Compressible viscous flow solver

Árpád Veress / *János* Molnár / *József* Rohács

Abstract

Nowadays, in spite of disadvantages of turbulence closure models for RANS (Reynolds Averaged Navier-Stokes equations), they are at present the only tools available for the computation of complex turbulent flows of practical relevance. Their popularity comes from high efficiency in terms of accuracy and computational cost, which makes them widely used in commercial codes and related multidisciplinary applications. Hence, for modelling compressible flow, as a framework of complex inverse design optimisation tool, Navier-Stokes solver is implemented by using k-ω turbulence model in C++ environment. The governing equations in conservative form are deduced by using Favre averaging to filter local fluctuations. The code is based on structured, density based cell centred finite volume method. The convective terms are discretized by Roe approximated Riemann method. Central discretization is applied for diffusive terms. MUSCL approach is implemented for higher order spatial reconstruction with Mulder limiter for monotonicity preserving. Wilcox k-ω two equations turbulence model is implemented for turbulence modelling. The explicit system of the equations is solved by the 4th order Runge-Kutta method. The numerical boundary conditions are based on the method of characteristics. The interest is mostly in high speed aeronautical applications with the possibility of extension for surface optimisation. Hence, the applied validational test cases are in transonic and supersonic flow regime: circular bump in the transonic channel and compression corner.

Keywords

Navier-Stokes solver · k-ω turbulence modelling · CFD

Árpád Veress

Department of Aircraft and Ships, BME, H-1111 Budapest, Sztoczek u. 6, J ép. 426, Hungary
e-mail: averess@rht.bme.hu

János Molnár
József Rohács

Department of Aircraft and Ships, BME, H-1111 Budapest, Sztoczek u. 6, J ép. 426, Hungary

1 Introduction

The industrial developments must be more and more cost efficient. There are many engineering applications amongst them, in which the fluid dynamics play a significant role. However, the accurate prediction of the flow physics has always been a challenging problem. Generally, the product development process requires measurements, which are more cost and time consuming compared with numerical analyses and, in certain problems, can give results only for a limited number of situations, if it is possible to implement at all without any disturbances. On contrary, the virtual reality – beside the reduction of efforts turned to the measurements – has promising advances in modelling and visualizing freedom with causal economical advantages and without significant limitation on geometry, material, physics and boundary conditions. It has a wide range of possibilities for modelling and simulating hard, expensively or non reproducible phenomena. Moreover, in certain cases, it has also been allowed to complete approval process and preparing for homologating in industrial applications.

In CFD (Computational Fluid Dynamics) a great research effort has been devoted to the development of accurate and efficient numerical algorithms suitable for solving flows in the various Reynolds and Mach number regimes. The type of convection scheme to be used in a given application depends on the value of Reynolds number. On the other hand, the Mach number value dictates the type of algorithm to be utilized in the solution procedure. These algorithms can be classified into two groups: density-based methods and pressure-based methods, generally, the former is used for high Mach number flows (M>0.3), and the latter for low Mach number flows (M<0.3).

The pressure-based methods were originally developed to solve incompressible flows, adopting pressure as a primary variable. With this approach, the magnitude of the pressure gradient remains finite, irrespective of Mach number, rendering computation tractable throughout the entire spectrum of Mach number, hence circumventing the shortcomings of density-based methods [1]. The first implementation of pressure-based schemes for compressible flow is widely attributed to the early contribution of Harlow and Amsden based on a semi-implicit finite differ-

ence algorithm [2,3]. Pressure-correction or projection methods are pressure-based fractional-staged schemes with correction for velocity and pressure introduced through the pioneering work of Chorin and Temam [4,5]. Such methods have been employed effectively within several finite volume implementations, through the SIMPLE (Semi-Implicit Pressure Linked Equations) family of schemes for example [6]. Karki and Patankar developed the SIMPLER method for compressible flows, applicable for a wide range of problem speeds [1]. Munz et al. extended the SIMPLE scheme for low Mach number flow employing multiple pressure variables, each being associated with different physical response [7]. Similar procedures have been adopted by others [?8–10]. Pressure-correction was taken forward within finite differences to a second-order by Van Kan [11]. Alternatively, within finite elements, Donea et al. introduced a pressure-correction fractional-step method, designed to significantly reduce computational cost in transient incompressible viscous flow situations [12]. More recently, in the finite element context, Zienkiewicz and co-workers have introduced the characteristic-based-split procedure (CBS) [13]-[16]. This implementation is a Taylor-Galerkin/Pressure-Correction scheme, suitable for both incompressible and compressible flow regimes. The problem here is to split the equation system into two parts: a part of convection-diffusion type (discretized via a characteristic-Galerkin procedure) and one of self-adjoint type. With the CBS-scheme, one may solve both parts of the system in an explicit manner. Alternatively, one may use a semi-implicit scheme for the first part, allowing for much larger time-steps, and solve the second part implicitly, with its advantage of unconditional stability. The CBS procedure has been tested successfully on a number of scenarios, for example, transonic and supersonic flows, low Mach number flows with low and high viscosity and in addition, on shallow-water wave problems [23].

