Chapter 6

Riemann solvers

The numerical hydrodynamics algorithms we have devised in Chapter 5 were based on the idea of operator splitting between the advection and pressure force terms. The advection was done, for all conserved quantities, using the gas velocity, while the pressure force and work terms were treated as source terms. From Chapter 2 we know, however, that the characteristics of the Euler equations are not necessarily all equal to the gas velocity. We have seen that there exist an eigenvector which indeed has the gas velocity as eigenvectors, $\lambda_0 = u$, but there are two eigenvectors which have eigenvalues $\lambda_\pm = u \pm C_s$ which belong to the forward and backward sound propagation. Mathematically speaking one should do the advection in these three eigenvectors, using their eigenvalues as advection velocity. The methods in Chapter 5 do not do this. By extracting the pressure terms out of the advection part and adding them as a source term, the advection part has been reduced essentially to *Burger's equation*, and the propagation of sound waves is entirely driven by the addition of the source terms. Such methods therefore do not formally propagate the sound waves using advection, even though mathematically they should be. All the effort we have done in Chapters 3 and 4 to create the best advection schemes possible will therefore have no effect on the propagation of sound waves. Once could say that for two out of three characteristics our ingeneous advection scheme is useless.

Riemann solvers on the other hand keep the pressure terms within the to-be-advected system. There is no pressure source term in these equations. The mathematical character of the equations remains intact. Such solvers therefore propagate all the characteristics on equal footing. We shall see that Riemann solvers are based on the concept of the *Riemann problem*, so we will first dig into this concept. We will then cover the purest version of a Riemann solver: the Godunov solver, but we will then quickly turn our attention to linearized Riemann solvers, which are simpler to program and are conceptually more closely linked to the concept of characteristic transport. Perhaps the most powerful linear Riemann solver is the *Roe solver* which has the particular advantage that it recognizes shock waves and transports all characteristics nicely.

As we shall see, Riemann solvers tend to have advantages, but also some disadvantages. One can therefore not say that they are always the method of choice. However, for problems involving shock waves, contact discontinuities and other high-resolution flow features, Riemann solvers remain unparallelled in keeping these flow features sharp. For that reason they are becoming ever more popular.

Many of the things covered in this chapter were inspired by the book of Randall LeVeque, "Finite Volume Methods for Hyperbolic Problems".

6.1 Simple waves, integral curves and Riemann invariants

Before we can delve into the concepts of Riemann problems and, lateron, Riemann solvers, we must first solidify our understanding of characteristic families, and the associated concepts of simple waves, integral curves and Riemann invariants. Since these concepts are important conceptually, but not of too great importance quantitatively, we shall remain brief here. Let us recall the following form of the Euler equations (cf. Eq. 6.1):

$$\partial_t \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} + \begin{pmatrix} 0 & 1 & 0 \\ \frac{\gamma-3}{2}\rho u^2 & (3-\gamma)u & (\gamma-1) \\ -\left\{\gamma e_{\rm tot} u + (\gamma-1)u^3\right\} & \left\{\gamma e_{\rm tot} + \frac{3}{2}(1-\gamma)u^2\right\} & \gamma u \end{pmatrix} \partial_x \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} = 0 \quad (6.1)$$

where $q_1 = \rho$, $q_2 = \rho u$ and $q_3 = \rho e_{\rm tot}$ The eigenvalues are

$$\lambda_1 = u - C_s \quad (6.2)$$
$$\lambda_2 = u \quad (6.3)$$
$$\lambda_3 = u + C_s \quad (6.4)$$

with eigenvectors:

$$e_1 = \begin{pmatrix} 1 \\ u - C_s \\ h_{\rm tot} - C_s u \end{pmatrix} \qquad e_2 = \begin{pmatrix} 1 \\ u \\ \frac{1}{2}u^2 \end{pmatrix} \qquad e_3 = \begin{pmatrix} 1 \\ u + C_s \\ h_{\rm tot} + C_s u \end{pmatrix} \quad (6.5)$$

where $h_{\rm tot} = e_{\rm tot} + P/\rho$ is the total specific enthalpy and $C_s = \sqrt{\gamma P/\rho}$ is the adiabatic sound speed.

The definition of the eigenvectors depend entirely and only on the state $q = (q_1, q_2, q_3)$, so in the 3-D state-space these eigenvectors set up three vector fields. We can now look for set of states $q(\xi) = (q_1(\xi), q_2(\xi), q_3(\xi))$ that connect to some starting state $q_s = (q_{s,1}, q_{s,2}, q_{s,3})$ through integration along one of these vector fields. These constitute *inegral curves of the characteristic family*. Two states q_a and q_b belong to the same 1-characteristic integral curve, if they are connected via the integral:

$$q_b = q_a + \int_a^b de_1 \quad (6.6)$$

The concept of integral curves can be understood the easiest if we return to linear hyperbolic equations with a constant advection matrix: in that case we could decompose q entirely in eigencomponents. A 1-characteristic integral curve in state-space is a set of states for which only the component along the e_1 eigenvector varies, while the components along the other eigenvectors may be non-zero but should be non-varying. For non-linear equations the decomposition of the full state vector is no longer a useful concept, but the integral curves are the non-linear equivalent of this idea.

Typically one can express integral curves not only as integrals along the eigenvectors of the Jacobian, but also curves for which some special scalars are constant. In the 3-D parameter space of our $q = (q_1, q_2, q_3)$ state vector each curve is defined by two of such scalars. Such scalar fields are called *Riemann invariants* of the characteristic family. One can regard these integral curves now as the crossling lines between the two contour curves of the two Riemann invariants. The value of each of the two Riemann invariants now identifies each of the characteristic integral curves.

For the eigenvectors of the Euler equations above the Riemann invariants are:

$$\begin{array}{lll} \text{1-Riemann invariants:} & s, & u + \frac{2C_s}{\gamma-1} \\ \text{2-Riemann invariants:} & u, & P \\ \text{3-Riemann invariants:} & s, & u - \frac{2C_s}{\gamma-1} \end{array} \tag{6.7}$$

The 1- and 3- characteristics represent sound waves. Indeed, sound waves (if they do not topple over to become shocks) preserve entropy, and hence the entropy s is a Riemann invariant of these two families. The 2- characteristic represents an *entropy wave* which means that the entropy can vary along this wave. This is actually not a wave in the way we know it. It is simply the comoving fluid, and adjacent fluid packages may have different entropy. The fact that u and P are Riemann invariants of this wave can be seen by integrating the vector $(1, u, u^2/2)$ in parameter space. One sees that ρ varies, but u does not. Also one sees that $q_3 = \rho(e_{\rm th} + u^2/2)$ varies only in the kinetic energy component. The value $\rho e_{\rm th}$ stays constant, meaning that the pressure $P = (\gamma - 1)\rho e_{\rm th}$ remains constant. So while the density may increase along this integral curve, the $e_{\rm th}$ will then decrease enough to keep the pressure constant. This means that the entropy goes down, hence the term "entropy wave".

In time-dependent fluid motion a wave is called a *simple wave* if the states along the wave all lie on the same integral curve of one of the characteristic families. One can say that this is then a pure wave in only one of the eigenvectors. A simple wave in the 2-characteristic family is a wave in which u =const and P =const, but in which the entropy may vary. A simple wave in the 3-characteristic family is for instance an infinitesimally weak sound wave in one direction. In Section 6.3 we shall encounter also another simple wave of the 1- or 3- characteristic family: a *rarefaction wave*.

As we shall see in the section on Riemann problems below, there can exist situations in which two fluids of different entropy lie directly next to each other, causing an entropy jump, but zero pressure or velocity jump. This is also a simple wave in the 2-family, but a special one: a discrete jump wave. This kind of wave is called a *contact discontinuity*.

Another jump-like wave is a *shock wave* which can be either from the 1-characteristic family or from the 3-characteristic family. However, shock waves are waves for which the Riemann invariants are no longer perfectly invariant. In particular the entropy will no longer be constant over a shock front. Nevertheless, shock fronts can still be associated to either the 1-characteristic or 3-characteristic family. The states on both sides of the shock front however, do not lie on the same integral curve. They lie instead on a *Hugoniot locus*.

