INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Fluid Dynamics Computation with NSC2KE An User-Guide Release 1.0

Bijan Mohammadi

N° RT-0164

Mai 1994

PROGRAMME 6

Calcul scientifique,

modélisation

et logiciel numérique

Fluid Dynamics Computation with NSC2KE
An User-Guide
Release 1.0

Bijan Mohammadi

Programme 6 — Calcul scientifique, modélisation et logiciel numérique
Projet MENUSIN

Rapport technique n° RT-0164 — Mai 1994 — 70 pages

Abstract: NSC2KE is a Finite-Volume Galerkin program computing 2D and axisymmetric flows on unstructured meshes. To solve the Euler part of the equations, a Roe, an Osher and a Kinetic solvers are available. To compute turbulent flows a $k-\varepsilon$ model is available. Near-wall turbulence is computed either by wall-laws or by a two-layer approach. Time dependant problems can also be considered as a fourth order Runge-Kutta solver has been used.

Key-words: CFD, Compressible Navier-Stokes Equations, Turbulence, $k-\varepsilon$ Two-equation Model, Two-layer Approach, Wall-Laws.

(Résumé : tsvp)

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
Téléphone : (33 1) 39 63 55 11 – Télécopie : (33 1) 39 63 53 30

Simulation de la Dynamique des Fluides avec NSC2KE
Notice d' utilisation
Version 1.0

Résumé : NSC2KE est un solveur Volumes-Finis-Galerkin pour le calcul d'écoulements 2D et axisymetriques utilisant des maillages non-structurés. Pour résoudre la partie Euler des equations, le flux de Roe, de Osher et un flux cinetique sont disponibles. Pour le calcul d'écoulements turbulents un modèle $k-\varepsilon$ est disponible. La modélisation à bas nombre de Reynolds s'effectue soit par une technique bi-couche, soit à travers des lois de paroi. L'utilisation d'un shéma Runge-Kutta à 4 pas permet aussi la résolution de problèmes instationnaires.

Mots-clé : CFD, Equations de Navier-Stokes Compressibles, Turbulence, Modèle à deux Equations $k-\varepsilon$, Approche Bi-couche, Lois de Paroi.

Distribution Rights

For more informations, please write to:

Bijan MOHAMMADI
INRIA-MENUSIN
Domaine de Voluceau
BP 105,
78153 Le Chesnay, Rocquencourt
FRANCE.
Email : Bijan.Mohamadi@inria.fr
Tel: (33).(1).39-63-55-11
Fax: (33).(1).39-63-53-30

Contents

1 Introduction

Major advances in computer technologies and numerical techniques have made possible to propose an alternative or at least a complementary approach to the classical technique of laboratory experiments. More and more, Computational Fluid Dynamics becomes part of the design process. In fact, even if it does not remove the necessity of doing experiments by realizing prototypes, the CFD would help to reduce the number of cases to test.

Unstructured Finite Volume Galerkin

The NSC2KE fluid dynamics software uses a combinasion of the finite-volume and finite-element methods to simulate a wide range of flow fields going from subsonic to hypersonic regime for two-dimensional and axisymmetric configurations. As real configurations often involve complex geometries, unstructured meshes seem to be the best tool to correctly approach the reality.

Pre and Post-Processing

The only pre-processing task is to define a mesh. The natural pre-processing tool for NSC2KE is the automatical mesher EMC2 of F.Hecht and E.Saltel [HS]. The version 2.0 of NSC2KE will allow the user to define meshes using a language interpretor and an automatic mesh generator. NSC2KE generates one and two dimensional graphics which can directly be visualize using the well-known GNUPLOT graphic tool.

Class of Problems

NSC2KE simulates a wide range of 2D and axisymmetric flow fields including

- External and Internal Flows.
- Subsonic to Hypersonic Inviscide Flows: At the time of writing, the code have been tested for Mach number between 0.1 and 10.
- Low-Reynolds Subsonic to Hypersonic Viscous Flows.
- Subsonic to Hypersonic fully separated Turbulent Flows.
- Steady and Unsteady Flows.

2 The Governing Equations

Let ρ be the density, $u = (u_1, u_2)$ the velocity, T the temperature, $E = T + \frac{\|u\|^2}{2}$ the total energy, $p = (\gamma - 1)\rho T$ the pressure, $\nabla u = u_{i,j}$ the gradient of u, $D = u_{i,i}$ its divergence, $S = (\nabla u + \nabla u^t) - \frac{2}{3} DI$ the deformation tensor.

2.1 The Navier-Stokes Equations

The Navier-Stokes equations in the nondimensional form are:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) = 0$$

$$\frac{\partial \rho u}{\partial t} + \nabla \cdot (\rho u \otimes u) + \nabla p = \nabla \cdot ((\mu + \mu_t) S) \tag{2.1}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot ((\rho E + p) u) = \nabla \cdot ((\mu + \mu_t) S u) + \nabla ((\kappa + \kappa_t) \nabla T)$$

with

$$\kappa = \frac{\gamma \mu}{Pr}, \quad \kappa_t = \frac{\gamma \mu_t}{Pr_t},$$

$$\gamma = 1.4, \quad Pr = 0.72 \quad \text{and} \quad Pr_t = 0.9.$$

Where $\mu = 1/Re_{lam}$ is the inverse of the laminar Reynolds number given by Sutherland law

$$\mu = \mu_\infty \left(\frac{T}{T_\infty}\right)^{1.5} \left(\frac{T_\infty + 110.}{T + 110.}\right),$$

where ∞ denotes reference quantities.

For turbulent application, $\mu_t = 1/Re_t$ is the inverse of the turbulent Reynolds number. In what follows, we will call μ and μ_t the laminar and turbulent viscosity.

2.2 The High-Reynolds $k - \varepsilon$ Model

The high-Reynolds $k - \varepsilon$ equations are

$$\frac{\partial \rho k}{\partial t} + \nabla.(\rho u k) - \nabla((\mu + \mu_t) \nabla k) = S_k, \tag{2.2}$$

$$\frac{\partial \rho \varepsilon}{\partial t} + \nabla.(\rho u \varepsilon) - \nabla((\mu + c_\varepsilon \mu_t) \nabla \varepsilon) = S_\varepsilon. \tag{2.3}$$

The right hand sides of (2.2)-(2.3) contain the production and the destruction terms for ρk and $\rho \varepsilon$:

$$S_k = \mu_t P - \frac{2}{3} \rho k D - \rho \varepsilon, \tag{2.4}$$

$$S_\varepsilon = c_1 \rho k P - \frac{2c_1}{3c_\mu}\rho\varepsilon D - c_2 \rho \frac{\varepsilon^2}{k} \tag{2.5}$$

where $c_\mu, c_1, c_2, c_\varepsilon$ are respectively $0.09, 0.129, 1.83, 0.07$.

The eddy viscosity is given by:

$$\mu_t = c_\mu \rho \frac{k^2}{\varepsilon}. \tag{2.6}$$

By definition, $P = R : \nabla u$ which leads to the following expression for P in 2D.

$$P = (u_{1,2} + u_{2,1})^2 + \frac{4}{3}(u_{1,1}^2 + u_{2,2}^2 - u_{1,1}u_{2,2}). \tag{2.7}$$

In [MO3], we have shown that this choice leads to an overprediction of the eddy viscosity by the model and that a better choice is to drope terms in D in (2.2)-(2.3) and to only keep the shear-based components in (2.7). More precisely, we take for P

$$P = (u_{1,2} + u_{2,1})^2. \tag{2.8}$$

2.3 The Low-Reynolds Model

The classical $k - \varepsilon$ model is valid under the hypothesis that the local Reynolds number is high. Therefore, it is not adequate to describe regions close to a solid wall. NSC2KE proposes two alternative to deal with this problem. A two-layer approach [PA1] [MO1] [MO2] and a classical wall-laws technique. We recommand to use as often as possible the first approach because it enables the user to compute the flow up to the wall with almost no empirism. Of course, this requires more computational ressources as a finer mesh should be used.

2.3.1 The Two-Layer Approach

The idea is to couple the $k - \varepsilon$ model to a one-equation model in an automatical way. This is done by introducing a local Reynolds number y^+ and to compute for $y^+ < 200$, k using the following transport equation:

$$\frac{\partial \rho k}{\partial t} + \nabla.(\rho u k) - \nabla.((\mu + \mu_t)\nabla k) = \mu_t P - D_{iss}, \tag{2.9}$$

where

$$D_{iss} = \rho \frac{k^{\frac{3}{2}}}{l_\varepsilon}, \tag{2.10}$$

and

$$\mu_t = c_\mu \rho \sqrt{k} l_\mu. \tag{2.11}$$

l_μ and l_ε are two length scales containing the damping effects in the near wall regions.

$$l_\mu = \kappa c_\mu^{-3/4} y (1 - exp(\frac{-y^+}{c})),$$

$$l_\varepsilon = \kappa c_\mu^{-3/4} y(1 - exp(-\frac{y^+}{2\kappa c_\mu^{-3/4}})),$$

where $c = 70$ and $\kappa = 0.41$ and y^+ is defined by

$$y^+ = \frac{\sqrt{k\rho\rho_w}\, y}{\mu_w},$$

where w subscript means computed at the closest point of the wall and y is the distance of the current point to this point. In (2.9), P is given by (2.8).

2.3.2 Wall-Laws Technique

The most widely used approach when dealing with the near-wall regions difficulties is to avoid solving the Navier-Stokes equations (including the turbulence model) up to the wall. Instead, the edge of the computational domain is placed a small distance δ away from the wall in the high Reynolds number region (fully developed turbulent region). Empirical laws are then used to define boundary conditions at the edge of the domain (the subscript w and δ would mean at the wall and at the artificial wall boundary). If we denote $\vec{\tau}$ a unit vector tangential to the boundary Γ and $\vec{n}$ the unit outward normal to Γ, the boundary conditions for the mean velocity are:

$$\vec{u} \cdot \vec{n} = 0, \tag{2.12}$$

$$(\mu + \mu_t) S \cdot \vec{n} = -\rho u_\tau^2 \vec{\tau}. \tag{2.13}$$

These conditions are taken into account in the weak formulation of the problem.

