2D Euler Solver for Unstructured Grids

By Means of the AUSM+ Scheme

Author: Nout van den Bos

Contents

1 Introduction

In this report, the implemented two-dimensional Euler solver for unstructured grids is discussed. This project came to be in order to apply the concepts that were learned during the M.Sc. Aerodynamics at the Delft University of Technology, and to further learn new concepts in the field of scientific programming. During the M.Sc., mostly structured grids were used, thus an unstructured grid solver was implemented out of curiosity. Furthermore, during the M.Sc. advanced finite volume methods are discussed, but only applied in one dimension. These topics gave inspiration to implement this 2D Euler solver. It is planned for the future to rewrite the solver in C++, and make the solver parallel using OpenMPI.
After the introduction, the Euler equations are shortly discussed. After this, the numerical methodology is elaborated upon. Simulated testcases are discussed as well, and finally the report is concluded.

2 Governing Equations

The governing equations that are considered, are the Euler equations in 2 spatial dimensions. The Euler equations can be derived from the Navier-Stokes equations, with the limit of viscosity going to zero, $\mu \rightarrow 0$. Given this limit, it is found that the Reynolds number goes to infinity in the case of the Euler equations, $Re \rightarrow \infty$, and thus the Euler equations can be used to approximate flows at high Reynolds numbers. In these cases, the Euler equations can be used to approximate the pressure distribution and to find the location of shock-waves.

The Euler equations in the integral form are shown in eq. (2.1), where W is the state vector as defined in eq. (2.2), and F is the flux vector as defined in eq. (2.3). V denotes the velocity magnitude, n is the unit normal vector, E is the total energy and H is the total enthalpy. The Euler equations consist of the conservation of mass, momentum in x- and y- direction, and energy. To close this system of equations, an equation of state is needed. The ideal gas equation is used, as shown in eq. (2.4), where γ is the specific heat ratio.

$$\frac{\partial}{\partial t}\int_{\Omega} W d\Omega + \oint_{\partial\Omega} F dS = 0 \tag{2.1}$$

$$W = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix} \tag{2.2}$$

$$F = \begin{pmatrix} \rho V \\ \rho u V + n_x p \\ \rho v V + n_y p \\ \rho V H \end{pmatrix} \tag{2.3}$$

$$E = \frac{p}{\rho(\gamma - 1)} + \frac{1}{2}V^2 \tag{2.4}$$

3 Methodology

In eq. (2.1), the integral form of the Euler equations are shown. This form is considered because the finite volume method is used. In the finite volume method, the domain is divided into separate "volumes" through a mesh (in a 2 dimensional case, these "volumes" are actually surfaces). For each individual cell, eq. (2.1) can then be applied, with some additional assumptions.

3.1. Finite Volume Method

Rather than solving for W, the equations are solved for the cell-averaged value $\bar{W}$, then, the integral $\int_\Omega \bar{W} d\Omega = \bar{W}\Omega$. For time discretization, the forward Euler scheme is used. It is assumed that the flux is constant along a cell's face (in the 2d case, the cell faces are line segments). With these assumptions, the following discretized equation is created:

$$\bar{W}^{n+1} = \bar{W}^n - \frac{\Delta t}{\Omega} \sum_i \hat{F}_i d\Omega_i \tag{3.1}$$

This equation is solved in each cell of the domain. $\hat{F}_i$ is the numerical flux of face i at each cell. However, the flux vectors must be evaluated at the faces, while the state vector is known in the cell centres. To evaluate the flux vector at the cell faces, the AUSM+ (Advection Upstream Splitting Method) scheme is used, by Liou [1].

