrodynamics. The difference is that in the case of hydrodynamics the matrix is so large that storing the LU decomposition directly is out of the question. The reason is that the LU decomposition of a sparse matrix is dense, in general. For the nuclear network storage space is not an issue because the size of the matrix is negligible. This means computing the LU decomposition directly is possible and also desirable as the solution to several different vectors is needed in the Bader–Deuflhard method. The methods for sparse LU decomposition used in the context of this thesis are PARDISO (Schenk & Gärtner, 2004) and SuperLU (Demmel et al., 1999). For small nuclear networks with less than a few tens of species the use of the standard LU decomposition for dense matrix

[7] See Timmes (1999) for a comparison of different numerical methods for reaction networks.

is as efficient or even more efficient than the sparse methods. Consequently, the standard LU decomposition (e.g. from Galassi et al., 2009) is used these cases.

2.5.3 Coupling to Hydrodynamics

Nuclear reactions couple to the equations of hydrodynamics through two effects. First the changing composition has an influence on the equation of state, through the mean molecular weight or degeneracy. The second effect is the release of binding energy to thermal energy of the plasma. The latter effect could be absorbed in the equation of state by including nuclear binding energy in the total energy. We do not follow this approach in this work as it is expected that errors in the advection of the composition variables would lead to large fluctuations in the internal energy and thereby in temperature and pressure.

There are several possibilities for numerically coupling the reaction terms to hydrodynamics. The most straight-forward one is simple operator splitting. Here, one complete hydrodynamics step is performed—including all intermediate stages of the high-order Runge–Kutta methods. The resulting fluid state is used to apply the nuclear reaction network to every individual grid cell. The resulting state, changed in energy and composition, is then used as initial data for the next hydrodynamics step and the process is repeated. This type of operator splitting is called *Godunov splitting*.

It is quite obvious that this splitting strategy introduces an error that depends on the size of the time step. For example, nuclear burning releases energy in one cell, which increases temperature and pressure. The increased temperature in turn enhances the reaction rates. If there is an intermediate hydrodynamics step, the increased pressure would cause a fluid flow out of the cell. The temperature would subsequently drop and reduce the reaction rates. The shorter the interval between reaction and hydrodynamics is the closer it is to the unsplit solution. A rigorous, mathematical treatment of this matter is given by LeVeque (2002). He finds that the splitting error takes the form

$$q(x, \Delta t) - q^{**}(x, \Delta t) = \frac{1}{2}\Delta t^2 (\mathcal{AB} - \mathcal{BA}) q(x, 0) + \mathcal{O}(\Delta t^3). \tag{2.123}$$

$q(x, t)$ is the exact solution to the differential equation considered, $q^{**}(x, \Delta t)$ is the solution computed using Godunov splitting (not including any errors caused by the particular time stepping method). $\mathcal{A}$ and $\mathcal{B}$ are the two operators that are being split. This shows that the combined system is only first order accurate in time, with its error proportional to the commutator $\mathcal{AB} - \mathcal{BA}$.

One suggestion to remove the Δt^2 term in the error is the so-called *Strang splitting*. Here, one complete time step is split into a step of length $\Delta t/2$ with the operator $\mathcal{A}$, a step of length Δt with the operator $\mathcal{B}$, and another step of length $\Delta t/2$ using $\mathcal{A}$. This scheme can be further simplified by combining two successive $\mathcal{A}$ steps to one of length Δt. This makes Strang splitting identical to Godunov splitting except for the first and last steps, which are half-steps using $\mathcal{A}$. It seems surprising that the errors of the two methods should differ greatly even though they are virtually identical except for the very first and last step. Indeed tests shown in LeVeque (2002) indicate that the coefficient for the Δt^2 error term is very small and for all practical resolutions the two methods deliver nearly identical results. In the limit of $\Delta t \to 0$ the difference in order will become apparent but this range is not reached in many cases.

In the light of the marginal benefits of Strang splitting and the additional complexity introduced by the constraint that two consecutive time steps must have the same length, it is not pursued in the work of this thesis. Instead we investigate more accurate coupling methods to judge the accuracy of Godunov splitting.

It is possible to completely remove any splitting error by using an *unsplit method*. The errors are then solely determined by the errors of spatial discretization (Section 2.1.2) and of time integration (Section 2.1.3). In an unsplit method the reaction terms are added to the spatial residual function, which already includes the hydrodynamical fluxes, and then integrated in time together. This coupling method adds several additional difficulties to the solution of the system which are discussed in the following section.

All species that are part of the reaction network have to be treated as active scalars. That means they increase the number of variables per grid cell for which the system is solved simultaneously from 5 (in the three-dimensional case) by the number of species. This also affects the Jacobian matrix which normally consists of 5×5 blocks. Including a rather small 5 species network would quadruple the size of the Jacobian matrix. As the size of the Jacobian is already one of the most restrictive constraints on computational resources, the use of large networks in the unsplit method is only possible for limited test scenarios. The feasibility must be checked for the individual cases.

Another issue is the fact that nuclear reaction networks are stiff systems and need to be integrated by implicit methods as explained in Section 2.5.2. This prevents the use of explicit time stepping methods for the unsplit reactive hydrodynamics equations. This is unproblematic for the type of systems targeted in this thesis as they are at very low Mach numbers and benefit from implicit methods anyway. But even if stability is not a concern, accuracy has to be considered. If the time step is simply chosen according to the advective CFL condition, it might be too large to resolve some of the fast reactions and

yield inaccurate results, even compared to a split method. The error estimator present in the ESDIRK time stepping schemes can be used in this case to ensure time steps stay small enough. In certain cases, however, this time step might be prohibitively small. We will investigate the feasibility for a number of test cases in the following section.

If the unsplit methods are computationally too demanding and Godunov splitting turns out to be too inaccurate, other methods of coupling could be investigated. A possibility for this is running the nuclear burning under the assumption of constant pressure (i.e. add the energy release to the enthalpy of the system and compute the change in temperature from density and enthalpy) and release the energy from burning not instantaneously at the beginning of a time step but rather at a constant rate during the step.

2.5.4 Test on a One-Dimensional Deflagration Front

We try the different coupling methods on a simple test problem, to see what the impact of the different coupling schemes is. Additionally, we make a comparison with another hydrodynamics code to check for any obvious errors in the integration of the nuclear network into SLH. The tests are performed in one spatial dimension.

One problem in which the coupling between hydrodynamics, nuclear reactions, and also thermal conduction are important is that of a deflagration front in white dwarf matter. This problem appears in the context of Type Ia supernovae.[8] The problem is clearly in the subsonic regime and consequently well suited to the hydrodynamics code used in this thesis. For comparison we use a hydrodynamics code that was originally developed for the purpose of simulating thermonuclear flames by Edelmann (2010). We call it Hydro1d in the following. It combines the explicit Piecewise-Parabolic Method (PPM) of Colella & Woodward (1984) with a Riemann solver for real gases (Colella & Glaz, 1985). The equation of state is the same as in SLH (Timmes & Swesty, 2000). The nuclear reaction network is identical as well. Thermal conduction in Hydro1d is handled using the Crank-Nicolson method, which is an implicit method that is second-order accurate in time and space. Hydro1d uses Godunov splitting to couple hydrodynamics, nuclear reactions, and thermal conduction. SLH also uses a second-order spatial discretization of the thermal conduction term but couples it to hydrodynamics in an unsplit way. Both codes use exactly the same formula for the conductivity of degenerate electrons.

The initial setup is a homogeneous plasma of 50% ^{12}C and 50% ^{16}O at a constant density of $10^9\,\mathrm{g/cm^3}$ and temperature of $10^8\,\mathrm{K}$. To ignite the flame,

[8] See Hillebrandt & Niemeyer (2000, Section 5) for a review of flames in Type Ia supernovae.

the temperature in a small region on one side is raised to 5×10^9 K. This also increases pressure and causes a shock to propagate through the cold material. The boundary condition allows the outflow of material, so pressure equilibrium is quickly reestablished. This leaves the hot region at a lower density. This ignition procedure is merely a convenient way of setting up a steadily burning deflagration flame and does not reflect the real physics of ignition.

The initial relaxation phase is computed in Hydro1d and the result is then transferred to SLH. We do this to avoid any differences in the results that are simply due to the treatment of the initial shock wave, in which we are not interested in here. The propagation of the flame is then followed in Hydro1d as well as in SLH, using different coupling and time-stepping for the latter. Godunov splitting together with passive treatment of composition variables in the sense of Section 2.1.3 is termed "passive scalars" in the plots. The unsplit method combined with active treatment of the composition is denoted by "active scalars". Time-stepping was performed implicitly in all calculations with SLH, except for one. The choice of time step was either due to the CFL_u criterion or by setting a relative error threshold for the adaptive time-stepper (see Section 2.1.3). Hydro1D is a purely explicit code, thus the time step was chosen according to the CFL_{uc} criterion. The same criterion was used the explicit SLH calculation.

Figure 2.13 shows the profile of the ^{12}C mass fraction for the different methods. The region with the strongest deviation of the different models is at the front of the deflagration, where the ^{12}C content starts to drop. Hydro1D differs, if just slightly, from the SLH results. The hydrodynamics method in Hydro1D uses parabolic reconstruction and is therefore of higher order than the SLH runs. They were performed using linear reconstruction. This is the probable explanation for the small separation of results. It is verified with an explicit, operator-split simulation, whose method and time step is identical to Hydro1D. The only difference lies in the order of the hydrodynamics method. This explicit calculation also deviates from Hydro1D and lies very close to the unsplit result.

