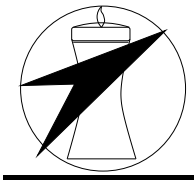
ARDB Centre of Excellence for Aerospace CFD
Department of Aerospace Engineering
Indian Institute of Science

Fluid Mechanics Report

Kinetic Meshless Method

Praveen C and S. M. Deshpande
ARDB Centre of Excellence for Aerospace CFD
Department of Aerospace Engineering
Indian Institute of Science

Report 2003-FM-10



September 2003
Bangalore 560 012

Fluid Mechanics Report

Kinetic Meshless Method

Praveen C and SM Deshpande
CFD Center, Department of Aerospace Engg.,
Indian Institute of Science, Bangalore 560012
Email: (praveen,suresh)@aero.iisc.ernet.in

Abstract

We propose a new method called the Kinetic Meshless Method (KMM), for the solution of time-dependent conservation laws that can be derived from a Boltzmann-type equation by taking suitable moments. An important characteristic of this method is its *grid-free* nature; it is able to use any type of grid or combination of grids, or even some distribution of nodes. All that the method requires is the specification of the *connectivity*, which consists of a *cloud* of neighbouring points around each point. For hyperbolic conservation laws, the method is stabilized by introducing an upwind-bias using the kinetic representation.

A numerical order of accuracy of two is shown for a scalar conservation law even though the formal order of the method is one. Numerical second order accuracy is obained even on a highly non-uniform grid.

We have applied the above method for the solution of the Euler equations by using the Maxwell-Boltzmann distribution function. In this case, it has been previously called KRIME, Kinetic Rotationally Invariant Method for Euler equations. We present a number of results on point distributions obtained from unstructured grids. Results are obtained on adapted grids and demonstrate the ability of the method to capture discontinuities sharply.

Keywords: Conservation laws, Euler equations, kinetic method, meshless approximation, upwind scheme.

1 Introduction

The basic ingredient in all grid-free methods is the recovery of a function and/or its derivatives from arbitrarily scattered data. One of the earliest works is that of Liszka and Orkisz [22, 23] who used a *least squares approximation* to evaluate derivatives from arbitrarily scattered data. They pointed out many applications:

- Extraction of derivatives of FEM/FDM solution of boundary value problems.
- Interpolation of the solution from one FE/FD mesh to another.
- Construction of shape functions used in FEM (like in the DEM and EFGM).
- Interpolation of experimental data.

These are known as generalised finite difference formulae since they contain the usual finite difference formulae on regular grids as special cases. Liszka et al. [23] applied the least squares approximation to solve some elliptic and parabolic problems. Lancaster and Salkauskas [21] proposed the *moving least squares* (MLS) approximation for data fitting which involves approximating an unknown function using some set of basis functions. This was used by Nayroles et. al. [27] in the Diffuse Element Method (DEM) and by Belytschko et. al. [9] in their Element Free Galerkin Method (EFGM). These methods have been applied in fracture mechanics to study crack growth and propagation. The grid has to track the crack propagation and adapt to resolve the crack geometry, and this is easily achieved due to the grid-free nature of the methods. The MLS approximation reduces to the least squares approximation of Liszka under some conditions. Duarte and Oden, and Babuska and Melenk have advocated the *partition of unity* methods which generalise the MLS approximation. Duarte and Oden call their method as the *hp*-cloud method. This has been recently modified and applied to the solution of the incompressible Navier-Stokes equations by Baumann et. al. [8]. A review of these methods is contained in Belytschko [10].

All the above methods have been used for the solution of elliptic or parabolic equations, while applications in CFD involving hyperbolic equations are very few. Ghosh and Deshpande [15, 16] used the least squares approach and the CIR splitting technique for the Boltzmann equation, to develop an upwind kinetic scheme called LSKUM (Least Squares Kinetic Upwind Method) for the solution of Euler equations. The LSKUM has been applied to many problems in gas dynamics with different types of grids and combination of grids [31]. Ghosh and Deshpande [15, 16] have also proposed a novel way of achieving second order accuracy in the least squares approximation which does not require calculation of the higher derivatives;

this leads to a Pade type approximation with a smaller support than is normally required for second order accuracy. Batina [7] has also proposed a grid-free method for Euler equations using artificial dissipation and used it to solve flow about a complete aircraft configuration. Morinishi [25, 26] used a method similar to the MLS approximation but with a non-compact weight function, together with artificial dissipation for solving Navier-Stokes equations, and has applied it to a variety of point distributions and for multi-element airfoils. Balakrishnan [3, 4, 32] has proposed a family of upwind meshless schemes known as LSFD-U, which again use least squares in combination with any flux-vector or flux-difference splitting technique. Note that the methods of Ghosh and Deshpande, Batina, Morinishi, and Balakrishnan can be considered as generalized finite difference methods. Yagawa et al. [38, 39] have developed the Free Mesh Method (FMM) which uses a local Delaunay triangulation for a finite element solution. They have applied it to heat conduction, stress and fracture analyses, and incompressible flows, and are now tackling compressible flows with the SUPG finite element method.

There is also the class of particle and Lagrangian methods for fluid problems, which includes Smooth Particle Hydrodynamics, Reproducing Kernel Particle Methods, Finite Mass Method, Finite Volume-Particle Method [20], etc. Some of these methods have similarities with the MLS approximation and are reviewed in Belytschko [10].

Another line of attack that is being pursued by many people is the use of radial basis functions for interpolation and recovery of functions from arbitrarily scattered data. It involves a Lagrange interpolation using a basis of radial functions and/or polynomials, and does not require a least squares approximation. Rigorous error and regularity results have been derived and they have been shown to lead to optimal recovery in certain function spaces. Sonar [34] has used radial basis functions for ENO-type reconstruction in finite volume methods and has shown better performance compared to pure polynomial reconstruction. Wendland [37] has used radial basis functions in a meshless finite element formulation and has derived some error estimates.

The least squares approximation has also been used by many researchers in the reconstruction step in high-order finite volume methods, see Barth [6]. An interesting development in this area is the use of data-dependent weights which avoids the need to use a limiter, and provides a more rational basis for the least squares approximation, see Gooch [18].

The kinetic meshless method presented here was inspired by the work of Balakrishnan [3, 4, 32] and has some features similar to LSFD-U. It is the outcome of combining the use of Boltzmann equation like in LSKUM, with the least squares approximation used in LSFD-U based on mid-point data. The new least squares approximation as used here and by Balakrishnan are slightly different from the usual approximation, and will be refered to as the *dual least squares approximation*. The advantage of the dual least squares approximation is that it allows the introduction of upwinding on an unsplit stencil, which is required to derive a stable scheme for hyperbolic equations.

We start with a discription of the kinetic representation for conservation laws and the kinetic scheme using finite differences. This is generalized to arbitrary grids using least squares giving rise to LSKUM. We then explain the dual least squares approximation, using which we derive the KMM. The wall boundary conditions which are based on the strong formulation are discussed next. The numerical order of accuracy of the scheme is shown by solving a scalar conservation law. Finally some results of compressible 2-D flows are presented.

2 Kinetic representation

The kinetic meshless method can be applied to any system of conservation laws which have a kinetic representation. By this we mean that the system of conservation laws can be obtained by taking suitable moments of a Boltzmann-type scalar equation. The conservation law in the divergence form is[1]

$$\frac{\partial U}{\partial t} + \sum_{\alpha=1}^{d} \frac{\partial F^\alpha}{\partial x^\alpha} = 0 \tag{1}$$

where $U \in \mathbb{U}_{ad} \subset \mathbb{R}^m$ is the vector of conserved variables, $\mathbb{U}_{ad}$ being the set of admissible values of U, and $F^\alpha : \mathbb{U}_{ad} \to \mathbb{R}^m$, $1 \leq \alpha \leq d$, are the Cartesian components of the flux. The above system of equations can be obtained by taking moments of the Boltzmann equation,

$$\frac{\partial f}{\partial t} + \sum_{\alpha=1}^{d} v_\alpha \frac{\partial f}{\partial x^\alpha} = J(f) \tag{2}$$

[1]The position of the Greek sub/super-scripts is not intended to convey the tensorial nature of the quantities but is adopted for notational convenience.

The moments are defined as,

$$\langle \cdot \rangle = \int_{\mathbb{R}^d \times \mathbb{R}^+} (\cdot) \psi \mathrm{d}\vec{v} \mathrm{d}I \tag{3}$$

where $\psi(\vec{v}, I) \in \mathbb{R}^m$ are the collisional-invariants of the Boltzmann equation and I is a variable introduced to take account of non-translational degrees of freedom [12]. The additional conditions which relate (2) to (1) are,

$$U = \langle f \rangle, \quad F^\alpha = \langle v_\alpha f \rangle \tag{4}$$

together with the compatibility condition,

$$\langle J(f) \rangle = 0 \tag{5}$$

The last equation in fact defines the collisional invariants. For linear hyperbolic equations, a kinetic representation can be easily written. In the non-linear case, a kinetic representation exists if the fluxes can be written as $F^\alpha(U) = A^\alpha(U)U$, where the matrices $A^\alpha(U)$ have only real eigenvalues and a complete set of eigenvectors. Though this condition seems to be restrictive, Harten, Lax, van Leer [17] have shown that such a representation of the flux is possible whenever the system (1) admits a convex entropy.

In the case of Euler equations the physically relevant distribution function is the Maxwell-Boltzmann distribution function [12], given by

$$f(\vec{v}, I; \varrho, \vec{u}, \beta) = \frac{\varrho}{I_o} \left(\frac{\beta}{\pi} \right)^{d/2} \exp\left(-\beta |\vec{v} - \vec{u}|^2 - \frac{I}{I_o} \right) \tag{6}$$

and the set of collisional invariants is given by

$$\psi(\vec{v}, I) = \begin{bmatrix} 1 \\ \vec{v} \\ I + \frac{1}{2}|\vec{v}|^2 \end{bmatrix} \tag{7}$$

where $\beta = (2RT)^{-1}$ and $I_o = RT(2 + d - \gamma d)/2(\gamma - 1)$.

It is also possible to use other distribution functions like that of Kaniel or of Sanders and Prendergast, which might lead to simpler flux formulae.

3 Kinetic scheme

A kinetic scheme for a conservation law that has a kinetic representation is derived in the following way: Since the Boltzmann equation is a scalar convection equation,

its characteristics and hence its upwind directions are clearly known. We first derive an upwind scheme for the Boltzmann equation and then take its moments to obtain a discretization of the conservation law. This is known as moment-method strategy. We explain the kinetic scheme in one dimension, the equations being given by (1), (2) by retaining only the x-components. The exact solution of the Boltzmann equation without the collision term is

$$f(v; x, t + \Delta t) = f(v; x - v\Delta t, t)$$

i.e., the solution is simply advected with speed $|v|$ and in a direction that depends on the sign of v. We first set

$$v^+ = \frac{v + |v|}{2}, \quad v^- = \frac{v - |v|}{2}$$

and write equation (2) in CIR-split form,

$$\frac{\partial f}{\partial t} + v^+ \frac{\partial f}{\partial x} + v^- \frac{\partial f}{\partial x} = 0 \tag{8}$$

where we have neglected the collision term $J(f)$. Since $v^+ \geq 0$ and $v^- \leq 0$, we write the difference equation for (8) as,

$$\frac{\mathrm{d}f_i}{\mathrm{d}t} + v^+ \frac{f_i - f_{i-1}}{\Delta x} + v^- \frac{f_{i+1} - f_i}{\Delta x} = 0$$

which is the standard upwind scheme for the convection equation. Taking moments of this discrete equation, we get an update equation for the Euler system.

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} + \frac{F_i^+ - F_{i-1}^+}{\Delta x} + \frac{F_{i+1}^- - F_i^-}{\Delta x} = 0$$

where we have set

$$F^\pm = \langle v^\pm f \rangle \tag{9}$$

This is essentially a flux vector split scheme, with the split fluxes being defined by (9). Kinetic schemes are very robust at all Mach numbers because they are entropy consistent and satisfy positivity of pressure and density under a CFL-like condition.

4 Least squares approximation

We next explain the least squares method for determining the derivative of a smooth function from arbitrarily scattered data. Let $f : \mathbb{R} \to \mathbb{R}$ be a sufficiently smooth

function whose values are known at some arbitrarily spaced locations $\{x_i\}$. For each point i we define a set of points $\mathcal{C}_i$ as its connectivity. Using Taylor's formula, we can write,

$$f_j = f_i + \Delta x_{ij} \left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)_i + O(\Delta x_{ij}^2), \quad \Delta x_{ij} = x_j - x_i$$

In the least squares method, the derivative is determined by solving a weighted minimization problem:

$$\text{minimize} \sum_{j \in \mathcal{C}_i} w_{ij} \left[f_j - f_i - \Delta x_{ij} \left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)_i \right]^2, \quad \text{w.r.t.} \quad \left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)_i \tag{10}$$

The resulting formula is first order accurate in general but second order accurate on a symmetric centered connectivity. In the above, the $w_{ij} = w(|\vec{x}_j - \vec{x}_i|)$ are positive weight functions and a standard choice is $w(r) = r^{-p}$ with $p \geq 0$. We have used this weight function with $p = 2$ in all the computations presented here. A better choice might be to use an exponential weight function as in SPH or cubic/quartic polynomials used in EFGM, which remain bounded even as $r \to 0$.