6.2 Riemann problems

A Riemann problem in the theory of hyperbolic equations is a problem in which the initial state of the system is defined as:

$$q(x, t = 0) = \begin{cases} q_L & \text{for} \quad x \leq 0 \\ q_R & \text{for} \quad x > 0 \end{cases} \tag{6.8}$$

In other words: the initial state is constant for all negative x, and constant for all positive x, but differs between left and right. For hydrodynamic problems one can consider this to be a 1-D hydrodynamics problem in which gas with one temperature and density is located to the left of a removable wall and gas with another temperature and density to the right of that wall. At time $t = 0$ the wall is instantly removed, and it is watched what happens.

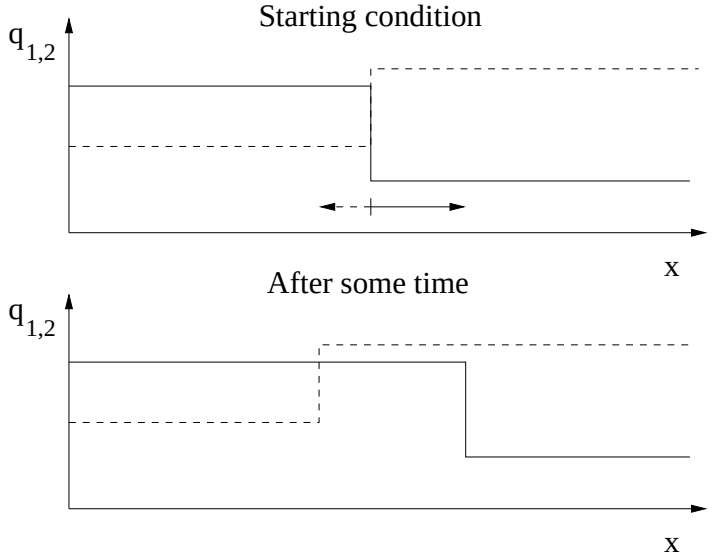

Figure 6.1. Example of the solution of a linear Riemann problem with constant and diagonal advection matrix. Top: initial condition (solid line is q_1, dashed line is q_2). Bottom: after some time, the q_1 component has moved to the right ($\lambda_1 > 0$) while the q_2 component has moved to the left ($\lambda_2 < 0$).

For hydrodynamic problems such *shock tube tests* are used to test the performance of numerical hydrodynamics algorithms. This was first done by (Sod 1978, J. Comp. Phys 27, 1), hence the name *Sod shock tube tests*. But such tests were also carried out in the laboratory (see e.g. the book by Liepmann & Roshko).

6.2.1 Riemann problems for linear advection problems

The simplest Riemann problems are those of linear advection problems with constant advection velocity, or constant Jacobian matrix. Consider the following equation:

$$\partial_t \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} + \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \partial_x \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = 0 \tag{6.9}$$

Consider the following Riemann problem for this set of equations:

$$q_1(x,0) = \begin{cases} q_{1,l} & \text{for } x < 0 \\ q_{1,r} & \text{for } x > 0 \end{cases} \tag{6.10}$$

$$q_2(x,0) = \begin{cases} q_{2,l} & \text{for } x < 0 \\ q_{2,r} & \text{for } x > 0 \end{cases} \tag{6.11}$$

Clearly the solution is:

$$q_1(x,t) = q_1(x - \lambda_1 t, 0) \tag{6.12}$$

$$q_2(x,t) = q_2(x - \lambda_2 t, 0) \tag{6.13}$$

which is simply that the function $q_1(x)$ is shifted with velocity λ_1 and the function $q_2(x)$ is shifted with velocity λ_2. An example is shown in Fig. 6.1 A very similar solution is found if the matrix is not diagonal, but has real eigenvalues: we then simply decompose (q_1, q_2) into eigenvectors, obtaining $(\tilde{q}_1, \tilde{q}_2)$, shift $\tilde{q}_1$ and $\tilde{q}_2$ according to their own advection velocity (eigenvalue of the matrix), and then reconstruct the q_1 and q_2 from $\tilde{q}_1$ and $\tilde{q}_2$.

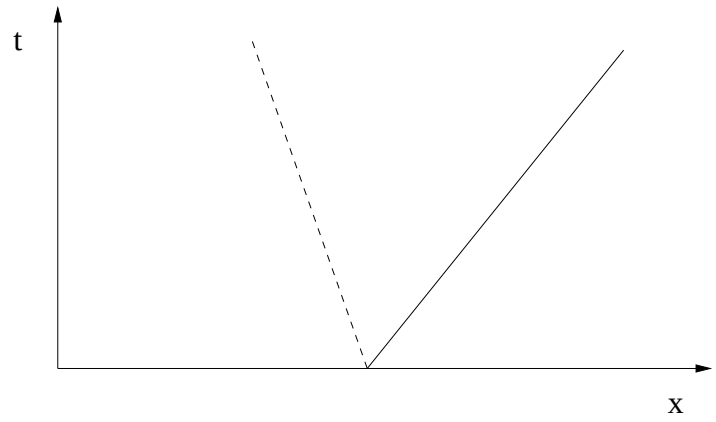

Figure 6.2. The characteristics of the problem solved in Fig. 6.1.

→ **Exercise:** Solve in this way the general Riemann problem for the equation

$$\partial_t \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \partial_x \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = 0 \tag{6.14}$$

These examples are for hyperbolic equations with two characteristics, but this procedure can be done for any number of characteristics.

Note that if we look at this problem in an (x, t) diagram, then we see two waves propagating, one moving with velocity λ_1 and one with velocity λ_2. We also see that the solution is self-similar:

$$q_{1,2}(x, t_b) = q_{1,2}(xt_a/t_b, t_a) \tag{6.15}$$

6.3 Riemann problems for the equations of hydrodynamics

Riemann problems for the Euler equations are much more complex than those for the simple linear hyperbolic equations shown above. This is because of the strong non-linearity of the equations. A Riemann problem for the equations of hydrodynamics is defined as:

$$\rho, u, P = \begin{cases} \rho_l, u_l, P_l & \text{for } x < 0 \\ \rho_r, u_r, P_r & \text{for } x > 0 \end{cases} \tag{6.16}$$

The general solution is quite complex and even the qualitative shape of the solution depends strongly on the Riemann problem at hand. In this section we will discuss two special cases.

6.3.1 Special case: The converging flow test

The simplest Riemann problem for the hydrodynamic equation is that in which $P_l = P_r$, $\rho_l = \rho_r$ and $u_l = -u_r$ with $u_l > 0$. This is a symmetric case in which the gas on both sides of the dividing line are moving toward each other: a converging flow. From intuition and/or from numerical experience it can be said that the resulting solution is a compressed region that is expanding in the form of two shock waves moving away from each other. Without a-priori proof (we shall check a-posteriori) let us assume that the compressed region in between the two shock waves has a constant density and pressure, and by symmetry has a zero velocity. We also assume that the converging gas that has not yet gone through the shock front is undisturbed.

The problem we now have to solve is to find the shock velocity v_s (which is the same but opposite in each direction) and the density and pressure in the compressed region: ρ_c, P_c. For a

given v_s the Mach number $\mathcal{M}$ of the shock is:

$$\mathcal{M} = \frac{u_l + v_s}{C_{s,l}} = (u_l + v_s)\sqrt{\frac{P_l}{\gamma\rho_l}} \tag{6.17}$$

(we take by definition $v_s > 0$). We now need the Rankine-Hugoniot adiabat in the form of Eq. (1.97),

$$\frac{\rho_1}{\rho_c} = \frac{(\gamma-1)\mathcal{M}^2+2}{(\gamma+1)\mathcal{M}^2} = \frac{\gamma-1}{\gamma+1} + \frac{2}{(\gamma+1)\mathcal{M}^2} \tag{6.18}$$

as well as the condition for mass conservation

$$\rho_l(u_l + v_s) = \rho_c v_s \tag{6.19}$$

By writing $v_s = v_s + u_l - u_l = (\mathcal{M} - u_l/C_{s,l})C_{s,l}$ in the latter equation we can eliminate ρ_l/ρ_c in both equations to obtain

$$\frac{\gamma-1}{\gamma+1}\mathcal{M}^2 + \frac{2}{\gamma+1} = \mathcal{M}^2 - \frac{u_l}{C_{s,l}}\mathcal{M} \tag{6.20}$$

which reduces to

$$\mathcal{M}^2 - \frac{\gamma+1}{2}\frac{u_l}{C_{s,l}}\mathcal{M} - 1 = 0 \tag{6.21}$$