The friction velocity u_τ is evaluated using $u_\tau = \frac{u}{y^+}$, if $y^+ \leq 10$ and from $u_\tau = u/(\frac{1}{\kappa} log(y^+) + \beta)$ if $y^+ \geq 10$ (with $\kappa = 0.41, \beta = 5.5$). Once u_τ is computed, the boundary conditions for k and ε are :

$$k_\delta = \frac{u_\tau^2}{\sqrt{c_\mu}}, \quad \varepsilon_\delta = \frac{u_\tau^3}{\kappa\delta}. \tag{2.14}$$

2.4 The global Formulation

We consider the following form of the conservation Navier-Stokes equations (2.1) and $k - \varepsilon$ equations without source terms

$$\frac{\partial W}{\partial t} + \nabla.(F(W) - N(W)) = 0 \tag{2.15}$$

where W is the vector of the conservation variables (i.e. $W = (\rho, \rho u_1, \rho u_2, \rho E, \rho k, \rho\varepsilon)^t$), F and N represent respectively the advective and viscous operators.

2.5 Boundary Conditions

The boundary conditions available in NSC2KE are rather simple. As we have tried not to make a too complicate code, a complete but not exhaustive set of boundary conditions is available which let the user to compute a wide range of flow fields including internal and external flows.

Solid Bodies

For laminar computations, the classical boundary condition for the velocity is namely the no-slip boundary condition $\vec{u} = 0$ while for the temperature either a Newmann or Dirichlet b.c. can be used dependind on the physic of the problem.

For turbulent computations, the user can use either wall laws or a two-layer approach. In the first case a weak boundary condition is used for the velocity and Newmann condition for the temperature and non-homogenous Dirichlet b.c. for k and ε. In the second case, the same conditions as before are applied for u and T while homogenous Dirichlet condition is used for k and Neumann for ε.

Inflow and Outflow Weak Boundaries

NSC2KE treats infinite boundaries by a characteristic technique. This means that along these boundaries the fluxes are splitted in positive and negative parts following the sign of the eigenvalues of the jacobian A of the convective operator F.

$$\int_{\Gamma_\infty} F.n d\sigma = \int_{\Gamma_\infty} (A^+ W_{in} + A^- W_\infty).n d\sigma,$$

where W_{in} is the computed (internal) value at the previous iteration and W_∞ the external value, given by the flow configuration through file **DATA**. More details can be found in [SW1].

Inlet Profiles

In internal flow computations, we often have to give an inlet profile coming either from experiences or from another computation. NSC2KE enables the user to give such a profiles.

Symmetry or Slip Boundaries

Non penetration ($\vec{u}.\vec{n} = 0$) boundary condition is necessary for Euler computations or sometimes in viscous cases. This condition is applied on solid boundaries for Euler computations and on symmetry lines depending on the physic of the problem.

2.6 External Forces

At this time, NSC2KE only takes into account of gravitational forces. This is done by adding

$$-\frac{\rho}{Fr}$$

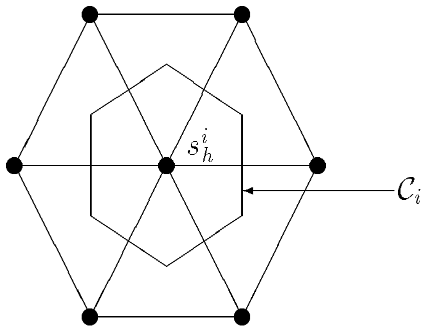
to the right hand side of the third equation (i.e. ρu_2). Here Fr is the Froude number.

3 Numerical Techniques

Our algorithm is explicit in time. To reach the steady solution an iterative scheme is used. The Navier-Stokes and $k-\varepsilon$ equations are solved by a Finite Volume-Galerkin upwind technique using an Osher [OS1], Roe [RO1] or Kinetic [PE1] Riemann solver for the convective part of the equations. The viscous terms are treated using a standard Galerkin technique.

3.1 The Navier-Stokes Solver

Let $\Omega_h = \cup_j T_j$ be a discretization by triangles of the computational domain Ω and let $\Omega_h = \cup_i C_i$ be its partition in cells.



Thus, we can associate to each $w_h \in V_h$, where V_h is the set of the continuous P^1 functions on our triangulation, a w'_h piecewise constant function on cells by

$$w'_h|C_i = \frac{1}{|C_i|}\int_{C_i} w_h.$$

Conversely, knowing w'_h piecewise constant, w_h is obtained as $w_h(S_i) = w'_h|C_i$.

If we suppose that F varies linearly on each triangle, the weak formulation of (2.15) involving V_h:

Find $W_h \in (V_h)^6$ such that, $\forall \phi_h \in V_h$

$$\int_\Omega \frac{\partial W_h}{\partial t}\Phi_h - \int_\Omega (F_h - N_h)(W_h)\nabla(\phi_h) + \int_{\partial\Omega}(F_h - N_h)\cdot n\phi_h = 0 \tag{3.16}$$

is equivalent to the following weak formulation obtained by taking in the convective part of (3.16) for ϕ_h the characteristic function of C_i and by using an explicit time integration:

$$|C_i|\frac{W^{n+1} - W^n}{\Delta t} + \int_{\partial C_i} F_d(W^n)\cdot n = R.H.S. \tag{3.17}$$

We use a centered scheme to compute the right hand side:

$$R.H.S. = -\int_{\Omega_h} N(W^n)\nabla(\phi_h) + \int_{\partial\Omega} N(W^n)\cdot n\phi_h.$$

Moreover, $F_d(W_h^n) = F(W_{\partial\Omega})$ on $\partial C_i \cap \partial\Omega$ and elsewhere F_d is a piecewise constant approximation of $F(W)$ verifying

$$\int_{\partial C_i} F_d \cdot n = \sum_{j\neq i} \Phi(W'|_{C_i}, W'|_{C_j}) \int_{\partial C_i \cap C_j} n. \tag{3.18}$$

We compute Φ using the following formula

$$\Phi(u,v) = \frac{1}{2}(F(u)+F(v)) - d(u,v)$$

where d is the numerical diffusion. In NSC2KE, the user can choose between the Roe [RO1], Osher [OS1] and Kinetic [PE1] propositions for d.

The previous formulation is only first order accurate in space. Spatial second order accuracy is obtained by using a MUSCL like extension involving combinaisons of upwind and centred gradients. More precisely, let ∇W_i be an approximation of the gradient of W at node i. We define on the segment $[i,j]$ the following quantities

$$W_{ij} = W_i + 0.5Lim(\beta(\nabla W)_i\vec{ij}, (1-\beta)(W_i - W_j))$$

and

$$W_{ji} = W_j - 0.5Lim(\beta(\nabla W)_j\vec{ij}, (1-\beta)(W_j - W_i) \tag{3.19}$$

with Lim a Van Albada type limitor [VAN1]

$$Lim(a,b) = 0.5(1+sgn(ab))\frac{(a^2+\alpha)b+(b^2+\alpha)a}{a^2+b^2+2\alpha} \quad \text{with} \quad 0<\alpha<<1$$

β contains the amount of upwinding at each point. Now, the second order accuracy in space is obtained by replacing W_i' and W_j' in (3.18) by W_{ij} and W_{ji}.

This approach does not guarantee the positivity of ρk and $\rho\varepsilon$, therefore the convective fluxes for the turbulent equations are computed using the scheme proposed in [LA1] for positivity preserving for chemical species.

More precisely, once the density fluxes computed, the turbulent convective fluxes are deduced by

$$\int_{\partial C_i \cap C_j} \rho uk.n = k_i(resp.k_j)\int_{\partial C_i \cap C_j} \rho u.n \quad \text{if} \quad \int_{\partial C_i \cap C_j} \rho u.n > 0(resp. < 0).$$

The source terms of the $k-\varepsilon$ model have been taken into account in an explicit way. For low-Reynolds points ε is deduced from k using (2.10).

3.2 The Time Integration Procedure

Lets rewrite (3.17) as

$$\frac{\partial W}{\partial t} = RHS(W).$$

In (3.17) the time integration has been done by an explicit scheme. NSC2KE rather uses the following fourth stage Runge-Kutta scheme [DE1][LAL1].

$$W^0 = W^n$$

$$W^k = W^0 + \alpha_k \Delta t RHS(W^{k-1}) \quad \text{for} \quad k = 1, .., 4$$

$$W^{n+1} = W^4$$

where the optimum choice for α_k [LAL1] is the following

$$\alpha_1 = 0.11, \alpha_2 = 0.2766, \alpha_3 = 0.5, \alpha_4 = 1.0.$$

This scheme enables us to also compute time dependent flows.

3.3 The Local Time Stepping Procedure

NSC2KE uses the following formula to compute the local time step for a given node s_i

$$\Delta t(s_i) = min(\frac{\Delta x}{|u| + c}, \frac{\rho Pr \Delta x^2}{2(\mu + \mu_t)}) \tag{3.20}$$

where Δx is the minimum height of the triangles having the node s_i in common. As this result is a generalization of a 1D stability analysis, we will see that in fact it is too strong. For instance, we will see through examples that often for a Navier-Stokes computation, the Euler time step also works. This means we can take only the first term in (3.20). But this is not always true, therefore the user should be carefull when choosing the time step strategy. So, for computing steady state solution, the user should always use a local time stepping strategy while for unsteady solution of course a global time step is necessary. In the latter case, the time step is the minimum of the local time steps.