5 Kinetic scheme on arbitrary grids: LSKUM

The LSKUM [15, 16] of Ghosh is obtained by generalising the kinetic scheme to arbitrary grids using the least squares approximation. Starting with equation (8) we now replace the two spatial derivatives with least squares approximations.

$$\frac{\partial f}{\partial t} + v^+ \left(\frac{\partial f}{\partial x}\right)_{\mathcal{C}_i^+} + v^- \left(\frac{\partial f}{\partial x}\right)_{\mathcal{C}_i^-} = 0 \tag{11}$$

where

$$\mathcal{C}_i^+ = \{j \in \mathcal{C}_i : x_j < x_i\}, \quad \mathcal{C}_i^- = \{j \in \mathcal{C}_i : x_j > x_i\}$$

and the subscripts indicate the stencil to be used in the least squares procedure. Note that this corresponds to using generalised backward and forward difference formula instead of standard finite difference formulae. Following the moment method strategy, we take moments of (11) to obtain an upwind discretization of the Euler equations. In 2-D we need four sub-sets of connectivity for each node i:

$$\mathcal{C}_i^{x+} = \{j \in \mathcal{C}_i : x_j < x_i\}, \quad \mathcal{C}_i^{x-} = \{j \in \mathcal{C}_i : x_j > x_i\}$$

$$\mathcal{C}_i^{y+} = \{j \in \mathcal{C}_i : y_j < y_i\}, \quad \mathcal{C}_i^{y-} = \{j \in \mathcal{C}_i : y_j > y_i\}$$

The Boltzmann equation is discretized as

$$\frac{\partial f}{\partial t} + v_x^+ \left(\frac{\partial f}{\partial x}\right)_{\mathcal{C}_i^{x+}} + v_x^- \left(\frac{\partial f}{\partial x}\right)_{\mathcal{C}_i^{x-}} + v_y^+ \left(\frac{\partial f}{\partial y}\right)_{\mathcal{C}_i^{y+}} + v_y^- \left(\frac{\partial f}{\partial y}\right)_{\mathcal{C}_i^{y-}} = 0 \qquad (12)$$

which is a generalised version of the CIR-split scheme in 2-D. Note that this is a fully upwind method and has been found to be very robust at all Mach numbers. In 1-D it has been shown to be positivity preserving under a CFL-like condition and numerical experience indicates that this property holds in higher dimensions also.

6 Dual least squares approximation

A variant of the least squares approximation, which will be called the dual least squares approximation, has been used by Morinishi, and Balakrishnan. We explain it in an abitrary dimension d. Let $f : \mathbb{R}^d \to \mathbb{R}$ be a smooth function. If $j \in \mathcal{C}_i$ then let f_{ij} denote the value of f at the midpoint of the line segment $\vec{ij}$. Using Taylor's formula

$$f_{ij} = f_i + \frac{1}{2}\Delta\vec{x}_{ij} \cdot \nabla f_i + O(|\Delta\vec{x}_{ij}|^2)$$

The dual least squares approximation is obtained from the following weighted minimization:

$$\text{minimize} \sum_{j\in\mathcal{C}_i} w_{ij} \left[f_{ij} - f_i - \frac{1}{2}\Delta\vec{x}_{ij} \cdot \nabla f_i \right]^2, \quad \text{w.r.t. } \nabla f_i \qquad (13)$$

This gives a system of $d \times d$ equations

$$\mathbb{A}_i \nabla f_i = \mathrm{b}_i$$

where

$$\mathbb{A}_i = \frac{1}{4}\left[\sum_{j\in\mathcal{C}_i} w_{ij}\Delta x_{ij}^\alpha \Delta x_{ij}^\beta\right]_{1\le\alpha,\beta\le d}$$

$$\mathrm{b}_i = \frac{1}{2}\left[\sum_{j\in\mathcal{C}_i} w_{ij}\Delta x_{ij}^\alpha (f_{ij} - f_i)\right]_{1\le\alpha\le d}$$

The matrix $\mathbb{A}$ is symmetric and positive definite as long as the connectivity is non-degenerate; hence a unique solution always exists. The solution of this linear system can be written as

$$\text{Dual Least Squares:} \quad \left.\frac{\partial f}{\partial x^\alpha}\right|_i = \sum_{j\in\mathcal{C}_i} \mu_{ij}^\alpha (f_{ij} - f_i) \qquad (14)$$

Note that since we have neglected terms of $O(|\Delta\vec{x}_{ij}|^2)$, we must use atleast a second order estimate [4] of f_{ij}, i.e., $f_{ij} = f\left(\frac{\vec{x}_i + \vec{x}_j}{2}\right) + O(|\vec{x}_i - \vec{x}_j|^2)$. We also mention the least squares approximation of the derivative,

$$\text{Least Squares: } \left.\frac{\partial f}{\partial x^\alpha}\right|_i = \frac{1}{2}\sum_{j\in\mathcal{C}_i} \mu_{ij}^\alpha (f_j - f_i) \tag{15}$$

The fact that there is a factor of $1/2$ in the above formula shows that the least squares approximation satisfies the following property:

$$\bar{x}^\alpha = 2x^\alpha \Longrightarrow \frac{\partial f}{\partial \bar{x}^\alpha} = \frac{1}{2}\frac{\partial f}{\partial x^\alpha}$$

Theorem 1 *The formulae (14) and (15) are first order accurate estimates of the derivative.*

7 Kinetic mesh-less method

The KMM is built on the dual least squares approximation described in the last section and the moment-method strategy. In the LSKUM which uses the least squares approximation, all the nodal values of f are given and the only way of introducing upwinding is by stencil sub-division. On the contrary, in the dual least squares, the value of f_{ij} is not known a priori, and we can introduce upwinding into the scheme by defining f_{ij} in a consistent and upwind manner. Substituting the dual least squares estimate given by equation (14) into the Boltzmann equation, we obtain

$$\frac{\mathrm{d}f_i}{\mathrm{d}t} + \sum_\alpha \sum_{j\in\mathcal{C}_i} \mu_{ij}^\alpha (v_\alpha f_{ij} - v_\alpha f_i) = 0 \tag{16}$$

Taking moments we obtain a semi-discrete form of the conservation law

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} + \sum_\alpha \sum_{j\in\mathcal{C}_i} \mu_{ij}^\alpha (F_{ij}^\alpha - F_i^\alpha) = 0 \tag{17}$$

where

$$F_{ij}^\alpha = \int_{\mathbb{R}^d\times\mathbb{R}^+} \psi v_\alpha f_{ij} \mathrm{d}\vec{v}\mathrm{d}I \tag{18}$$

In order to upwind-bias the approximation given by equation (17), (18), we construct f_{ij} from the particle distribution functions of nodes i and j, depending on whether the particle velocity has a component towards node i or towards node j. Let V

be a complete set of macroscopic variables which uniquely define[2] f. Then the upwind-biased approximation of f_{ij} is given by

$$f_{ij}(\vec{v}, I) = \begin{cases} f(\vec{v}, I; V_{ij}^{+}), & \vec{v} \cdot \hat{e}_{ij} \geq 0 \\ f(\vec{v}, I; V_{ij}^{-}), & \vec{v} \cdot \hat{e}_{ij} \leq 0 \end{cases} \tag{19}$$

where $\hat{e}_{ij} = (\cos\theta_{ij}, \sin\theta_{ij})$ is the unit vector which is directed from node i to node j and θ_{ij} is the angle made by $\vec{ij}$ with the x-axis. The $V_{ij}^{\pm}$ must be atleast second-order accurate interpolants [4] of V from $\vec{x}_i$, $\vec{x}_j$ to the midpoint of $\vec{ij}$. This is achieved by using linear interpolation

$$V_{ij}^{+} = R_{ij}(V_i + \frac{1}{2}\Delta\vec{x}_{ij} \cdot \nabla V_i), \quad V_{ij}^{-} = R_{ij}(V_j - \frac{1}{2}\Delta\vec{x}_{ij} \cdot \nabla V_j) \tag{20}$$

The matrix R_{ij} effects a rotation of the coordinate system such that the x^1-axis becomes parallel to $\hat{e}_{ij}$ upon rotation. This transformation makes it easier to use equation (20) in (19). The derivatives of V are calculated using the least squares approximation given by equation (15).

Remarks: An essential difference between KMM and other least squares based methods is that in KMM we apply the dual least squares approximation at the Boltzmann level so that we have to handle only one scalar function, the distribution function. On the other hand, Balakrishnan and Morinishi both apply the dual least squares at the level of the conservation laws so that they deal with flux vectors or tensors.

Theorem 2 *The scheme given by (17), (18), (19), (20) is rotationally invariant, i.e., if U^{n+1}, $\bar{U}^{n+1}$ are the updated values in two Cartesian coordinate frames, then they are related by*

$$\bar{U}^{n+1} = RU^{n+1}$$

where R is the coordinate transformation operator.

Remarks: A necessary condition for the above theorem to hold is that the distribution function should be a true scalar, i.e., it should be coordinate invariant. The Maxwell-Boltzmann distribution satisfies this condition since it depends on the velocity only in the form $|\vec{v}-\vec{u}|$. The proof of the theorem makes use of the transformation properties of the first order least squares approximation [29, 30, 13]. The use of minmax or Venkatakrishnan limiter in the presence of discontinuities, will make the interpolation to be coordinate dependent. This has been remedied by adopting

[2] In the case of the Navier-Stokes equations, the distribution function depends on the spatial derivatives of V also.

a MUSCL-type approach which is described below.

Remarks: The property of rotational invariance is important for two reasons. The first is that this property is also shared by the partial differential equations that we are trying to solve. Secondly, it allows us to use stretched grids in boundary layers or near shocks (which could come from anisotropic adaptation) without leading to loss of accuracy, even if the stretching is not aligned with the coordinate axes.

Theorem 3 *The scheme given by (17), (18), (19), (20) has formal first order accuracy.*

Remarks: This result follows very easily because we have retained only terms upto $O(h)$ in the least squares approximation [30]. Inspite of the above theorem, the numerical results show very good accuracy and are comparable with second order results. The use of unsplit stencil twice in each update, for interpolation and also for the flux derivatives, might have a beneficial effect. We give numerical evidence that the method has an effective order of accuracy of two even on a highly random point distribution in two dimensions.

7.1 Higher order formulation

The route to higher order formulation of KMM is straight-forward. To obtain a p-th order scheme, we must retain terms upto $O(h^p)$ in the least squares approximation and use an interpolation which is p-exact for determining $V_{ij}^{\pm}$, i.e., $V_{ij}^{\pm} = V((\vec{x}_i + \vec{x}_j)/2) + O(|\vec{x}_i - \vec{x}_j|^{p+1})$ in smooth regions. If we use a formally second order method, then the least squares procedure leads to a system of five coupled equations in 2D for $\partial_x f$, $\partial_y f$, $\partial_{xx} f$, $\partial_{xy} f$ and $\partial_{yy} f$. This system has to be solved only once and we can store the values of μ that are required in formulae (14) and (15). Note that the structure of the equations (14), (15) remains same even when the order of the least squares approximation is changed except that the values of μ will change. The computer code does not require any modification; only the subroutine for calculating the μ's has to be changed according to the order of the least squares approximation.

7.2 KMM for Euler equations

The Euler equations for compressible flows can be obtained as moments of the Boltzmann equation using the Maxwell-Boltzmann distribution function. In the case of the Euler equations, we can take the primitive variables, $V = [\varrho \;\; u_x \;\; u_y \;\; p]^T$,

for interpolation and the fluxes are given by

$$F_{ij}^x = R_{ij}^{-1}\left[\cos\theta_{ij}F(V_{ij}^+, V_{ij}^-) - \sin\theta_{ij}\tilde{G}(V_{ij}^+, V_{ij}^-)\right]$$

$$F_{ij}^y = R_{ij}^{-1}\left[\sin\theta_{ij}F(V_{ij}^+, V_{ij}^-) + \cos\theta_{ij}\tilde{G}(V_{ij}^+, V_{ij}^-)\right]$$

In the above equations

$$F(V,W) = F^+(V) + F^-(W), \quad \tilde{G}(V,W) = \tilde{G}^+(V) + \tilde{G}^-(W)$$

where $F^\pm = \langle v_x^\pm f\rangle$ and

$$\tilde{G}^\pm(V) = \int_{\mathbb{R}^\pm\times\mathbb{R}\times\mathbb{R}^+} v_y\psi f(\vec{v}, I; V)\mathrm{d}v_x\mathrm{d}v_y\mathrm{d}I \tag{21}$$

Note that these split fluxes are unconventional; they are fluxes in the y-direction but the splitting is with respect to the x-component of the molecular velocity. When the distribution function is a Maxwellian, the expressions for these split fluxes are given in the appendix. We can also show that if $V_{ij}^+ = V_{ij}^- = V_i$, then $F_{ij}^x = F_i^x$ and $F_{ij}^y = F_i^y$. This follows easily from the tranformation properties of the Euler fluxes. This property was used in [30] to prove the rotational invariance of the method.

7.3 Boundary conditions for Euler equations

Wall boundary conditions based on the strong formulation [5] can be easily implemented in a meshless framework. In the strong formulation, the following conditions are enforced at a solid wall point.

$$\begin{aligned}
\vec{u}\cdot\vec{n} &= 0 \\
\frac{\partial p}{\partial n} &= -\varrho\vec{n}\cdot\vec{u}\cdot\nabla\vec{u} \\
\frac{\partial S}{\partial n} &= 0 \\
\frac{\partial H}{\partial n} &= 0
\end{aligned}$$

where $S = p/\varrho^\gamma$ and $H = c^2/(\gamma-1) + u_t^2/2$, and we have neglected the vorticity source term in entropy equation, which is a valid assumption for subsonic and transonic flows. These conditions are enforced using the least squares approximation; if ϕ denotes any of the variables (p, S, H) at a wall boundary point b, then we

determine ϕ_b from

$$\frac{n_b^x}{2} \sum_{j \in \mathcal{C}_b} \mu_{bj}^x (\phi_j - \phi_b) + \frac{n_b^y}{2} \sum_{j \in \mathcal{C}_b} \mu_{bj}^y (\phi_j - \phi_b) = S_\phi$$

An alternative, which we have not tried, is to determine ϕ_b by solving a constrained least squares minimization problem:

$$\text{minimize} \quad \sum_{j \in \mathcal{C}_b} w_{bj} (\phi_j - \phi_b - \Delta \vec{x}_{bj} \cdot \nabla \phi_b)^2 + (\vec{n}_b \cdot \nabla \phi_b - S_\phi)^2, \quad \text{wrt } \phi_b, \nabla \phi_b$$

assuming that ϕ_j, $j \in \mathcal{C}_b$ are known. The solution satisfies the boundary condition only in the limit of the steady solution. A trick that we have successfully used is to update the flow at a solid wall point using the interior scheme and then kill the normal component of the velocity. This has been used in the case of supersonic flow over a cylinder and the results are presented here. A variant of this which works well for subsonic and transonic flows is to correct the updated pressure and density by solving the normal momentum and normal entropy equations listed above.

For subsonic and transonic flows the solution on the outer boundary is updated by using the point vortex model of Thomas and Salas [35].

8 Interpolation schemes

Interpolation of some set of variables is required to determine the fluxes F_{ij}^α. Any set of variables like the conserved variables, primitive variables, characteristic variables or entropy variables can be used. In the present work, we have used primitive variables for this purpose.