Another property we want to investigate is how well the hydrodynamics time-stepper can compute the pure network without the influence of hydrodynamics. For this, we look at the temporal evolution of the ^{12}C mass fraction in the cell directly at the left boundary, shown in Fig. 2.14. This cell is near to completely free of the effects of hydrodynamics and of thermal conduction. The curves for the three unsplit calculations, and Hydro1D lie essentially on top of each other. The operator-split calculations show a slight deviation from this curve, except for the calculation with explicit time-stepping. The operator-split calculation using adaptive time-stepping with a threshold of 10^{-4} deviates most from the others. The cause becomes obvious with Fig. 2.15.

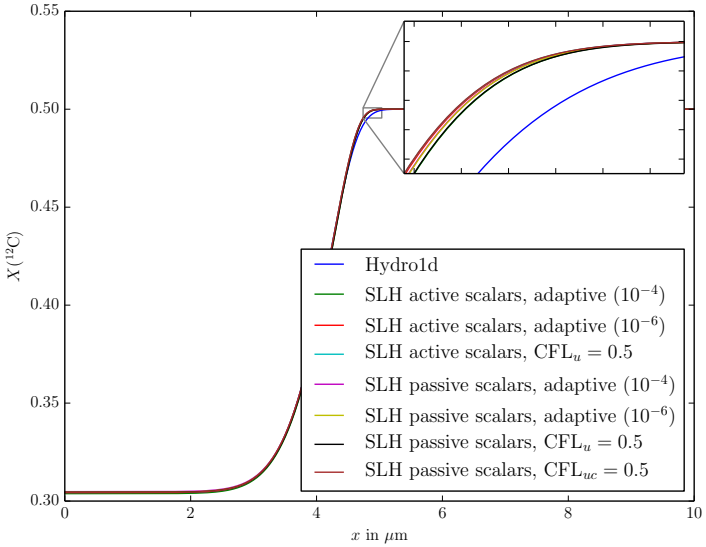


Figure 2.13: Profile of the ^{12}C mass fraction of a one dimensional deflagration for different methods. The number in parentheses after adaptive is the relative error threshold of the adaptive time-stepper.

Larger time steps are correlated with larger deviations for the "passive scalars", while the "active scalars" are accurate even for the largest time steps. The reason for this behavior is that the operator-split calculations only increase the temperature between the steps, while the unsplit method accounts for the change in temperature also during one step. A similar behavior could be achieved in the operator-split case if the temperature change were tracked during a network time step assuming a predetermined thermodynamic behavior. Pakmor et al. (2012) discusses this possibility.

For the simulations presented in Chapter 3 we use the operator-split method in combination with the CFL_u time step. It is significantly cheaper computationally and delivers results that are still accurate enough for these first proof-of-concept calculations.

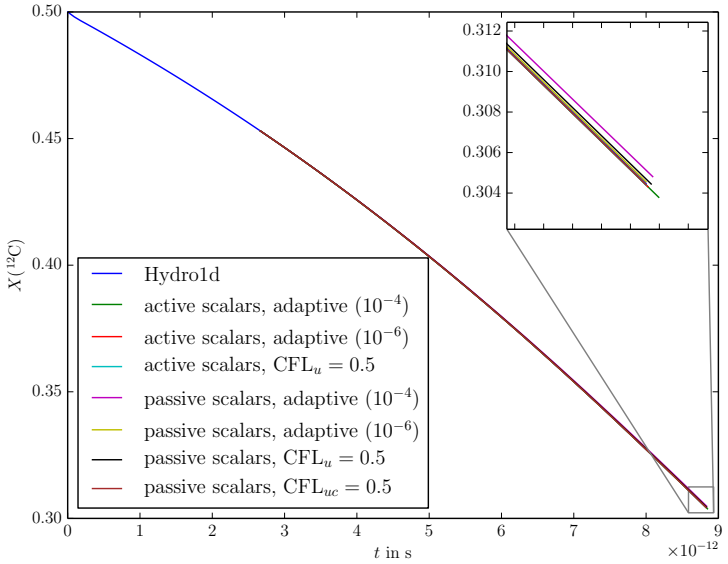


Figure 2.14: Temporal evolution of the ^{12}C mass fraction at the left boundary of the domain.

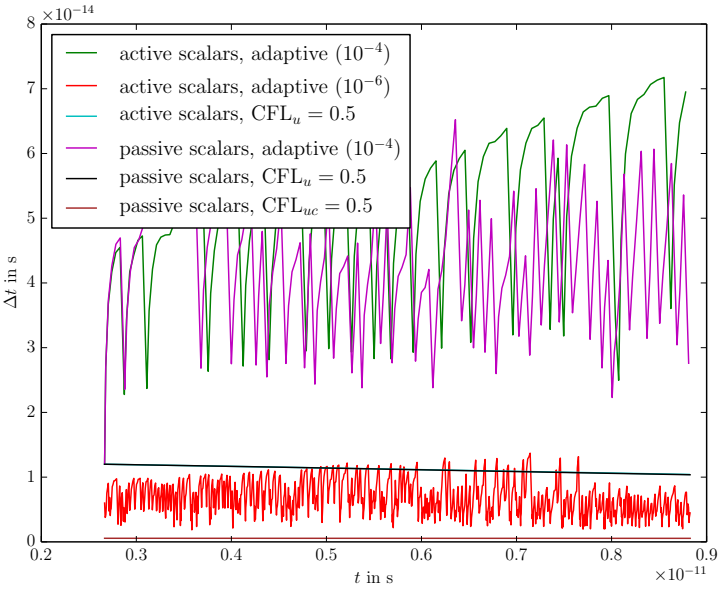


Figure 2.15: Time-step size in the 1D deflagration simulation according to different criteria.

3 Convective Mixing in Massive Population III Stars

As a first astrophysical application of the methods developed in the previous parts of this thesis, we turn to convective mixing in massive Population III (Pop-III) stars. More precisely, we simulate the boundary between the convective helium burning core and the non-convective hydrogen shell on top of it. As Pop-III stars are completely deficient of metals, all hydrogen burning occurs through the pp chain. When carbon is mixed from the core into the hydrogen shell, the CNO cycle is enabled, which is orders of magnitude more efficient in burning hydrogen. There is doubt in the stellar evolution community on what the correct treatment of mixing at this interface is. The goal is to gain more insight into this process using multidimensional hydrodynamics simulations including nuclear burning.

3.1 Introduction to Massive Population III Stars

Pop-III stars are the first stars that formed in the universe. They are identified by their complete lack of elements heavier than helium, except for a tiny fraction of Li. As no other stars could have enriched the interstellar medium with their burning products previously, Pop-III stars form with the primordial abundance of the elements. This has important implications for the structure and evolution of these stars.

Stars of solar metallicity ($Z = 0.02$) become pulsationally unstable above a mass of about $60\,M_\odot$ (Schwarzschild & Härm, 1959). The underlying phenomenon is called the *ε-mechanism* caused by an increase in nuclear energy generation during contraction of the star (for details see e.g. Kippenhahn et al., 2013). Stars considerably heavier than this limit are expected to shed their mass in shells at each pulsation until it is reduced to the critical value of $60\,M_\odot$. However, it is still unresolved whether the growth rate of these instabilities is large enough to have a significant impact during the main-sequence lifetime of these stars.

The situation is different for Pop-III stars. These lack elements heavier than helium and therefore cannot perform hydrogen burning through the CNO cycle,

but only through the pp chain, even though they have core temperatures that would make the former much more efficient. Consequently, their hydrogen burning region is much hotter and more compact than in metal-enriched, massive stars. Baraffe et al. (2001) performed an analysis of pulsational instabilities in very massive primordial stars ($M > 120\,M_\odot$). They found that the lack of metals causes the instabilities to grow on much longer timescales and the growth is stopped while the star is still in the hydrogen burning phase. Even stars with masses of $300\,M_\odot$ only lose $8.5\,M_\odot$ until they become pulsationally stable. This leads to the conclusion that if star formation produces such objects, they can survive all burning stages up to core collapse without losing much of their mass due to pulsations. As very massive Pop-III stars reach different regimes of density and temperature at the core, their final fate is very different from their metal-enriched counterparts. The reason is that electron–positron pairs can be created, which take up part of the internal energy stored in gas and radiation, in their rest mass energy. This reduces the pressure and leads to the collapse of the star. The contraction triggers explosive thermonuclear oxygen and silicon burning, which produces enough energy to disrupt the whole star, leaving no compact remnant behind. This scenario, called *pair instability supernova*, was first described by Barkat et al. (1967) and explored in the context of very massive stars by Bond et al. (1984). As these explosions occur in the first generation of stars, which formed even before the first galaxies, and produce large amounts of ^{56}Ni (which ultimately decays to ^{56}Fe), they play an important role in the early chemical evolution of the Universe. Heger & Woosley (2002) performed a systematic study of the nucleosynthetic signature of Pop-III stars with masses from $140\,M_\odot$ to $260\,M_\odot$. This covers the whole range of stars that are expected to explode as pair instability supernovae. Lower-mass stars would undergo core collapse with direct black hole formation. More massive stars ($\gtrsim 260\,M_\odot$) are still subject to the pair instability, but the energy released in the following thermonuclear explosion is not sufficient to overcome the collapse and the star forms a black hole (Fryer et al., 2001). A distinct feature of nucleosynthesis of pair instability supernovae is that all isotopes above ^{14}N with odd charge number and neutron-rich isotopes are significantly underproduced. There is basically no production of elements beyond the iron group either, since the conditions for neither the r- nor the s-process are fulfilled in Pop-III stars. Heger & Woosley (2002) suggest that pair instability supernovae could have left a detectable signature in metal-poor stars ($[\mathrm{Fe/H}] = -4$).