3.4 Axisymmetric Corrections

The differences between the axisymmetric and 2D weak form of the Navier-Stokes equations we use come from the RHS $H(W)$ which appear in 7.21 (see appendix 1) and from the infinitesimal area in the integrals ($rdrdz$ instead of $dxdy$) [MO2]. So, to adapt the solver to axisymmetric computations, we multiply the cell areas $|C_i|$, the triangle areas $|T_i|$ and the edge lengths $|E_i|$ by some radius r obtained from the radius of the nodes r_i:

- $|C_i|$ becomes $|C_i| * r_i$ where r_i is the second coordinate of the node s_i.

- $|T_i|$ becomes $|T_i| * \frac{r_1+r_2+r_3}{3}$ with r_1, r_2, r_3 the radius corresponding to the vertices of the triangle T_i.
- $|E_i|$ becomes $|E_i| * \frac{r_1+r_2}{2}$ where r_1 and r_2 are the radius of the extremities of E_i.

The other modifications come from the additional source terms occuring in the equation 7.21. More precisely, for the momentum equation in u_z, we add to the right hand side:

$$-\int_{\partial C_i} \frac{2}{3}\mu_{tot} u_r n_z r d\gamma.$$

In the same way, to the RHS of the momentum equation in u_r we add:

$$\int_{C_i} \frac{P}{r}.rdrdz - \int_{C_i} \frac{\tau_{\theta\theta}}{r} r.drdz - \int_{\partial C_i} \frac{2}{3}\mu_{tot} u_r n_r d\gamma.$$

Finally, to the RHS of the energy equation we add:

$$-\int_{\partial C_i} \frac{2}{3}\mu_{tot} u_z u_r n_z r d\gamma - \int_{\partial C_i} \frac{2}{3}\mu_{tot} u_r^2 n_r r d\gamma.$$

4 The Code Description

This section is to give an idea of what might be a right choice of the parameters for a given computation when using NSC2KE.

The code is on non-dimensional form. To define the computational configuration, the user have to define two adimensional quantities, namely the Mach and the Reynolds numbers, which will describe the flow field. To better understand the parameters required by the code, several test cases have been reproduced at the end of this manual. We give an example of file **DATA** in appendix.

4.1 The Scheme Choice

The user has the possibility of defining the kind of solver he likes to use. At the time of writing, Roe, Osher and kinetic solvers are available for the Euler part of the equations while for the viscous part only a centred scheme is available. There is no rule on how to choose the convective solver. For subsonic or slightly transonic flows all three solvers can be used. But with the Mach number growing, the Roe solver will not be robuste enough.

The user can choose the solver he wants through the parameter **iflux** in the file **DATA**.

- iflux = 1 Roe solver,
- iflux = 2 Osher solver,
- iflux = 3 Kinetic solver.

The version 2.0 of NSC2KE will propose a widest range of scheme including an hybrid scheme (combination of Osher and Kinetic) [COQ1] and an SUPG scheme.

4.2 The Spatial Precision of the Scheme

NSC2KE proposes schemes with first and second order spatial precision. The user has to choose the precision of the scheme. This is done through the parameter **nordre**

- nordre = 1 first order scheme,
- nordre = 2 second order scheme,
- nordre = 3 second order scheme with Van Albada limitor (3.1).

The rule is to start the computation with first order scheme and to switch to second order after a while. For low Mach number flows, the user can try the complete second order scheme without limiting process. Unfortunately, this is not viable when the Mach number is higher.

Remark:

The previous choices are analyzed through the shock-tube problem as tutorial example.

4.3 The Turbulence Model

As we said earlier, the user can choose between a two-layer approach or a more classical wall laws technique. The best choice is to use the two-layer approach, as it gives the flow description up to the wall with almost no empirism. This is in fact a question of computational time as the user has to use a much finer grid in this case.

4.4 Boundary Conditions Implementation

In this section we describe how the user can apply different boundary conditions presented in the previous section at different points of the mesh.

Each node pocesses an integer parameter, called **logic**, which define the kind of boundary condition the user want to assign to this node.

These logics are available in the code

- **logic** = 0 denotes an internal node,
- **logic** = 2 denotes a symmetry or slipping boundary point,
- **logic** = 3 denotes a solid boundary point,
- **logic** = 4 denotes an outflow boundary point,
- **logic** = 5 denotes an inflow boundary point,
- **logic** = 6 denotes an inlet forced profile point.

LOGIC = 2

If the user choose **logic** = 2 for a node the code will consider this node as being part of a slipping boundary and will introduce the following boundary conditions in the weak formulation

$$\vec{u}.\vec{n} = 0.$$

LOGIC = 3

If the user choose **logic** = 3 for a node the code will consider this node as being part of a no-slipping boundary and will introduce the following boundary conditions in the weak formulation

$$\vec{u} = 0 \quad \text{and} \quad T = T_w \quad \text{if} \quad \text{iecc} = 0,$$
$$\vec{u} = 0 \quad \text{and} \quad \frac{\partial T}{\partial n} = 0 \quad \text{if} \quad \text{iecc} = 1.$$

iecc is a parameter in file **DATA** which enables the user to ask the body to be isothermal or adiabatic. For turbulent computations, the previous rule is true when the two-layer approach is used. When wall-laws are used, only adiabatic boundary condition is available for the temperature and for the velocity, conditions (2.12)-(2.13) are applied.

LOGIC = 4 or LOGIC = 5

If the user choose **logic** = 5 or 4 for a node the code will consider this node as being part of an inflow or outflow boundary. These boundaries are treated by a Stegger-Warming [SW1] approach which consists of using information from outside following the sign of the eigenvalues of the jacobian of the convective operator.

LOGIC = 6

If the user choose **logic** = 6 for a node, the code will consider this node as being part of a boundary where the user wants all the variables remaining unchanged during the computation (for instance a given inlet profile). This is specially usefull when defining an initial condition through the files **INIT_NS** and **INIT_KE**.

4.5 The Mesh :

In a computation, the first point is the definition of the computational domain. This is done by giving a file containing the grid points and their **logic**. We also need a connectivity table, describing how these points are connected between them. NSC2KE reads the mesh using the following format from the file **MESH**:

```
open(1,file='MESH',status='old')
read(1,*) ns,nt   !number of points, number of triangles
do i=1,ns  !loop on the mesh points
read(1,*) ii,x(i),y(i),logic(i) !ii is the number of the node
enddo
do it=1,nt !loop on the triangles
read(1,*) ii,(conexion(i,it),i=1,3) !ii is the number of triangle
enddo
close(1)
```

As we have said, this can be done using EMC2 [HS1]. The version 2.0 of NSC2KE will allow the user to define meshes without passing by an external support and this by using a language interpretor and an automatic mesh generator.

4.6 Initial Conditions

To initialize the computation, NSC2KE proposes two strategies. The user can either start the computation from an uniform state defined by the inflow conditions (Inflow Mach number and temperature through file **DATA**) or restart from a solution obtained before or defined by the user. In this case, NSC2KE reads the initial condition for the conservation mean flow variables from file **INIT_NS** using the following format

```
open(1,file='INIT_NS',status='old')
do is=1,ns  !loop on the mesh points
read(1,*) (var(i,is),i=1,4) !read conservation variables
enddo
close(1)
```

where $var = (\rho, \rho u, \rho v, \rho E)^t$ is the vector of conservation variables.

To use the $k-\varepsilon$ turbulence model, again the user can either start from an initial uniform state for k and ε or to use a previous solution from **INIT_KE** using the following format

```
open(1,file='INIT_KE',status='old')
do is=1,ns  !loop on the mesh points
read(1,*) (vart(i,is),i=1,4) !read turbulent quantities
enddo
close(1)
```

where $vart = (\rho k, \rho\epsilon, \mu_{total}, \mu_{turb})^t$. Therefore, for particular applications, the user can write a program defining these files and using **logic = 6** for nodes where he wants to keep a particular profile.

4.7 Visualization

NSC2KE produces 1D and 2D outputs for GNUPLOT. The choice of GNUPLOT is related to the fact that it is one of the most accessible visualization tool in research labs. Quantities like pressure, friction and heat flux distributions on the body or the L^2 norm of the residual during the computation are 1D curves. On the other hand, the mesh, the velocity field, the Mach number and the pressure distribution in the flow are 2D pictures. The user can visualize these quantities by typing

$$gnuplot < GNU.DATA.$$

To give an idea, **GNU.DATA** might be of the following form:

```
#set term postscript
set term x11
set title "PRESSURE COEFFICIENT"
set xlabel "X"
plot [0.:1.][0:2.] "WALL.DATA"
pause 50
set title "FRICTION COEFFICIENT"
set xlabel "X"
plot [0.:1.][-3.:3.] "WALL.DATA" using 1:3
pause 50
set title "MESH"
set xlabel "X"
plot [-1:3][-1.2:1.2] "GNU.MESH" w l
pause 50
set title "P/PINF"
plot [-1:3][-1.2:1.2] "GNU.PRES" w l
pause 50
```

```
set title "VELOCITY FIELD"
plot [-1:3][-1.2:1.2] "GNU.VECT" w l
pause 50
set title "MACH NUMBER"
plot [-1:3][-1.2:1.2] "GNU.MACH" w l
pause 50
set title "NORMALIZED L2 NORM OF THE RESIDUAL"
set xlabel "ITERATION"
plot [0:3000][0.:1.0]"RESIDUAL"  w l
pause 25
```

The files used here by GNUPLOT are directly produced by NSC2KE. All the pictures presented at the end of this manual have been obtained by this way (after puting in **postscript** format).

Of course, the user can (should) adapt the output to his more specific post-processing tool (if any).

GNUPLOT is a graphic tool developped by Thomas Williams and Colin Kelley.

For more informations write to info-gnuplot@dartmouth.edu.

5 Tutorial Examples

In this section, we will present some configurations of external and internal flows computed with NSC2KE. The range of the Mach number goes through examples from subsonic to hypersoinc. The aim of this chapter is that the user, by trying these examples, becomes familiar with NSC2KE.