8.1 Linear interpolation with Venkatakrishnan limiter

Venkatakrishnan proposed a limiter for high order unstructured finite volume computations based on a smooth modification of the minmax limiter of Barth and Jespersen. The linear interpolants (20) are modified by multiplying the gradients with Venkatakrishnan limiter function s, which is defined for any node i as follows

$$s_i = \min_{j \in \mathcal{C}_i} s_{ij}$$

and

$$s_{ij} = \frac{(\Delta_{ij}^+)^2 + 2\Delta_{ij}^+ \Delta_{ij}^- + \varpi}{(\Delta_{ij}^+)^2 + 2(\Delta_{ij}^-)^2 + \Delta_{ij}^+ \Delta_{ij}^- + \varpi}$$

where $\Delta_{ij}^- = \frac{1}{2}\Delta x_{ij} \cdot \nabla V_i$ and

$$\Delta_{ij}^+ = \begin{cases} V_i^{\max} - V_i, & \text{if } \Delta_{ij}^- > 0 \\ V_i^{\min} - V_i, & \text{if } \Delta_{ij}^- < 0 \end{cases}$$

In the above equation $V_i^{\max}$, $V_i^{\min}$ are the maximum and minimum values of V in the connectivity and $\varpi = (Kh_i)^3$, where h_i can be taken to be the minimum length in the connectivity. The value of K determines the amount of limiting with a large value causing almost no limiting. In the present computations, it is set to 3.

8.2 Rotationally invariant interpolant

Another way of obtaining a limited interpolant is to follow the MUSCL approach [19] which can be generalized to multidimensions and non-structured grids [24]. We first define

$$\Delta_{ij}^- = R_{ij}\Delta\vec{x}_{ij} \cdot \nabla V_i, \quad \Delta_{ij}^+ = R_{ij}(V_j - V_i)$$
$$\Delta_{ji}^+ = R_{ij}\Delta\vec{x}_{ij} \cdot \nabla V_j, \quad \Delta_{ji}^- = R_{ij}(V_j - V_i)$$

The interpolants to the midpoint of $\vec{ij}$ are given by

$$V_{ij}^+ = R_{ij}V_i + \frac{s_{ij}}{4}\left[(1 - \kappa s_{ij})\Delta_{ij}^- + (1 + \kappa s_{ij})\Delta_{ij}^+\right]$$
$$V_{ij}^- = R_{ij}V_j - \frac{s_{ji}}{4}\left[(1 - \kappa s_{ji})\Delta_{ji}^+ + (1 + \kappa s_{ji})\Delta_{ji}^-\right]$$

where s is a Van Albada limiter which is defined as

$$s_{ij} = \max\left(0, \frac{2\Delta_{ij}^+\Delta_{ij}^- + \epsilon}{(\Delta_{ij}^+)^2 + (\Delta_{ij}^-)^2 + \epsilon}\right), \quad 0 < \epsilon \ll 1$$

with a similar expression for s_{ji}. The parameter κ determines the amount of upwinding; if $\kappa = 0$ then this corresponds to taking the arithmetic average in smooth regions. If $\kappa < 0$ then there is more upwind-bias in the interpolation. This way of defining $V_{ij}^{\pm}$ makes it rotationally-invariant even under the action of the limiter, and the scheme has the property stated in theorem (1).

9 Numerical order of accuracy

To investigate the numerical order of accuracy of KMM, we have applied it to a scalar conservation law of the form

$$\frac{\partial \varrho}{\partial t} + \sum_\alpha \frac{\partial F^\alpha}{\partial x^\alpha} = 0, \quad F^\alpha(\vec{x}, t, \varrho) = u_\alpha(\vec{x}, t, \varrho)\varrho(\vec{x}, t)$$

This equation can be obtained by taking moments (with $\psi = 1$) of the Boltzmann equation (2) with a Maxwellian-type distribution given by

$$f(\vec{v};\vec{x},t) = \varrho(\vec{x},t)\left(\frac{\beta}{\pi}\right)^{d/2} \exp\left(-\beta|\vec{v}-\vec{u}|^2\right)$$

for any $\beta > 0$. We choose a two-dimensional problem with $u_x = y$, $u_y = -x$, i.e., we solve

$$\frac{\partial \varrho}{\partial t} + \frac{\partial}{\partial x}(y\varrho) + \frac{\partial}{\partial y}(-x\varrho) = 0, \quad t > 0, \ (x,y) \in (0,1)\times(0,1) \tag{22}$$

with initial conditions

$$u(x,y,0) = \begin{cases} u_o(y), & \text{on } x = 0 \\ 0, & \text{elsewhere} \end{cases}$$

We choose u_o to be smooth and compactly supported in $[0,1]$ so that the appropriate boundary conditions are

$$u(0,y,t) = u_o(y), \quad u(x,1,t) = u(1,y,t) = 0, \quad \forall t > 0$$

The characteristics are concentric circles centered at the origin so that the steady state solution is

$$u_s(x,y) = u_o(\sqrt{x^2+y^2})$$

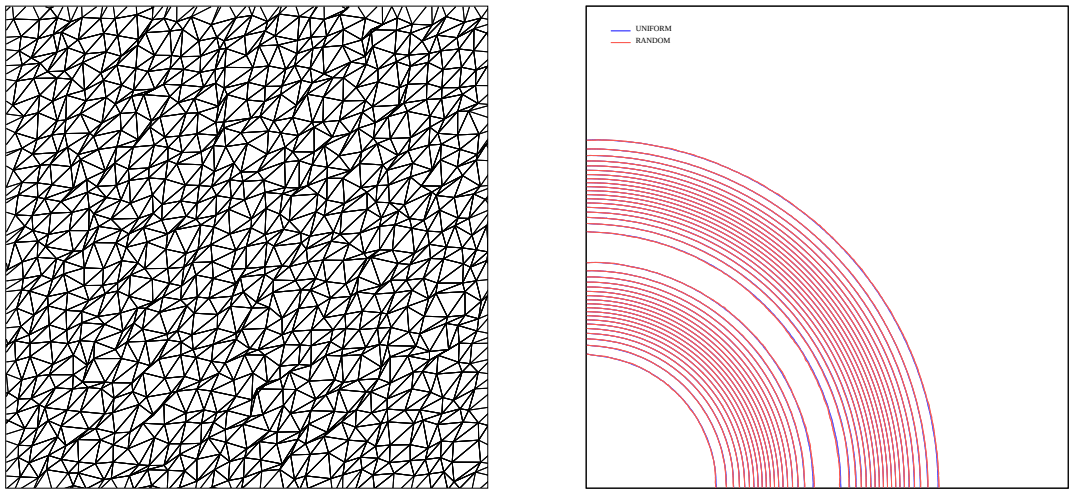
We use the following compactly supported data

$$u_o(y) = \begin{cases} \sin^2(2\pi(y+1/4)), & \frac{1}{4} < y < \frac{3}{4} \\ 0, & \text{elsewhere} \end{cases}$$

Equation (22) with $\beta = 0.1$ is solved on a sequence of uniform Cartesian and non-uniform/random grids which become successively finer. The non-uniform grid is obained by randomly perturbing the uniform Cartesian grid. The perturbed coordinates $(\bar{x},\bar{y})$ are given by $\bar{x} = x + 0.4\zeta\Delta x$ and $\bar{y} = y + 0.4\eta\Delta y$, where ζ, $\eta \in [-1,1]$ are random numbers. Figure (1) shows a non-uniform grid of 2601 points which is obtained by perturbing a Cartesian grid of 51×51 points. For each grid, the L_1, L_2 and L_∞ error are calculated and these are plotted in figure (2) against a grid size h which is defined as

$$h = \max_i \max_{j\in\mathcal{C}_i} |\vec{x}_j - \vec{x}_i|$$

The slope of the best straight line fit for the error as a function of h in log-log scale



(a) (b)

Figure 1: (a) Part of the random grid of 2601 points and (b) comparison of solution on uniform and random grid of 10201 points.

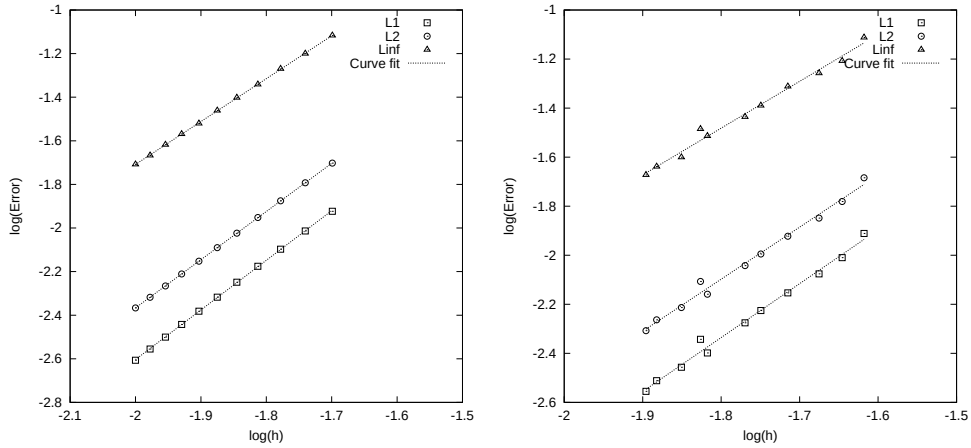
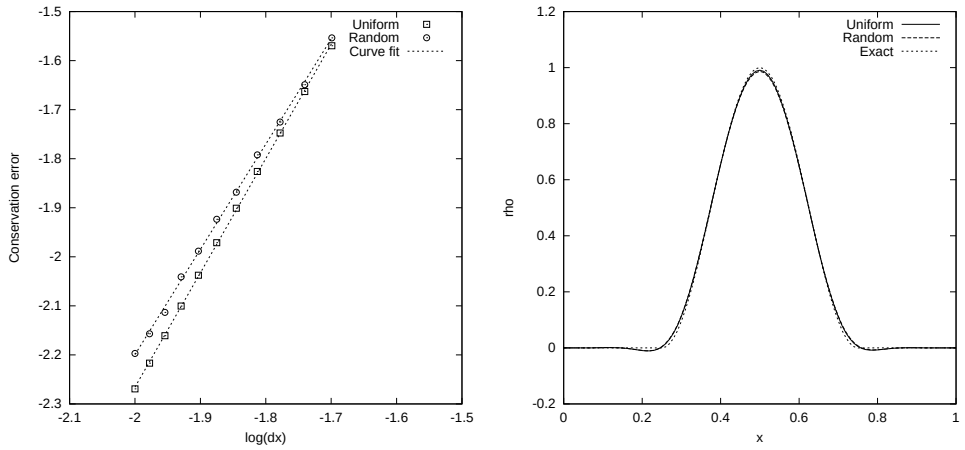


Figure 2: Error versus grid size for uniform and random grids.

Grid	L_1	L_2	L_∞
Uniform	2.27	2.21	1.97
Random	2.19	2.12	1.90

Table 1: Computed order of accuracy on uniform and non-uniform grids.



(a) (b)

Figure 3: (a) Conservation error versus point spacing, and (b) solution on the line $y = 0$.

gives the order of accuracy and these are listed in table. The best fit lines are also shown in figure (2). These results show an effective second order accuracy for the method. The initial condition specified on $x = 0$ is convected along circles centered at the origin, and the same profile should be obtained on $y = 0$ at steady state. Any difference in the two is a measure of loss of conservation. Hence to investigate the conservation property of the numerical method, we compute the L_1 norm of the solution error on the line $y = 0$ and this is plotted in figure (3a) against the point spacing on that line. The slope of these lines are found to be 2.33 and 2.14 for the uniform and random grids, respectively. The solution on the line $y = 0$ obtained on the final grid of 10201 points is shown in figure (3b), and is compared with the exact solution. This also shows an effective second order accuracy. Numerical results for transonic and supersonic flows indicate that the method has good conservation property since otherwise shocks would not have been captured correctly. Note that such methods cannot be conservative in the finite-volume sense; the best that can be done is to show that the conservation error decreases with the discretization size h.

The above numerical evidence suggests that the method given by (17), (18), (19), (20), might have effective second order accuracy even though the formal accuracy is one. The numerical test on highly non-uniform grids indicates that formal second order accuracy might be obtained on smooth grids and slightly non-uniform grids

though a proof of this is not available at present. The selection of the connectivity will also play an important role in determining the accuracy. For arbitrary distribution of points, a good connectivity can be obtained by performing a local delaunay triangulation on a local cloud of points and selecting the nearest neighbours. If the given point distribution is not smooth then it is also possible to use some grid improvement strategies to obtain a better point distribution. Alternately, non-uniform Cartesian grids based on quadtree technique can be generated relatively easily even for complex geometries and provide an ideal way of obtaining the point distribution required for meshless methods. These grids are smooth since the local point spacing jumps only by a factor of two.

10 Numerical results

The numerical results presented here are computed on unstructured grids since they are very easy to generate in 2-D and many public domain softwares are available. The vertices of the unstructured grid (triangles) provide the point distribution for the domain of the problem. The grid information is only used to generate the connectivity information and no other grid information is used in the solution process. In all the results presented below (except for subsonic flows), the limiter of Venkatakrishnan is used and time integration is performed using a 3-stage scheme of Shu and Osher [33] which for the equation

$$\frac{\mathrm{d}U}{\mathrm{d}t} = L(U)$$

is given by

$$\begin{aligned}
U^{(0)} &= U^n \\
U^{(1)} &= U^n + \Delta t L(U^{(0)}) \\
U^{(2)} &= \frac{3}{4}U^n + \frac{1}{4}(U^{(1)} + \Delta t L(U^{(1)})) \\
U^{(3)} &= \frac{1}{3}U^n + \frac{2}{3}(U^{(2)} + \Delta t L(U^{(2)})) \\
U^{n+1} &= U^{(3)}
\end{aligned}$$

The time-step is chosen based on the maximum local wave-speed $(|\vec{u}| + c)$, the minimum point spacing in the connectivity, and a CFL number of 2.

10.1 Subsonic flow over cylinder

Subsonic flow over a cylinder is computed at a free-stream Mach number of 0.38 with a total 4111 points with 250 points on the cylinder; a close-up view is shown in figure (4). The computed Mach number on cylinder shows very good left-right symmetry in figure (5) The pressure and Mach contours around the cylinder are shown in figure (6) which also show good symmetry in the solution. Any asymmetry will be seen more prominently in the Mach number contours and this is found to be very less in this case. The entropy production is very less with $S_{\min} = 1.59731$, $S_{\max} = 1.59886$ while the free-stream value is $S_\infty = 1.59869$, the maximum change being 0.097% of the freestream value. The lift and drag coefficients are 0.0006 and 0.0012 respectively.