The solution is:

$$\mathcal{M} = \frac{1}{2}\left\{\left(\frac{\gamma+1}{2}\right)\frac{u_l}{C_{s,l}} \pm \sqrt{\frac{(\gamma+1)^2}{4}\frac{u_l^2}{C_{s,l}^2} + 4}\right\} \tag{6.22}$$

For our purpose we need to choose the positive root. One sees that there is always a solution, and that one can find two limits:

- Limit 1, $u_l \ll C_{s,l}$: The solution is $\mathcal{M} = 1$. This means that in this limit the shock wave reduces to a sound wave.
- Limit 2, $u_l \gg C_{s,l}$: The solution is $\mathcal{M} = (u_l/C_{s,l})(\gamma+1)/2$. This is the strong shock limit: the compression reaches its maximum of $\rho_c \to \rho_l(\gamma+1)/(\gamma-1)$. The strong shock limit is the limit in which the pre-shock thermal energy is so small compared to the post-shock value that it can be regarded as being zero.

6.3.2 Special case: Sod's shock tube tests

A special case of a Riemann problem in Eulerian hydrodynamics is when the initial state has zero velocity on both sides of the dividing point, but the pressure has a jump. We shall discuss these solutions here following the book by Bodenheimer et al. (2007). The complete solution of the Sod shock tube test is rather difficult to derive, so here we shall derive only the most obvious relations, and write down other relations without derivation.

The most important first step is to find out what the qualitative form of the solution is. Here we rely on experience: If one solves such problems using numerical hydrodynamics or laboratory experiments one finds that the self-similar solution that follows from such a problem typically has 5 regions which we shall call region 1,2,3,4,5 as depicted in Fig. 6.3. Region 1 and 5 have states which correspond to the left and the right initial state respectively. Regions 3 and 4 have steady states (independent of x within the region) and region 2 has an x-dependent state. This

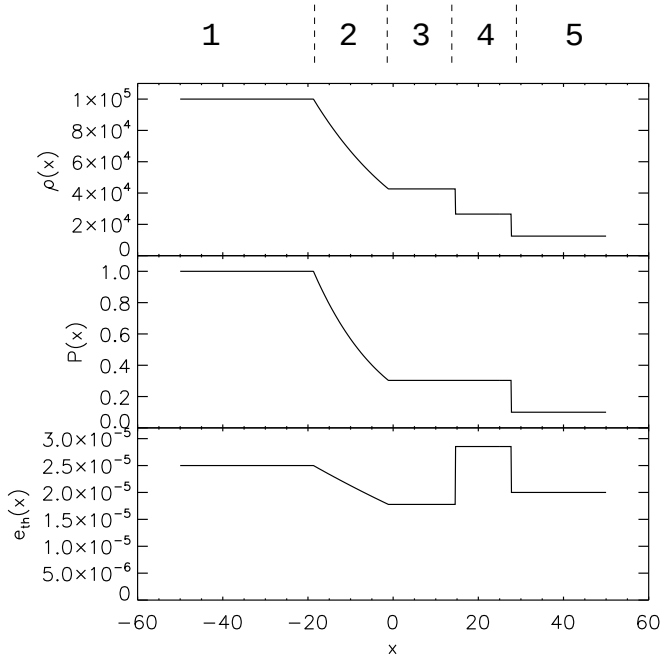

Figure 6.3. The solution to the shock tube problem of Sod for $\gamma = 7/5$, $\rho_l = 10^5$, $P_l = 1$, $\rho_r = 1.25 \times 10^4$ and $P_r = 0.1$, shown at time $t = 5000$. The regions 1 to 5, as mentioned in the text, are annotated at the top.

region 2 represents an *expansion wave*, also called *rarefaction wave*. This is a simple wave of the left-going characteristic family (the 1-characteristic family in the terminology of Section 6.1). It is the only non-constant region in the solution. The dividing line between region 3 and 4 is a *contact discontinuity*, i.e. a line separating two fluids of different entropy but the same pressure and the same velocity. This is a "wave" of the middle characteristic family (the 2-characteristic family in the terminology of Section 6.1). Therefore $u_3 = u_4$ and $P_3 = P_4$. The propagation speed of the contact discontinuity is therefore also $u_c = u_4$ and the location of this discontinuity at some time t is $x_{\text{contact}} = u_c t$. Regions 4 and 5 are separated by a forward moving shock wave. This is a jump in the forward moving characteristic family (the 3-characteristic family in the terminology of Section 6.1). Since $u_5 = 0$ one can invoke mass conservation to write the shock propagation speed u_s in terms of the velocity u_4 and the densities in both regions:

$$u_s = u_4 \frac{\rho_4}{\rho_4 - \rho_5} \tag{6.23}$$

The location of the shock wave at time t is therefore $x_{\text{shock}} = u_s t$. According to the Rankine-Hugoniot conditions derived in Section 1.9 we can also relate the density ratio and the pressure ratio over the shock:

$$\frac{\rho_4}{\rho_5} = \frac{P_4 + m^2 P_5}{P_5 + m^2 P_4} \tag{6.24}$$

where $m^2 = (\gamma - 1)/(\gamma + 1)$. From these relations we can derive the velocity in region 4, because we know that $u_5 = 0$. We obtain

$$u_4 = (P_4 - P_5)\sqrt{\frac{1 - m^2}{\rho_5 (P_4 + m^2 P_5)}} \tag{6.25}$$

This is about as far as we get on the shock front. Let us now focus on the expansion wave (region 2). The leftmost onset of the expansion wave propagates to the left with the local sound speed. So we have, at some time t, this point located at $x_{\rm wave} = -C_{s,1}t$, where $C_{s,1} = \sqrt{\gamma P_1/\rho_1}$ is the sound speed in region 1. Without further derivation (see Hawley et al. 1984) we write that the gas velocity in region 2 can be expressed as

$$u_2 = \sqrt{\frac{(1-m^4)P_1^{1/\gamma}}{m^4\rho_1}}\left(P_1^{\frac{\gamma-1}{2\gamma}} - P_2^{\frac{\gamma-1}{2\gamma}}\right) \tag{6.26}$$

To find the dividing line between regions 2 and 3 we now solve the equation $u_2 = u_4$, i.e. Eq.(6.26) $-$ Eq.(6.25)$= 0$, for the only remaining unknown P_3. We do this using a numerical root-finding method, for instance the `zbrent` method of the book *Numerical Recipes* by Price et al.. This will yield us a numerical value for $P_3 = P_4$. Then Eq.(6.25) directly leads to $u_c = u_3 = u_4$. The Hugoniot adiabat of the shock (Eq. 6.24) now gives us ρ_4. Now with Eq. (6.23) we can compute the shock velocity u_s, and thereby the location of the shock front $x_{\rm shock} = u_s\, t$. The density in region 3 can be found by realizing that none of the gas to the left of the contact discontinuity has ever gone through a shock front. It must therefore still have the same entropy as the gas in region 1. Using the law for polytropic gases $P = K\rho^\gamma$ we can say that K is the same everywhere left of the contact discontinuity (i.e. in regions 1,2 and 3). Therefore we can write that $\rho_3 = \rho_1(P_3/P_1)^{1/\gamma}$. At this point we know the density, the pressure and the gas velocity in regions 1,3,4,5. We can therefore easily calculate any of the other quantities in these regions, such as the sound speed C_s or the internal thermal energy $e_{\rm th}$. The remaining unknown region is region 2, and we also do not yet know the location of the separation between regions 2 and 3. Without derivation (see Hawley et al. 1984) we state that in region 2:

$$u(x,t) = (1-m^2)\left(\frac{x}{t} + C_{s,1}\right) \tag{6.27}$$

which indeed has the property that $u = 0$ at $x = -C_{s,1}t$. We can find the location of the separation between regions 2 and 3 by numerically solving $u(x,t) = u_3$ for x. The expression for the sound speed $C_s(x,t)$ in region 2 is now derived by noting that region 2 is a classical *expansion fan*, in which the left-moving characteristic λ_- must, by nature of self-similar solutions, have the form $\lambda_- \equiv u(x,t) - C_s(x,t) = x/t$. This yields for region 2:

$$C_s^2(x,t) \equiv \gamma\frac{P(x,t)}{\rho(x,t)} = \left(u(x,t) - \frac{x}{t}\right)^2 \tag{6.28}$$

Also here we know that $P(x,t) = K\rho(x,t)^\gamma$ with the same K as in region 1. Therefore we obtain:

$$\rho(x,t) = \left[\frac{\rho_1^\gamma}{\gamma P_1}\left(u(x,t) - \frac{x}{t}\right)^2\right]^{1/(\gamma-1)} \tag{6.29}$$

from which $P(x,t)$ can be directly derived using $P(x,t) = K\rho(x,t)^\gamma$, and the sound speed and internal thermal energy follow then immediately. We now have the total solution complete, and for a particular example this solution is shown in Fig. 6.3. Later in this chapter we shall use these solutions as simple test cases to verify the accuracy and performance of hydrodynamic algorithms.