All the meshes used for these computations have been obtained using EMC2, the automatic mesher of F. Hecht and E. Saltel [HS] developped at INRIA.

All the pictures presented here have been obtained using GNUPLOT from the standard output of NSC2KE.

The following configurations will be presented:

- The Sod Shock-Tube : a Comparison Between the Schemes,
- Flat Plate Turbulent Boundary Layer,
- Natural Convection in a Square Cavity,
- Inviscide Transonic Flow over a NACA 0012 : The Fish Tail,
- Subsonic Unsteady Turbulent Flow over a NACA 0012,
- Transonic Turbulent Flow over a 2D Bump: The λ-Shock,
- Axisymmetric Transonic Turbulent SOC Projectile,
- Axisymmetric Supersonic Turbulent Air Intake,
- Hypersonic Turbulent Compression Corner.

All these computations except the second one have been done without the influence of the gravity.

5.1 The Sod Shock-Tube

This test problem is due to Sod [SO1]. We compute the solution at time $T = 0.25$ starting from the following initial condition

$$\rho = 1, \quad u = 0, \quad p = 1 \quad \text{if} \quad 0 \leq x \leq 0.5,$$

$$\rho = 0.125, \quad u = 0, \quad p = 0.1 \quad \text{if} \quad 0.5 < x \leq 1.$$

The computational domain is a tube of length 1 and of height 0.1. These computations have been done with 100 points in the x and 5 points in the y direction. Symmetry condition is assumed for the top and bottom boundaries (**logic** = 2). At the left boundary the data have been frozen (**logic** = 6) while free boundary condition is imposed at the right boundary (**logic**= 4).

Through this example, we will compare the Euler solvers (Roe, Osher and Kinetic) available in NSC2KE. Solutions obtained with different level of precision are presented. This case also shows the ability of NSC2KE to compute time dependant solutions.

Pictures 1, 2 and 3 show the distribution of the density, pressure and the velocity in the x direction using first-order schemes. As expected, the solutions are too smooth.

Pictures 4, 5 and 6 show the distribution of the density, pressure and the velocity in the x direction using the full second-order schemes. The fitting between the computed and the exact solution is quite satisfactory, but under and overshoots appear.

Pictures 7, 8 and 9 show the distribution of the density, pressure and the velocity in the x direction using the limited second-order schemes. As expected, the schemes are more diffusive when using limiters, but under and overshoots have disappeared.

5.2 Turbulent Flat Plate Boundary Layer

Through this test case, we will compare the two-layer and wall-laws techniques available in NSC2KE. The mesh used for this case has 1196 nodes and 2250 triangles. The first point is at 5.10^{-5} meter from the plate. This is too fine for the wall-laws computation but necessary for the two-layer technique. The plate has a length of one meter. The Reynolds number is 10^6 and the inflow Mach number is 0.1. We have compared in Fig. 10, the computed friction coefficients to the theoretical curv of $0.0375 Re_x^{-1/6}$. The velocity profiles through the boudary layer have been compared in Fig. 11.

5.3 Natural Convection in a Square Cavity

The geometry is a square cavity of side one meter. The left and right sides are isothermal and the bottom and top are adiabatic walls. The starting point is a given difference of temperature between isothermal walls and zero velocity is assumed in all the domain. In our case, the cold (left) wall is at a temperature of 300 Kelvin and the right wall is at 360 Kelvin. The Reynolds number is $Re = 10^4$ and the Froude number is one. This makes a Rayleigh number of order $Ra = Re(T_2 - T_1) = 6.10^5$.

The mesh has 1681 nodes and 3200 triangles. The results presented here are after 50000 iterations. The CFL number is 1.5, local Euler time stepping and the Roe solver have been used. Fig. 12, 13, 14 and 15 show the mesh, the velocity field, the normalized iso-temperature and the iso-Mach number contours. One of the difficulty of this case is the long time necessary for the solution to be established as the velocity is small and the flow is only driven by the temperature and gravity.

5.4 Inviscide Transonic Flow over a NACA 0012: The Fish Tail

This is a transonic inviscide case. A complexe shock structure, formed by two oblic shocks and another straight making a fish tail form, is present in the tail region. The computational configuration is as follow (defined through the file **DATA**):

- 2D Euler Computation,
- Inflow Mach number = 0.95,
- Angle of Attack = 0,
- Roe Euler Solver,
- Local Euler Time Stepping,
- C.F.L. = 1.5,
- Number of Time Step = 500.

There are other parameters in the file **DATA** but they do not influence this case. The mesh presented here have been obtaind after three local refinements. It has 2122 nodes and 4136 triangles. The mesh is shown in Fig. 16, the iso-pressure in Fig. 17 and the iso-Mach number lines in Fig. 18.

5.5 Subsonic Unsteady Turbulent Flow over a NACA 0012

This is an usteady subsonic case to show the ability of NSC2KE to also compute low-Mach number flows without being too long to get the result. In fact, some compressible solvers when used in low-speed cases take more time to converge to the steady state. In unsteady case, the situation is even worse as the time step should be global, not local when steady solutions are aimed. The mesh used has 2719 nodes and 5280 triangles. We present the results for time $T = 10$ (the length of the computational domain is 4.5 meters). This means we have done enough iterations to get there (approximately 10000 iterations). The wall-laws technique have been used for this case. If the two-layer technique were used, the situation would not be so simple as the global time step would be at least 10 times smaller and the number of mesh points 10 times higher. This give an idea of the difficulties existing in the computation of such flows.

The computational configuration is as follow (defined through the file **DATA**):

- 2D Navier-Stokes Computation,
- Inflow Mach number = 0.1,
- Reynolds number by meter = 10^6,
- Angle of Attack = 30,
- Roe Euler Solver,
- Global Euler Time Stepping,
- C.F.L. = 1,
- make enough time step to reach a Total Time of 10,
- $k - \varepsilon$ turbulence model,
- wall-laws technique with $\delta = 0.05m$.

The mesh is shown in Fig. 19, iso-pressure lines in Fig. 20, iso-Mach number lines in Fig. 21 and the turbulent kinetic energy contours in Fig. 22.

5.6 Transonic 2D Turbulent Bump: The λ-Shock

This is a classical test case for people doing turbulence modeling. The λ-shock is a characteristic of the turbulent nature of the flow and it is due to the recirculation present after the bump. This computation have been done using the $k - \varepsilon$ model with classical wall-laws technique. The mesh used has 5661 nodes and 11000 triangles.

The computational configuration is as follow (defined through the file **DATA**):

- 2D Navier-Stokes Computation,
- Inflow Mach number = 0.65,
- Reynolds number by meter = 1.3510^7,
- POUT / PIN = 0.635 the ratio between inlet and outlet pressures,
- Osher Euler Solver,
- Local Euler Time Stepping,
- C.F.L. = 1.,
- Number of time steps = 10000,
- $k - \varepsilon$ turbulence model,
- wall-laws technique with $\delta = 7.10^{-5}m$.

The mesh is shown in Fig. 23, iso-pressure lines in Fig. 24, iso-Mach number lines in Fig. 25 and the turbulent kinetic energy contours in Fig. 26.

5.7 Axisymmetric Transonic Turbulent SOC Projectile

The secant-ogive-cylinder (SOC) configuration is one of the classical models for artillery projectile. The difficulty in this computation is the presence of a blunt base and a complex shock structure on the projectile. For this case, we have used the Kinetic solver. In fact, due to the base configuration, difficulties appear with the pressure or density becoming too small in the base region. The Roe solver is not enough robuste. The user should try with the Osher solver. The computation has been started with first-order accuracy in space (**nordre** =1) before going to limited second-order (**nordre** =3) after 10000 iterations. The $k-\varepsilon$ model has been used with wall-laws technique. The computational configuration is as follow (defined through the file **DATA**):

- Axisymmetric Navier-Stokes Computation,
- Inflow Mach number = 0.96,
- Reynolds number by meter = 1.3510^7,
- Kinetic Euler Solver,
- Local Euler Time Stepping,
- C.F.L. = 0.1,
- Number of time steps = 20000,
- $k-\varepsilon$ turbulence model,
- wall-laws technique with $\delta = 0.05m$.

The mesh is shown in Fig.27. It has 3131 nodes and 6000 triangles. Iso-Mach number and pressure are shown in Fig. 28 and 29. Iso-k contours are given in Fig. 30.

5.8 Axisymmetric Supersonic Turbulent Air Intake

This is a fully turbulent supersonic flow. $k-\varepsilon$ model and wall-laws have been used. This enables us to use quite a coarse mesh. The mesh is shown in Fig. 31. It has 3111 nodes and 6000 triangles. The computational configuration is as follow (defined through the file **DATA**).

- AXI, Navier-Stokes Computation,
- Inflow Mach number = 2.,
- Reynolds number by Meter = 10^6,
- Osher Euler Solver,
- Global Euler Time Stepping,

- C.F.L = 1.0,
- $k - \varepsilon$ Turbulence Model,
- Wall-Law Technique with $\delta = 0.05$,
- Number of Time Step = 2000.

Iso-pressure lines are shown in Fig. 32 and iso-Mach number contours in Fig. 33. The turbulent kinetic energy distribution, iso-k contours is given in Fig. 34.

5.9 Hypersonic Turbulent Compression Corner

This test case consists of a 35° hypersonic compression corner. ch number is 5 and the Reynolds number $4.10^7/m$. The free stream temperature is $T_\infty = 83.6K$ and the wall temperature $T_w = 288K$. The two-layer approach have been used for this case. We aim to obtain a precise description of wall quantities (i.e. pressure, friction and heat flux coefficients) for which experimental results are available [DC1]. Another reason for using the two-layer approach is the presence of isothermal wall for which the wall-laws are inapropriate.