10.2 Subsonic flow over Williams airfoil

The Williams airfoil is a two element airfoil for which potential solution is available. Figure (7) shows a close-up view of the grid and the computed streamlines through the gap between the airfoil and the flap. Figure (8) shows very good comparison between the computed pressure coefficient and the potential solution for both the main airfoil and the flap, while figure (9) shows the pressure and Mach contours. The minimum and maximum entropy in the domain are $S_{\min} = 3.45724$, $S_{\max} = 3.46488$ while the free-stream value is $S_\infty = 3.45776$, the maximum change being 0.22% of the freestream value. The lift and drag coefficients are 3.7608 and -0.0069 while the potential solution gives 3.736 and -0.0001 respectively.

10.3 Supersonic flow over cylinder

Supersonic flow over a cylinder at $M_\infty = 3.0$ is computed on a grid of 12857 points. The interior scheme is applied on the wall boundary points also and no special treatment is done except to kill the normal component of the velocity. The pressure and Mach contours are shown in figure (10); the Mach contours reveal a slip line on the leeward side of the cylinder. This can be seen in the velocity vector plot in figure (11). There are two counter-rotating vortices on the leeward side and one of them is seen in the streamline pattern in figure (12), while both of them can be seen in the vector plot (11) and entropy contours in figure (13). At the stagnation point behind the shock, the computed total pressure and total temperature are 0.3234 and 1.0006, while the exact values are 0.3283 and 1.0000 respectively.

10.4 Transonic flow over NACA-0012

We have computed the transonic flow over NACA-0012 at $M_\infty = 0.85$, AOA=1^o on a series of solution adapted grids. The grid generation and adaptation was performed using BAMG [2, 14]. The adaptation is based on all the conserved variables and figure (14) shows the sequence of adapted grids with five levels of adaptation, with 1989 (initial grid), 1636, 2222, 2999, 4335 and 6938 points. Figures (15), (16) show the pressure and Mach contours on the adapted grids, while figure (17) shows the contours on the final grid. In figure (18) the pressure coefficient on the final grid is compared with that on a grid of 4385 points. The lift and drag coefficients on the final grid are 0.389 and 0.054 respectively.

10.5 Transonic flow over RAE-2822

A similar adaptation sequence is carried out for transonic flow over RAE-2822 airfoil at $M_\infty = 0.75$ and AOA=3^o. The initial and adapted grids are shown in figure (19) which contain 1966 (initial grid), 1624, 2027, 2696, 3828 and 5482 points. The pressure and Mach contours are shown in figure (20) and (21). The contours on the final grid are shown in figure (22). The lift and drag coefficients are 1.141 and 0.0445 respectively.

11 An alternative formulation

We can give an alternative formulation of KMM which has some resemblance to finite volume methods. For this we follow the shape function approach that is common in finite element methods and approximate f around the node i by

$$f_i(\vec{x}) = A_i + \sum_\alpha A_i^\alpha (x^\alpha - x_i^\alpha)$$

Note that in the previous formulation the value of A_i is fixed by the known value of f at node i, while here it is left as an additional variable. Performing a least squares minimization as before, we obtain the following equations for the derivatives at node i

$$\left.\frac{\partial f}{\partial x^\alpha}\right|_i = \sum_{j \in \mathcal{C}_i} \nu_{ij}^\alpha f_{ij}$$

where the coefficients, which have the units of inverse-length, sum to zero

$$\sum_{j \in \mathcal{C}_i} \nu_{ij}^\alpha = 0$$

Using this approximation in the Boltzmann equation and taking moments we obtain a semi-discrete equation

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} + \sum_\alpha \sum_{j \in \mathcal{C}_i} \nu_{ij}^\alpha F_{ij}^\alpha = 0 \tag{23}$$

where the flux F_{ij}^α is defined as before. The finite volume approximation can be written as

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} + \sum_\alpha \sum_{j \in \mathcal{C}_i} \frac{n_{ij}^\alpha S_{ij} F_{ij}^\alpha}{V_i} = 0$$

assuming the points in $\mathcal{C}_i$ define a finite volume around node i. Note that the coefficients in the above equation have the property

$$\sum_{j \in \mathcal{C}_i} n_{ij}^\alpha S_{ij} = 0$$

and hence we have the following structural similarity between the two methods

$$\nu_{ij}^\alpha \sim \frac{n_{ij}^\alpha S_{ij}}{V_i}$$

We have not explored this approach or its further similarities with the finite volume method, but it is possible that the above similarity might have much deeper consequences. For a related discussion, see the paper by Junk [20].

In the shape function approach, it is not necessary to use only polynomial basis functions; some authors have used radial basis functions to better represent some local behaviour which has cylindrical or spherical symmetry. Interpolation using a combination of polynomial and radial basis functions has been shown to be superior to pure polynomial-based interpolation [34].

12 Summary

A kinetic meshless method for the numerical solution of hyperbolic conservation laws which are derivable from a Boltzmann-type equation has been presented and applied to many 2D problems involving subsonic to supersonic flows. These results amply demonstrate the ability of the method to model compressible flows and capture the important flow features. While formally the method is only first order accurate, the results indicate much better accuracy; and a numerical order of accuracy of approximately two has been demonstrated for a scalar convection equation even on highly non-uniform grids. The extension of the method to formally higher order accuracy is straight-forward. Future work is aimed at using the alternative formulation described in the previous section together with higher order basis functions.

Acknowledgements

We have used two grid generators, Delaundo [1] of JD Mueller and BAMG [2] of Frederic Hecht, for generation of unstructured grids. The latter also has grid adaptation capability which has been used in some of the computations. We thank the respective authors for making these software available in the public domain.

Appendix: Expressions for split fluxes

The expressions for the split fluxes as defined in equation (21) are given below.

$$\tilde{G}^{\pm} = \begin{bmatrix} \varrho u_y A_x^{\pm} \\ \varrho u_y (u_x A_x^{\pm} \pm B_x) \\ (p + \varrho u_y^2) A_x^{\pm} \\ (E+p) u_y A_x^{\pm} \pm \dfrac{\varrho u_x u_y}{2} B_x \end{bmatrix}$$

The fluxes $F^{\pm}$ are the usual kinetic x-split fluxes, which are given by,

$$F^{\pm} = \begin{bmatrix} \varrho (u_x A_x^{\pm} \pm B_x) \\ (p + \varrho u_x^2) A_x^{\pm} \pm \varrho u_x B_x \\ \varrho u_y (u_x A_x^{\pm} \pm B_x) \\ (E+p) u_x A_x^{\pm} \pm (E + p/2) B_x \end{bmatrix}$$

where

$$E = \frac{p}{\gamma - 1} + \frac{\varrho |\vec{u}|^2}{2}$$

and for $\alpha = x, y$,

$$\begin{aligned} s_\alpha &= u_\alpha \sqrt{\beta} \\ A_\alpha^{\pm} &= \frac{1}{2}[1 \pm \mathrm{erf}\,(s_\alpha)] \\ B_\alpha &= \frac{\exp(-s_\alpha^2)}{2\sqrt{\pi\beta}} \end{aligned}$$

References

[1] `http://www.cerfacs.fr/ muller/delaundo.html`

[2] `http://pauillac.inria.fr/cdrom/ftp/bamg`

[3] N Balakrishnan and Praveen. C, "A new upwind least squares finite difference scheme (LSFD-U) for Euler equations of gas dynamics", Finite Volumes for Complex Applications-II, Hermes Science Publications, pp. 331, Paris, 1999.

[4] N Balakrishnan, "New least squares based finite difference methods", 99-FM-9, Dept. of Aerospace Engg., Indian Institute of Science, Bangalore.

[5] N Balakrishnan and G Fernandez, "Wall boundary conditions for inviscid compressible flows on unstructured meshes", *Int. Jl. Num. Meth. Fluids*, 28:1481-1501, 1998.

[6] TJ Barth, "Aspects of unstructured grids and finite-volume solvers for the Euler and Navier-Stokes equations", VKI Lecture Series 1994-05.

[7] JT Batina, "A gridless Euler/Navier-Stokes solution algorithm for complex aircraft applications", AIAA Paper 93-0333, 1993.

[8] CE Baumann, CA Duarte, ON Hamzeh, and TJ Liszka, "The mathematical theory and implementation of hp adaptive meshless analysis for interdisciplinary applications and high-performance computing", May 2000.

[9] T Belytschko, YY Lu, and L Gu, "Element-free Galerkin methods", *Int. Jl. Num. Meth. Engg.*, 37:229-256, 1994.

[10] T Belytschko, Y Krongauz, D Organ, M Flemming and P Krysl, "Meshless methods: An overview and recent developments", *Comput. Methods Appl. Mech. Engg.*, Vol. 139, pp. 3-47, 1996.

[11] KC Chung, "A generalised finite difference method for heat transfer problems of irregular geometries", *Num. Heat Transfer*, 4:345-357, 1981.

[12] SM Deshpande, "On the Maxwellian distribution, symmetric form, and entropy conservation for the Euler equations", NASA-TP-2583, 1986.

[13] SM Deshpande, "Meshless methods, accuracy, symmetry breaking, upwinding and LSKUM", FM Report, 2003-FM-1, Dept. of Aerospace Engg., IISc, Bangalore, 2003.

[14] MJ Castro Diaz, F Hecht, B Mohammadi, "New progress in anisotropic grid adaptation for inviscid and viscous flow simulations", RR-2671, October 1995, INRIA.

[15] AK Ghosh, SM Deshpande, "Least squares kinetic upwind method for inviscid compressible flows", AIAA paper 95-1735,1995.

[16] AK Ghosh, "Robust least squares upwind method for inviscid compressible flows", PhD Thesis, Dept. of Aerospace Engg., Indian Institute of Science, Bangalore, June 1996.

[17] A Harten, PD Lax and B van Leer, "On upstream differencing and Godunov-type schemes for hyperbolic conservation laws", ICASE Report No. 82-5, 1982.

[18] Carl-Ollivier Gooch, "High-order ENO schemes for unstructured meshes Based on least-squares reconstruction", AIAA 35th Aerospace Sciences Meeting, Reno, NV, January 1997

[19] C Hirsch, *Numerical computation of internal and external flows. Vol. 2*, John Wiley and Sons, 1992.

[20] M Junk, "Do finite volume methods really need a mesh ?", Int. Workshop on Meshfree Methods for PDE, Bonn, September, 2001.

[21] P Lancaster and K Salkauskas, "Surfaces generated by moving least squares methods", *Math. Comp.*, 37(155):141-158, 1981.

[22] T Liszka, "An interpolation method for an irregular net of nodes", *Int. Jl. for Num. Meth. in Engg.*, 20:1599-1612, 1984.

[23] T Liszka and J Orkisz, "The finite difference method at arbitrary irregular grids and its application in applied mechanics", *Comp. Struct.*, 11:83-95, 1980.

[24] R Lohner, *Applied CFD Techniques: An Introduction based on Finite Element Methods*, John Wiley and Sons, 2001.

[25] K Morinishi, "Gridless type solution for high Reynolds number multi-element flow fields", AIAA Paper 95-1856, 1995.

[26] K Morinishi, "An implicit gridless type solver for the Navier-Stokes equations", Int. Symp. on CFD, Bremen, September 1999.

[27] B Nayroles, G Touzot, and P Villon, "Generalizing the finite element method: diffuse approximations and diffuse elements", *Computational Mechanics*, 10:307-318, 1992.

[28] Praveen C, "A new upwind least squares finite difference scheme for compressible flows", ME Thesis, Dept. of Aerospace Engg., Indian Institute of Science, Bangalore, January 2000.

[29] Praveen C, "Some results on the least squares formula", FM Report, FM 2001-06, Dept. of Aerospace Engg., IISc, Bangalore.

[30] Praveen C, SM Deshpande, "Rotationally invariant grid-less upwind method for Euler equations", 2001-FM-08, Dept. of Aerospace Engg., IISc, Bangalore.

[31] V Ramesh and SM Deshpande, "Euler computations on arbitrary grids using LSKUM", First ICCFD, July 2000, Kyoto, Japan.

[32] Sridar D and Balakrishnan N, "An upwind finite difference scheme for meshless solvers", *FM Report*, No. 2002-FM-01, Dept. of Aerospace Engg., IISc, Bangalore, 2002.

[33] CW Shu and S Osher, "Efficient implementation of essentially non-oscillatory schemes", *Jl. Comp. Phy.*, 77:439, 1988.

[34] Th. Sonar, "Optimal recovery using thin plate splines in finite volume methods for the numerical solution of hyperbolic conservation laws", IMA J. Num. Anal., 16:549-581, 1995.

[35] Thomas JL and Salas MD, "Far-field boundary condition for transonic lifting solutions to the Euler equations", AIAA Paper No. 85-0020, 1985.

[36] V Venkatakrishnan, "Convergence to steady state solutions of the Euler equations on unstructured grids with limiters", *Jl. Comp. Phy.*, 118:120-130, 1995.

[37] Wendland H, "Meshless Galerkin methods using radial basis functions", *Math. Comp.*, 68:1521-1531, 1999.

[38] Yagawa G and Yamada T, "Free mesh method: A new meshless finite element method", *Comp. Mech.*, 18:383-386, 1996.

[39] T Fujisawa and G Yagawa, "A virtually meshless formulation for compressible high speed flows with free mesh method", First MIT conference on Computational Fluid and Solid Mechanics, June 2001.

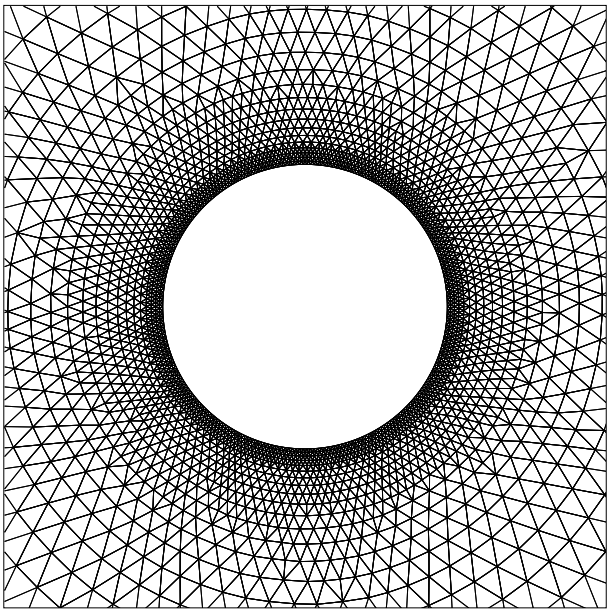


Figure 4: Unstructured grid for a 2-D cylinder.