The predictions made above depend on stellar evolution models, which employ simplified prescriptions for all phenomena that cannot be treated in a one-dimensional, hydrostatic code. Two of these are the treatment of mixing in convection zones and that of convective overshooting. If a region is determined

to be convectively unstable by either the Schwarzschild or Ledoux criterion (see Section 2.3.2), the composition is evenly distributed over all mass shells in the region. This is a reasonable assumption if the timescales of nuclear burning are much longer than those of convection. In certain situations, however, this is not valid. Marigo et al. (2001) notice this problem in their simulations of Pop-III stars. During the central He-burning phase the boundary of the convective part of the core moves outward until it reaches the surrounding H-shell. When this happens, hydrogen is mixed into and distributed over the whole core, not just its outermost regions. At this point, the core has already produced a sufficient fraction of ^{12}C through the 3α-reaction to support the CNO cycle. The addition of hydrogen causes a flash, i.e. a sudden release of energy by nuclear burning. This flash causes the core to expand the convective region to recede below the H-shell. This stops the mixing of hydrogen into the core, the structure slowly returns to the pre-flash state, and the process is repeated when the convective zone grows to reach the H-shell again. The result are quasi-periodic H-flashes. Marigo et al. (2001) argue that these flashes are unrealistic because they are only caused by the unphysically strong mixing of hydrogen into the core. For lack of a more realistic solution they suppress these flashes by artificially constraining the core convective region to the bottom of the H-shell. It is clearly desirable to investigate the interplay of convective mixing and nuclear burning at this interface to find out whether prescriptions as mentioned above are justified or have to be replaced with a more physical model.

3.2 The Stellar Model

As a basis for the three-dimensional modeling of the boundary of a convective core we received several snapshots of a stellar evolution calculation of a non-rotating 250 $M_\odot$ Pop-III star (Alexander Heger, private communication). The stellar model was evolved using the one-dimensional, implicit hydrodynamics code KEPLER (Weaver et al., 1978; Heger et al., 2000). Figure 3.1 shows the model during core He-burning. A fraction of about 0.2 of ^{4}He has already been converted to ^{12}C by the 3α-process. The center of the core is convective but the convective region does not yet extend to the H-shell. This shell is completely devoid of any of the elements that act as catalysts for the CNO cycle. H-burning therefore only proceeds through the pp chain despite the temperature of more than 100 MK at the bottom of the shell. During the following evolution of the star, the convective region in the core grows and reaches the H-shell at some point. When this happens, a small fraction of ^{12}C is mixed into the hydrogen-rich region and enables burning via the CNO cycle. The increased

energy release changes the temperature profile, so that a small convectively stable region forms at the upper boundary of the core, which separates the convective zones at the center and at the bottom of the H-shell.

Figure 3.2 shows the model about 20 000 years after the state displayed in the previous figure. Here, the bottom of the H-shell is convective and is burning via the CNO cycle due to a ^{12}C mass fraction of 10^{-5}. This convective zone is separated from the convective core through a small radiative layer. The region above the convective part of the H-shell is still devoid of CNO isotopes and therefore burning hydrogen via the pp chain.

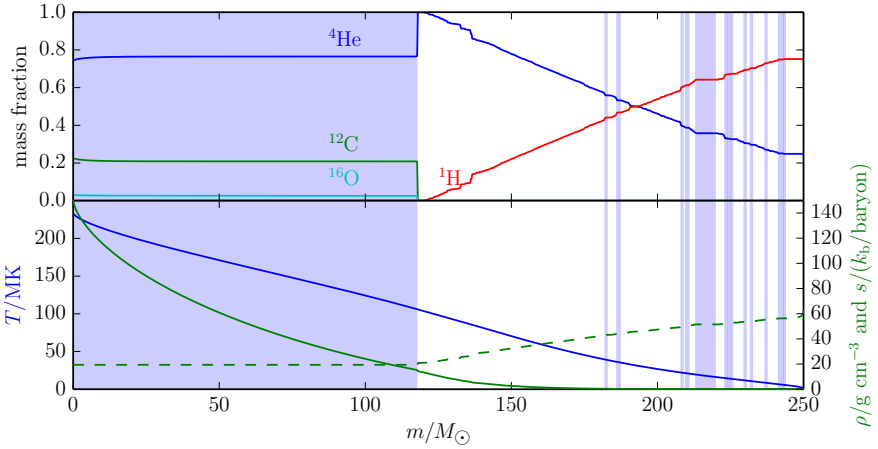


Figure 3.1: A 250 $M_\odot$ Pop-III star model during core He-burning, before the central convective region reaches the H-burning shell. The upper panel shows the composition structure in mass coordinates. Convective regions are shaded in blue. The lower panel shows temperature (blue), density (green), and entropy (green dashed).

3.3 Setup of the Three-Dimensional Simulation

In order to study the mixing of species at the boundary between the He-burning core and the H-burning layer above, we pick a section of the model that encompasses the top part of the convective core and the whole convective region at the bottom of the H-shell. This section is mapped to the computational grid of the multidimensional hydrodynamics code SLH. In the model from Fig. 3.1 the convective part of the core has not yet reached the H-shell, so no mixing is

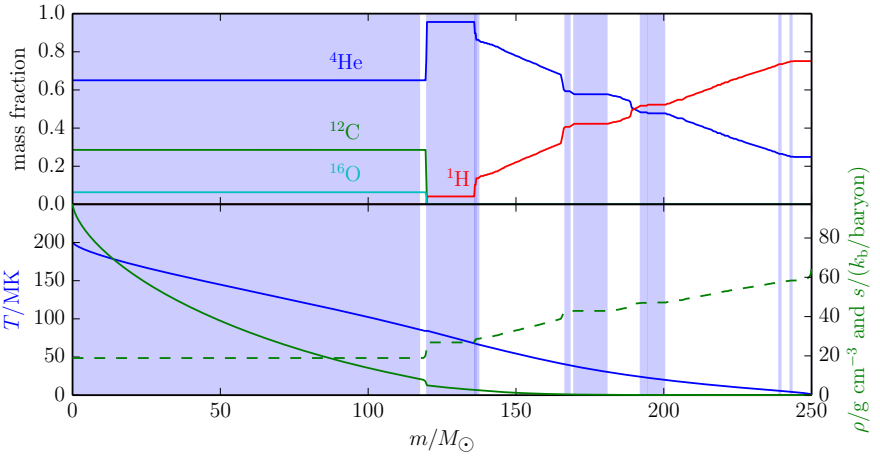


Figure 3.2: The model from Fig. 3.1 in a state roughly 20 000 years later. The central convective region now partially encompasses the H-burning shell. The upper panel shows the composition structure of a 250 $M_\odot$ Pop-III star in mass coordinates. Convective regions are shaded in blue. The lower panel shows temperature (blue), density (green), and entropy (green dashed).

expected here. Starting with this model and waiting for the convective region of the core to expand to the H-shell is not feasible either, because the growth is tied to the nuclear timescale of He-burning, which is much larger than the hydrodynamical timescale. Hydrodynamical simulations with the required resolution can be performed for a physical time of several weeks at most. A significant growth of the core would, however, take tens to hundreds of years, as illustrated by the comparison of Figs. 3.1 and 3.2, which are 20 000 years apart. For this reason we choose the later one of the two snapshots, which has already undergone significant mixing at the interface.

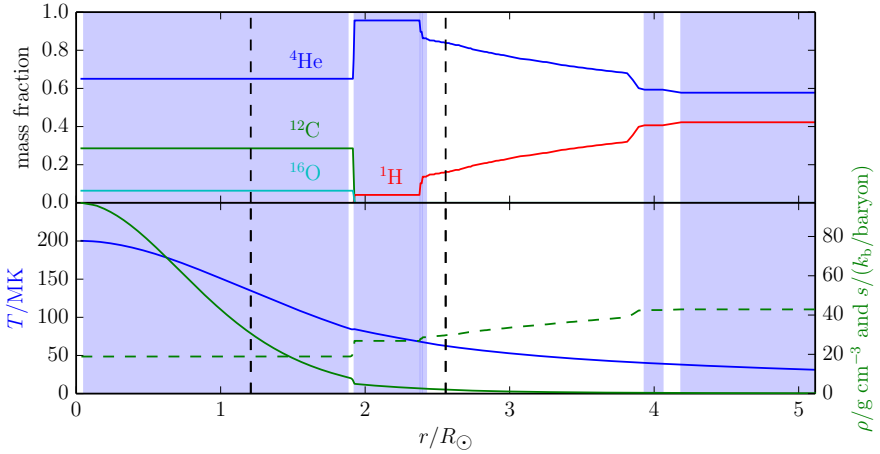


Figure 3.3: The inner part of the stellar model from Fig. 3.2 is shown in radial instead of mass coordinates. The vertical dashed lines indicate the size of the domain chosen for multidimensional simulations.

The computational domain is centered at the top of the core and extends radially in both directions for $2\,H_p$. This uses the pressure scale height H_p, defined by Eq. (2.80), at the center of the domain, for reference. This domain includes the whole convective zone at the bottom of the H-shell. Figure 3.3 shows where the chosen region is located in the stellar model. It is now displayed in radial coordinates instead of mass. The radius of the whole star is $90\,R_\odot$. This illustrates that the core carries about half of the mass of the star while, due to the high density contrast between core and envelope, its radial extent is only 2% of the stellar radius. The inner and outer radii of the domain differ by a factor of 2, which makes the plane parallel approximation to this problem clearly invalid. We still use a Cartesian box as the computational

grid for the first proof-of-concept simulations as it reduces the complexity of the problem and allows us to focus on the interaction of hydrodynamics and nuclear burning. This means, of course, that before any definite conclusions about stellar models can be drawn the simulations should be repeated with a more realistic grid geometry. This could either be a spherical wedge in which the core and the poles have been cut out to avoid the singularities or even a grid of the type shown in Fig. 2.1. The latter geometry would enable us to simulate the whole convective core of the star, consequently removing the inner boundary and thereby avoid numerical artifacts there. As the implementation of a gravity solver in SLH is still pending, full-core simulations have to be postponed to future investigations outside the scope of this thesis.

For the horizontal extent of the computational domain we chose the same as for the vertical in order to not suppress or enhance convective motions through boundary effects. This is probably an overestimate since the convective structures are much smaller than the domain, but it serves as a conservative estimate.