- 2D Navier-Stokes Computation,
- Inflow Mach number = 5,
- Reynolds number by Meter = 4.10^7,
- Inflow Temperature = 83.6 K,
- Wall Temperature = $288K$,
- Osher Euler Solver,
- Local Navier-Stokes Time Stepping,
- C.F.L = 0.1,
- $k - \varepsilon$ Turbulence Model,
- Two-Layer technique with $\delta = 0.01m$,
- Number of Time Step = 20000.

The mesh has 6630 nodes and 12898 triangles. A partial view of the mesh is given in Fig. 35. The corner is at $x = 0.25m$ from the leading edge. The first mesh spacing from the wall is $3.10^{-6}m$. This shows that a two-layer computation requires clearly more computational ressources.

The pressure nad Heat flux coefficients (Stanton number) are compared with experimental values in Fig. 37 and 39. The friction coefficient is shown in Fig. 38. The recirculation bubble corresponds to region of negative Cf. The wall quantities are defined by:

$$C_p = 2\frac{p - p_\infty}{\rho_\infty u_\infty^2},$$

$$C_f = \frac{2\tau_w}{\rho_\infty u_\infty^2},$$

and

$$S_t = \frac{-2(\kappa + \kappa_t)\nabla T.n}{c_{p\infty}(T_0 - T_w)\rho_\infty u_\infty}$$

where T_0 is the stagnation temperature and $c_{p\infty}$ is the specific heat at constant pressure at infinity.

In the same way, Iso-Mach contours are plotted in Fig. 36.

6 References

[COQ1] F. Coquel, M-S Liou, *Hybrid Upwind Splitting by a Field-by-Field Decomposition*, TO appear.

[DC1] J. Delery, M.C. Coet, *Experiments on shock-wave/boundary-layer interactions produced by two-dimensional ramps and three-dimensional obstacles*, Proc. on Workshop on Hypersonic Flows for Reentry Problems, Springer-Verlag, Antibes 22-26 jan. 1990.

[FE1] L. Fezoui, A. Dervieux, *Finite Element Non Oscillatory Schemes for Compressible Flows,* Comp. Math. and Applic. 8th France-U.S.S.R.-Italy Joint Sympos. Pavia. 1989.

[FE2] L. Fezoui, B. Stoufflet, *A Class of Implicit Upwind Schemes for Euler Simulations with Unstructured Meshes,* Journ. of. Comp. Phys., 84, pp. 174-206, 1989.

[HS] F. Hecht, E. Saltel, *EMC2, un Logiciel d'Edition de Maillages et de Contours Bidimensionnels*, INRIA technical report num. 118, 1990.

[LS1] B.E. Launder, D.B. Spalding, *Mathematical Models of Turbulence.* Academic Press, 1972.

[LAL1] M.H.Lallemand,*Schemas Decentrés Multigrilles pour la Résolution des Equations D'Euler en Eléments Finis*, Thesis, Univ. of Provence-Saint Charles, 1988.

[LA1] B. Larrouturou, *How to Preserve the Mass Fraction Positivity when Computing Compressible Multi-Component Flows,* INRIA report 1080, 1989.

[MO1] B. Mohammadi, *Complex turbulent flows computation with a two-layer approach* , 1992, Int. J. Num. Meth. Fluids, Vol. 15, 747-771.

[MO2] B. Mohammadi, J.H.Saiac, *Turbulent Compressible Axisymmetric flows Computation with the $k-\varepsilon$ Model,* Computational Fluid Dynamics, Vol. 1, pp.115-133.

[MO3] B. Mohammadi, O. Pironneau, *Two-Layer Approach and Compressibility Corrections for Hypersonic Flows*, to appear.

[OS1] S. Osher, S. Charravarthy,*Upwind Difference Schemes for the Hyperbolic systems of conservation laws'* mathematics of Computation, April 1982.

[PE1] B. Perthame,*Boltzmann Type Schemes for Gas Dynamics and Entropy Property,* SIAM Num. Anal. 27 (6), 1990.

[PI1] O. Pironneau, *Finite Element in Fluids*, Masson and Wiley Int. Eds. 1990.

[PI2] O. Pironneau, B.Mohammadi, *Analysis of the $k-\varepsilon$ Turbulence Model,* Masson and Wiley Int. Eds., 1994.

[RO1] P.L.Roe, *Approximate Riemann Solvers, Parameters Vectors and Difference Schemes,* J.C.P. Vol.43, 1981.

[SO1] Sod. *A Survey of Finite-Difference Methods for Systems of Nonlinear Hyperbolique Conservation Laws,* J. of Comp. Phys., Vol. 27, 1978.

[SW1] J. Steger, R.F. Warming, *Flux Vector Splitting for the Inviscid gas dynamic with Applications to Finite-Difference Methods*, J. Comp. Phys. 40, pp:263-293.
(1983).

[VAN1] G.D.Van Albada, B. Van Leer,*Flux Vector Splitting and Runge-Kutta Methods for the Euler Equations*, ICASE 84-27, June 1984.

7 Appendix 1: Axisymmetric Form of the Navier-Stokes equations

Let $\rho, u_z, u_r, E, p, T, k, \varepsilon$ be the density, the axial and radial components of the velocity, the total energy, the pressure, the temperature, the kinetic turbulent energy and the rate of dissipation of k. We suppose that the fluid motion is axisymmetric. That means that all of the θ-derivatives are zero. Moreover, the swirl or θ-component of the velocity is supposed to be zero. So, $W = (\rho, \rho u_z, \rho u_r, \rho E, \rho k, \rho\varepsilon)$ being the vector of conservation variables, in cylindrical coordinates (z, θ, r), the Navier-Stokes equations in conservation forms are:

$$\frac{\partial}{\partial t}(W) + \nabla.F(W) = \nabla.N(W) + H(W) \tag{7.21}$$

with

$$W = \begin{pmatrix} \rho \\ \rho u_z \\ \rho u_r \\ \rho E \\ \rho k \\ \rho\varepsilon \end{pmatrix},$$

$$F_z(W) = \begin{pmatrix} \rho u_z \\ \rho u_z^2 + p \\ \rho u_z u_r \\ (\rho E + p)u_z \\ \rho u_z k \\ \rho u_z \varepsilon \end{pmatrix},$$

$$F_r(W) = \begin{pmatrix} \rho u_r \\ \rho u_r u_z \\ \rho u_r^2 + p \\ (\rho E + p)u_r \\ \rho u_r k \\ \rho u_r \varepsilon \end{pmatrix},$$

$$N_z(w) = \begin{pmatrix} 0 \\ \tau_{zz} \\ \tau_{zr} \\ \kappa_{tot}\frac{\partial T}{\partial z} + u_z\tau_{zz} + u_r\tau_{zr} \\ (\mu + \mu_t)\frac{\partial k}{\partial z} \\ (\mu + c_\varepsilon\mu_t)\frac{\partial \varepsilon}{\partial z} \end{pmatrix},$$

$$N_r(w) = \begin{pmatrix} 0 \\ \tau_{rz} \\ \tau_{rr} \\ \kappa_{tot}\frac{\partial T}{\partial r} + u_z\tau_{zr} + u_r\tau_{rr} \\ (\mu+\mu_t)\frac{\partial k}{\partial r} \\ (\mu + c_\varepsilon\mu_t)\frac{\partial \varepsilon}{\partial r} \end{pmatrix},$$

and

$$H(w) = \begin{pmatrix} 0 \\ 0 \\ -\frac{\tau_{\theta\theta}}{r} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

with the following expressions for the tensor components:

$$\tau_{zz} = \mu_{tot}[2\frac{\partial u_z}{\partial z} - \frac{2}{3}(\nabla.u)],$$

$$\tau_{zr} = \mu_{tot}[\frac{\partial u_z}{\partial r} + \frac{\partial u_r}{\partial z}],$$

$$\tau_{rr} = \mu_{tot}[2\frac{\partial u_r}{\partial r} - \frac{2}{3}(\nabla.u)],$$

$$\tau_{\theta\theta} = \mu_{tot}[2\frac{u_r}{r} - \frac{2}{3}(\nabla.u)]$$

where

$$\nabla.u = \frac{\partial}{\partial z}u_z + \frac{1}{r}\frac{\partial}{\partial r}(ru_r),$$

$$p = (\gamma-1)\rho E, E = T + \frac{1}{2}u^2,$$

$$\kappa_{tot} = \mu\frac{\gamma}{Pr} + \mu_t\frac{\gamma}{Pr_t},$$

$$\mu_{tot} = \mu + \mu_t,$$

$$\gamma = 1.4, Pr = 0.72, Pr_t = 0.9.$$

μ is the viscosity of the fluid given by the Sutherland law's, κ_{tot} is the total thermal conductivity, P_r and Pr_t are respectively the laminar and turbulent Prandtl numbers.

8 Appendix 2: The file DATA

This is the file **DATA** used for the fish tail computation.

```
0      --> =0 2D, =1 AXISYMMETRIC
0      --> =0 Euler, =1 Navier-Stokes
1.e4   --> Reynolds by meter (mesh in meter)
1.     --> inverse of the Froude number (if = 0 no gravity)
0.95   --> inflow Mach number
1.     --> ratio pout/pin
1      --> wall =1 adiabatic wall, =2 isothermal wall
300.   --> inflow temp(in Kelvin) for Sutherland laws
300.   --> if isothermal walls, wall temp (in Kelvin)
0.0    --> angle of attack
1      --> Euler fluxes =1 roe, =2 osher,=3 kinetic
2      --> nordre = 1 first order, =2 second, =3 limited second
1      --> =0 global time (unsteady), =1 local Euler, =2 local N.S.
1.5    --> cfl
500    --> number of time step
500    --> frequence for the  solution to be saved
1.e10  --> maximum physical time for run (for unsteady problems)
-4.    --> order of magnitude for the residual (for steady problems)
0      --> =0 start with uniform solution, =1 restart from INIT_NS
cccc  turbulence  ccccccccccccccccccccccccccccccccccccccccccccccccccc
0      --> =0 no turbulence model, =1 k-epsilon model
0      --> =0 two-layer technique, =1 wall laws
1.e-2  --> delta in wall laws or limit of the one-eq. model. (in meter)
0      --> =0 start from uniform solution for k-eps, =1 from INIT_KE
```

We can see that it could happen that some parameters do not influence a given situation. For instance, for an inviscide computation, the Reynolds number will not be used even if it is given.