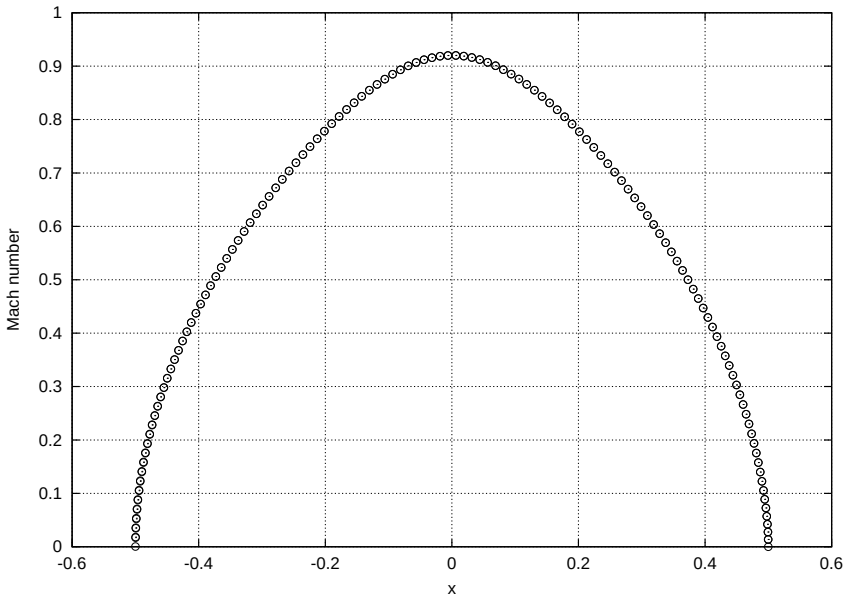
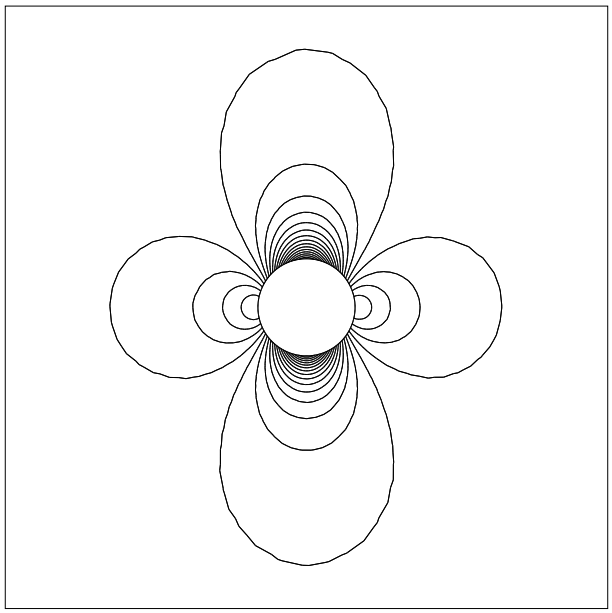


Figure 5: Mach number for flow over a cylinder at $M_\infty = 0.38$ and $AOA = 0^o$.

Pressure



Mach number

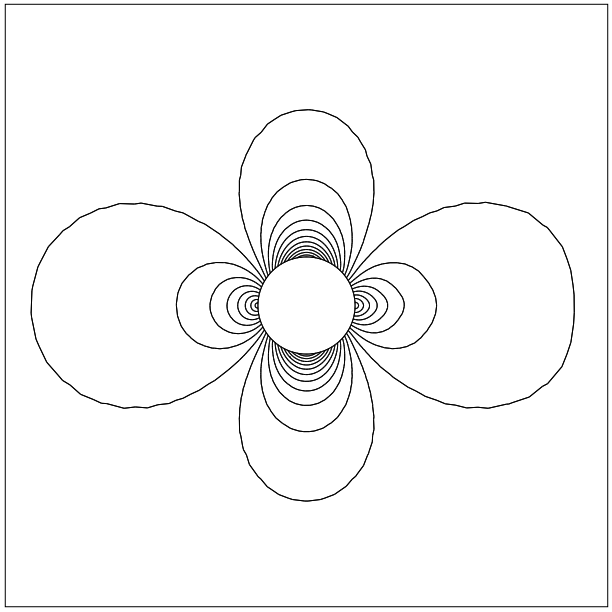


Figure 6: Pressure and Mach contours for flow over a cylinder at $M_\infty = 0.38$ and $AOA = 0^o$.

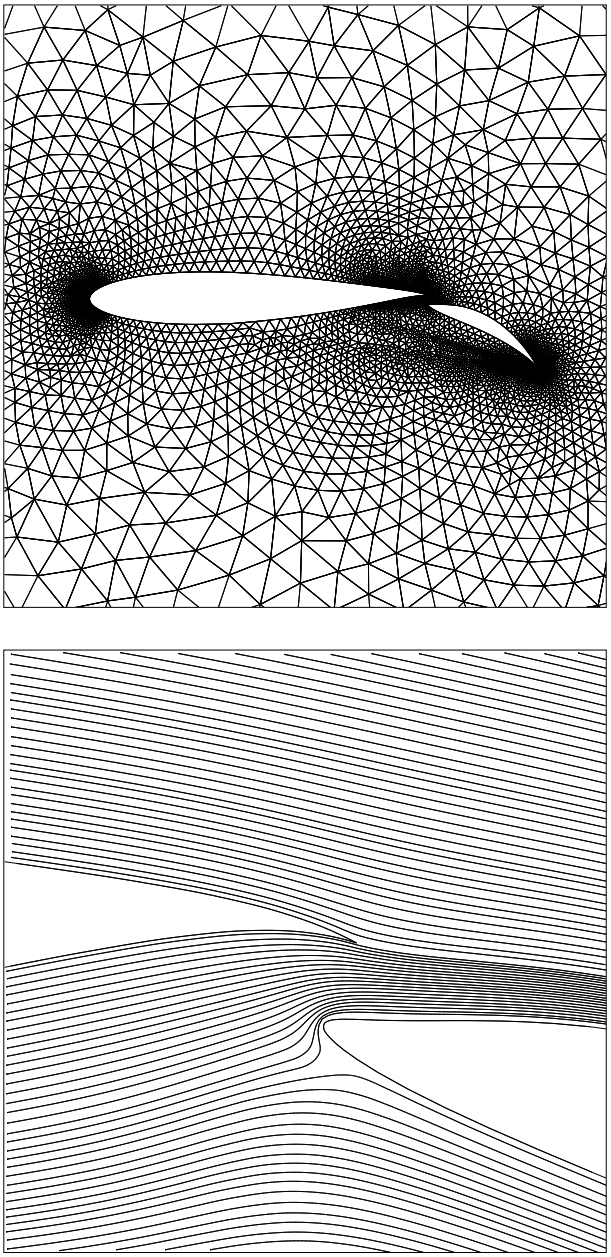
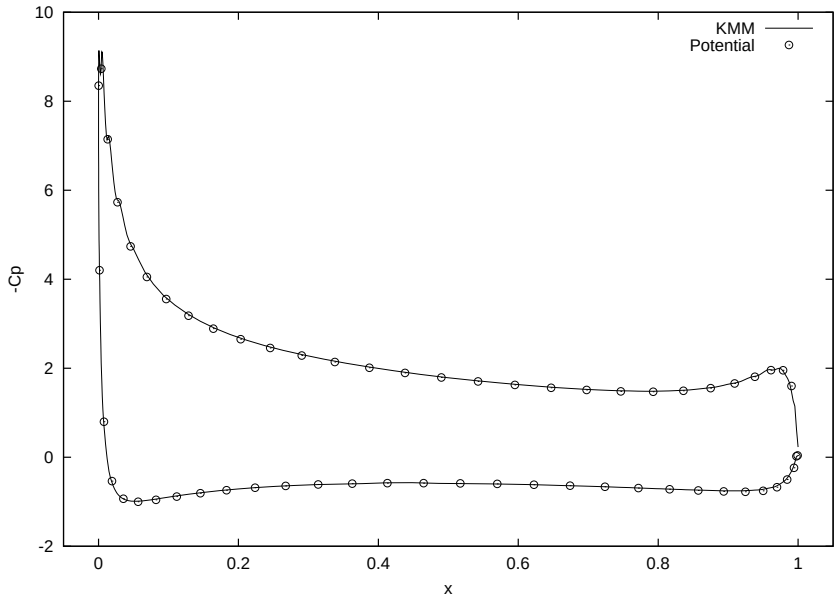


Figure 7: Grid and streamlines around Williams airfoil, $M_\infty = 0.15$ and $AOA = 0^o$.

Main Airfoil



Flap

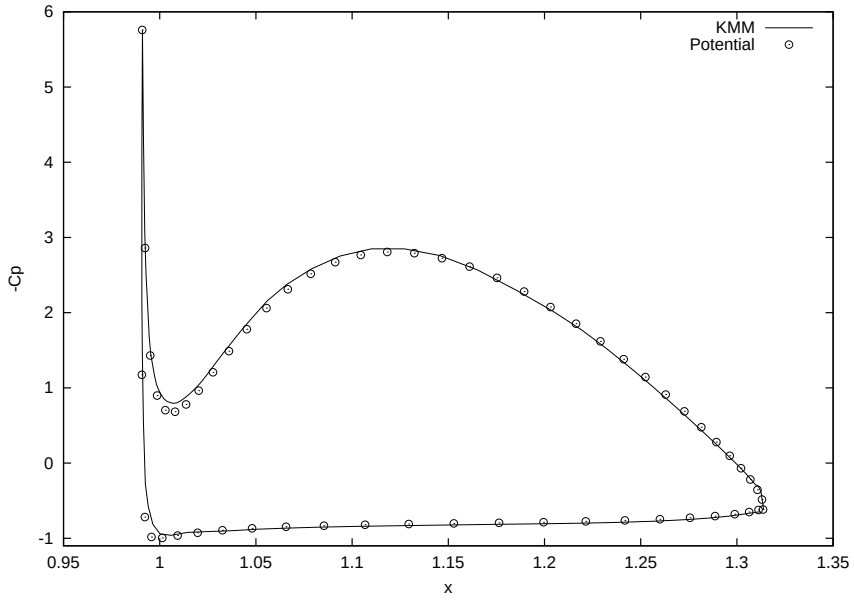
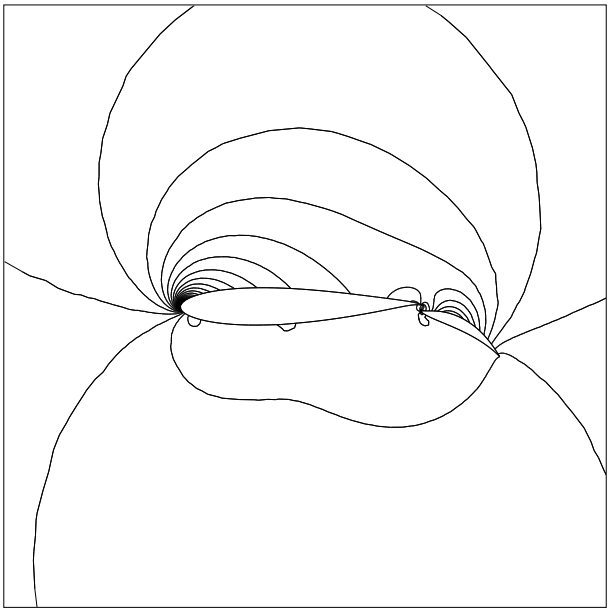


Figure 8: Pressure coefficient for Williams airfoil, $M_\infty = 0.15$ and $AOA = 0^o$.

Pressure



Mach number

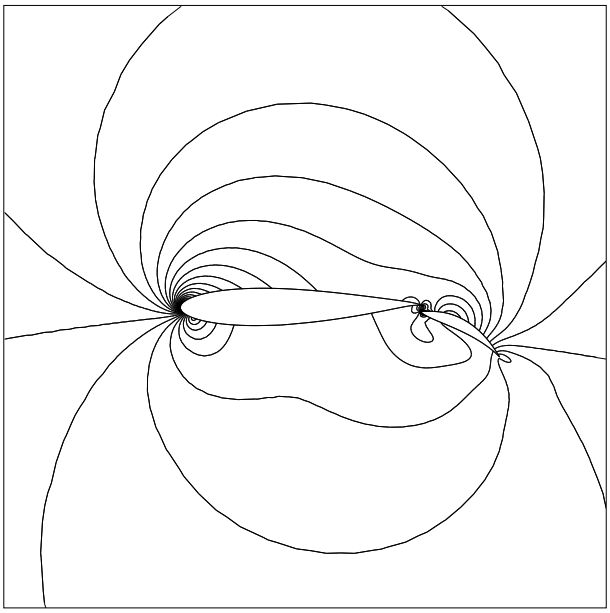
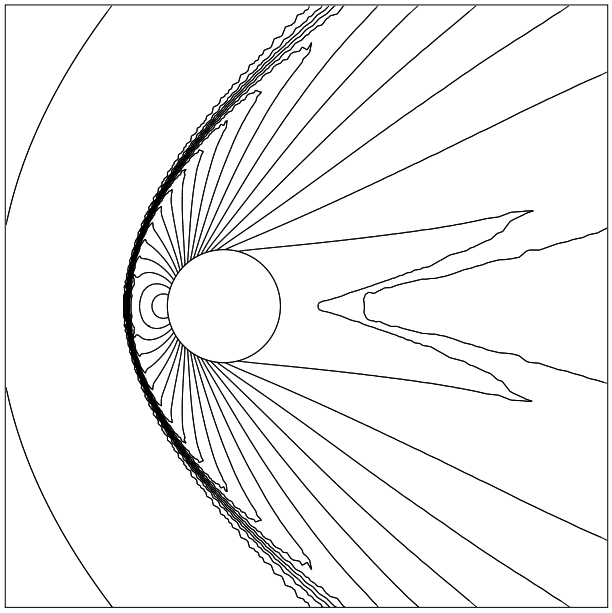


Figure 9: Pressure and Mach contours for Williams airfoil, $M_\infty = 0.15$ and $AOA = 0^o$.