Models of stellar structure are computed under the assumption of perfect *hydrostatic equilibrium* (see Section 2.3.1). We should also assume this for the initial state of the multidimensional simulation. If the quantities from the stellar model are just naïvely interpolated to the finer hydrodynamics grid, the resulting stratification is not in perfect equilibrium and will experience large accelerations at the beginning of the simulation.

We use a special technique to ensure that the initial conditions are in good agreement with hydrostatic equilibrium. We start from the ordinary differential equation (ODE) for the hydrostatic pressure, Eq. (2.73),

$$p'(y) = \rho\left(p(y), X_i(y), T(y)\right) g_y(y). \tag{3.1}$$

This ODE can be solved numerically using values for the composition X_i, temperature T, and gravity g_y from the stellar model. Alternatively, specific entropy s can be used instead of T. An even simpler method is to interpolate the values of density ρ directly and determine p by numerical integration. In the interpolated model, only the "given" quantity will match the original stellar model exactly; the other quantities are determined by hydrostatic equilibrium and the equation of state.

The results of using these different methods are shown in Fig. 3.4. An important feature of the original model is the location of the convectively unstable zones. Each panel shows the original convective regions at the bottom and the convective regions of the interpolated model at the top. We find that the models with given density and given entropy reproduce the non-convective region at the entropy jump equally well; the model with given temperature

makes this region significantly larger. The model with given density introduces a few additional convective zones at large radii which are not present in the original model; for the other two models this is not the case. These observations imply that it is best to perform the interpolation in specific entropy and to compute density and temperature from that, according to the equation of state and hydrostatic equilibrium.

The first multidimensional tests found the entropy jump occurring at the upper end of the stable region at the grid center to cause convergence problems in the hydrodynamics code. In turn, the cause of this discontinuity lies in the treatment of convection in stellar evolution codes. Constant values of composition and entropy are enforced in the convective regions. This leads to abrupt changes in the entropy profile at the interfaces of these regions. As the jump is only resolved within a few mass shells, we do not expect this to be the true physical state. In order ease the initial mapping of the model onto the multidimensional grid we employ *smoothing* of the data. It is performed by averaging over a moving window of 11 data points, weighted with the Hann function.[1] Again all three choices of interpolation quantity were tried using smoothing. The results are shown in Fig. 3.5. The position of the non-convective zone at the center is still best preserved by the models with given density or entropy but it is understandably not as good as in the unsmoothed case. At large radii the models are nearly indistinguishable. No additional convective zones are created in the model with given ρ this time, but the two small convective regions of the original model are no longer reproduced, either. It might be possible to improve the mapping of the model by applying smoothing only in selected regions or changing the size of the smoothing window, but for the initial proof-of-concept calculations here we settle on interpolation in entropy using the smoothing described above.

The choice of proper *boundary conditions* is very important for hydrodynamics simulations, especially when the computational grid is immersed in a larger object, like a star, and does not cover it in its entirety. Many boundary conditions are expressed using *ghost cells*. These are cells just outside the computational grid that influence the flux through the cells at the boundary. The boundary conditions for the vertical fluxes are of particular importance for hydrostatic atmospheres. Outflow boundaries that extrapolate from the state right at the boundary are not suitable because the atmosphere starts to flow "out" of the grid then. This happens because of a self-amplifying effect caused by vertical velocities at the boundary. These are extrapolated into the ghost cells, which then trigger an increase of the velocity at the boundary. An alter-

[1] The Hann function is given by $w_j = \frac{1}{2}\left[1 - \cos(2\pi j/(N-1))\right]$, where j is the index of the point in a window of size N (Press et al., 1992).

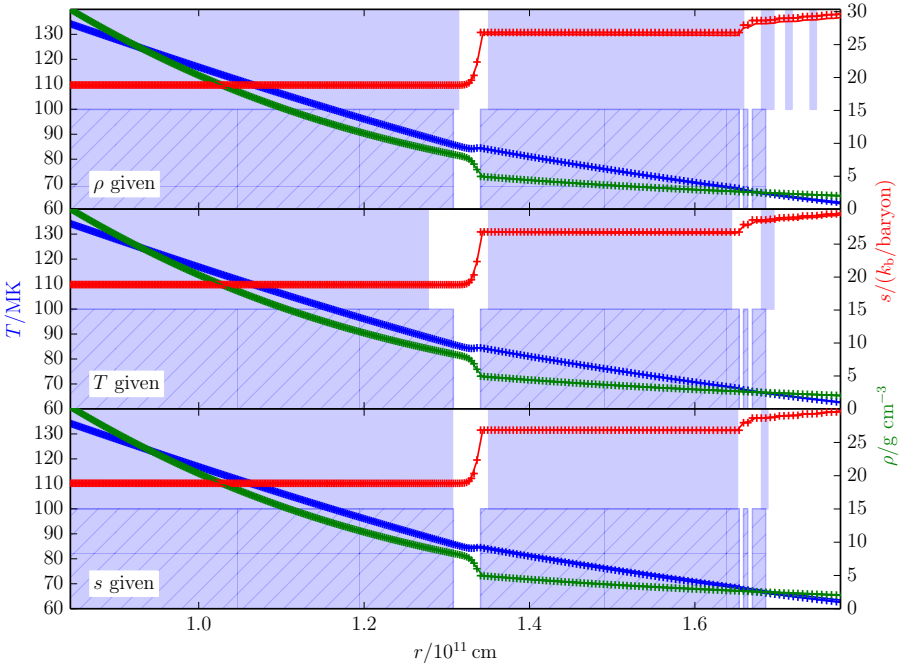


Figure 3.4: This shows the vertical profile of temperature (blue), density (green), and specific entropy (red) in the multidimensional simulation box. The points marked (+) stem from the one-dimensional model; the line is the interpolated profile. The shaded areas at the bottom indicate regions in which the original stellar model was convectively unstable. The shaded areas on top show the regions that are unstable in the multidimensional model, according to the Ledoux criterion. The different panels show the result of using ρ, T, or s as input variable for the integration of hydrostatic equilibrium.

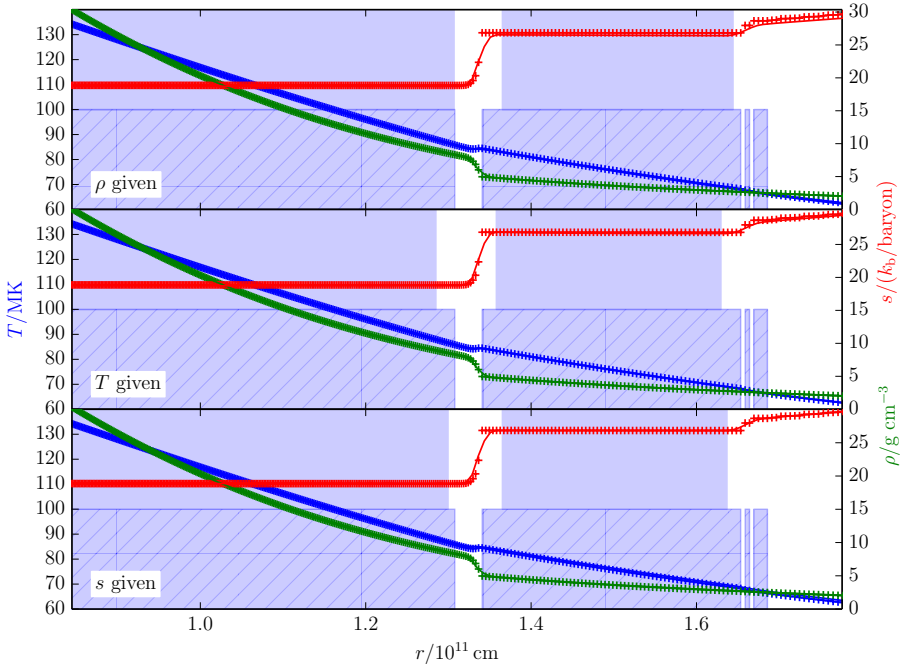


Figure 3.5: This plot is identical to Fig. 3.4 except that smoothing was used to remove steep jumps from the stellar evolution model.

native are boundary conditions that enforce zero flux through the boundary. Possibilities implemented in SLH are reflective and wall boundary conditions. Reflective boundaries fill the ghost cells with mirrored values from the grid and invert the direction of velocity. The wall boundary is a *flux boundary* condition, i.e. it does not use ghost cells but computes the flux at the boundary directly. Both of these boundary conditions show spurious oscillations at the vertical boundary that span a few grid cells. Another simple yet very effective type of boundary condition are *constant ghost cells*. Here the values in the ghost cells are set at the beginning using values from the initial profile and zero velocity. They are then left untouched for the whole simulation. This boundary condition can keep the initial hydrostatic profile very accurately without any obvious artifacts. It allows in- and outflow as the last grid cell before the boundary can have a non-zero velocity. Still the velocities in the cells adjacent to the boundary are dampened because of the zero velocity ghost cells. There are also so-called *far-field boundary conditions* that allow hydrodynamically consistent in- and outflow according to a prescribed background state of the fluid far away from the grid. While these are excellent boundary conditions, they cannot be used in the case of a stratified atmosphere, because the far-field state depends on the height in the atmosphere. In the future hydrostatic far-field boundaries could be designed, possibly using an approach similar to the well-balancing technique from Section 2.4.4. For the simulations in this thesis we choose constant ghost cells.

The boundary conditions for the horizontal fluxes also deserve some consideration. Periodic boundary conditions seem physically reasonable and are numerically well behaved. Yet we noticed in numerical tests that shear flows develop on the grid. These are not expected from the stellar model, which did not include rotation effects. As the simplified plane-parallel geometry of the simulation box does not accurately represent the geometry of the stellar core, it is not justified to say that the shear naturally develops in this part of the star. To investigate this, future simulations should employ a more realistic geometry either using spherical coordinates with a cut-out center or even a curvilinear grid as shown in Fig. 2.1. To prevent these shear flows from developing we use the wall boundary condition described above.