3.2. The AUSM+ Scheme

With the AUSM+ scheme, the flux is split up into a velocity flux and a pressure flux, as shown in eq. (3.2).

$$\hat{F} = V \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho H \end{pmatrix} + p \begin{pmatrix} 0 \\ n_x \\ n_y \\ 0 \end{pmatrix} \tag{3.2}$$

In the AUSM+ scheme, each face has a "left" cell and a "right" cell, which is determined by the normal direction of the velocity. Furthermore, the AUSM+ makes use of "interface" values, which are combining values of the left and right state. The left state is defined with a L subscript, the right state with a R and the interface state with a L/R subscript. The flux at the interface between the left and the right cell is then defined in the AUSM+ scheme as in eq. (3.3). Here $\hat{f}_L$ and $\hat{f}_R$ are the velocity flux at the left and right cell respectively, as shown in eq. (3.4).

$$\hat{F}_{L/R} = \frac{1}{2} M_{L/R} a_{L/R} (\hat{f}_L + \hat{f}_R) - \frac{1}{2} |M_{L/R}| a_{L/R} (\hat{f}_L - \hat{f}_R) + p_{L/R} \begin{pmatrix} 0 \\ n_x \\ n_y \\ 0 \end{pmatrix} \tag{3.3}$$

$$\hat{f}_{L(R)} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho H \end{pmatrix}_{L(R)} \tag{3.4}$$

The interface speed of sound $a_{L/R}$ is the average of the adjusted speed of sound from the left and right cell, as shown in eq. (3.5). The adjusted speed of sound is calculated by eq. (3.6) and eq. (3.7), this is used as to limit overshoots, which occurred in the AUSM scheme.

$$a_{L/R} = \frac{a_L + a_R}{2} \tag{3.5}$$

$$a = \frac{a_{crit}^2}{/} max(a_{crit}, V) \tag{3.6}$$

$$a_{crit} = \sqrt{2H\frac{\gamma - 1}{\gamma + 1}} \tag{3.7}$$

Finally, the interface Mach number and pressure are also combinations of the left and the right state, but Mach splitting functions are used. These splitting functions are used to increase the accuracy of the upwind scheme. The functions have several properties such as consistency, monotonically increasing, symmetry and continuously differentiable for at least 1 order, among others. The definitions of the interface Mach number and pressure are shown in eq. (3.8) and eq. (3.9) respectively, with the splitting functions in eq. (3.10) and eq. (3.11).

$$M_{L/R} = \mathscr{M}^+(M_L) + \mathscr{M}^-(M_R) \tag{3.8}$$

$$p_{L/R} = \mathscr{P}^+(M_L)p_L + \mathscr{P}^-(M_R)p_R \tag{3.9}$$

$$\mathscr{M}^{\pm}(M) = \begin{cases} \frac{1}{2}(M \pm |M|) & \text{if } |M| > 1 \\ \pm\frac{1}{4}(M \pm 1)^2 \pm \frac{1}{8}(M^2 - 1)^2 & \text{otherwise} \end{cases} \tag{3.10}$$

$$\mathscr{P}^{\pm}(M) = \begin{cases} \frac{1}{2}(1 \pm \text{sign}(M)) & \text{if } |M| > 1 \\ \pm\frac{1}{4}(M \pm 1)^2(2 \mp M) \pm \frac{3}{16}(M^2 - 1)^2 & \text{otherwise} \end{cases} \tag{3.11}$$

3.3. Boundary Conditions

Finally, the methods for the boundary conditions are discussed. For now, two types of boundary conditions are available, the freestream boundary condition, and the slip wall boundary condition. The boundary conditions are implemented by means of ghost cells. The freestream boundary condition is implemented by means of the assumption that the state vector of the boundary face is equal in both cells, i.e. with constant extrapolation.

The slip wall is slightly more complex. While the density and the pressure are extrapolated to the ghost cell, this can not be done for the velocity. This is because at the boundary face, the normal velocity must be equal to zero. This is arranged by prescribing the reflection of the boundary cell velocity to the ghost cell. Since the velocity vector is reflected over the boundary face, the velocity at the face will be fully tangential, with a zero normal component.

4 Results

The method has been implemented in MATLAB, and two test cases have been studied to verify the solver. Namely, Sod's shocktube and a forward step at Mach 3. These cases have been used in the past to verify other Euler solvers and numerical schemes, and the results are readily available.