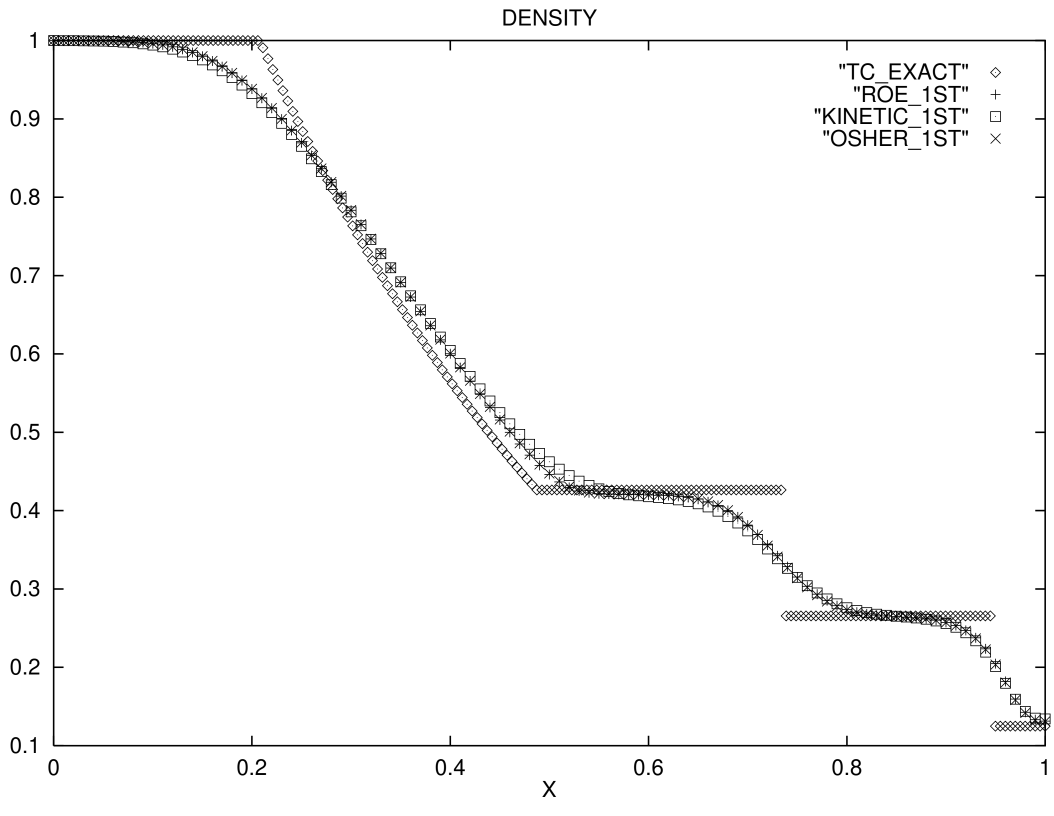


Figure 1: Shock Tube: Density distribution in the x direction, computed with first-order Roe, Osher and Kinetic schemes.

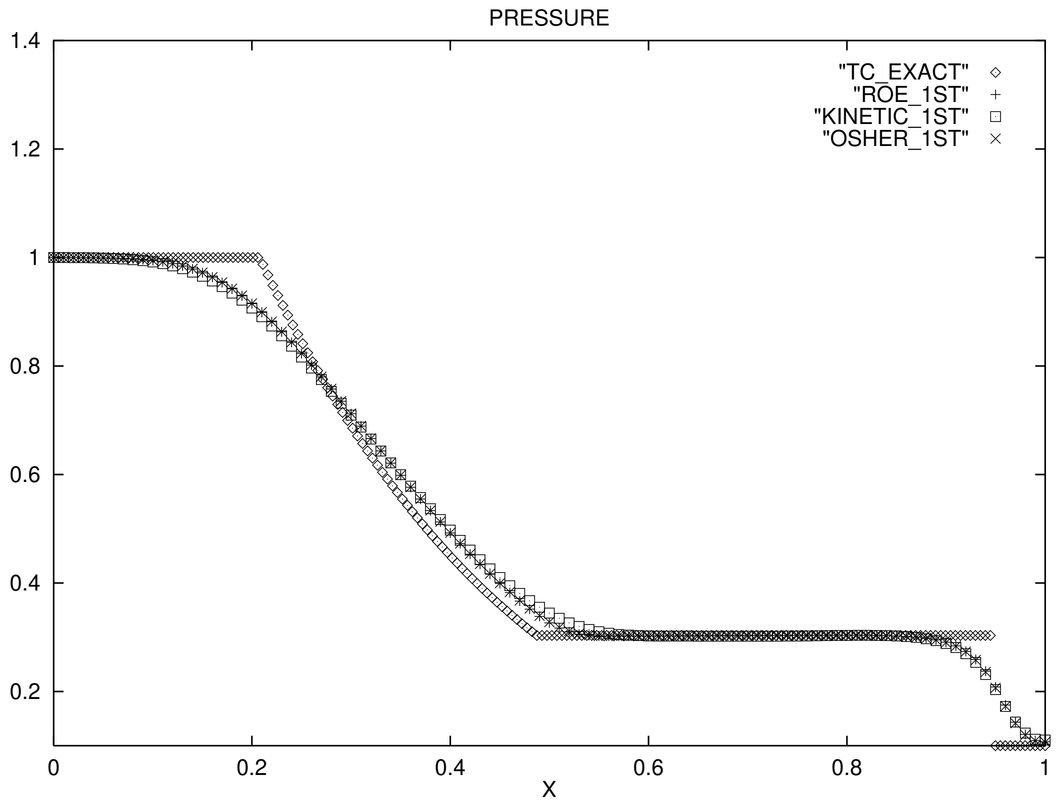


Figure 2: Shock Tube: Pressure distribution in the x direction, computed with first-order Roe, Osher and Kinetic schemes.

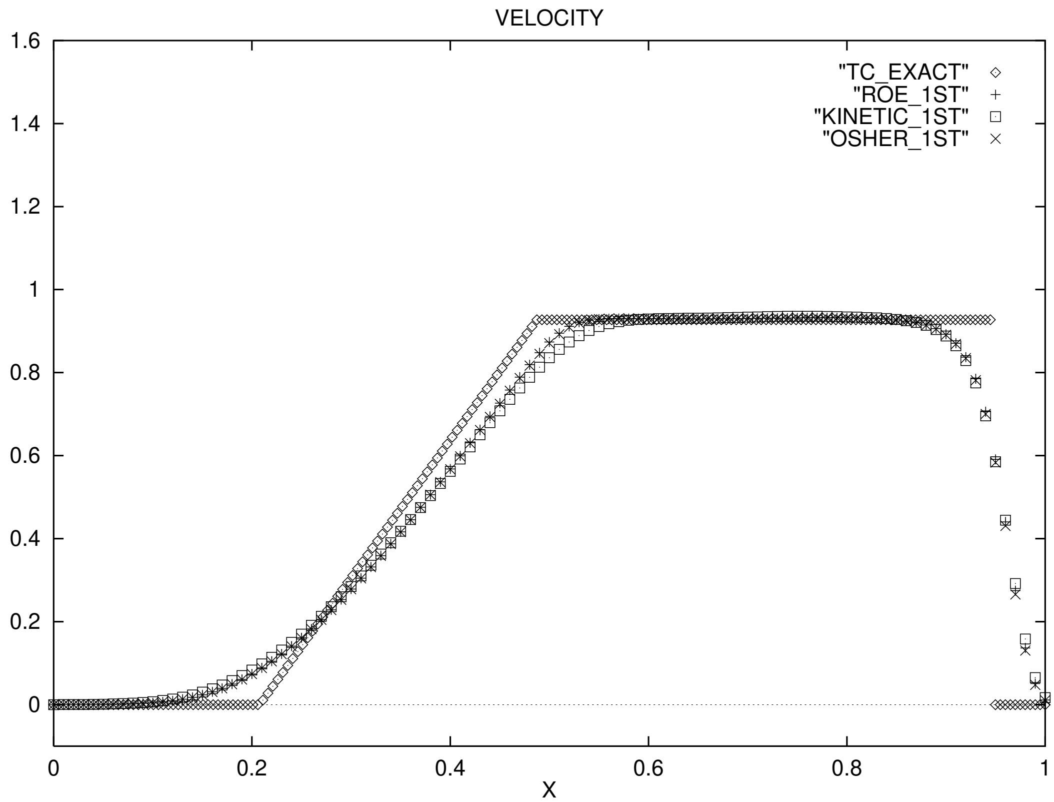


Figure 3: Shock Tube: Velocity distribution in the x direction, computed with first-order Roe, Osher and Kinetic schemes.