As the hydrodynamics method we choose the AUSM$^+$-Lowmach scheme including Cargo–Leroux well-balancing. This method resolves hydrostatic equilibrium very accurately and shows no checkerboard instabilities as seen in the test in Section 2.4.5. We use the Helmholtz equation of state (see Section 2.1.1) including the effects of an ideal gas of ions, radiation, and degenerate electrons. The contribution of degeneracy is negligible. It is included nevertheless because the additional computational cost of the equation of state

has only a minor impact on the overall efficiency, as implicit time stepping needs fewer calls to the equation of state.

A suitable *nuclear reaction network* needs to be chosen. The network included in SLH automatically chooses reaction rates according to the chemical species given. So the choice of network is in fact determined by the choice of included species. There are three main nuclear processes occurring in the selected part of the stellar model. In the lower half of the atmosphere, which consists of ^{4}He and ^{12}C, ^{4}He is fused to ^{12}C by the triple α process. In the upper half of the atmosphere, where ^{1}H is present, the pp chain is turning ^{1}H to ^{4}He. It does not need the presence of any other element as a catalyst. The remaining process is the CNO cycle, which only occurs in regions where ^{1}H and at least a small fraction of CNO elements (i.e. carbon, nitrogen, or oxygen) are present. The CNO cycle is a catalytic process, which involves the following reactions (e.g. Kippenhahn et al., 2013):

$$\begin{aligned}
^{12}\mathrm{C} + {}^1\mathrm{H} &\longrightarrow {}^{13}\mathrm{N} + \gamma \\
^{13}\mathrm{N} &\longrightarrow {}^{13}\mathrm{C} + e^+ + \nu \\
^{13}\mathrm{C} + {}^1\mathrm{H} &\longrightarrow {}^{14}\mathrm{N} + \gamma \\
^{14}\mathrm{N} + {}^1\mathrm{H} &\longrightarrow {}^{15}\mathrm{O} + \gamma \\
^{15}\mathrm{O} &\longrightarrow {}^{15}\mathrm{N} + e^+ + \nu \\
^{15}\mathrm{N} + {}^1\mathrm{H} &\longrightarrow {}^{12}\mathrm{C} + {}^4\mathrm{He}
\end{aligned}$$

This is known as the CNO-I cycle. Branching is possible in the last reaction:

$$\begin{aligned}
^{15}\mathrm{N} + {}^1\mathrm{H} &\longrightarrow {}^{16}\mathrm{O} + \gamma \\
^{16}\mathrm{O} + {}^1\mathrm{H} &\longrightarrow {}^{17}\mathrm{F} + \gamma \\
^{17}\mathrm{F} &\longrightarrow {}^{17}\mathrm{O} + e^+ + \nu \\
^{17}\mathrm{O} + {}^1\mathrm{H} &\longrightarrow {}^{14}\mathrm{N} + {}^4\mathrm{He}
\end{aligned}$$

This is the CNO-II cycle. The ^{14}N produced in the last reaction can then enter the cycle again at the fourth line of the CNO-I cycle. This cycle is about 10^3 times less probable than the CNO-I cycle (Kippenhahn et al., 2013) thus it has only a minor impact on the reaction rate. It shows that the presence of ^{16}O alone is enough to activate the CNO cycle.

To decide which processes are relevant on the dynamical timescale of this setup, we calculate the energy release rate of the different processes. The sound crossing time through the simulation box is about 8 min. Even the assumption of conservatively low Mach numbers in the convective zone of 0.01 results in a dynamical timescale of only 13 days. For the triple alpha process under its most favorable conditions, at the very bottom of the grid, the energy release rate is 0.3 erg/(g s). The energy release rate from the pp chain would amount to 300 erg/(g s), under the extreme assumption that

the highest mass fraction of ^{1}H on the grid were mixed down to the bottom of the domain. The CNO cycle is more efficient than the pp chain above temperatures of $\sim 1.5 \times 10^7$ K (Kippenhahn et al., 2013). Indeed, under the same assumption as above, and adding a ^{12}C mass fraction of just 10^{-8}, it results in an energy release rate of 2×10^8 erg/(g s). These are just thought experiments to explore the limits of the different processes. The process that will have the largest energy release in the actual simulations is burning of ^{1}H in a very ^{12}C-rich environment. The ^{12}C abundance in the core is 0.28. The ingestion of $X(^1\mathrm{H}) = 10^{-5}$ into the core already leads to an energy release of 6×10^9 erg/(g s). The largest energy release is expected directly at the boundary between the ^{12}C-rich core and the ^{1}H-rich shell above. This effect can be understood with the formula for the energy release from the CNO cycle (e.g. Kippenhahn et al., 2013)

$$\begin{aligned} \epsilon_{\mathrm{CNO}} &= 8.24 \times 10^{25} g_{14,1} X_{\mathrm{CNO}} X_{^1\mathrm{H}} \rho T_9^{-2/3} \\ &\quad \cdot \exp\left(-15.231 T_9^{-1/3} - (T_9/0.8)^2\right), \end{aligned} \tag{3.2}$$

$$g_{14,1} = (1 - 2.00T_9 + 3.41T_9^2 - 2.43T_9^3), \tag{3.3}$$

where $T_9 = 10^{-9}T$. The quantities ϵ_{CNO} and ρ are in CGS units. In the original stellar evolution model, the mass fraction of ^{12}C in the upper convective zone was 10^{-5}. If X_{CNO} is raised to 0.1 by mixing, the reaction rate increases by a factor of 10^4.

We conclude that it is sufficient to reproduce the CNO-I cycle. Even in regions without CNO elements, where only the pp chain can take place, it is so slow that we do not need to follow it in the dynamical simulation. The same is true for the triple α process. We choose to use 10 species in the 3D simulation: ^{1}H, ^{4}He, ^{12}C, ^{13}C, ^{13}N, ^{14}N, ^{15}N, ^{15}O, and ^{16}O. ^{4}He and ^{16}O are just included because in the original model, they are present in a non-negligible mass fraction, that we want to preserve.

We also include radiation transport in the diffusion limit. This is important because it ensures that the convective regions from the stellar model stay convective, where expected. Without radiation transport, convection would flatten the unstable temperature gradient until the region becomes stable. We use the OPAL Rosseland mean opacities tabulated by Iglesias & Rogers (1996). The opacity varies from 0.16 cm^2 g^{-1} to 0.18 cm^2 g^{-1} over the whole simulation domain. For simplicity, we do not implement a live table look-up during the 3D simulation, but instead assume a mean value of 0.167 cm^2 g^{-1} for the opacity. This approximation has no notable effect on the properties of radiation transport and convection.

3.4 Results

To get an overview of the effects to be expected in the 3D simulation, a 2D simulation was performed priorly. A simulation box with a resolution of 128 grid cells in the vertical direction and 96 cells in the horizontal direction was used. The horizontal size of the domain was chosen so that the cells are of square shape, in order for the resolutions in horizontal and vertical direction to be the same. Due to the relatively low computational cost of the 2D simulation it could be followed for 58 days of physical time.

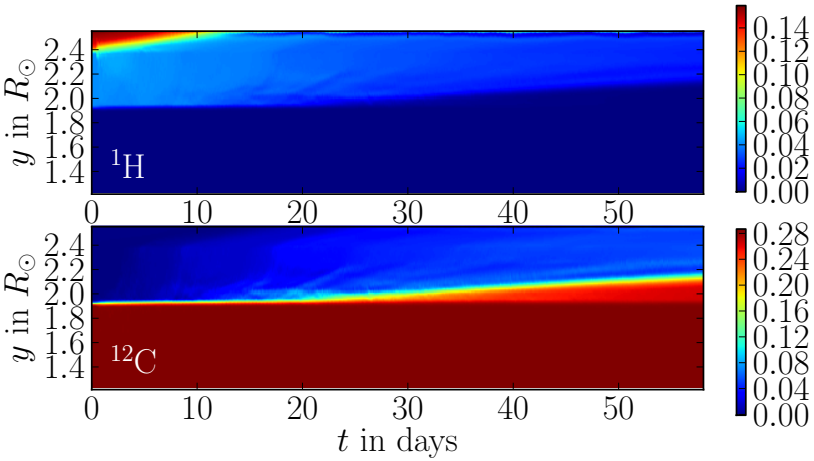


Figure 3.6: Temporal evolution of the horizontally averaged mass fractions of ^{1}H (top panel) and ^{12}C (bottom panel) against the vertical coordinate y given in solar radii. The data are from the 2D simulation.

Figure 3.6 shows the temporal evolution of the horizontal average of the abundance profiles of ^{1}H and ^{12}C. After a few days, ^{12}C is getting mixed into the ^{1}H-rich layers. After about 20 days, the boundary between the core and the shell above begins to move noticeably outward. The cause is the fast burning of ^{1}H in the ^{12}C-enriched environment. The shell is consumed from below, where the temperature is higher, because the efficiency of the CNO cycle is highly sensitive to temperature.

Initially, there is a higher fraction of ^{1}H in the convectively stable region at the top of the domain, which disappears after about 10 days. This is not due to nuclear burning; the temperature in that region is not high enough. Instead, mixing with the convective layer below is the cause.

These effects are in contradiction to the predictions of the 1D stellar evolution model, where much less ^{12}C is mixed through convective overshooting. Hydrogen burning in the shell above the core progresses much more efficiently in the 2D hydrodynamical simulation. Still, no definite conclusions should be drawn from this first study. The end of this section discusses required improvements to the simulation setup.