4.1. Sod's Shocktube

Sod's shocktube is a shock tube problem where the shock tube has a discontinuity in the pressure and the density at the initial conditions, the initial velocity is zero. In principle is Sod's shocktube problem a one-dimensional problem, as the solution should only vary in the lengthwise direction. Furthermore, the analytical solution is known, which makes it a good candidate for a comparison with the implemented Euler solver. For the sake of brevity, only the solution for the pressure field is investigated.
As the implemented solver can handle unstructured meshes, an unstructured grid is used to test the solver. The mesh is shown in fig. 4.1.

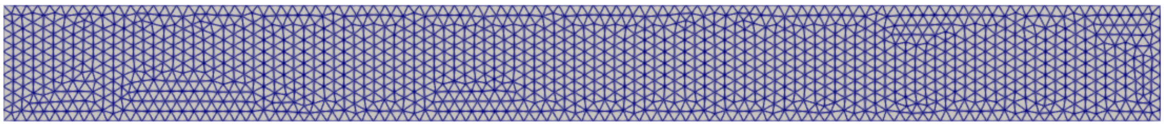

Figure 4.1: Structured mesh for Sod's shocktube.

The shock tube has a left state and a right state, separated at $x = 0.5$. The left state has a pressure of 1 bar and a density of 1 $\frac{kg}{m^3}$, whereas the right state has a pressure of 0.1 bar and a density of 0.125 $\frac{kg}{m^3}$. The initial pressure is shown in fig. 4.2. Note that the discontinuity in the middle is not exactly straight, this is due to the fact that an unstructured mesh is used.

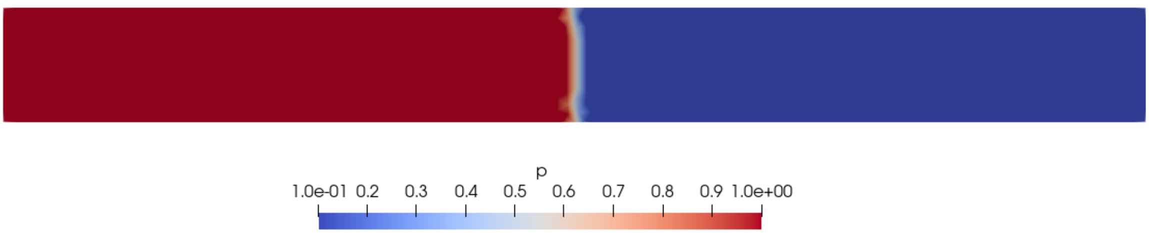

Figure 4.2: The initial pressure for Sod's shocktube.

The simulation is run for 0.17 seconds. Because of the discontinuity in pressure and density, an information wave will travel both in the left and the right direction, while the flow moves solely from the high pressure side to the low pressure side. The results for the pressure are shown in fig. 4.2. As mentioned, the exact solution is also known. In order to compare the simulation to the exact solution, the pressure is averaged in the y-direction, such that the data is also one-dimensional. In fig. 4.4, the pressure of the 2D Euler solver is compared to the analytical solution. As can be seen, the AUSM+ scheme gives an accurate shape of the pressure, with only a small discrepancy at large discontinuities.

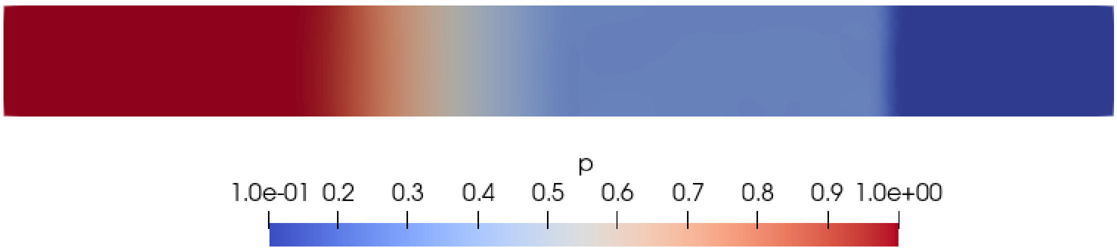

Figure 4.3: The pressure for Sod's shocktube at t = 0.17s.