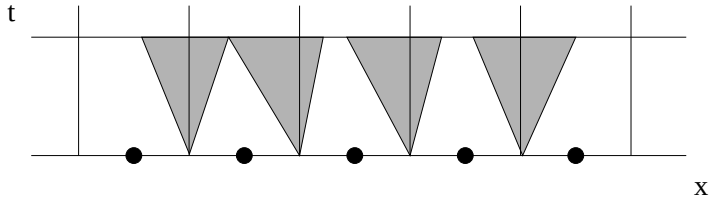

Figure 6.4. Godunov's method: solving a self-similar Riemann problem at each interface (grey), and making sure that the time step is small enough that they do not overlap. The two leftmost self-similar Riemann solutions just manage to touch by the end of the time step, which means that the time step can not be made larger before they will interfere.

6.4 Godunov's method

We can now apply what we learned about the solution of Riemann problems to devise a new numerical method for numerical hydrodynamics. Consider our numerical solution at some time t_n to be given by q_i^n. These are values of q given at the cell centers located at $x = x_i$. We define cell interfaces $x_{i+1/2}$ in the usual way (see Chapter 4) to be located in between the cell centers x_i and x_{i+1}. As our subgrid model we assume that at the start of the time step the state within each cell is strictly constant (piecewise constant method, see Chapter 4). At each interface the state variables now describe a jump. If we zoom in to the region around this interface we see that this is precisely the definition of a Riemann problem, but this time locally within the two adjacent cells. We can now calculate what the self-similar solution of the Riemann problem at each cell interface $i + 1/2$ would be. This is a subgrid analytic evolution of the hydrodynamic system *within each pair of cells*. This self-similar solution is calculated at each interface, so in order to preserve the self-similar character of these solutions we must prevent the solutions from two adjacent interfaces to overlap. This is depicted in Fig. 6.4. The time step is therefore restricted to

$$\Delta t \leq \min(\Delta t_i) \tag{6.30}$$

where

$$\Delta t_i = \frac{x_{i+1/2} - x_{i-1/2}}{\max(\lambda_{i-1/2,k+}) - \min(\lambda_{i+1/2,k-})} \tag{6.31}$$

where $\lambda_{i-1/2,k+}$ denotes the maximum positive eigenvalue at interface $i - 1/2$, and will be 0 in case no positive eigenvalues exist at that interface. Likewise $\lambda_{i+1/2,k-}$ denotes the smallest (i.e. most negative) negative eigenvalue at interface $i + 1/2$, or 0 if no negative eigenvalues exist.

How to proceed from here, i.e. how to create a numerical algorithm from this concept, can be seen in two different way, which we will highlight in the two next subsections.

6.4.1 One way to look at Godunov's method

At the end of the time step each cell i consists of three regions: a left region which is affected by the Riemann solution at interface $i - 1/2$, a middle region which is not yet affected, and a right region which is affected by the Riemann solution at interface $i + 1/2$. Since we know the (semi-)analytic solutions of the Riemann problems and we of course know the unaffected state in the middle region, we can (semi-)analytically average all state variables over the cell. This averaging then results in the cell-center value of q_i^{n+1}. This averaging procedure is very similar to what was done in the donor-cell algorithm, but this time the state in the cell at the end of the time step is far more complex than in the simple donor-cell algorithm. Because of this complexity we shall not work this out in this chapter.

6.4.2 Another way to look at Godunov's method

Another way to look at Godunov's method is by looking at the flux at the interface. We know that the Riemann solutions around the cell interfaces are self-similar in the dimensionless space variable $\xi = (x - x_{i+1/2})/(t - t_n)$. This means that the state at the interface in this solution is constant in time (at least between $t = t_n$ and $t = t_{n+1}$). This then implies that the flux $f_{i+1/2}$ is also constant in this time interval. We can therefore then write:

$$q_i^{n+1} = q_i^n - \Delta t \frac{f_{i+1/2}^n - f_{i-1/2}^n}{x_{i+1/2} - x_{i-1/2}} \tag{6.32}$$

where $f_{i+1/2}^n$ and $f_{i-1/2}^n$ are the fluxes calculated from the Riemann problems at the cell interfaces. Note that this is true as much for linear sets of hyperbolic equations as well as for non-linear ones. The complexity still remains in determining the state in the Riemann problem at the cell interfaces, but that is already much less difficult than determining the entire Riemann solution and averaging over it. Nevertheless for hydrodynamics the method remains complex and we will therefore not go into the Godunov method for these equations.

6.5 Godunov for linear hyperbolic problems: a characteristic solver

6.5.1 Example for two coupled equations

Instead of demonstrating how a Godunov solver works for the full non-linear set of equations of hydrodynamics, we show here how it works for linear hyperbolic sets of equations. The nice thing is that in this case the Riemann problem at each cell interface can be solved analytically. Moreover, we will then naturally be led to a new concept: that of a *characteristic solver*. For linear problems Riemann solvers and characteristic solvers are identical. Later, when dealing with the full set of non-linear hydrodynamics equations, we shall be using both concepts.

Let us consider the following equation:

$$\partial_t \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} + \begin{pmatrix} a & b \\ c & d \end{pmatrix} \partial_x \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = 0 \tag{6.33}$$

where the matrix is diagonizable and has two real eigenvalues. We wish to solve this numerically. Since the advection matrix, in this example, is constant, we were able to bring it out of the ∂_x operator without flux conservation violation. The way we proceed is first to define the state vector on the left- and right- side of the interface $i + 1/2$:

$$q_{i+1/2,L} \equiv q_i \tag{6.34}$$

$$q_{i+1/2,R} \equiv q_{i+1} \tag{6.35}$$

Now define the eigenvalues and eigenvectors of the problem:

$$\lambda_{(\pm)} = \frac{1}{2}\left\{(a+d) \pm \sqrt{(a-d)^2 + 4bc}\right\} \tag{6.36}$$

and

$$e_{(\pm)} = \begin{pmatrix} \lambda_{(\pm)} + 2d \\ 2c \end{pmatrix} \tag{6.37}$$

Note that we use indices $(-)$ and $(+)$ because in this special case the eigenvalues are clearly identifiable with left- and right-moving characteristics unless the problem is "supersonic" in that

both eigenvalues are negative or both are positive. In general we would simply use λ_1, λ_2 etc. This is just a notation issue. Now, any state

$$q = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} \tag{6.38}$$

can be decomposed into these eigenvectors:

$$\tilde{q}_{(-)} = \frac{1}{\lambda_{(+)} - \lambda_{(-)}} \left\{ \frac{\lambda_{(+)} + 2d}{2c} q_2 - q_1 \right\} \tag{6.39}$$

$$\tilde{q}_{(+)} = \frac{1}{\lambda_{(-)} - \lambda_{(+)}} \left\{ \frac{\lambda_{(-)} + 2d}{2c} q_2 - q_1 \right\} \tag{6.40}$$