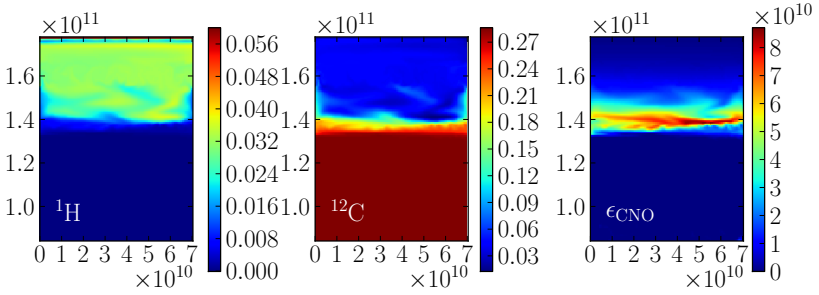


Figure 3.7: Mass fraction of ^{1}H (left panel), mass fraction of ^{12}C (center panel), and nuclear energy release rate due to the CNO cycle in erg/(g s). The axes are the Cartesian coordinates in cm. The data are from the 2D simulation, they show a snapshot after 30 days. The artifact at the upper edge results from the constant ghost cell boundary condition.

In Fig. 3.7, a snapshot from the 2D simulation shows the mass fractions of ^{1}H and ^{12}C and the nuclear energy release rate due to the CNO cycle. In the ^{1}H mass fraction, an artifact can be seen at the top boundary. It stems from the constant ghost cell boundaries. A different boundary type could be chosen to avoid this in future simulations, but as discussed in Section 3.3 there are no good alternatives at the moment. Another solution is simply an extension of the simulation box.

After the 2D simulation, we mapped the exact same model to a 3D grid with 128 grid cells in each spatial dimension. Due to the increased computational cost the model was only followed for 4 days.

Figure 3.8 shows the Mach number at 4 days for both simulations. We see the typical characteristics of 2D and 3D turbulence. In two dimensions turbulence forms large structures. In the 3D simulation, the eddies can decay in an additional dimension and form smaller structures. Also, the turbulent energy cascade is fundamentally different in both cases. This alone is a strong argument that 3D simulations are necessary to get a reliable result for convective mixing.

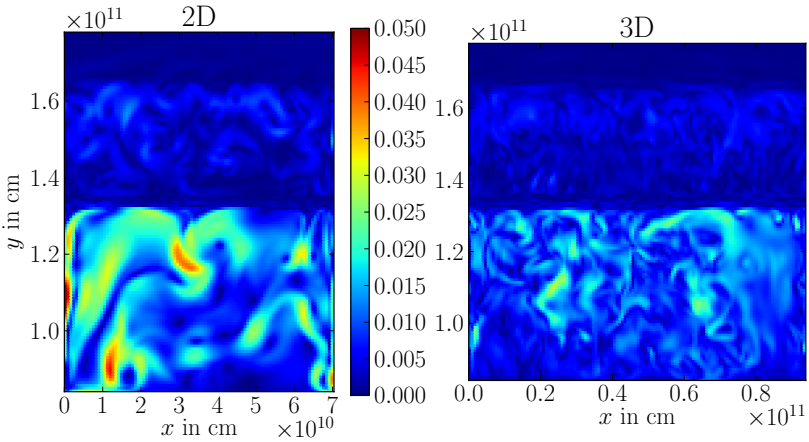


Figure 3.8: Mach number 4 days after the start of the simulation. The left panel shows the 2D simulation, the right panel a cut through the 3D simulation. The colorscale is the same for both panels.

Figure 3.9 compares the evolution of the horizontally averaged Mach number from the 2D and 3D cases. They are qualitatively similar but the 3D case is considerably smoother. This is probably a consequence of the smaller scales of the structures in 3D turbulence and the fact that the averaging is performed over a larger sample in 3D. The temporal evolution over the whole time span of the 2D simulation is shown in Fig. 3.10. For illustration, a 3D volume rendering of the Mach number, the ^{14}N, and the energy release rate is depicted in Fig. 3.11. In the image of the energy release the ingestion of ^{1}H into the core is visible.

Regrettably, the shorter simulated time of the 3D run means it does not progress to a stage where enough ^{12}C has been mixed into the upper layers for a discernable effect. So no statement can be made as to whether hydrogen shell burning will be increased in efficiency compared to the 2D case.

The 3D simulation still has several shortcomings that should be addressed before continuing the simulation for longer times in future work. The size of the simulation box is too small to properly capture the behavior in the convectively stable region at the top. It should be increased for the next simulations. The resolution of 128^3 does not resolve the interface at the center of the box well enough. It remains to be seen if the observed increase of mixing with respect to the 1D stellar evolution model is upheld in a simulation of higher resolution. As a last point, the plane-parallel geometry is clearly not even close to the real geometry of the core. Using a more appropriate geometry and including

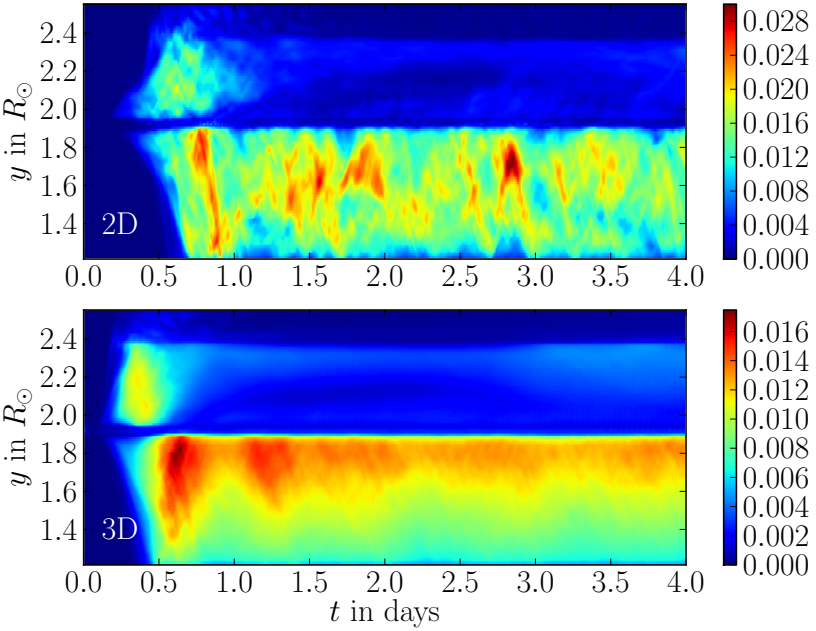


Figure 3.9: Temporal evolution of the horizontally averaged Mach number. The top panel shows the 2D results, the bottom panel the 3D results. The y-axis shows the vertical coordinate in solar radii. The panels do not share the same colorscale.

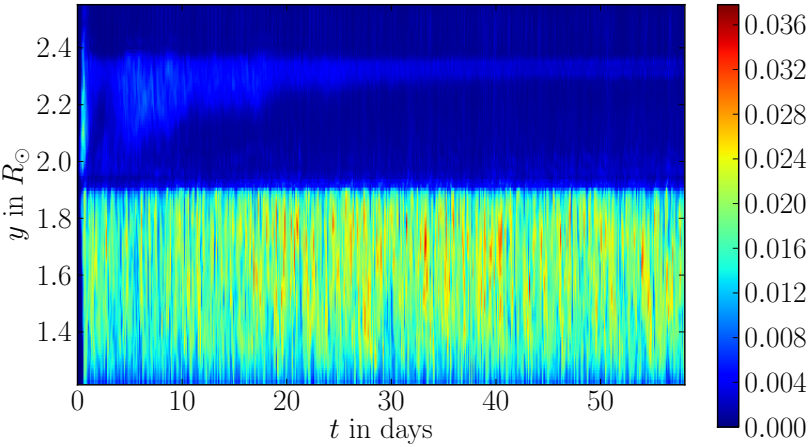


Figure 3.10: Temporal evolution of the horizontally averaged Mach number of the 2D simulation. The y-axis shows the vertical coordinate in solar radii.

the full core on the grid would treat core convection and possible shear flows much more realistically.

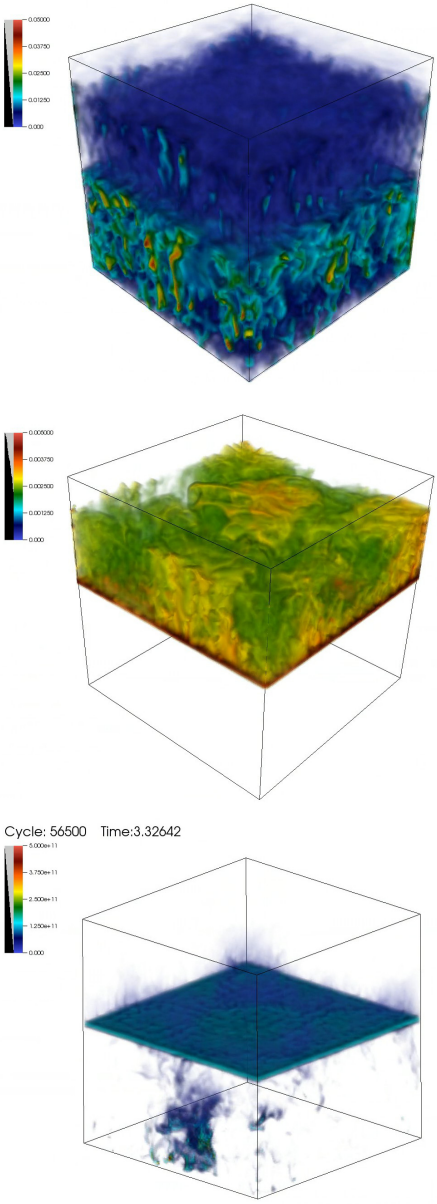


Figure 3.11: Volume rendering of the 3D simulation at 3.3 days. The top panel shows the Mach number, the central panel the ^{14}N mass fraction, and the bottom panel the energy release rate.