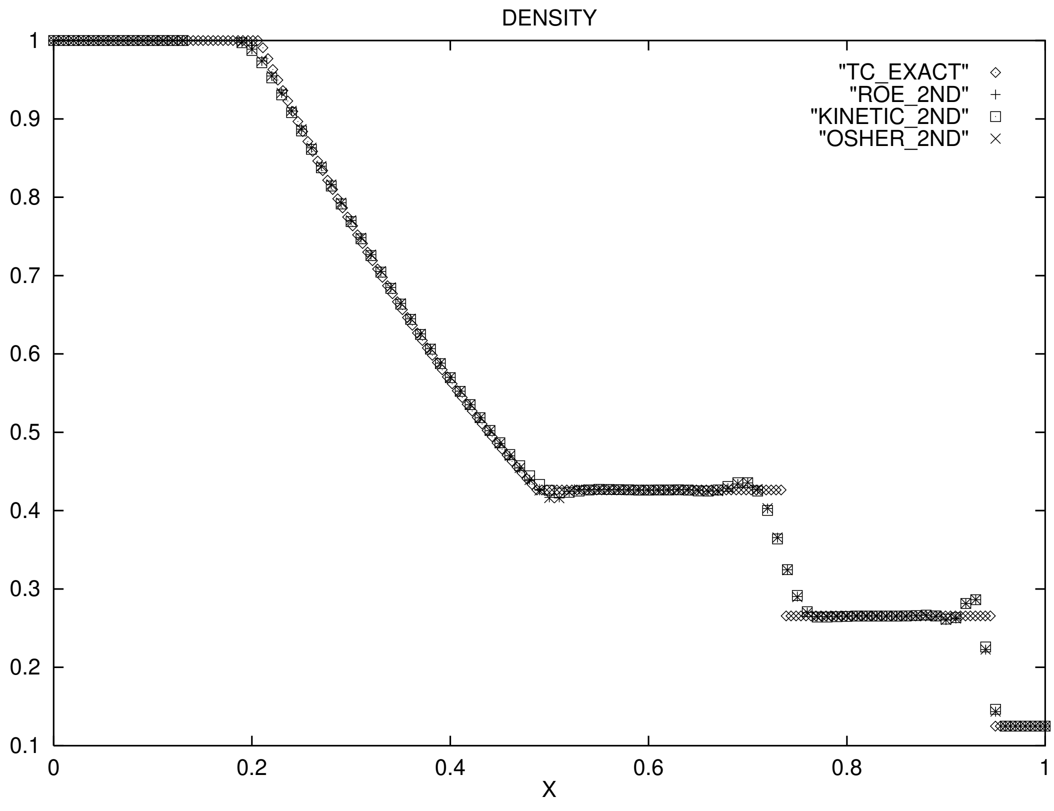


Figure 4: Shock Tube: Density distribution in the x direction, computed with full second-order Roe, Osher and Kinetic schemes without limiting process.

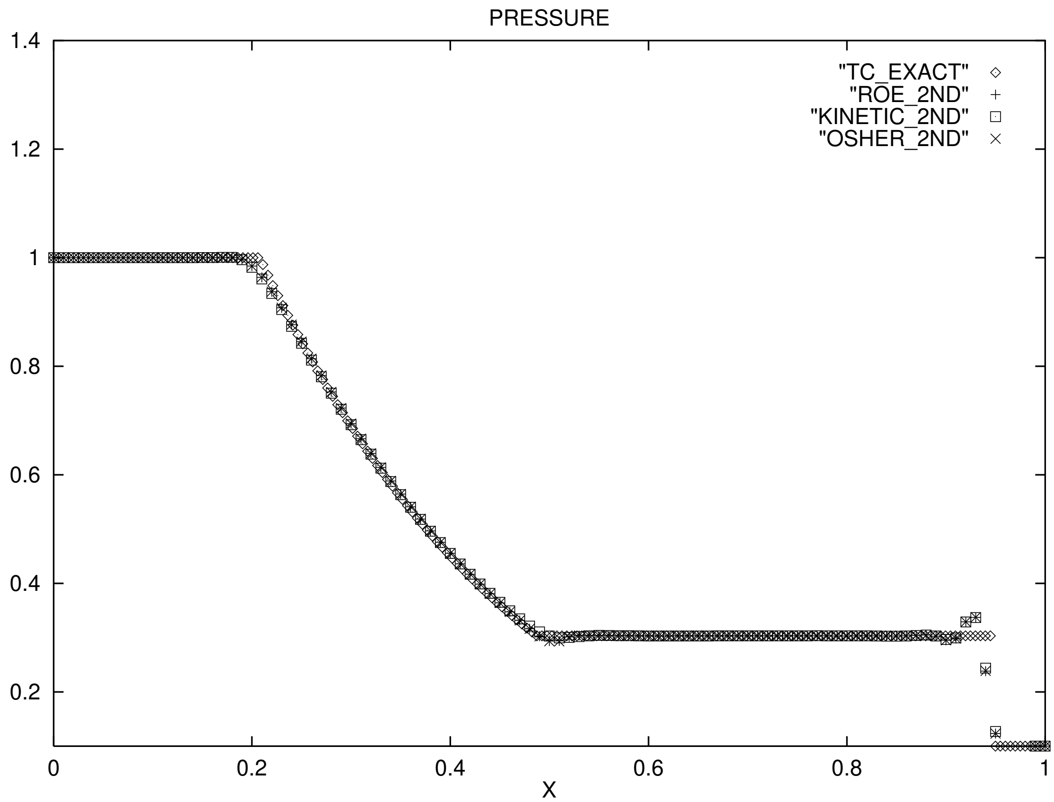


Figure 5: Shock Tube: Pressure distribution in the x direction, computed with full second-order Roe, Osher and Kinetic schemes without limiting process.

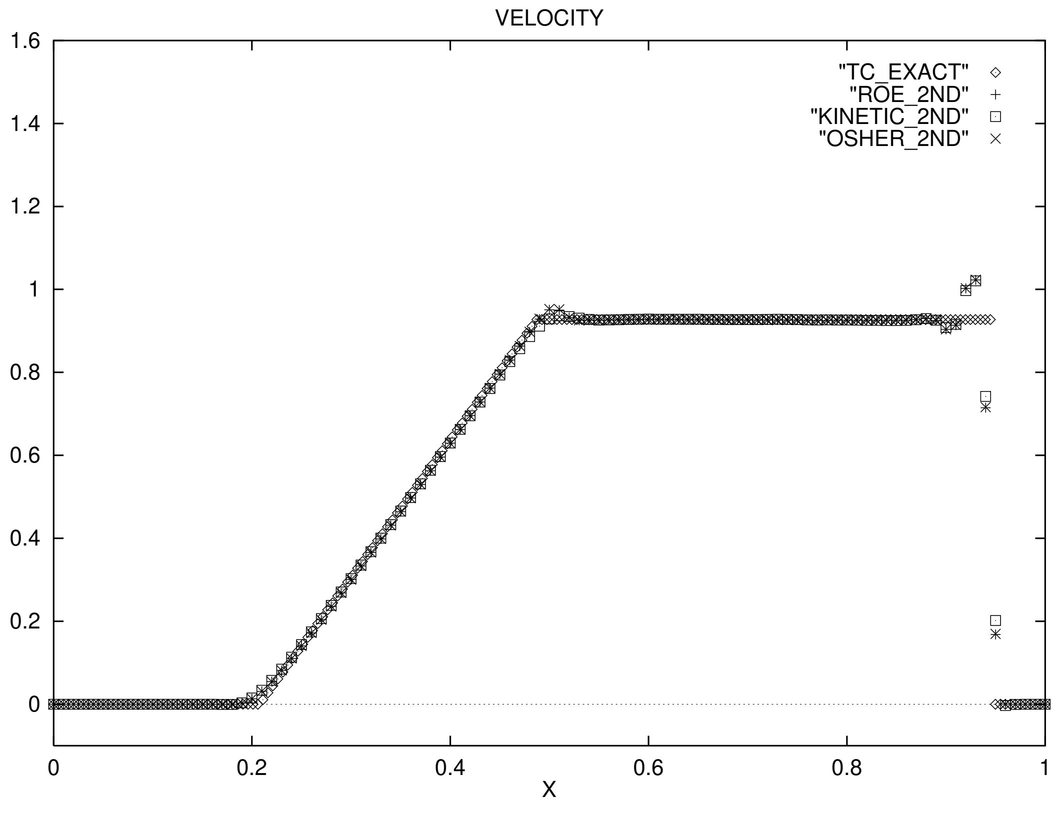


Figure 6: Shock Tube: Velocity distribution in the x direction, computed with full second-order Roe, Osher and Kinetic schemes without limiting process.

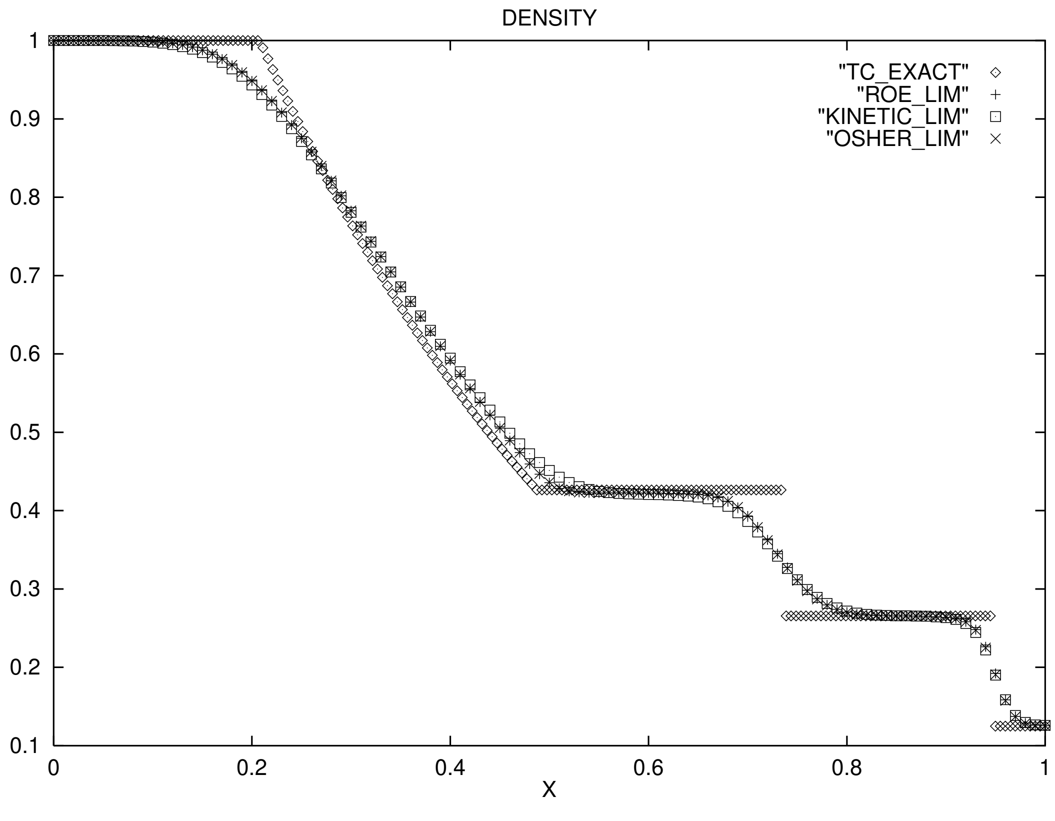


Figure 7: Shock Tube: Density distribution in the x direction, computed with second-order Roe, Osher and Kinetic schemes with Van Albada limiting process.