The time-marching density-based schemes are employed widely in computational fluid dynamics for computation of steady and transient transonic, supersonic and hypersonic flows. In the subsonic regime, when the magnitude of the flow-velocity is small, in comparison with the acoustic wave-speed, dominance of convection terms within the time dependent equation system renders the system stiff and solvers converge slowly [17]. The time-marching procedures may suffer severe stability and accuracy restrictions and become inefficient for low Mach-number flow regimes. Here, for explicit schemes, the time-step must satisfy the Courant-Freidrichs-Lewy (CFL) conditions, where numerical stability considerations lead to small time-steps, due to the prevailing acoustic wave-speeds. On the other hand, implicit methods suffer from stiffness due to large disparity in the eigenvalues of the system. There, the condition number is high and eigenvalues may vary by orders of magnitude. As a consequence, the unpreconditioned algebraic system is illconditioned, rendering iterative solutions excessively time consuming. The effect of system stiffness on solution convergence is well known, for both explicit and implicit schemes [23].

2 Numerical Method

Due to the high speed aeronautical applications, the density based method is preferred in the followings. The conservative form of the unsteady compressible Navier-Stokes Equations in 2 spatial dimensions without body forces and internal heat generation in a Cartesian coordinate system is given by (1).

$$\frac{\partial U}{\partial t} + \frac{\partial (F - F_v)}{\partial x} + \frac{\partial (G - G_v)}{\partial y} = \vec{0}, \qquad (1)$$

in which the conservative variables and convective fluxes are given by (2).

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, \quad F(U) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u h^{to} \end{pmatrix},$$

$$G(U) = \begin{pmatrix} \rho v \\ \rho v u \\ \rho v^2 + p \\ \rho v h^{to} \end{pmatrix}. \qquad (2)$$

The diffusive fluxes are described by (3).

$$F_v(U) = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx} + v\tau_{xy} - q_x \end{pmatrix},$$

$$G_v(U) = \begin{pmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ u\tau_{yx} + v\tau_{yy} - q_y \end{pmatrix}. \qquad (3)$$

Concerning the level of modelling, there are different approaches for fluid dynamics. Today's intensively developing CFD methods such as DNS (Direct Numerical Simulation) [18] and LES (Large Eddy Simulation) [19] for example are confined to low Reynolds number from purely computational consideration. They require a significant amount of computational time and memory comparing to the conventional flow solvers. On a physical standpoint, the two-equation closure modelling can be properly employed such as to capture the basic mechanisms related to turbulence and separation. Hence, k-ω turbulence model – originally developed by Wilcox in 1993 – is implemented after the Reynolds and Favre averaging, which are addressed to remove local uncertainty comes from turbulent fluctuation meanwhile the global effect of the turbulence is preserved [21].

The balance of turbulent kinetic energy is given by (4).

$$\bar{\rho}\frac{\partial k}{\partial t} + \bar{\rho}\tilde{u}_j \frac{\partial k}{\partial x_j} = -\overline{\rho u''_i u''_j} \frac{\partial \tilde{u}_i}{\partial x_j} - \beta^* \bar{\rho} k\omega +$$
$$\frac{\partial}{\partial x_j}\left[\left(\mu + \sigma^* \mu_t \right) \frac{\partial k}{\partial x_j} \right] \qquad (4)$$

The transport equation of the specific dissipation rate of turbu-

lent kinetic energy is described by (5).