4 Conclusions and Outlook

Traditional stellar evolution models are computed in spherical symmetry under many simplifying assumptions. Especially the treatment of convection and the related mixing processes is not adequate. Multidimensional modeling of the full evolution of a star following hydrodynamical processes is firmly out of reach of current computational capabilities due to disparate timescales involved. Yet what can be done using today's resources is the hydrodynamical simulation of limited regions of a star on dynamical timescales. Such calculations can serve as a check of the "recipes" employed in stellar evolution codes. In a next step careful evaluation of the multidimensional results could be used to devise improved prescriptions that would make stellar models more realistic.

In some of these convective mixing processes nuclear reactions are enabled that have an effect on the dynamical timescale. Thus the aim of this thesis was to discuss the coupling of nuclear reaction networks and multidimensional hydrodynamics to be used in the simulation of stellar interiors. The basis of this work is the Seven-League Hydro (SLH) code. This code is particularly well suited for the hydrodynamics of low Mach number flows due to its low-dissipation discretizations and implicit time-stepping. Yet it solves the full Euler equations, which makes it also applicable to high Mach number flows. In the work leading to this thesis the code was extended with a more general equation of state and a flexible nuclear reaction network that can be easily be applied to different nuclear processes occurring in stars.

In Section 2.2.3 the numerical Reynolds number of the code was examined using the Taylor–Green vortex as a test problem. It was found that the numerical Reynolds number of the low-Mach discretization used in SLH is largely independent of the Mach number, whereas the usual discretizations fail to reach the turbulent flow regime. Reynolds numbers of roughly 1000 are reached with resolutions of 240^3. While this is still far from the regime of stellar Reynolds numbers, we see that the energy dissipation rate is converged and the flow is in the turbulent regime. This allows us to apply the code to simulations of stellar convection. In the future, improvements could be made with the inclusion of an explicit sub-grid scale model.

In past work with SLH it was found that convectively stable atmospheres in hydrostatic equilibrium experience a spurious instability that causes flows with Mach numbers of roughly 10^{-2}. This is unacceptable for application since there

is no physical mechanism that should cause this instability. In Section 2.4.2 we give a possible explanation of this phenomenon. There is literature about checkerboard-like modes in pressure that decay more slowly than physically expected. This is a problem of many low Mach number discretizations. In some schemes special terms are introduced to implement pressure–velocity coupling that counteracts this effect. Since these solutions typically come from the engineering community, they are not easily applicable to the stratified atmospheres that are common in stellar astrophysics. In Section 2.4.4 we present a well-balanced method that removes the pressure gradient. As the original method was only developed for the one-dimensional case, we develop a modification that makes it usable in multidimensional hydrodynamics. Even though this extension is not universal, it is applicable for many of the problems we want to address. In Section 2.4.5 a survey of the impact of the well-balanced method on the instability is conducted for several different discretizations. It is found that the AUSM$^+$-Lowmach method shows no sign of the instability provided that the hydrostatic pressure gradient is removed by the well-balanced method.

Section 2.5.3 discusses two methods for coupling the reaction network to hydrodynamics. One of them is Godunov splitting, which is the simplest case of operator splitting. The other is a fully unsplit treatment, in which the combined set of equations is evolved in time simultaneously. In Section 2.5.4 both methods are applied to a simple test problem. For comparison the same problem is computed with a different hydrodynamics code. The result is that the difference between the two coupling methods is minor and Godunov splitting is used in applications due to its significantly lower computational cost.

As an application of the developed methods, we simulate a model of a $250\,M_\odot$ Population III star during core He-burning. The phase we model using multidimensional hydrodynamics is at the time when the convective zone in the core penetrates the H-rich shell above. This mixes CNO-elements into regions that could so far only burn hydrogen via the pp chain, which is much less efficient at the present temperatures. In a long-term 2D simulation we find, in contrast to the predictions of the 1D stellar evolution model, that large amounts of ^{12}C are mixed into the H-burning shell, which is then rapidly consumed from below. Three-dimensional calculations are also presented but these did not yet reach a state where conclusions on the mixing of ^{12}C could be drawn. It has to be noted that these first applications were run with limited resolution and with simplified geometry. Before any definite statements about the behavior of the convective boundary can be made, improved 3D simulations must be performed. Yet the present simulation demonstrates the potential

insight multidimensional models can provide to our understanding of stellar processes.

Before embarking on future astrophysical applications with the SLH code several numerical improvements are advised. First of all, the nature of the checkerboard mode should be studied in detail and a more comprehensive fix than the current method should be implemented. This could be done, for example, by modifying the p-term in the AUSM$^+$-up scheme so that it can cope with stratified atmospheres. A similar term could possibly also be added to the preconditioned Roe scheme to dampen the checkerboard mode. Another interesting possibility is the low Mach Roe scheme suggested by Rieper (2011), which has some intrinsic dampening of the checkerboard mode. It could be implemented in SLH with relative ease. If these two approaches do not yield the desired result, the momentum interpolation method by Li & Gu (2008) provides an alternative, although the prospective violation of zero velocity divergence makes it less favorable.

After these immediate concerns the next important step in development is the implementation of a gravity solver, so that setups including self-gravity can be simulated. A simple monopole approach would be good enough for a number of applications since many models are close to spherical symmetry. For reasons of flexibility it is preferable to have a solution that also works on general curvilinear grids, so that these can be used to simulate the whole core of a star. A suitable method for this might be an FFT gravity solver that works on an oversampled Cartesian grid.

The study of the turbulence properties of SLH should be extended with an analysis of the turbulent energy spectrum. If necessary, the code could be augmented with an explicit sub-grid scale model.

From an astrophysical point of view, an improvement of the Population III star model from this thesis is the next logical step. A more realistic geometry should be chosen in future simulations, either a spherical grid or even mapping the whole core to a curvilinear grid. The size of the computational domain must be increased at the top boundary to be able to investigate how far convection penetrates into the stable layer above. After these modifications, a more highly resolved 3D simulation of the model that is run for a time comparable to the 2D model of this thesis can provide a robust picture of the convective boundary in this star.

Another phase of stellar evolution that is well suited to multidimensional modeling is Si-burning in massive stars. This phase lasts only several days, which means it can probably be simulated in its entirety. This phase is of particular importance since it is the last burning phase before core collapse and thus determines the progenitor model other groups use to simulate core-collapse supernovae. Strong deviations from spherical symmetry are expected

in such models (Arnett & Meakin, 2011). To treat the physics of this problem accurately, to nuclear network of SLH must to be extended with the energy losses via neutrinos.

In the form presented in this thesis, SLH constitutes a versatile tool, suited to the simulation of dynamical phenomena in stellar atmospheres. Its future application will hopefully shed more light on the workings of these intricate objects known as stars.

Bibliography

Alexander, R. 1977, SIAM Journal on Numerical Analysis, 14, pp. 1006

Almgren, A. S., Bell, J. B., & Zingale, M. 2007, Journal of Physics Conference Series, 78, 012085

Arnett, W. D. & Meakin, C. 2011, ApJ, 733, 78, [1101.5646]

Bader, G. & Deuflhard, P. 1983, Numerische Mathematik, 41, 373

Baraffe, I., Heger, A., & Woosley, S. E. 2001, ApJ, 550, 890, [astro-ph/0009410]

Barkat, Z., Rakavy, G., & Sack, N. 1967, Physical Review Letters, 18, 379

Bazán, G., Dearborn, D. S. P., Dossa, D. D., et al. 2003, in Astronomical Society of the Pacific Conference Series, Vol. 293, 3D Stellar Evolution, ed. S. Turcotte, S. C. Keller, & R. M. Cavallo, [astro-ph/0306151]

Böhm-Vitense, E. 1958, ZAp, 46, 108

Bond, J. R., Arnett, W. D., & Carr, B. J. 1984, ApJ, 280, 825

Calhoun, D. A., Helzel, C., & Leveque, R. J. 2008, SIAM Review, 50, 723

Canuto, V. M. & Mazzitelli, I. 1991, ApJ, 370, 295

Cargo, P. & Le Roux, A. 1994, Comptes rendus de l'Académie des sciences. Série 1, Mathématique, 318, 73

Colella, P. & Glaz, H. M. 1985, Journal of Computational Physics, 59, 264

Colella, P. & Woodward, P. R. 1984, Journal of Computational Physics, 54, 174

Courant, R., Friedrichs, K. O., & Lewy, H. 1928, Math. Ann., 100, 32

Dellacherie, S. 2009, in Proceedings of ALGORITMY, Vol. 2009, 71–80

Demmel, J. W., Eisenstat, S. C., Gilbert, J. R., Li, X. S., & Liu, J. W. H. 1999, SIAM J. Matrix Analysis and Applications, 20, 720

Drikakis, D., Fureby, C., Grinstein, F. F., & Youngs, D. 2007, Journal of Turbulence, N20

Edelmann, P. V. F. 2010, Diplomarbeit, Technische Universität München

Edwards, J. R. & Liou, M.-S. 1998, AIAA journal, 36, 1610

Freytag, B., Steffen, M., Ludwig, H.-G., et al. 2012, Journal of Computational Physics, 231, 919, [1110.6844]

Frisch, U. 1995, Turbulence (Cambridge: Cambridge Universtiy Press)

Fryer, C. L., Woosley, S. E., & Heger, A. 2001, ApJ, 550, 372, [astro-ph/0007176]

Fryxell, B. A., Müller, E., & Arnett, W. D. 1989, Hydrodynamics and nuclear burning, MPA Green Report 449, Max-Planck-Institut für Astrophysik,