Pressure



Mach number

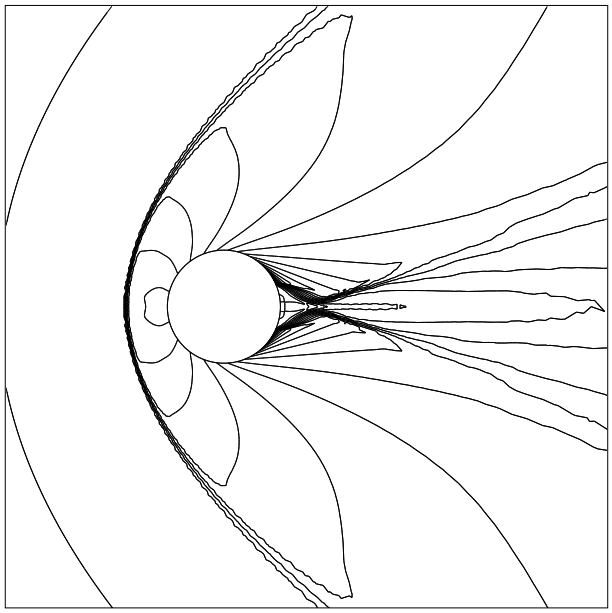


Figure 10: Pressure and Mach contours for cylinder, $M_\infty = 3$ and $AOA = 0^o$.

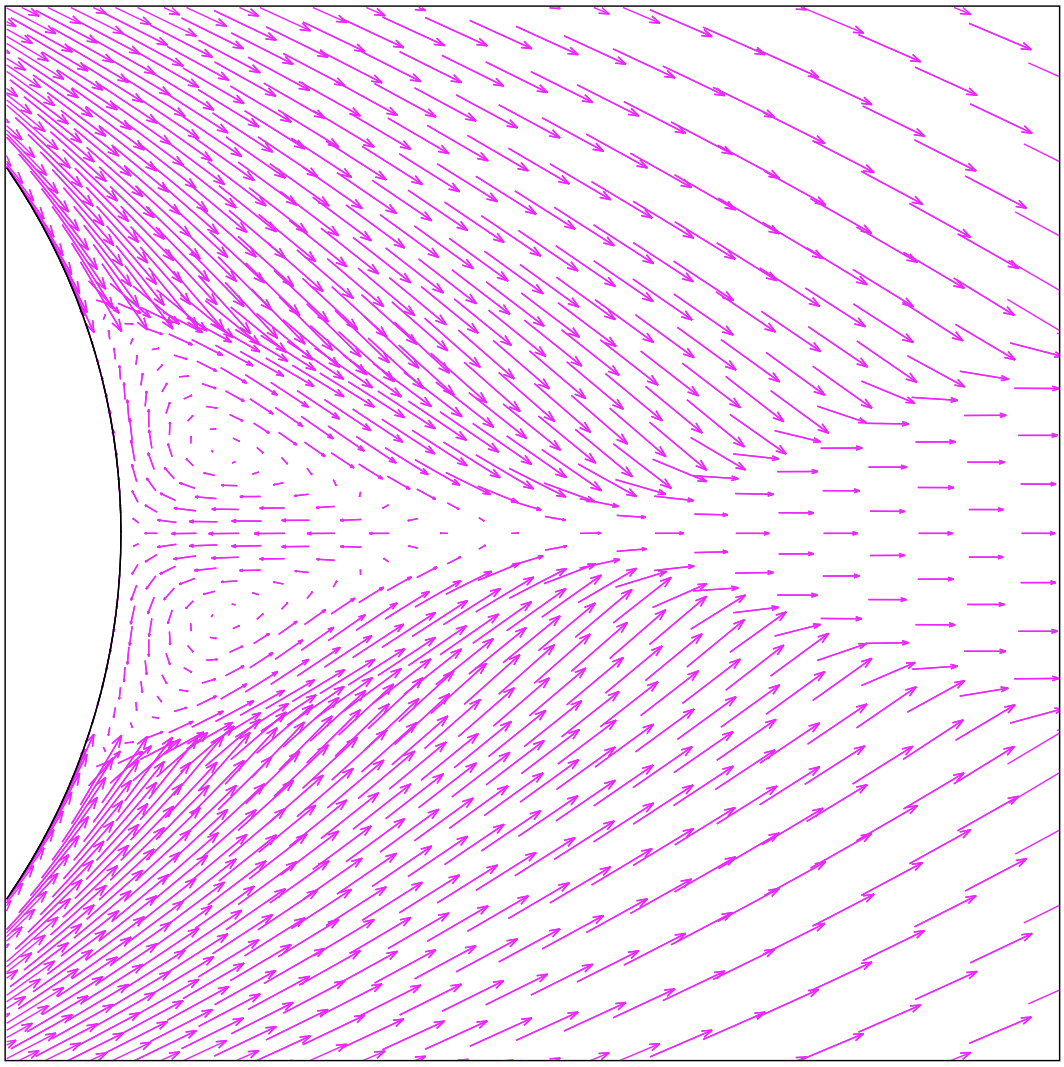


Figure 11: Velocity vectors on leeward side of cylinder, $M_\infty = 3$ and $AOA = 0^o$.

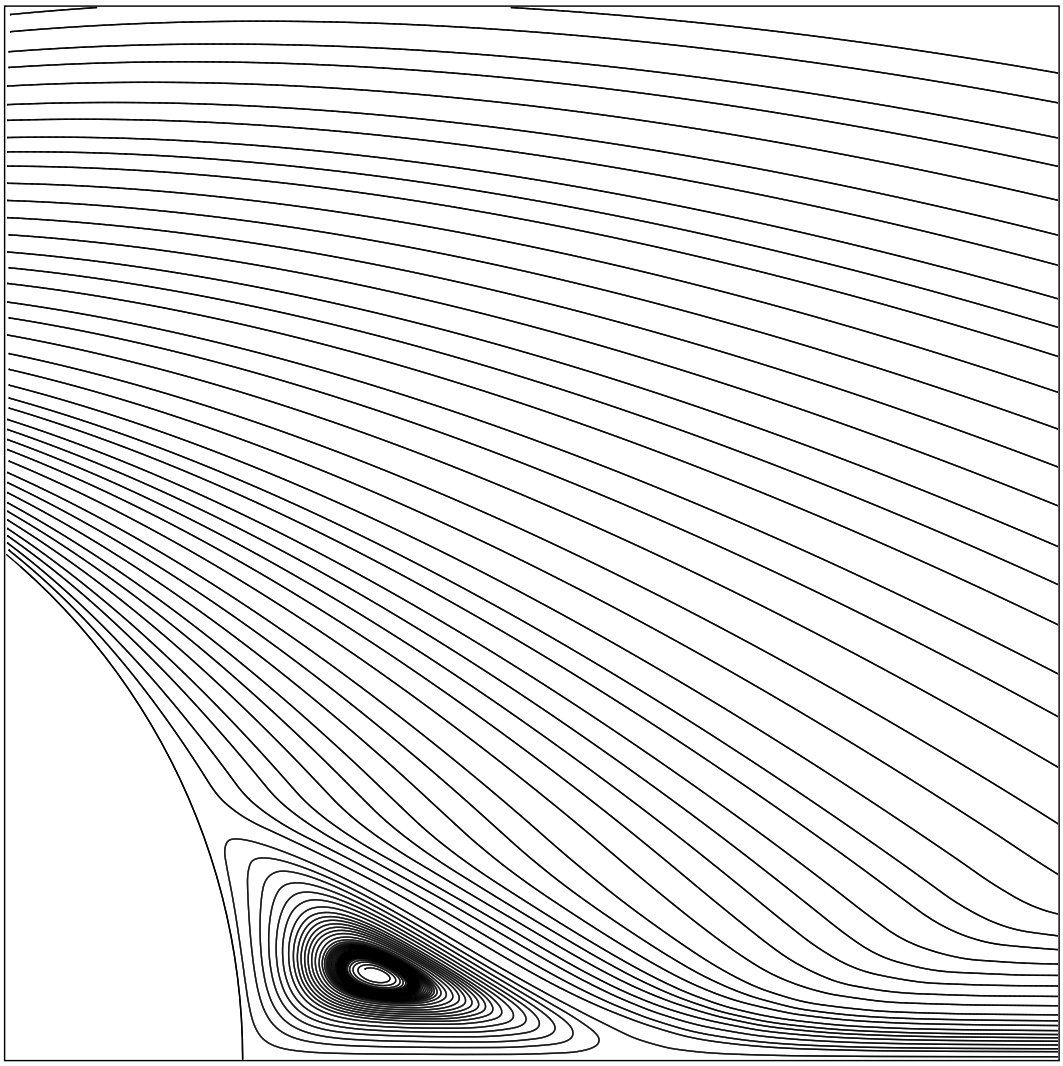


Figure 12: Streamline pattern on leeward side of cylinder, $M_\infty = 3$ and $AOA = 0^o$.

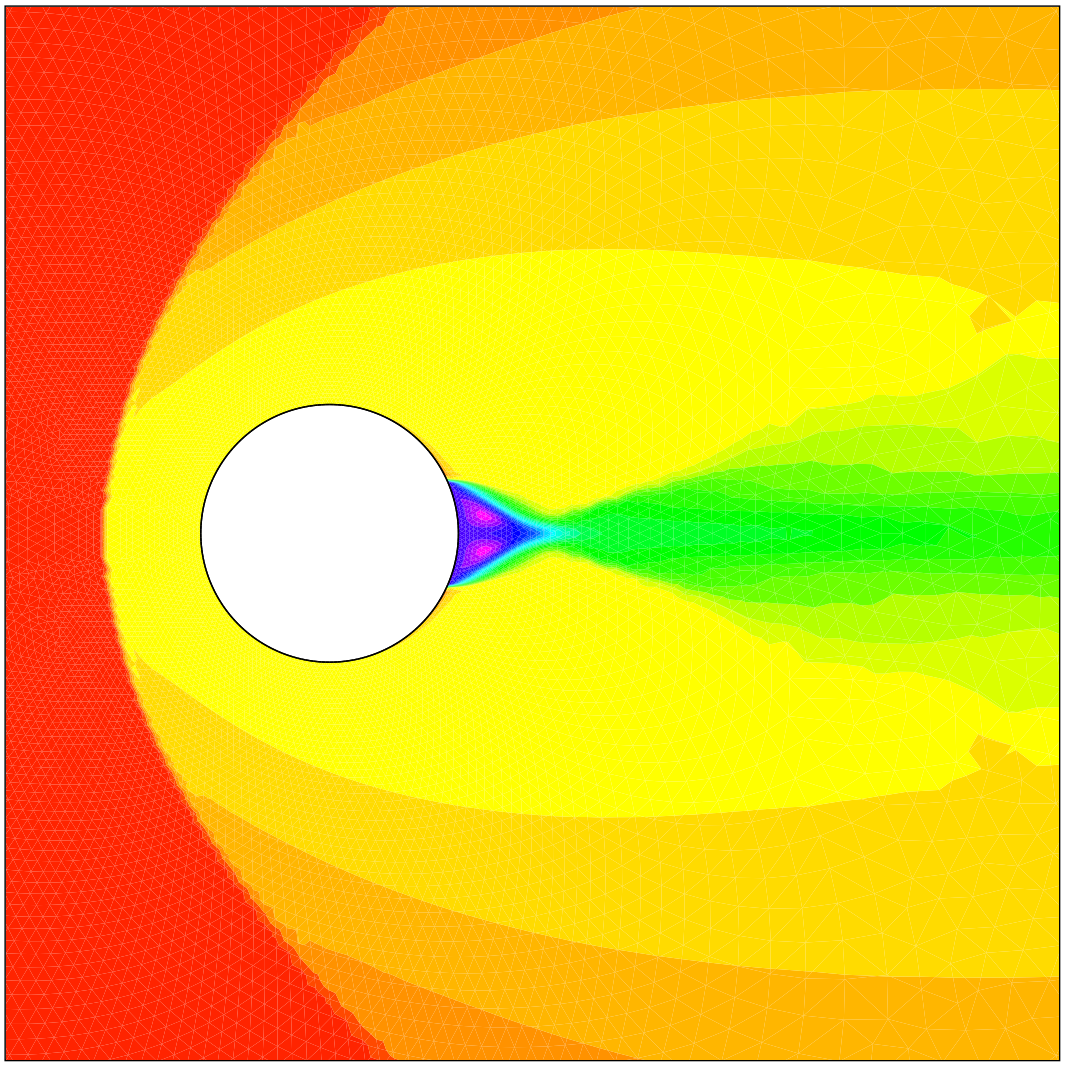


Figure 13: Entropy contours around cylinder, $M_\infty = 3$ and $AOA = 0^o$.

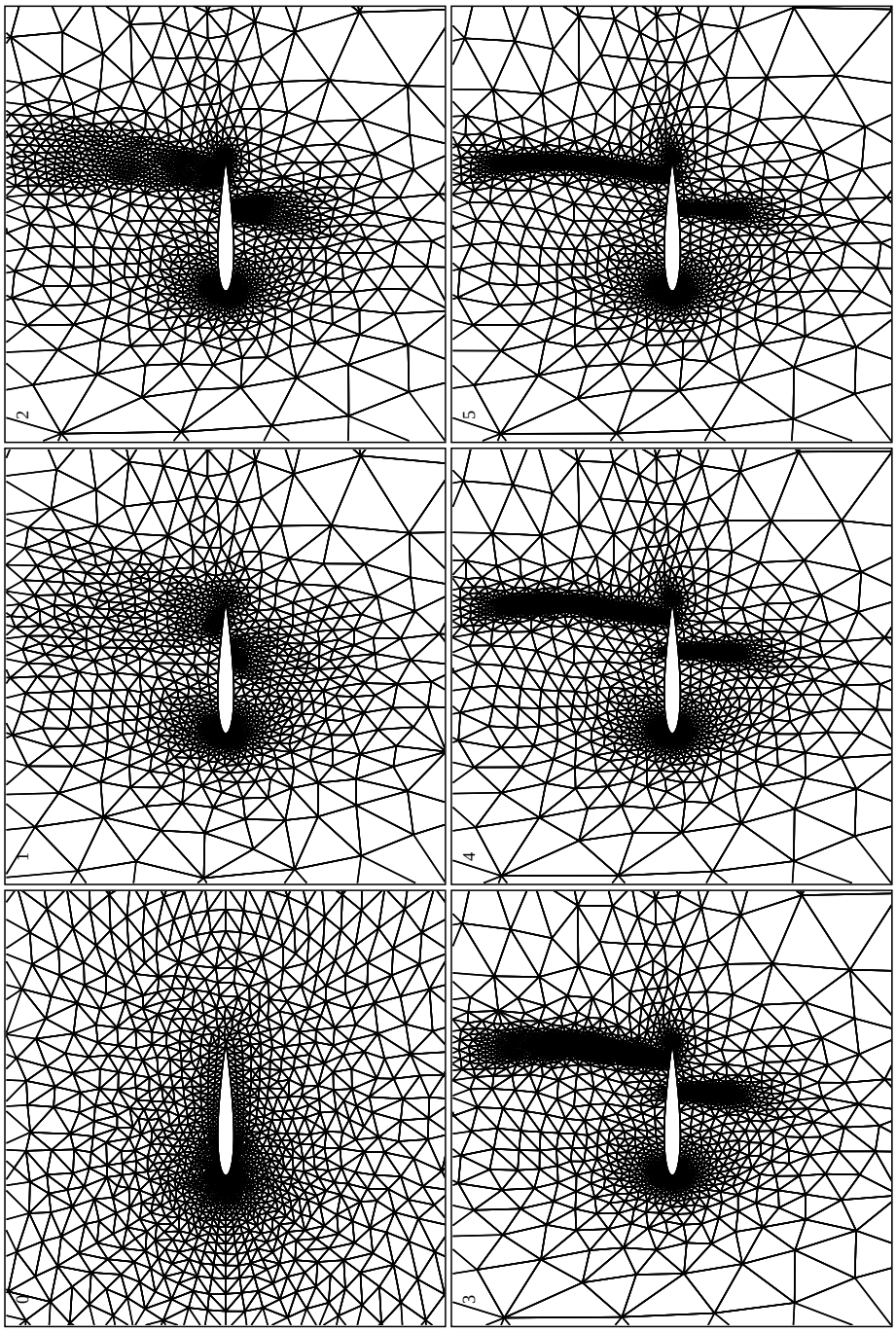


Figure 14: Sequence of adapted grids for NACA-0012, $M_\infty = 0.85$ and $AOA = 1.0^o$.