$$\bar{\rho}\frac{\partial \omega}{\partial t} + \bar{\rho}\tilde{u}_j\frac{\partial \omega}{\partial x_j} = -\overline{\rho u''_i u''_j}\alpha\frac{\omega}{k}\frac{\partial \tilde{u}_i}{\partial x_j} - \beta\bar{\rho}\omega^2 +$$
$$\frac{\partial}{\partial x_j}\left[(\mu + \sigma\mu_t)\frac{\partial \omega}{\partial x_j}\right] \qquad (5)$$

The closure forms, empirical expressions, correlations and other related formulas are given by (6)-(19).

$$\mu_t = \bar{\rho}\frac{k}{\omega}, \qquad (6)$$

$$\alpha = \frac{13}{25}, \quad \sigma^* = \frac{1}{2}, \quad \sigma = \frac{1}{2}, \qquad (7)$$

$$\beta^* = \beta_0^* f_{\beta^*}\left[1 + \xi^* F(M_t)\right], \qquad (8)$$

$$\beta = \beta_0 f_\beta - \beta_0^* f_{\beta^*}\xi^* F(M_t), \qquad (9)$$

$$\beta_0^* = \frac{9}{100}, \quad \beta_0 = \frac{9}{125}, \qquad (10)$$

$$f_{\beta^*} = \begin{cases} 1 & if \quad \chi_k \leq 0 \\ \frac{1+680\chi_k^2}{1+400\chi_k^2} & if \quad \chi_k > 0 \end{cases}, \quad \chi_k = \frac{1}{\omega^3}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, \qquad (11)$$

$$f_\beta = \frac{1+70\chi_\omega}{1+80\chi_\omega}, \quad \chi_\omega = \left|\frac{\Omega_{ij}\Omega_{jk}S_{ki}}{(\beta_0^*\omega)^3}\right|, \quad \chi_\omega = 0, \qquad (12)$$

$$\xi^* = \frac{3}{2}, \quad M_{t0} = \frac{1}{4}, \qquad (13)$$

$$F(M_t) = \left[M_t^2 - M_{t0}^2\right]\mathrm{H}(M_t - M_{t0}), \qquad (14)$$

in which

$$\mathrm{H}(x) = \begin{cases} 0 & if \quad x \leq 0 \\ 1 & if \quad x > 0 \end{cases} \qquad (15)$$

$$M_t^2 = \frac{2k}{a^2}, \qquad (16)$$

where 'a' is the sound speed,

$$\varepsilon = \beta^*\omega k, \qquad (17)$$

$$l = \frac{k^{1/2}}{\omega}, \qquad (18)$$

$$S_{ij} = \frac{1}{2}\left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i}\right) \text{ and } \Omega_{ij} = \frac{1}{2}\left(\frac{\partial \tilde{u}_i}{\partial x_j} - \frac{\partial \tilde{u}_j}{\partial x_i}\right). \qquad (19)$$

Although mathematical rigour and elegance of the FEM (Finite Element Method) will remain responsible for a larger growth with respect to both FDM (Finite Difference Method) and FVM (Finite Volume Method), the finite volume discretization is applied in this paper, because it tries to combine the geometrical flexibilities of the finite element method with the discretization flexibility of the finite difference method. In order to pass from a continuous to a discrete form, the unknown in a general finite volume of the partitioned computational domain is defined as:

$$U_j = \frac{1}{\Omega_j}\int_\Omega\int U d\Omega. \qquad (20)$$

Roe's approximate Riemann solver is used to evaluate numerical flux functions (21) at the cell interfaces [22]. The Godunov based procedure is highly non-dissipative and more closely linked to the concept of characteristic transport. It is one of the most powerful linear Riemann solvers due to the particular advantage that it recognizes shock waves and transports all the characteristics precisely.

$$H_n^{L,R} = \frac{1}{2}\left\{H_n^L + H_n^R - \sum_{i=1}^{N_f}\left|\hat{\lambda}_n^{(i)}\right|\left|\hat{r}_n^{(i)}\right|\Delta W_n^{(i)}\right\}. \qquad (21)$$