So we can define the decomposed state on each side of the interface:

$$\tilde{q}_{(-),i+1/2,L} = \tilde{q}_{(-),i} \tag{6.41}$$

$$\tilde{q}_{(+),i+1/2,L} = \tilde{q}_{(+),i} \tag{6.42}$$

$$\tilde{q}_{(-),i+1/2,R} = \tilde{q}_{(-),i+1} \tag{6.43}$$

$$\tilde{q}_{(+),i+1/2,R} = \tilde{q}_{(+),i+1} \tag{6.44}$$

Now we can construct the flux at the interface. Suppose that $\lambda_1 > 0$, then clearly the flux for $\tilde{q}_{(+),i+1/2}$ is determined solely by $\tilde{q}_{(+),i+1/2,L}$ and not by $\tilde{q}_{(+),i+1/2,R}$ (the upwind principle):

$$\tilde{f}_{(-),i+1/2} = \begin{cases} \lambda_{(-)} \tilde{q}_{(-),i+1/2,L} & \text{for} \quad \lambda_{(-)} > 0 \\ \lambda_{(-)} \tilde{q}_{(-),i+1/2,R} & \text{for} \quad \lambda_{(-)} < 0 \end{cases} \tag{6.45}$$

and similar for $\tilde{f}_{(+),i+1/2}$. The total flux for q is then:

$$f_{i+1/2} = \tilde{f}_{(-),i+1/2} e_{(-)} + \tilde{f}_{(+),i+1/2} e_{(+)} \tag{6.46}$$

We see that Godunov's method for linear advection equations is nothing else than the donor-cell algorithm applied to each characteristic separately.

We can generalize this method to non-constant advection matrix. Consider the following problem:

$$\partial_t \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} + \partial_x \left[\begin{pmatrix} a(x) & b(x) \\ c(x) & d(x) \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} \right] = 0 \tag{6.47}$$

The procedure is now the same, except that we must do the eigenvector decomposition with the *local* matrix at the interface $i + 1/2$. Both the eigenvectors and the eigenvalues are now local to the interface, and so will the decomposition be. In this case $\tilde{q}_{(-),i+1/2,L} \neq \tilde{q}_{(-),i-1/2,R}$, while in the case of constant matrix we had $\tilde{q}_{(-),i+1/2,L} = \tilde{q}_{(-),i-1/2,R}$. For the rest we construct the fluxes in the same way as above for the constant matrix.

We see that in the simple case of linear advection problems, the Godunov method (based on the Riemann problem) is actually nothing else than a *characteristic solver*: the problem is decomposed into characteristics, which are advected each in their own directions. Indeed, for linear problems the principle of using Riemann problems at each interface to perform the numerical integration of the equations is identical to the principle of decomposing into the eigenvectors of the Jacobian and advecting each component with its own eigenvalue as characteristic speed. In other words: *For linear problems a Riemann solver is identical to a characteristic solver.* This is not true anymore for non-linear problems: as we shall see later on, a characteristic solver for hydrodynamics is not a true Riemann solver but an *approximate Riemann solver* or equivalently a *linearized Riemann solver*. However, let us, for now, stick to linear problems a bit longer.

6.5.2 General expressions for Godunov solvers for linear problems

We can make a general expression for the flux, for any number of characteristics:

$$f_{i+1/2} = \sum_{k=1\cdots K} \tilde{f}_{k,i+1/2} e_k \tag{6.48}$$

where K is the number of characteristics (i.e. number coupled equations, or number of eigenvectors and eigenvalues), and where

$$\tilde{f}_{k,i+1/2} = \frac{1}{2}\lambda_k \Big[(1+\theta_k)\tilde{q}^n_i + (1-\theta_k)\tilde{q}^n_{i+1} \Big] \tag{6.49}$$

where $\theta_k = 1$ if $\lambda_k > 0$ and $\theta_k = -1$ if $\lambda_k < 0$. This is identical to the expressions we derived in Subsection 6.5, but now more general.

6.5.3 Higher order Godunov scheme

As we know from Chapter 4, the donor-cell algorithm is not the most sophisticated advection algorithm. We therefore expect the solutions based on Eq. (6.49) to be smeared out quite a bit. In Chapter 4 we found solutions to this problem by dropping the condition that the states are piecewise constant (as we have done in the Godunov scheme so far) and introduce a piecewise linear subgrid model, possibly with a flux limiter. In principle, for linear problems the Godunov scheme is identical to the advection problem for each individual characteristic, and therefore we can apply such linear subgrid models here too. In this way we generalize the Godunov scheme in such a way that we have a slightly more complex subgrid model in each cell (i.e. non-constant), but the principle remains the same. At each interface we define $\tilde{r}_{k,i-1/2}$:

$$\tilde{r}^n_{k,i-1/2} = \begin{cases} \frac{\tilde{q}^n_{k,i-1} - \tilde{q}^n_{k,i-2}}{\tilde{q}^n_{k,i} - \tilde{q}^n_{k,i-1}} & \text{for } \lambda_{k,i-1/2} \geq 0 \\ \\ \frac{\tilde{q}^n_{k,i+1} - \tilde{q}^n_{k,i}}{\tilde{q}^n_{k,i} - \tilde{q}^n_{k,i-1}} & \text{for } \lambda_{k,i-1/2} \leq 0 \end{cases} \tag{6.50}$$

where again k denotes the eigenvector/-value, i.e. the characteristic. We can now define the flux limiter $\tilde{\phi}(\tilde{r}_{k,i-1/2})$ for each of these characteristics according to the formulae in Section 4.5 (i.e. Eqs. 4.39, 4.40, 4.41). Then the flux is given by (cf. Eq. 4.38)

$$\begin{aligned} \tilde{f}^{n+1/2}_{k,i-1/2} =& \frac{1}{2}\lambda_{k,i-1/2}\Big[(1+\theta_{k,i-1/2})\tilde{q}^n_{k,i-1/2,L} + (1-\theta_{k,i-1/2})\tilde{q}^n_{k,i-1/2,R} \Big] + \\ & \frac{1}{2}|\lambda_{k,i-1/2}| \left(1 - \left| \frac{\lambda_{k,i-1/2}\Delta t}{\Delta x} \right| \right) \phi(\tilde{r}^n_{k,i-1/2})(\tilde{q}^n_{k,i-1/2,R} - \tilde{q}^n_{k,i-1/2,L}) \end{aligned} \tag{6.51}$$

It is useful, for later, to derive an alternative form of this same equation, which can be obtain with a bit of algebraic manipulation starting from Eq. (6.51). We use the identity $|\lambda_{k,i-1/2}| = \theta_{k,i-1/2}\lambda_{k,i-1/2}$ and the definition $\epsilon_{k,i-1/2} \equiv \lambda_{k,i-1/2}\Delta t/(x_i - x_{i-1})$ and obtain:

$$\begin{aligned} \tilde{f}^{n+1/2}_{k,i-1/2} =& \tfrac{1}{2}\lambda_{k,i-1/2}(\tilde{q}^n_{k,i-1/2,R} + \tilde{q}^n_{k,i-1/2,L}) \\ & - \tfrac{1}{2}\lambda_{k,i-1/2}(\tilde{q}^n_{k,i-1/2,R} - \tilde{q}^n_{k,i-1/2,L})[\theta_{k,i-1/2} + \tilde{\phi}_{k,i-1/2}(\epsilon_{k,i-1/2} - \theta_{k,i-1/2})] \end{aligned} \tag{6.52}$$

where $\tilde{\phi}_{k,i-1/2} \equiv \phi(\tilde{r}_{k,i-1/2})$. If we define the decomposed fluxes at the left and right side of the interface as

$$\tilde{f}_{k,i-1/2,L} = \lambda_{i-1/2}\tilde{q}_{k,i-1/2,L} \tag{6.53}$$

$$\tilde{f}_{k,i-1/2,R} = \lambda_{i-1/2}\tilde{q}_{k,i-1/2,R} \tag{6.54}$$

then we obtain:

$$\begin{aligned}\tilde{f}^{n+1/2}_{k,i-1/2} =& \tfrac{1}{2}(\tilde{f}^{n}_{k,i-1/2,R} + \tilde{f}^{n}_{k,i-1/2,L}) \\ & - \tfrac{1}{2}(\tilde{f}^{n}_{k,i-1/2,R} - \tilde{f}^{n}_{k,i-1/2,L})[\theta_{k,i-1/2} + \tilde{\phi}_{k,i-1/2}(\epsilon_{k,i-1/2} - \theta_{k,i-1/2})]\end{aligned} \tag{6.55}$$

Now we can arrive at our final expression by adding up all the partial fluxes (i.e. the fluxes of all eigen-components):

$$\begin{aligned}f^{n+1/2}_{i-1/2} =& \tfrac{1}{2}(f^{n}_{i-1/2,R} + f^{n}_{i-1/2,L}) \\ & - \tfrac{1}{2}\sum_{k=1\cdots K}(\tilde{f}^{n}_{k,i-1/2,R} - \tilde{f}^{n}_{k,i-1/2,L})[\theta_{k,i-1/2} + \tilde{\phi}_{k,i-1/2}(\epsilon_{k,i-1/2} - \theta_{k,i-1/2})]\end{aligned} \tag{6.56}$$

This is our final expression for the (time-step-averaged) interface flux.