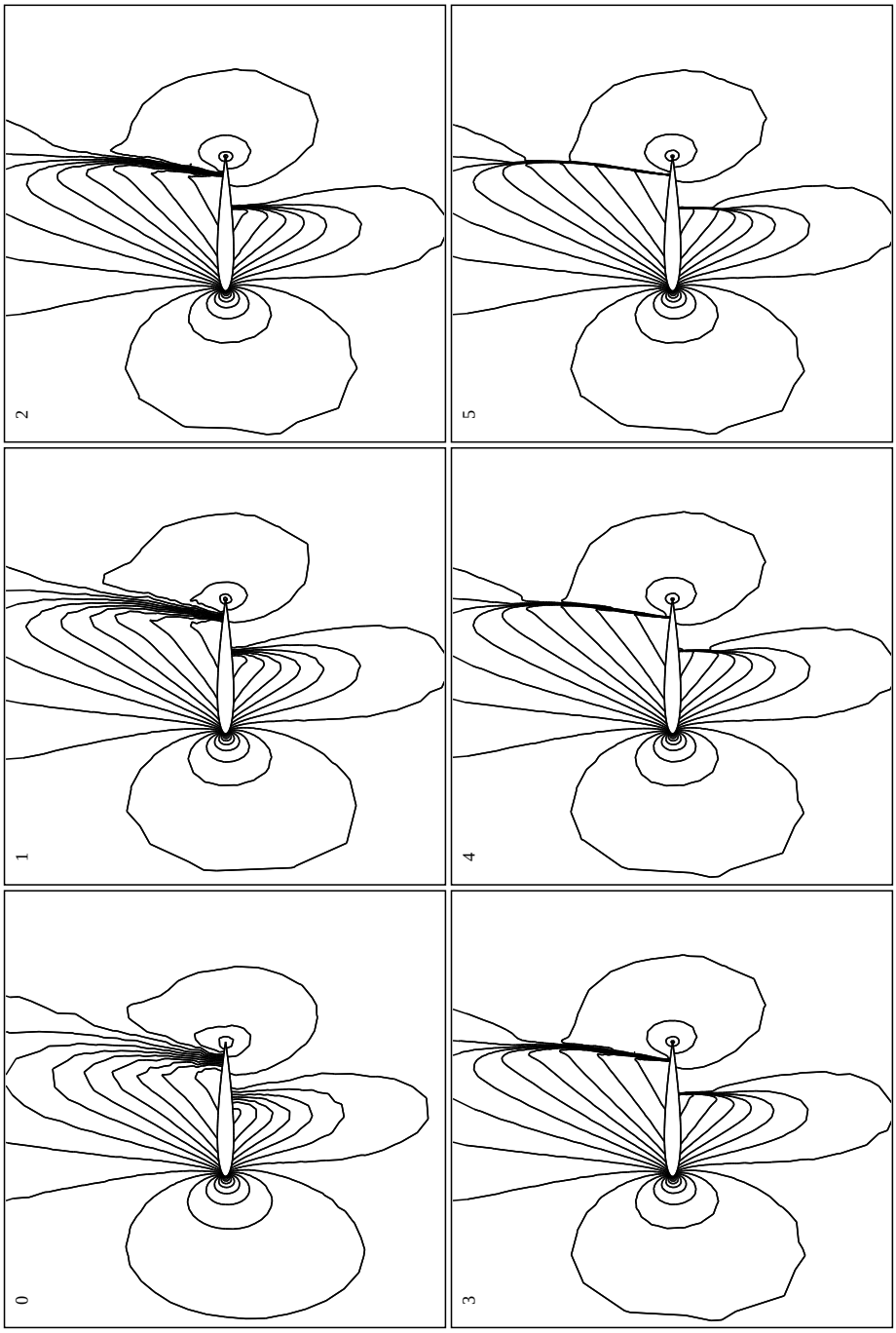


Figure 15: Sequence of pressure contours for NACA-0012, $M_\infty = 0.85$ and $AOA = 1.0^o$.

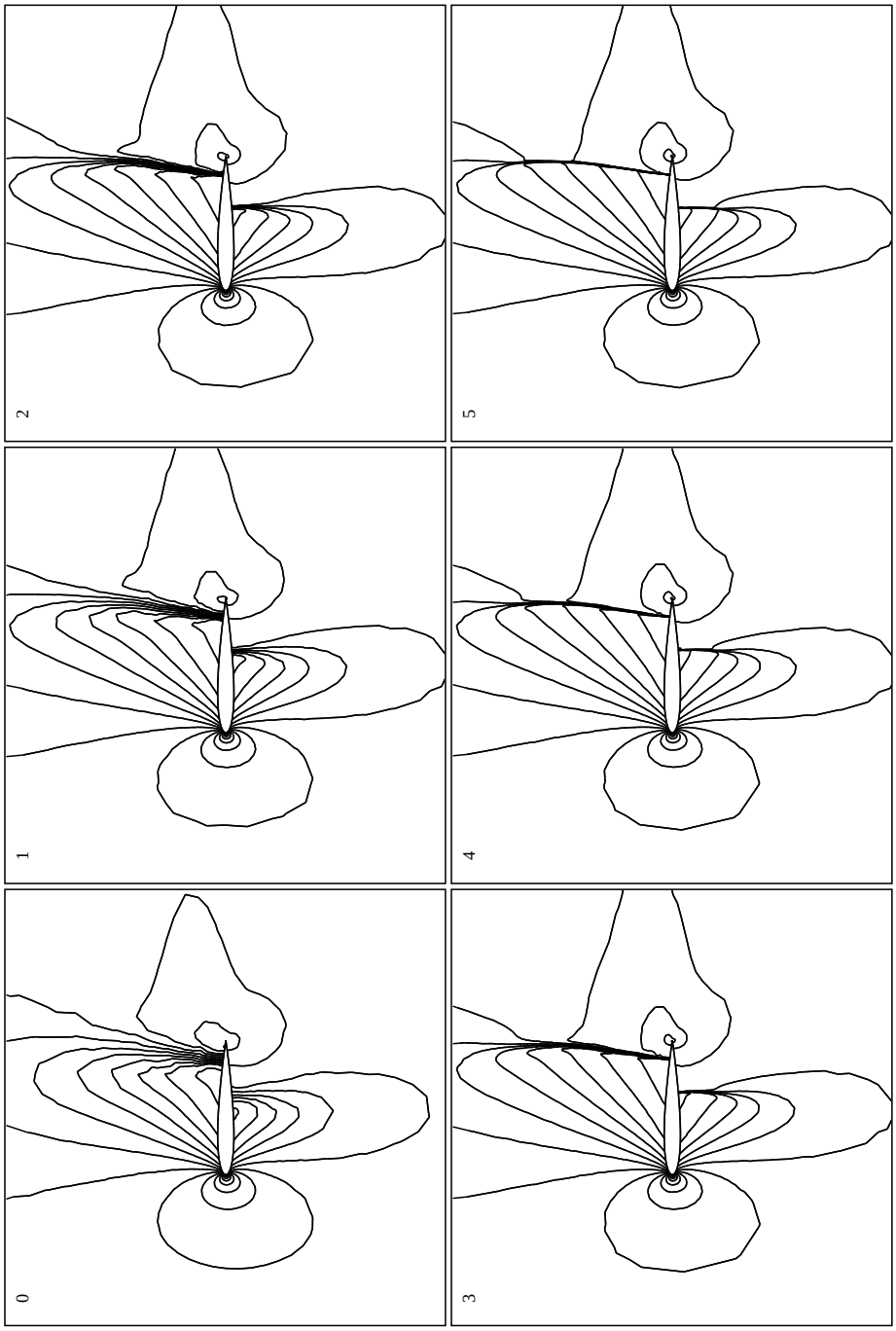
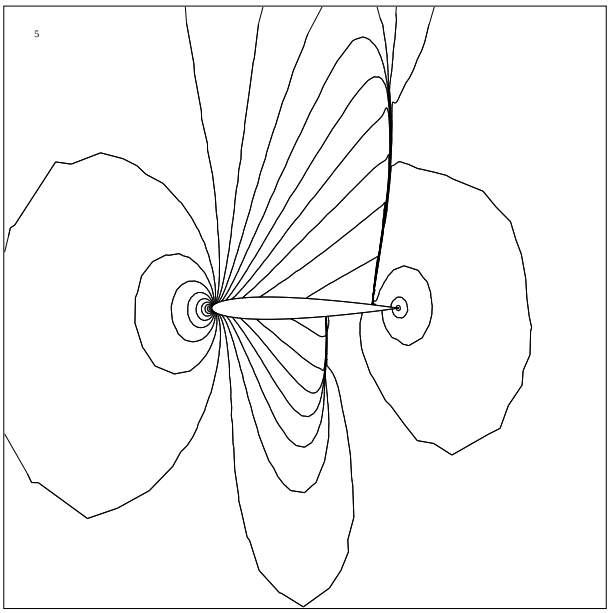


Figure 16: Sequence of mach contours for NACA-0012, $M_\infty = 0.85$ and $AOA = 1.0^o$.

Pressure



Mach number

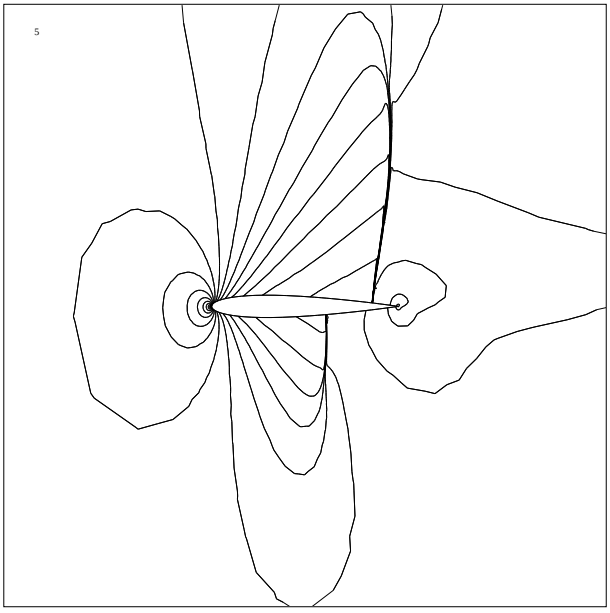


Figure 17: Pressure and Mach contours on final mesh for NACA-0012, $M_\infty = 0.85$ and $AOA = 1.0^o$.

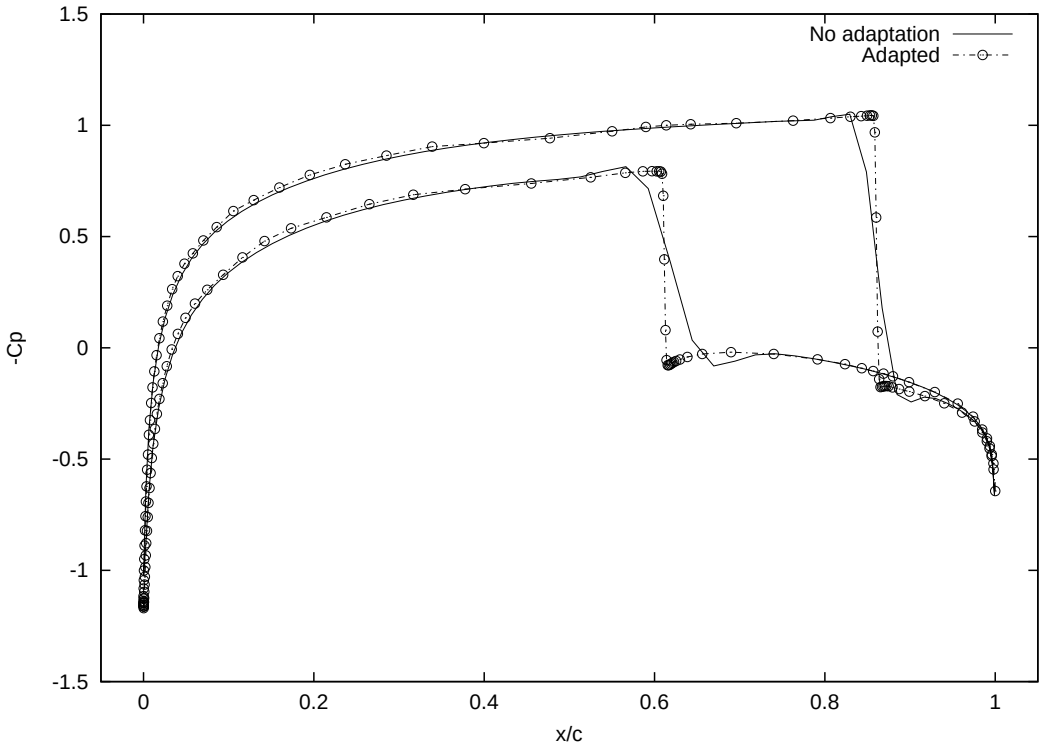


Figure 18: C_p for NACA-0012, $M_\infty = 0.85$ and $AOA = 1.0^o$.