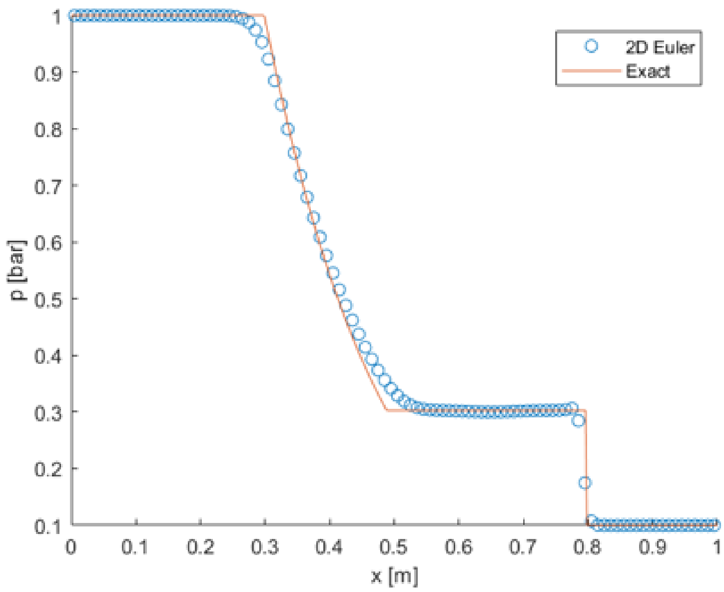

Figure 4.4: The pressure for Sod's shocktube at t = 0.17s, compared to the analytical solution.

4.2. Forward Step at Mach 3

While the previous test case has shown good agreement with the analytical solution, it does not test the 2D Euler solver to its full capabilities. The forward step at Mach 3 is a two-dimensional problem, that includes standing shocks and reflected shocks. For these reasons, this test case was also evaluated.

The domain of the forward step is shown in fig. 4.5. The domain consists of an inlet on the left side, an outlet on the right side, and walls at the top and bottom boundaries. The height of the inflow is h, the total length of the domain is $3h$, the step height is $0.2h$, and it starts at $0.6h$ from the inlet. The inlet conditions are $M = 3$, $p = 101320\ Pa$, and $\rho = 1\ kg/m^3$. The velocity direction is fully in x-direction (length-wise direction). For this case, $h = 1$ is taken.

Figure 4.5: The domain of the forward step test case.

The mesh consists of quadrilateral cells, with an average size of $\Delta x = \Delta y = 0.0125$, resulting in a mesh size of roughly 19,000 cells. A close up of the mesh is shown near the step, in fig. 4.6.

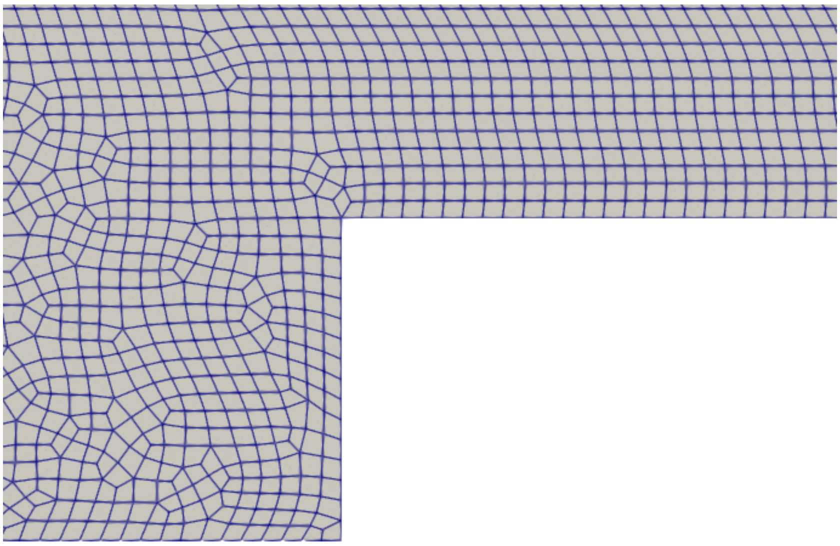

Figure 4.6: A close up of the mesh at the step.