There are a number of things we can learn from this expression:

1. The interface flux is the simple average flux plus a diffusive correction term. All the ingenuity of the characteristic solver lies in the diffusive correction term.

2. The flux limiter can be seen as a switch between donor-cell ($\tilde{\phi} = 0$) and Lax-Wendroff ($\tilde{\phi} = 1$), where we here see yet again another interpretation of Lax-Wendroff: the method in which the interface flux is found using a linear upwind interpolation (in contrast to upwinding, where the new *state* is found using linear upwind interpolation). Of course, if $\tilde{\phi}$ is one of the other expressions from Section 4.5, we get the various other schemes.

In all these derivations we must keep in mind the following caveats:

- Now that the states in the adjacent cells is no longer constant, the Riemann problem is no longer self-similar: The flux at the interface changes with time.
- In case the advection matrix is non-constant in space, the determination of the slopes becomes a bit less mathematically clean: Since the eigenvectors now change from one cell interface to the next, the meaning of $\tilde{q}_{k,i+1/2,R} - \tilde{q}_{k,i+1/2,L}$ is no longer identical to the meaning of $\tilde{q}_{k,i-1/2,R} - \tilde{q}_{k,i-1/2,L}$. Although the method works well, the mathematical foundation for this method is now slightly less strong.

6.6 Roe's linearized Riemann solver

6.6.1 The equations of hydrodynamics revisited

Before we construct our linearized Riemann solver, let us make a slightly modified definition of the state vector q, which allows us an easy generalization of our algorithms to 3-D. We define q to be

$$q = \begin{pmatrix} \rho e_{\mathrm{tot}} \\ \rho u \\ \rho v \\ \rho w \\ \rho \end{pmatrix} \tag{6.57}$$

For convenience we shall index it from 0 to 4:

$$q_0 = \rho e_{\mathrm{tot}} \quad q_1 = \rho u \quad q_2 = \rho v \quad q_3 = \rho w \quad q_4 = \rho \tag{6.58}$$

The full set of equations for 3-D hydrodynamics is then:

$$\partial_t q + \partial_x f_x(q) + \partial_y f_y(q) + \partial_z f_z(q) = 0 \tag{6.59}$$

where

$$f_x = \begin{pmatrix} \rho h_{\rm tot} u \\ \rho u^2 + P \\ \rho v u \\ \rho w u \\ \rho u \end{pmatrix} \qquad f_y = \begin{pmatrix} \rho h_{\rm tot} v \\ \rho u v \\ \rho v^2 + P \\ \rho w v \\ \rho v \end{pmatrix} \qquad f_z = \begin{pmatrix} \rho h_{\rm tot} w \\ \rho u w \\ \rho v w \\ \rho w^2 + P \\ \rho w \end{pmatrix} \tag{6.60}$$

6.6.2 Linearized Riemann solvers

We have seen that for linear problems the Riemann solver reduces to a characteristic solver. For the full non-linear set of equations of hydrodynamics this is no longer the case. A Riemann solver, such as Godunov's method, is then a rather complex solver because it involves complex and non-linear solutions to the Riemann problem at each cell interface. It is also rather costly to solve numerically. Cheaper and elegant simplifications are *linearized Riemann solvers*. The way this is done is by expressing our interface fluxes as much as possible only using the differences in the state variables:

$$\Delta q_{k,i-1/2} \equiv q_{k,i-1/2,R} - q_{k,i-1/2,L} \tag{6.61}$$

If these differences are small, then much of the algebra can be linearized to first order in $\Delta q_{k,i-1/2}$.

If we now set the state at the beginning of each time step constant within each cell, then the cell interfaces have jumps of the state, i.e. they define a Riemann problem. The way to linearize this is to define an average state at the interface $\hat{q}_{k,i-1/2}$ (note: here we retain the index k of the index notation) in some way:

$$\hat{q}_{k,i-1/2} = \text{Average}[q_{k,i}, q_{k,i-1}] \tag{6.62}$$

where the precise definition of the average will be defined later. For now we can simply set $\hat{q}_{k,i-1/2} = (q_{k,i} + q_{k,i-1})/2$ for instance. Now we can express the Riemann problem in the deviation from this average:

$$\delta q_{k,i-1/2,L} \equiv q_{k,i-1} - \hat{q}_{k,i-1/2} \tag{6.63}$$

$$\delta q_{k,i-1/2,R} \equiv q_{k,i} - \hat{q}_{k,i-1/2} \tag{6.64}$$

If $|\delta q_{k,i-1/2,L/R}| \ll |\hat{q}_{k,i-1/2}|$ then, locally, the Riemann problem can be regarded as a linear Riemann problem, which we have extensively discussed in Section 6.5. The advection matrix in x direction is now simply the Jacobian $\partial f_x(q)/\partial q$, so the equation, locally between $x_{i-1} < x < x_i$ becomes:

$$\partial_t \delta q + \left(\frac{\partial f_x}{\partial q}\right) \partial_x \delta q = 0 \tag{6.65}$$

The eigenvalues of the Jacobian $\partial f_x/\partial q$ at the interface $i - 1/2$ are (for convenience we leave out the $i - 1/2$ index):

$$\lambda_1 = \hat{u} - \hat{C}_s \tag{6.66}$$

$$\lambda_2 = \hat{u} + \hat{C}_s \tag{6.67}$$

$$\lambda_3 = \hat{u} \tag{6.68}$$

$$\lambda_4 = \hat{u} \tag{6.69}$$

$$\lambda_5 = \hat{u} \tag{6.70}$$

with eigenvectors:

$$e_1 = \begin{pmatrix} \hat{h}_{\rm tot} - \hat{C}_s\hat{u} \\ \hat{u} - \hat{C}_s \\ \hat{v} \\ \hat{w} \\ 1 \end{pmatrix} e_2 = \begin{pmatrix} \hat{h}_{\rm tot} + \hat{C}_s\hat{u} \\ \hat{u} + \hat{C}_s \\ \hat{v} \\ \hat{w} \\ 1 \end{pmatrix} \tag{6.71}$$

$$e_3 = \begin{pmatrix} \frac{1}{2}\hat{u}^2 \\ \hat{u} \\ \hat{v} \\ \hat{w} \\ 1 \end{pmatrix} e_4 = \begin{pmatrix} \hat{v}^2 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} e_5 = \begin{pmatrix} \hat{w}^2 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \tag{6.72}$$

where $\hat{h}_{\rm tot} = \hat{e}_{\rm tot} + \hat{P}/\hat{\rho}$ is the total specific enthalpy and $\hat{C}_s = \sqrt{\gamma\hat{P}/\hat{\rho}}$ is the adiabatic sound speed. In all symbols the caret $\hat{}$ indicates that these are the primitive variables as derived from the "average state" $\hat{q}_{i-1/2}$ at the location of interface $i-1/2$. Note that we can derive similar expressions for the advection in y and z direction.