MUSCL (Monotone Upstream Schemes for Conservation Laws) approach is used for higher order spatial extension and MinMod limiter for monotonicity preserving:

$$\overline{U}_{i\pm\frac{1}{2}}^{L,R} = U_i \pm \frac{1}{4}\left[(1 \pm \kappa)\bar{\Delta}_{i+\frac{1}{2}} + (1 \mp \kappa)\bar{\Delta}_{i-\frac{1}{2}}\right], \qquad (22)$$

where $\kappa = 1/3$, $\bar{\Delta}_{i\pm\frac{1}{2}} = MinMod\left(\Delta_{i\pm\frac{1}{2}}, \beta\Delta_{i\mp\frac{1}{2}}\right)$, $1 \leq \beta \leq \frac{3-\kappa}{1-\kappa}$ and $\Delta_{i+\frac{1}{2}} = U_{i+1} - U_i$, or $\Delta_{i-\frac{1}{2}} = U_i - U_{i-1}$. Simple central discretization is used for diffusive terms.

The final form of the discretized equations – in which the turbulence related terms are included – are given by (20).

$$\frac{\partial}{\partial t}[U]_{ij} = -\frac{1}{\Omega_{ij}}\left(\sum_{k=1}^{4}[H_n - H_{\upsilon n}]_{ij,k}\,\Gamma_{ij,k}\right) + [S]_{ij} = \mathfrak{R}. \qquad (23)$$

The 4th order Runge-Kutta method is used for time marching procedure: $U_{ij}^0 = U_{ij}^n$, $U_{ij}^k = U_{ij}^0 + \alpha_k\Delta t\mathfrak{R}_{ij}^{k-1}$, $U_{ij}^{n+1} = U_{ij}^k$, where $k = 1, 2, 3, 4$ and $\alpha_k = \frac{1}{4 - k + 1}$.
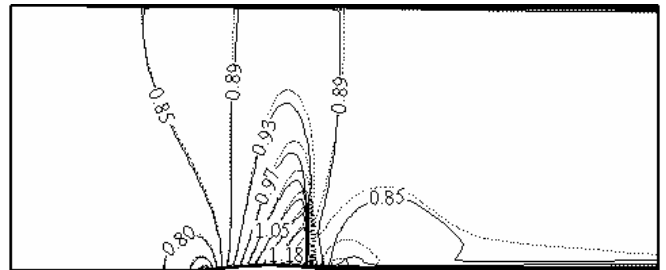

Fig. 1. Mach number distribution in transonic channel over circular bump test case (dotted line: FLUENT, continuous line: recent solver)

3 Validation

In the following sections numerical results are presented for the test problem of transonic channel over circular bump and compression corner for validation.
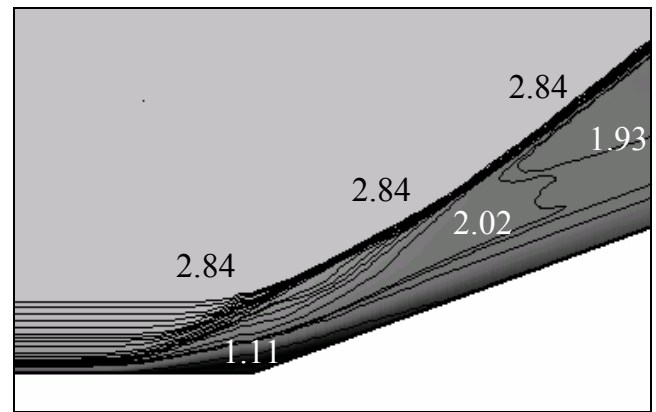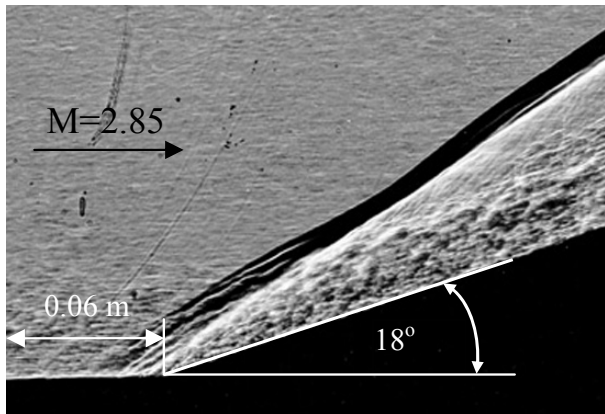
Fig. 2. Configuration and Schlieren photograph about compression corner at the inlet Mach number 2.85 (left side) [20] and Mach number distribution by calculation (right side)