The simulation is run until steady state is reached, which was after roughly 4,200 iterations. The Mach and density fields of the steady state solution are shown in fig. 4.7 and fig. 4.8, respectively. At $x = 0.3h$ a standing shockwave can be observed, which curves around the step, and reflects against the upper wall. This wave is again reflected on the step at roughly $x = 1.2h$, and once again at the top wall at roughly $x = 2.2h$.

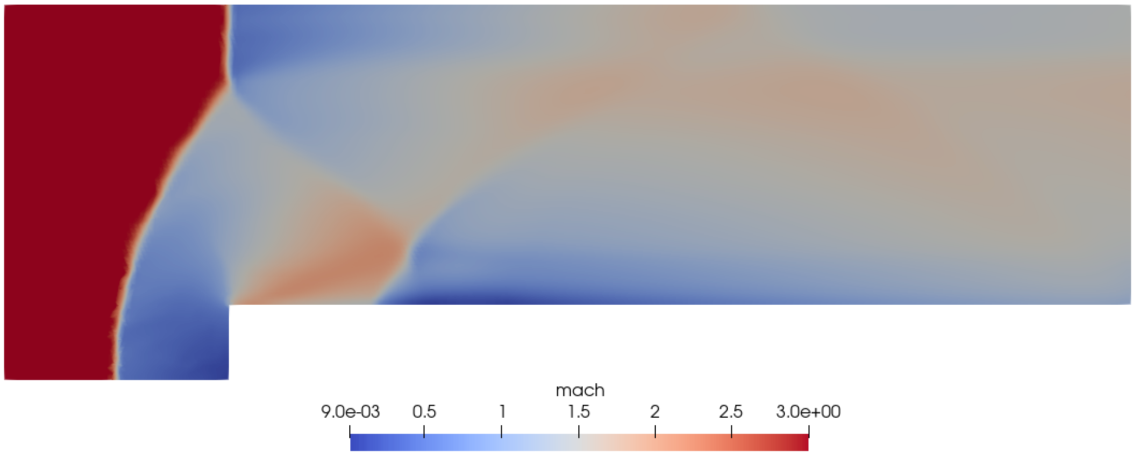

Figure 4.7: The Mach field for the forward step at Mach 3.

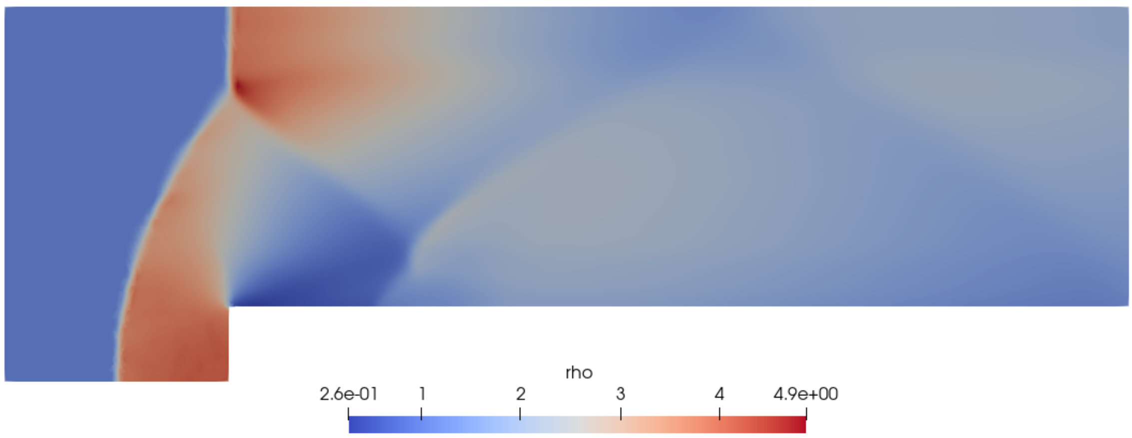

Figure 4.8: The density field for the forward step at Mach 3, given in kg/m^3.