We can now directly insert those formulae to Eq. (6.56) and apply this to the values of $\delta q_{k,i-1/2,L/R}$. Now the special form of Eq.(6.56) comes to our advantage, because since this expression (and the expressions for the flux limiters) only depend on the difference $\delta q_{k,i-1/2,R} - \delta q_{k,i-1/2,L} \equiv q_{k,i-1/2,R} - q_{k,i-1/2,L} \equiv \Delta q_{k,i-1/2}$, we can now forget about $\delta q_{k,i-1/2,L/R}$ and focus entirely on $\Delta q_{k,i-1/2}$, which is the jump of the state over the interface. We can now decompose $\Delta q_{k,i-1/2}$ into the eigenvectors Eq. (6.71...6.72):

$$\Delta q_{k,i-1/2} = \sum_{m=1\cdots5} \tilde{\Delta} q_{m,i-1/2} e_{m,k,i-1/2} \tag{6.73}$$

where (again for clarity we omit the index $i-1/2$):

$$\tilde{\Delta} q_1 = \frac{\gamma-1}{2\hat{C}_s^2}\left\{\hat{e}_{\rm kin}\Delta q_4 - \xi\right\} - \frac{\Delta q_1 - \hat{u}\Delta q_4}{2\hat{C}_s} \tag{6.74}$$

$$\tilde{\Delta} q_2 = \frac{\gamma-1}{2\hat{C}_s^2}\left\{\hat{e}_{\rm kin}\Delta q_4 - \xi\right\} + \frac{\Delta q_1 - \hat{u}\Delta q_4}{2\hat{C}_s} \tag{6.75}$$

$$\tilde{\Delta} q_3 = \frac{\gamma-1}{2\hat{C}_s^2}\left\{(\hat{h}_{\rm tot} - 2\hat{e}_{\rm kin})\Delta q_4 + \xi\right\} \tag{6.76}$$

$$\tilde{\Delta} q_4 = \Delta q_2 - v\Delta q_4 \tag{6.77}$$

$$\tilde{\Delta} q_5 = \Delta q_3 - w\Delta q_4 \tag{6.78}$$

where $\hat{e}_{\rm kin} = (\hat{u}^2 + \hat{v}^2 + \hat{w}^2)/2$ and $\xi \equiv u\Delta q_1 + v\Delta q_2 + w\Delta q_3 - \Delta q_0$. With these expressions for $\tilde{\Delta} q_{i-1/2}$ the flux at the interface becomes (cf. Eq. 6.56) becomes

$$\begin{aligned} f^{n+1/2}_{k,i-1/2} =& \tfrac{1}{2}(f^n_{k,i-1/2,R} + f^n_{k,i-1/2,L}) \\ & - \tfrac{1}{2}\sum_{m=1\cdots5} \lambda_{m,i-1/2}\tilde{\Delta} q_{k,i-1/2}[\theta_{m,i-1/2} + \tilde{\phi}_{m,i-1/2}(\epsilon_{m,i-1/2} - \theta_{m,i-1/2})] \end{aligned} \tag{6.79}$$

where we retained the index k in the expression for the flux: $f^{n+1/2}_{k,i-1/2}$ according to index notation. We now see that the interface flux for this linearized Riemann solver consists of the average of

the non-linear fluxes plus a correction term in which the difference of the flux over the interface is decomposed into eigenvectors and each component advected in its own upwind fashion.

A linearized Riemann solver is evidently not an exact Riemann solver, since the Riemann problem is solved in an approximate way only. This is why linearized Riemann solvers are part of the (larger) family of *approximante Riemann solvers*.

6.6.3 Roe's average interface state

The final missing piece of the algorithm is a suitable expression for the "average interface state", or better, the "average interface primitive variables" $\hat{u}_{i-1/2}$, $\hat{v}_{i-1/2}$, $\hat{w}_{i-1/2}$, $\hat{\rho}_{i-1/2}$, $\hat{h}_{\mathrm{tot},i-1/2}$. As long as the numerical solution is very smooth, i.e. that $|\Delta q_{k,i-1/2}| \ll |\hat{q}_{k,i-1/2}|$, then any reasonable average would do and would probably give the right results. However, when contact discontinuities and/or shock waves are present in the solution, then it becomes extremely important to define the proper average such that the "linearization" (which is then strictly speaking no longer valid) still produces the right propagation of these discontinuities.

A *Roe solver* is a linearized Riemann solver with a special kind of averaged state at the interface. These state variables are defined as:

$$\hat{u} = \frac{\sqrt{\rho_L}u_L + \sqrt{\rho_R}u_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{6.80}$$

$$\hat{v} = \frac{\sqrt{\rho_L}v_L + \sqrt{\rho_R}v_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{6.81}$$

$$\hat{w} = \frac{\sqrt{\rho_L}w_L + \sqrt{\rho_R}w_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{6.82}$$

$$\hat{h}_{\mathrm{tot}} = \frac{\sqrt{\rho_L}h_{\mathrm{tot},L} + \sqrt{\rho_R}h_{\mathrm{tot},R}}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{6.83}$$

With these expressions for the average interface state and the above defined eigenvector decomposition and interface flux expressions, as well as the flux limiter, the Roe solver is complete.

6.6.4 Background of Roe's average interface state

So where do these expressions for the interface state come from? Here we follow the book by LeVeque. The basic idea behind the Roe solver is that it should recognize when a jump of the state $q_R - q_L$ is a pure jump in one characteristic family only, and in that case produce the exact propagation velocity. So for a shock or for a contact discontinuity belonging to just one characteristic family the interface-average propagation matrix $\hat{A}$ (which is the interface average of the Jacobian $\partial f_x/\partial q$) should have the property:

$$\hat{A}(q_R - q_L) = f_{x,R} - f_{x,L} = s(q_R - q_L) \tag{6.84}$$

where s is the propagation speed of this wave. Note that this should not only hold for small $q_R - q_L$, but also when the jump is macroscopic. In case of a contact discontinuity it should get $s = u$ and for a shock wave it should obtain $s = v_s$ where v_s is the shock velocity.

There is an elegant mathematical derivation of how we obtain Roe's parameterization from

the above condition, which is based on the definition of a state vector W defined as:

$$W = \sqrt{\rho} \begin{pmatrix} h_{\text{tot}} \\ u \\ v \\ w \\ 1 \end{pmatrix} \tag{6.85}$$

also counted from 0 to 4. The nice property of this parameter vector is that the state vector is perfectly quadratic in the components of W:

$$q = \begin{pmatrix} \frac{1}{\gamma} W_0 W_4 + \frac{\gamma-1}{2\gamma}(W_1^2 + W_2^2 + W_3^2) \\ W_1 W_4 \\ W_2 W_4 \\ W_3 W_4 \\ W_4^2 \end{pmatrix} \tag{6.86}$$

and so is the flux f_x:

$$f_x = \begin{pmatrix} W_0 W_1 \\ \frac{\gamma-1}{\gamma} W_0 W_4 + \frac{\gamma+1}{2\gamma} W_1^2 - \frac{\gamma-1}{2\gamma}(W_2^2 + W_3^2) \\ W_1 W_2 \\ W_1 W_3 \\ W_1 W_4 \end{pmatrix} \tag{6.87}$$

and similar for f_y and f_z. This purely quadratic relation is very useful for the following argumentation.

If two states q_R and q_L are connected by a single wave (shock or contact discontinuity), then they obey

$$f_x(q_R) - f_x(q_L) = s(q_R - q_L) \tag{6.88}$$

where s is the propagation speed of this wave. We wish that our matrix $\hat{A}$ (our "Jacobian") recognizes this. So it should have the property that

$$\hat{A}(q_R - q_L) = s(q_R - q_L) \tag{6.89}$$

or in other words:

$$\hat{A}(q_R - q_L) = f_x(q_R) - f_x(q_L) \tag{6.90}$$

One way to obtain such a matrix is to integrate the Jacobian matrix $A(q)$ over a suitable path in parameter space from q_L to q_R. We can do this for instance by defining a straight path:

$$q(\xi) = q_L + (q_R - q_L)\xi \tag{6.91}$$

so that we get

$$\begin{aligned} f_x(q_R) - f_x(q_L) &= \int_0^1 \frac{\partial f_x(q(\xi))}{\partial \xi} d\xi \\ &= \int_0^1 \frac{\partial f_x(q(\xi))}{\partial q} \frac{\partial q(\xi)}{d\xi} d\xi \\ &= \left[\int_0^1 \frac{\partial f_x(q)}{\partial q} d\xi \right] (q_R - q_L) \end{aligned} \tag{6.92}$$

where we used $\partial q(\xi)/\partial\xi = (q_R - q_L)$. This the gives us the matrix $\hat{A}$:

$$\hat{A} = \int_0^1 \frac{\partial f_x(q)}{\partial q} d\xi \tag{6.93}$$

The problem with this integral is that it is not guaranteed that this produces a matrix which has real eigenvalues. Moreover, it is hard to evaluate, and therefore computationally costly.