The first test case is transonic channel over circular bump, in which the flow enters into the channel with Mach number 0.85 and a shockwave develops over the circular bump. The bump has 4.2 % maximum thickness. In the inlet plane the total pressure, total temperature, and flow angles are specified as physical boundary conditions. The static pressure corresponding an isentropic flow at Mach=0.85 is imposed at the outlet. Under these conditions the flow expands in the rear part of the bump up to a Mach number of about 1.2 and ends up into a week shock wave to allow the recovery of the free stream conditions. The results of FLUENT and own code are compared to each other. The Mach number iso-lines show reasonable deflections in Fig. 1, the present method predicts the shockwave earlier. The shape of the geometry and the thickness of the boundary layer have a dominant effect on the location of the shockwave. Different numerical methods have different inherent mechanism to model boundary layer and shock wave – boundary layer interaction. The boundary layer seems to be thinner at the downstream of the circular bump in case of the commercial code compared to the own one. Hence, the shockwave triggered earlier in case of our model. In the second test case a ramp with 18 degrees slope angle is located in the flow channel. The air enters into a channel with Mach number 2.85. Before the ramp an oblique shockwave develops. The geometry and Mach number distribution can be found in Fig. 2. The numerical results of the solver and the Schlieren photograph [20] are compared with each other without significant deviation. In the reality, the high intensity shock wave develops at the same position and with the same form as in the result of computation at the identical boundary conditions.

4 Conclusions

A new density-based viscous flow solver is developed and tested for compressible flow in C++ environment to be a base of the complex inverse design tool. The numerical code is based on structured, cell centered finite volume method. The governing equations in conservative form are deduced by using Favre averaging to filter local fluctuations. The convective terms are discretized by Roe approximated Riemann method. Central dis-

cretization is applied for diffusive terms. MUSCL approach is implemented for higher order spatial reconstruction with Mulder limiter for monotonicity preserving. Wilcox k-ω two equations turbulence model is used for turbulence modelling. The system of equation is solved by time accurate 4th order Runge-Kutta method. The validation of the code has been completed for circular bump in the transonic channel and compression corner.

References

1 **Karki K, Patankar S V**, *Pressure based calculation procedure for viscous flows at all speeds in arbitrary configurations*, AIAA journal **27** (1989), no. 9, 1167-1174.

2 **Harlow F H, Amsden A**, *Numerical calculation of almost incompressible flow*, Journal of Computational Physics **3** (1968), 80-93.

3 ———, *A numerical fluid dynamics calculation method for all flow speeds*, Journal of Computational Physics **8** (1971), 197-213.

4 **Chorin A J**, *Numerical solution of the Navier-Stokes equations*, Mathematics of Computation **22** (1968), 745-762.

5 **Temam R**, *Sur l'approximation de la solution de Navier-Stokes par la méthode des pas fractionnaires*, Archiv. Ration. Mech. Anal. **32** (1969), 377-385.

6 **Patankar S V**, *Numerical heat transfer and fluid flow*, McGraw-Hill, New York, 1980.

7 **Munz C D, Roller S, Klein R, Geratz K J**, *The extension of incompressible flow solvers to the weakly compressible regime*, Computer & Fluids **32** (2003), no. 2, 173-196.

8 **Bijl H, Wesseling P**, *A unified method for computing incompressible and compressible flows in boundary-fitted coordinates*, Journal of Computational Physics **141** (1998), no. 2, 153-173.

9 **Mary I, Sagaut P, Deville M**, *An algorithm for low Mach number unsteady flows*, Computers & Fluids **29** (2000), no. 2, 119-147.

10 **Roller S, Munz C D**, *A low Mach number scheme based on multi-scale asymptotics*, Computing and Visualization in Science **3** (2000), no. 1/2, 85-91.

11 **Van Kan J**, *A second-order accurate pressure-correction scheme for viscous incompressible flow*, SIAM Journal of Scientific and Statistical Computing **7** (1986), 870-891.

12 **Donea J**, *A Taylor-Galerkin method for convective transport problems*, International Journal for Numerical Methods in Engineering **20** (1984), 101-119.