In fig. 4.9, the density field as solved by the Discontinuous-Galerkin (DG) method by Gallego-Valencia et al. [2] is shown. As can be seen, the location of the normal shock wave is accurately predicted, as well as the shock attachment location at the upper wall, and the first reflection on the step. The currently used Euler solver predicts the last reflection earlier than [], and it can be seen that the solution is more smeared out. This is due to the fact that the current method is a first-order upwind method, which introduces numerical diffusion. The method used by Gallego-Valencia et al. [2] is a higher order method, and a mesh size of $\Delta x = \Delta y = 0.01$, meaning that their cells are roughly 40% smaller than the one used for the current simulation. Given these differences, a small discrepancy in the solution is expected.

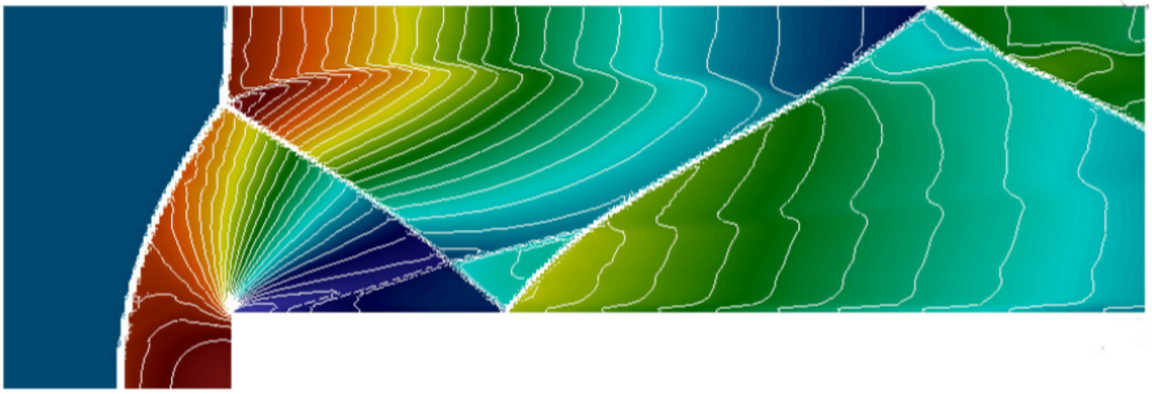

Figure 4.9: The density field contours for the forward step at Mach 3, as computed by the DG method from Gallego-Valencia et al. [2].

5 Conclusion

In conclusion, a 2D Euler solver for unstructured grids has been successfully implemented in MATLAB, using the finite volume method. The implemented method discretizes the temporal derivative through means of the Forward Euler scheme, and the flux is approximated at the cell faces through the AUSM+ scheme. For boundary conditions, ghost cells are implemented.
With this 2D Euler solver, both Sod's shocktube and a forward step at Mach 3 were simulated. From Sod's shocktube it was found that the pressure closely resembled the analytical solution, showing only minor deviations at large gradients. The forward step at Mach 3 was compared to a reference solution with a finer grid and a higher order method. It was found that the location of the normal shock and the first two reflections were well approximated. Further downstream the solution was more smeared out, and the accuracy of the shock reflection decreased. These discrepancies are due to the fact that a lower order method is implemented, which introduced numerical diffusion, and a coarser grid was used. Considering these factors, the implemented Euler solver gave satisfying results in terms of predicting the pressure, density and Mach number in both test cases.
For the future development of this code, the emphasis is put on the efficiency. The code is being ported to C++, and it is planned to implement a parallel version of the code.

Bibliography

[1] Meng-Sing Liou. A sequel to ausm: Ausm+. *Journal of Computational Physics*, 129(2):364–382, 1996.

[2] Juan Gallego-Valencia, Christian Klingenberg, and Praveen Chandrashekar. On limiting for higher order discontinuous galerkin method for 2d euler equations. *Bulletin of the Brazilian Mathematical Society, New Series*, 47:335–345, 03 2016.