Now here the parameter vector W comes in. Let us do the same trick of integration, but this time not in q space but in W space:

$$W(\xi) = W_L + (W_R - W_L)\xi \tag{6.94}$$

so that we get

$$\begin{aligned} f_x(q_R) - f_x(q_L) &= \left[\int_0^1 \frac{\partial f_x(W)}{\partial W} d\xi\right](W_R - W_L) \\ &\equiv \hat{C}(W_R - W_L) \end{aligned} \tag{6.95}$$

We can do the same for $q_R - q_L$:

$$\begin{aligned} q_R - q_L &= \left[\int_0^1 \frac{\partial q(W)}{\partial W} d\xi\right](W_R - W_L) \\ &\equiv \hat{B}(W_R - W_L) \end{aligned} \tag{6.96}$$

So we have now two matrices B and C. We can now construct the matrix $\hat{A}$:

$$\hat{A} = \hat{C}\hat{B}^{-1} \tag{6.97}$$

Now the nice thing of the parameter vector W is that both q and f_x are purely quadratic in W, and therefore the $\partial q/\partial W$ and $\partial f_x/\partial W$ are *linear* in W. This makes it much easier to evaluate the integral through W space! We shall not derive the final expressions. It suffices to say that if we would derive $\hat{A}$ in the above way we obtain a matrix with eigenvalues and eigenvectors as we have given above. Roe's choice of average interface state variables can now be understood as originating from the fact that this particular choice of parameter vector W makes q and f_x quadratic in W.

6.6.5 The complete recipe of a Roe solver

Although we have discussed the complete algorithm already, let us finish this section with a point-by-point recipe for a Roe solver:

1. Determine first the Δt using a CFL number of 0.5
2. Construct the state vector $q = (\rho e_{\rm tot}, \rho u, \rho v, \rho w, \rho)$ at each cell center. This q_i also automatically defines the states at each side of the interfaces: $q_{i-1/2,R} = q_i$, $q_{i-1/2,L} = q_{i-1}$.
3. Construct Roe's averages at the cell interfaces.
4. Using Roe's averages, create the eigenvectors and eigenvalues
5. Compute the flux jump over the interface and decompose this jump into the eigen-components to obtain the values of $\tilde{\Delta} q_{k,i-1/2}$.

6. Compute the flux limiter for each of the eigen-components.

7. Compute the symmetric flux average $(f_{i-1/2,L} + f_{i-1/2,R})/2$, and add the diffusive correction term using the flux limiter and the $\tilde{\Delta}q$. This creates the interface flux.

8. Now update the state vector using the interface fluxes and the Δt computed using the CFL condition.

9. End of time step; off to the next time step

6.7 Properties of the Roe solver

6.7.1 Strengths of the Roe solver

The Roe solver is a very powerful solver:

- It resolves shocks and contact discontinuities very tightly (in roughly 3 grid points). It is therefore significantly higher resolution than classical hydrodynamics solvers.
- It does not require any artificial viscosity for shocks because it treats shocks directly. It is, so to speak, a *shock-capturing scheme*.
- It has a very low numerical viscosity/diffusivity.
- It is strictly conserving in mass, momentum and energy.
- Because it treats pressure gradients as characteristics, sound waves are propagated with the same precision as moving matter.

6.7.2 Problems of the Roe solver

The Roe solver has excellent performance for many problems, but sometimes it can produce problems. Here is a list of known problems of the solver:

- Under some (rare) conditions it can try to create *expansion shocks* where it should create smooth expansion waves. This is because the Roe solver is built in such a way as to recognize Rankine-Hugoniot jump conditions even if they are the reverse. A Roe solver decomposes any smooth wave into a series of small contact discontinuities or shocks. Even for expansion waves it can happen that one of these shocks becomes strong and produces an expansion shock. This is clearly unphysical and violates the entropy condition. If this happens an *entropy fix* is necessary: a trick that disallows an expansion shock to form. We refer to LeVeque's book for details.
- Just behind strong shocks in 2-D or 3-D flows, when the shock is parallel to the grid, sometimes waves can appear. This is an odd-even decoupling problem, and fixes have been proposed which involve a minute amount of diffusion parallel to the shock, but only in the neiborhood of the shock.
- Also for grid-parallel shocks one can sometimes observe severe protrusions appearing which have a pyramid shape and a checkerboard pattern. This is known as the *carbuncle phenomenon*. This problem is rare, but if it happens it is hard to solve.
- Due to the fact that the total energy equations is used in Roe solvers, in case of extremely

supersonic flows it can happen that small errors in the momentum equation can yield negative thermal energies (because $e_{\rm th} = e_{\rm tot} - u^2/2$). This can be a potentially serious problem of all schemes that use the total energy equation. Various fixes can be considered, which can be useful in various regimes, but a generally valid solution is difficult.

- A rare, but possible problem of the Roe solver is that it does not guarantee that the interface flux that it produces is physical. For instance in the Riemann problem with $\gamma = 1.5$, $\rho_L = 1$, $u_L = -2$, $P_L = 4/3$ and $\rho_R = 4$, $u_R = 1$ and $P_R = 13/3$ the first order upwind flux has an energy flux but no mass flux (Eulerink & Mellema 1995). This is clearly unphysical, and can lead to numerical problems.

In general, though, the Roe solver is quite stable and very non-viscous. It is a truly high-resolution method.

6.8 A brief note on other Riemann solvers

In this chapter we have mainly concentrated on the Roe solver. However, there are a number of other popular solution methods. One of them is the HLLE solver (named after Harten, Lax, van Leer and Einfeldt) which constructs the interface flux directly from a suitable averaged interface state which is generated using the fasted leftward and the fastest rightward moving characteristic. Various versions of this algorithm were developed in the literature (HLLEM, HLLEC) and the complete family of methods is sometimes called HLLX where X stands for your favorite variant.

Another popular Riemann solver is the *Piecewise Parabolic Method*. This is a generalization of Godunov's method in which the states in the two cells adjacent to the interface $i + 1/2$ are not assumed to be constant (as in Godunov's method and Roe's method) but are assumed to be parabolic. We will not go into detail here.

6.9 Code testing: the Sod shock tube tests

The Sod shock tube test of Section 6.3.2 can be used to test the performance of our computer program. We leave it to the reader to text the algorithms of the previous chapter on this kind of test. Here we merely shock the performance of the Roe solver on a test with $\rho_L = 10^5$, $P_L = 1$, $u_L = 0$, $\rho_R = 1.25 \times 10^4$, $P_R = 0.1$, $u_R = 0$ and $\gamma = 7/5$. The result is shown in Fig. 6.5.

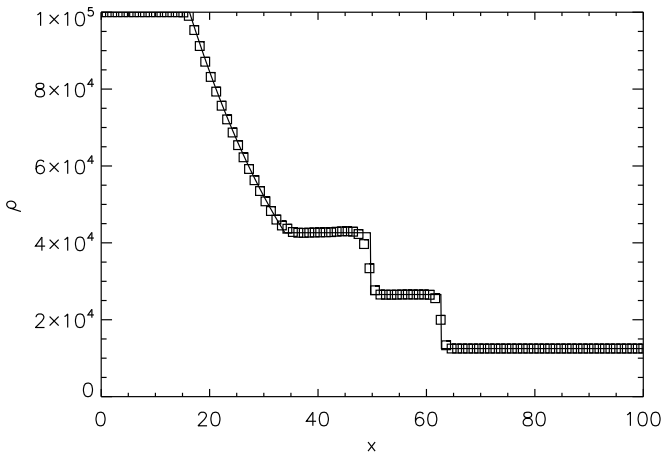

Figure 6.5. The result of the Roe solver with superbee flux limiter on a Sod shocktube test with $\rho_L = 10^5$, $P_L = 1$, $u_L = 0$, $\rho_R = 1.25 \times 10^4$, $P_R = 0.1$, $u_R = 0$ and $\gamma = 7/5$. Solid line: analytic solution; symbols: Roe solver.