Tab. 1. List of Symbols and Abbreviations

Variables (Latin)	**Variables (Greek)**
$c[m/s]$ absolute velocity	$\alpha_k[-]$ const. for the RK time it.
$C[m]$ camber length	$\alpha, \beta_0, \beta_0^*, \xi^*, \sigma, \sigma^*$ [-] Wilcox model constants
$E[J/kg]$ total energy per unit mass	δ $[m]$ thickness of the boundary layer
$F_B[N/m]$ blade force	Δ [-] difference
F, G[-] inviscid flux vectors	$\lambda_n^{(i)}$[-] eigenvalues
F_v, G_v[-] viscous flux vectors	μ [Ns/m^2] molecular viscosity
$h^{to}[J/kg]$ total enthalpy per unit mass	$\mu_t[Ns/m^2]$ turbulent viscosity
H_n[-] normal flux function	ρ [kg/m^3] density
i, j[-] spatial variables for indexing	$\tau_{ij}[N/m^2]$ stress tensor components
$k[m^2/s^2]$ turb. kinetic energy per unit mass	ω [$1/s$] specific dissipation rate
$L[m]$ length	Ω [m^2] area of the finite volume
n[-] normal to the surface	**Subscripts and Superscripts**
$n_x, n_y[m]$ components of normal vector	‾ vector, time average
$p[Pa]$ static pressure	~ Favre (mass) average
$p^*[Pa]$ static pressure, turbulent kinetic energy is included	ˆ Roe average state space
$q_i[J/m^2/s]$ components of heat flux vector	1,2 inlet and outlet
$r_n^{(i)}$[-] right normal eigenvectors	n variables normal to the surface
$R, \quad \Re$ [$J/kg/K$] specific gas constant, residual [-]	T turbulent
S[-] vector of source term	**Abbreviations**
$t[s, -]$ time, unit normal direction to n	DNS : Direct Numerical Simulation
$u[m/s]$ tangential velocity	L : left
U[-] vector of conservative variables, free stream velocity	LES : Large Eddy Simulation
$u, v[m/s]$ components of velocity vector	N_f : number of faces
$V_n[m/s]$ velocity normal to the surface	R : right
$w[m/s]$ relative velocity	RANS : Reynolds Averaged Navier-Stokes
W[-] vector of characteristic variables	RK : Runge-Kutta
$x, y[m]$ Cartesian coordinates in space	

13 **Zienkiewicz O. C, Codina R**, *A general algorithm for compressible and incompressible flow - Part I. The split characteristic-based scheme*, International Journal for Numerical Methods in Fluids **20** (1995), 887-913.

14 **Zienkiewicz O. C, Morgan K, Sataya Sal B. V. K, Vasquez M**, *A general algorithm for compressible and incompressible flow – Part II . Test on the explicit form*, International Journal for Numerical Methods in Fluids **20** (1995), 887-913.

15 **Zienkiewicz O. C, Nithiarasu P, Codina R, Vazquez M, Ortiz P**, *The characteristic-based-split procedure: An efficient and accurate algorithm for fluid problems*, International Journal for Numerical Methods in Fluids **31** (1999), 359.

16 **Zienkiewicz O. C, Taylor R L**, *The finite element method, 3: fluid dynamics*, Butterworth and Heinemann (2000).

17 **Choi Y. H, Merkle C L**, *The application of preconditioning in viscous flows*, Journal of Computational Physics **105** (1993), no. 2, 207-223.

18 **Orszag S. A, Patterson G S**, *Numerical Simulation of Three-Dimensional Homogeneous Isotropic Turbulence*, Phys. Rev. Lett. **28** (1972), 76-79.

19 **You D, Moin P**, *A Dynamic Global-Coefficient Subgrid-Scale Eddy-Viscosity Model for Large-Eddy Simulation in Complex Geometries*, Physics of Fluids **19** (2007), no. 6, 065110, DOI 10.1063/1.2739419.

20 **Gary Settles**, Princeton, 1975, `http://www.efluids.com/efluids/gallery/gallery_pages/18degramp.htm`. Gas Dynamics Lab, Penn State University.

21 **Wilcox David C**, *Turbulence Modelling for CFD*, DCW Industries Inc, 1998.

22 **Roe P. L**, *Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes*, Journal of Computational Physics **43** (1981), 357-372.

23 **Keshtiban I.J, Belblidia F, Webster M F**, *Compressible flow solvers for low Mach number flows*, 2004, available at `http://cs.swan.ac.uk/reports/yr2004/CSR2-2004.pdf`.