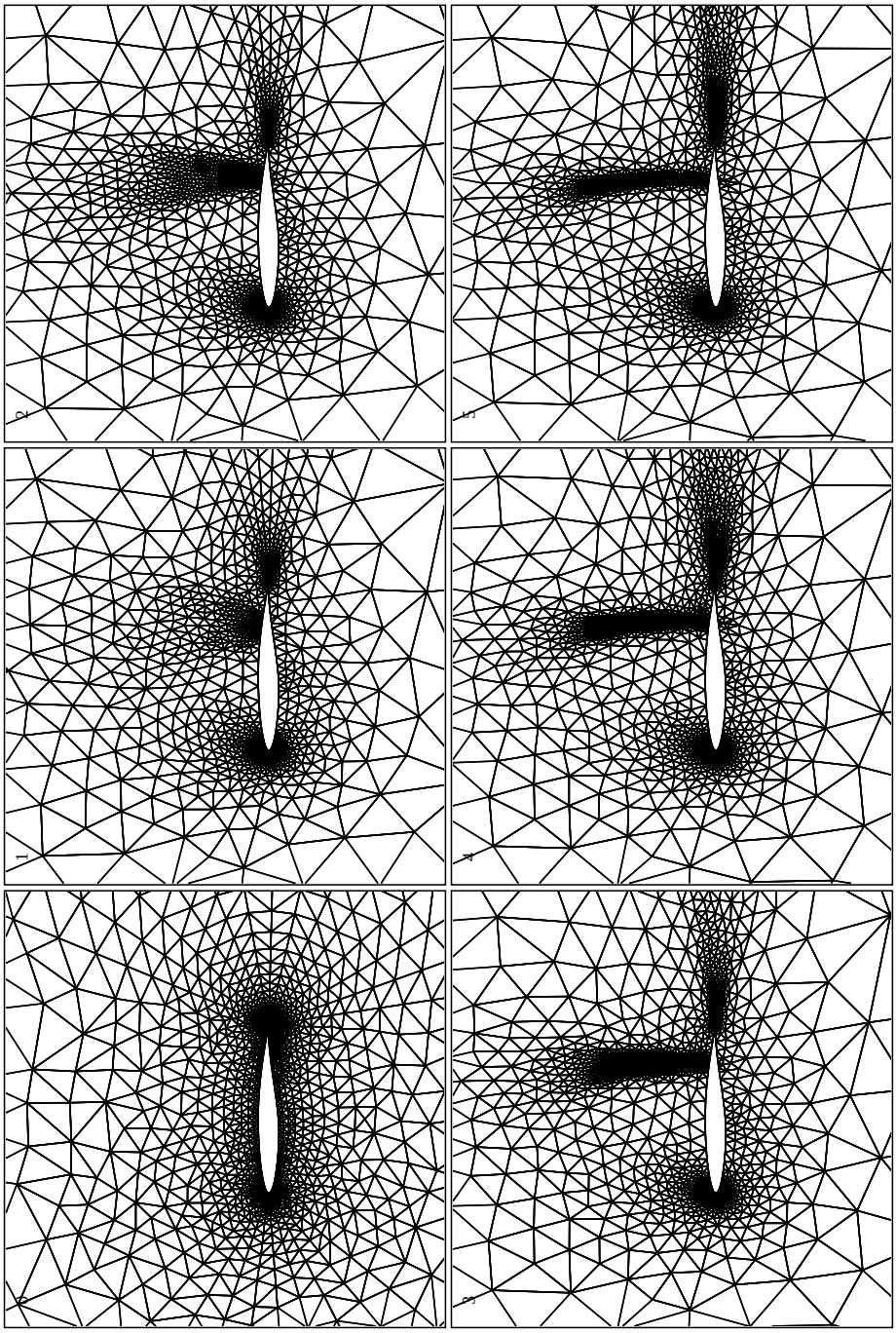


Figure 19: Sequence of adapted grids for RAE-2822

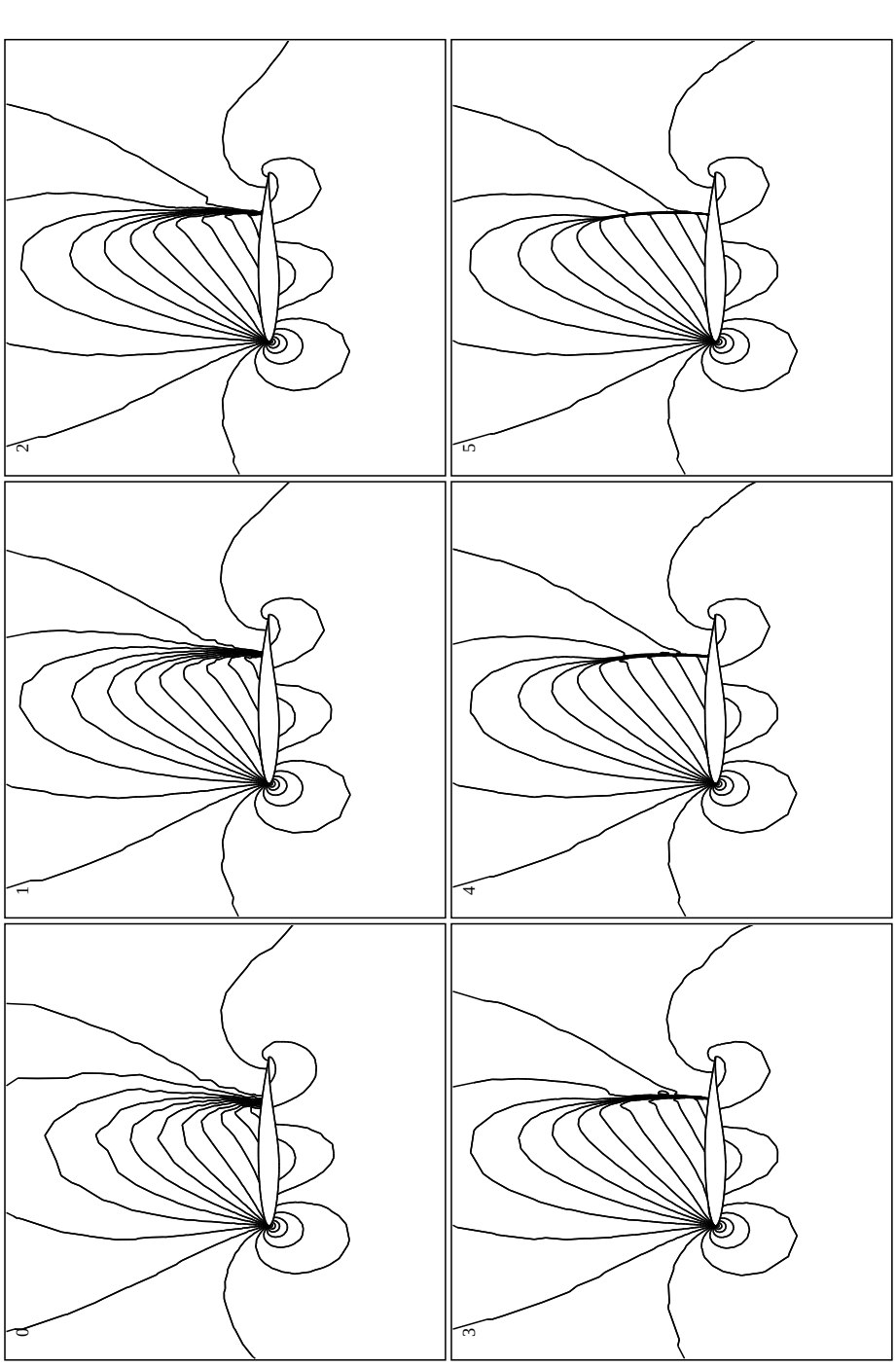


Figure 20: Pressure contours on adapted grids for RAE-2822

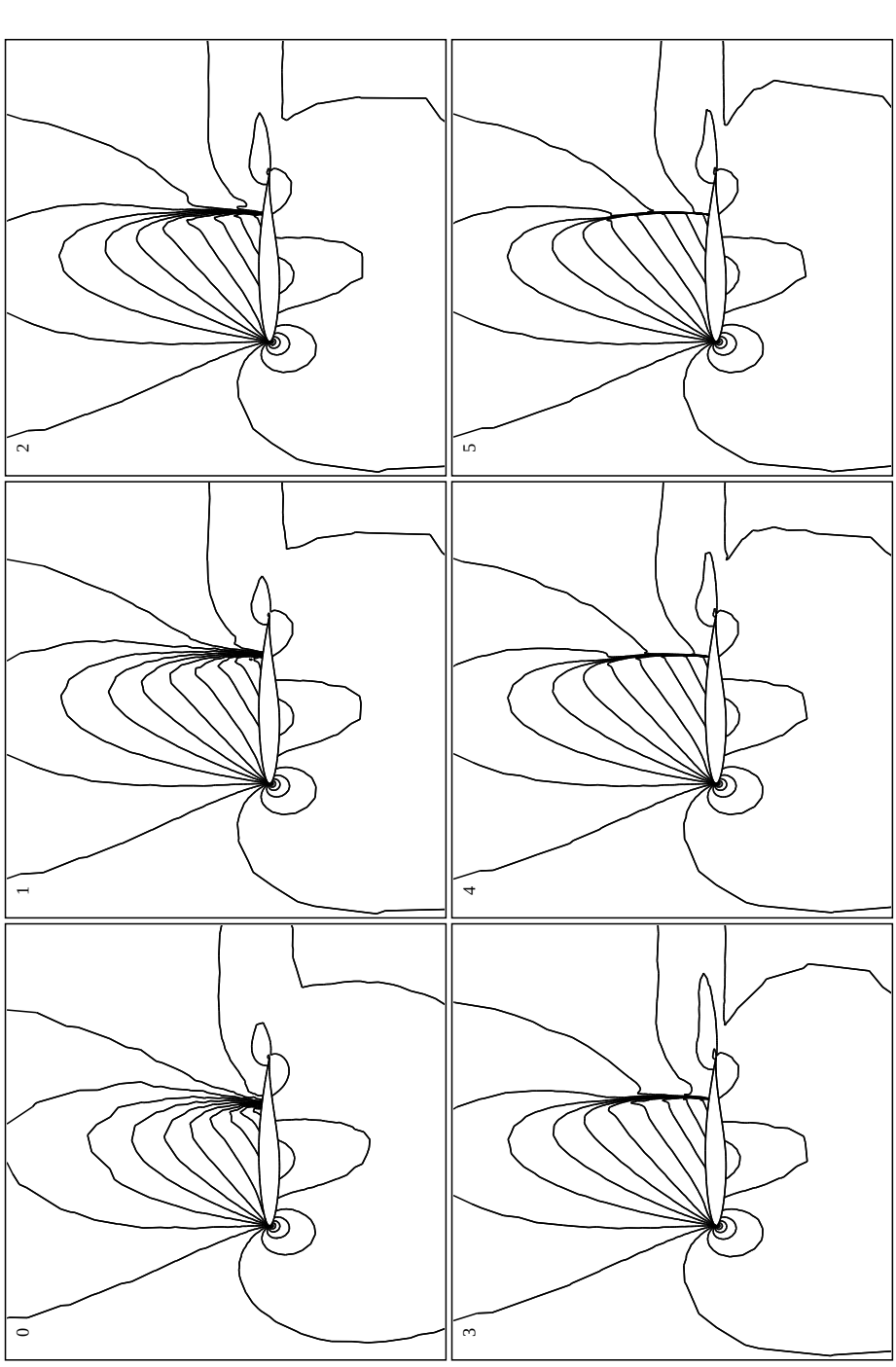
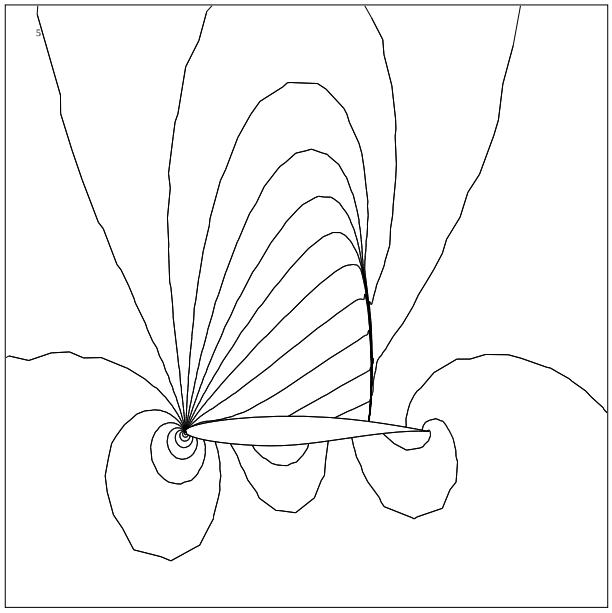


Figure 21: Mach contours on adapted grids for RAE-2822

Pressure



Mach number

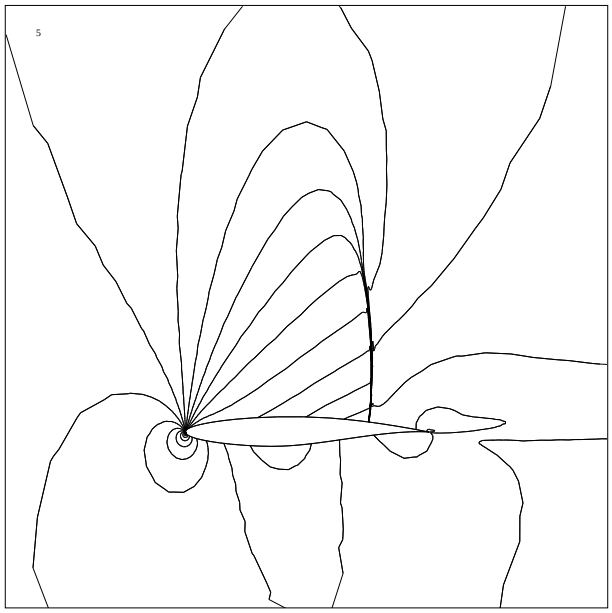


Figure 22: Pressure and Mach contours on final mesh for RAE-2822