Garching
Fryxell, B. A., Woodward, P. R., Colella, P., & Winkler, K.-H. 1986, Journal of Computational Physics, 63, 283
Galassi, M., Davies, J., Theiler, J., Gough, B., & Jungman, G. 2009, GNU Scientific Library - Reference Manual, Third Edition, for GSL Version 1.12 (3. ed.) (Network Theory Ltd), 1–573
Gilet, C., Almgren, A. S., Bell, J. B., et al. 2013, ApJ, 773, 137
Glatzmaier, G. A. 1984, Journal of Computational Physics, 55, 461
Grimm-Strele, H., Kupka, F., & Muthsam, H. J. 2013, Computer Physics Communications, [1308.3066]
Guillard, H. & Viozat, C. 1999, Computers & Fluids, 28, 63
Hairer, E., Nørsett, S. P., & Wanner, G. 1993, Solving Ordinary Differential Equations I: Nonstiff Problems, 2nd edn. (Springer-Verlag)
Happenhofer, N., Grimm-Strele, H., Kupka, F., Löw-Baselli, B., & Muthsam, H. 2013, Journal of Computational Physics, 236, 96
Heger, A., Langer, N., & Woosley, S. E. 2000, ApJ, 528, 368, [astro-ph/9904132]
Heger, A. & Woosley, S. E. 2002, ApJ, 567, 532, [astro-ph/0107037]
Henyey, L. G., Forbes, J. E., & Gould, N. L. 1964, ApJ, 139, 306
Herwig, F., Pignatari, M., Woodward, P. R., et al. 2011, ApJ, 727, 89, [1002.2241]
Hillebrandt, W. & Niemeyer, J. C. 2000, ARA&A, 38, 191, [arXiv:astro-ph/0006305]
Hix, W. R. & Meyer, B. S. 2006, Nuclear Physics A, 777, 188, [arXiv:astro-ph/0509698]
Hosea, M. & Shampine, L. 1996, Applied Numerical Mathematics, 20, 21 , method of Lines for Time-Dependent Problems
Iglesias, C. A. & Rogers, F. J. 1996, ApJ, 464, 943
Janka, H.-T. 2012, Annual Review of Nuclear and Particle Science, 62, 407, [1206.2503]
Jeong, J. & Hussain, F. 1995, Journal of Fluid Mechanics, 285, 69
Kennedy, C. A. & Carpenter, M. H. 2001, Additive Runge-Kutta Schemes for Convection-Diffusion-Reaction Equations, Tech. rep., NASA Technical Memorandum
Kifonidis, K. & Müller, E. 2012, A&A, 544, A47
Kifonidis, K., Plewa, T., Janka, H.-T., & Müller, E. 2003, A&A, 408, 621, [astro-ph/0302239]
Kippenhahn, R., Weigert, A., & Hofmeister, E. 1967, Methods in Computational Physics, 7, 129
Kippenhahn, R., Weigert, A., & Weiss, A. 2013, Stellar Structure and Evolution
Kuhlen, M., Woosley, W. E., & Glatzmaier, G. A. 2003, in Astronomical Society of the Pacific Conference Series, Vol. 293, 3D Stellar Evolution, ed.

S. Turcotte, S. C. Keller, & R. M. Cavallo, 147, [astro-ph/0210557]
Kwatra, N., Su, J., Grétarsson, J. T., & Fedkiw, R. 2009, Journal of Computational Physics, 228, 4146
Landau, L. D. & Lifshitz, E. M. 1981, Physical Kinetics: Volume 10 (Course of Theoretical Physics) (Butterworth-Heinemann)
Langanke, K. & Martínez-Pinedo, G. 2001, Atomic Data and Nuclear Data Tables, 79, 1
LeVeque, R. J. 2002, Finite volume methods for hyperbolic problems, Vol. 31 (Cambridge university press)
Li, X.-s. & Gu, C.-w. 2008, Journal of Computational Physics, 227, 5144
Liou, M.-S. 1996, Journal of Computational Physics, 129, 364
Liou, M.-S. 2006, Journal of Computational Physics, 214, 137
Liou, M.-S. & Steffen, C. J. J. 1993, Journal of Computational Physics, 107, 23
Liska, R. & Wendroff, B. 2003, SIAM Journal on Scientific Computing, 25, 995
Liu, X.-D., Osher, S., & Chan, T. 1994, Journal of Computational Physics, 115, 200
Magic, Z., Collet, R., Asplund, M., et al. 2013, A&A, 557, A26, [1302.2621]
Marigo, P., Girardi, L., Chiosi, C., & Wood, P. R. 2001, A&A, 371, 152, [astro-ph/0102253]
Meakin, C. A. & Arnett, D. 2006, ApJ, 637, L53, [astro-ph/0601348]
Meakin, C. A. & Arnett, D. 2007a, ApJ, 665, 690, [astro-ph/0611317]
Meakin, C. A. & Arnett, D. 2007b, ApJ, 667, 448, [arXiv:astro-ph/0611315]
Miczek, F. 2013, PhD thesis, Technische Universität München
Mocák, M., Campbell, S. W., Müller, E., & Kifonidis, K. 2010, A&A, 520, A114, [1003.3646]
Mocák, M., Müller, E., Weiss, A., & Kifonidis, K. 2008, A&A, 490, 265, [0805.1355]
Mocák, M., Müller, E., Weiss, A., & Kifonidis, K. 2009, A&A, 501, 659, [0811.4083]
Mundprecht, E., Muthsam, H. J., & Kupka, F. 2013, MNRAS, 435, 3191, [1209.2952]
Muthsam, H. J., Kupka, F., Löw-Baselli, B., et al. 2010, New Astronomy, 15, 460, [0905.0177]
Nonaka, A., Almgren, A. S., Bell, J. B., et al. 2010, ApJS, 188, 358, [1005.0112]
Nonaka, A., Aspden, A. J., Zingale, M., et al. 2012, ApJ, 745, 73, [1111.3086]
Nordlund, Å. & Galsgaard, K. 1995, Journal of Computational Physics,(in preparation)
Pakmor, R., Edelmann, P., Röpke, F. K., & Hillebrandt, W. 2012, MNRAS, 424, 2222, [1205.5806]
Paxton, B., Bildsten, L., Dotter, A., et al. 2011, ApJS, 192, 3, [1009.1622]

Plewa, T. & Müller, E. 1999, A&A, 342, 179, [astro-ph/9807241]
Press, W., Teukolsky, S., Vetterling, W., & Flannery, B. 1992, Numerical Recipes in C, 2nd edn. (Cambridge, UK: Cambridge University Press)
Rauscher, T. & Thielemann, F.-K. 2000, Atomic Data and Nuclear Data Tables, 75, 1, [arXiv:astro-ph/0004059]
Rieper, F. 2011, Journal of Computational Physics, 230, 5263
Roe, P. 1981, Journal of Computational Physics, 43, 357
Rogers, F. J. & Nayfonov, A. 2002, ApJ, 576, 1064
Schenk, O. & Gärtner, K. 2004, Future Generation Computer Systems, 20, 475
Schwarzschild, M. & Härm, R. 1959, ApJ, 129, 637
Shu, C.-W. & Osher, S. 1988, Journal of Computational Physics, 77, 439
Smagorinsky, J. 1963, Monthly Weather Review, 91, 99
Stancliffe, R. J., Dearborn, D. S. P., Lattanzio, J. C., Heap, S. A., & Campbell, S. W. 2011, ApJ, 742, 121, [1109.1289]
Taylor, G. I. & Green, A. E. 1937, Royal Society of London Proceedings Series A, 158, 499
Timmes, F. X. 1999, ApJS, 124, 241
Timmes, F. X. & Arnett, D. 1999, ApJS, 125, 277
Timmes, F. X. & Swesty, F. D. 2000, ApJS, 126, 501
Toro, E. F. 2009, Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction (Springer)
Turkel, E. 1999, Annual Review of Fluid Mechanics, 31, 385
Viallet, M., Baraffe, I., & Walder, R. 2011, A&A, 531, A86, [1103.1524]
Viallet, M., Baraffe, I., & Walder, R. 2013a, A&A, 555, A81, [1305.6581]
Viallet, M., Meakin, C., Arnett, D., & Mocák, M. 2013b, ApJ, 769, 1, [1212.6365]
Wada, Y. & Liou, M.-S. 1994, A flux splitting scheme with high-resolution and robustness for discontinuities, Tech. rep.
Wallace, R. K., Woosley, S. E., & Weaver, T. A. 1982, ApJ, 258, 696
Wallace, R. K., Woosley, S. E., & Weaver, T. A. 1983, ApJ, 264, 746
Weaver, T. A., Zimmerman, G. B., & Woosley, S. E. 1978, ApJ, 225, 1021
Weiss, J. M. & Smith, W. A. 1995, AIAA Journal, 33, 2050
Woodward, P., Herwig, F., Porter, D., et al. 2008, in American Institute of Physics Conference Series, Vol. 990, First Stars III, ed. B. W. O'Shea & A. Heger, 300–308, [0711.2091]
Woodward, P. R., Herwig, F., & Lin, P.-H. 2013, ArXiv e-prints, [1307.3821]

Acknowledgments

I want to thank my advisor Friedrich Röpke, who asked me almost 5 years ago if I was interested in writing a diploma thesis on astrophysics and has since then always given me exciting new aspects to work on and constructive advice when I needed it. Yet when I found something interesting, I was free to follow up on it.

The whole Type Ia supernova group, which is now partially scattered across the world, even to remote places like Lower Franconia, deserves my gratitude for making me feel at home at MPA since the first day. The same goes for the astrophysics and mathematics group in Würzburg.

The SESTAS group deserves credit for providing me with a new interesting problem from stellar astrophysics to think about on a weekly basis.

Special thanks goes to my colleagues from office 009, including all alumni, and all the other students from MPA for all those (scientific) discussions and welcome distractions, including the many questions about pdsoft.

This thesis would not have been possible without the diligent work of Fabian Miczek, who wrote most of the code formerly known as LHC.

I thank my parents for their continuous support during the last 29 years in achieving everything I wanted to.

I will always be grateful to my girlfriend Christina, who was always there for me and helped me keep calm even in stressful situations.