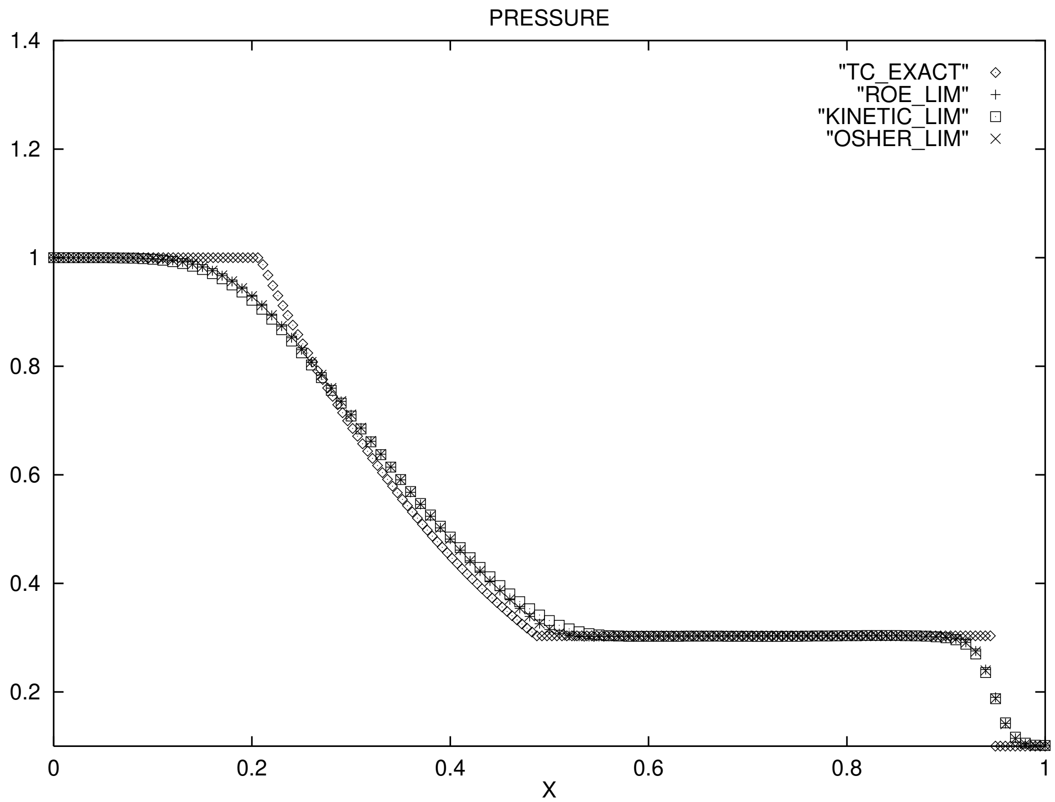


Figure 8: Shock Tube: Pressure distribution in the x direction, computed with second-order Roe, Osher and Kinetic schemes with Van Albada limiting process.

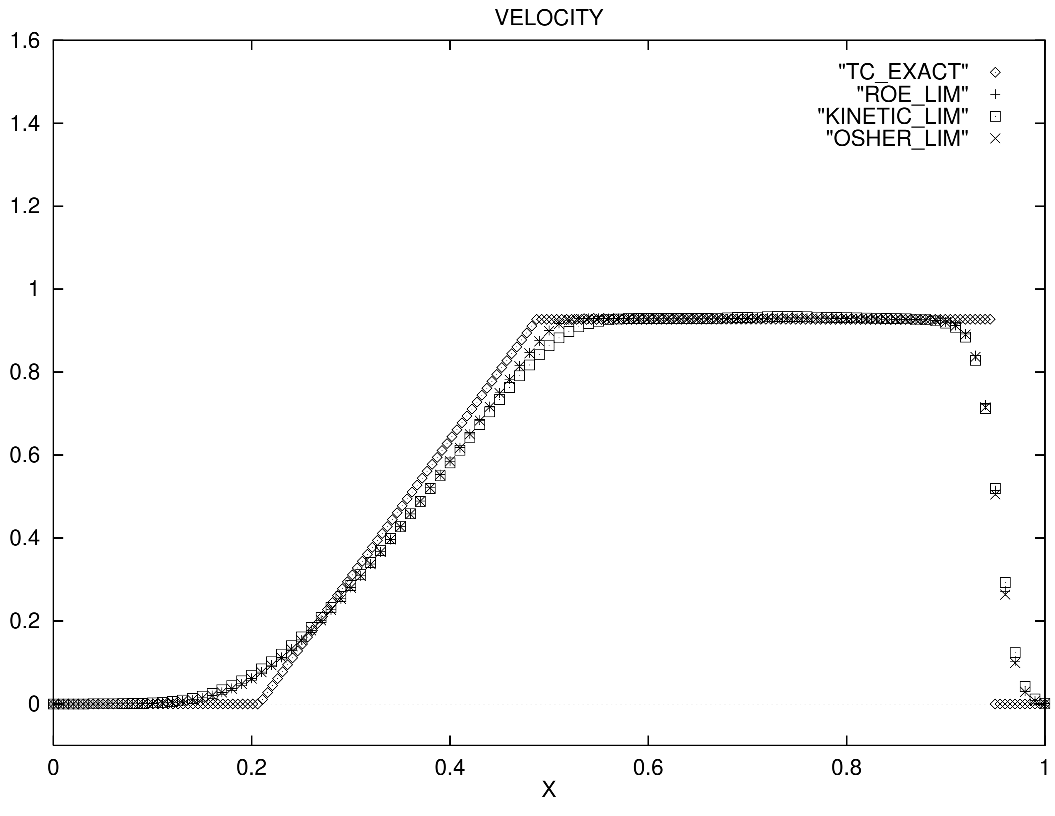


Figure 9: Shock Tube: Velocity distribution in the x direction, computed with second-order Roe, Osher and Kinetic schemes with Van Albada limiting process.

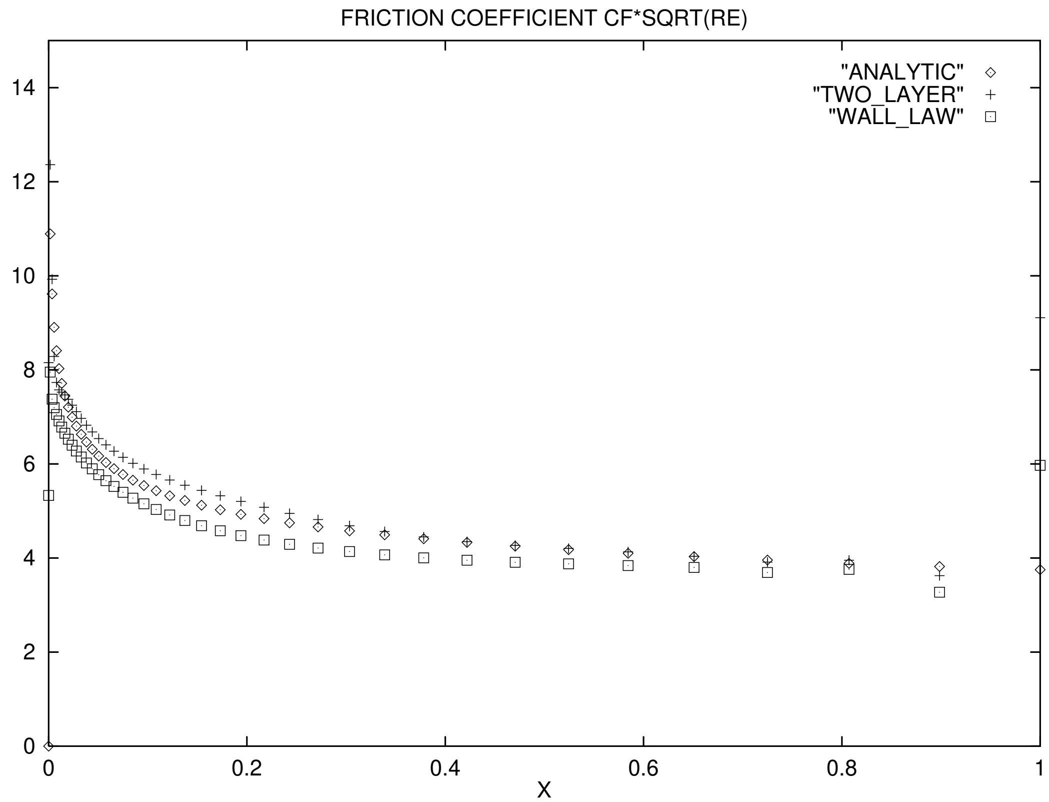


Figure 10: Flat Plate Boundary Layer, Friction coefficient, computed with wall-laws and two-layer and theoretical values.

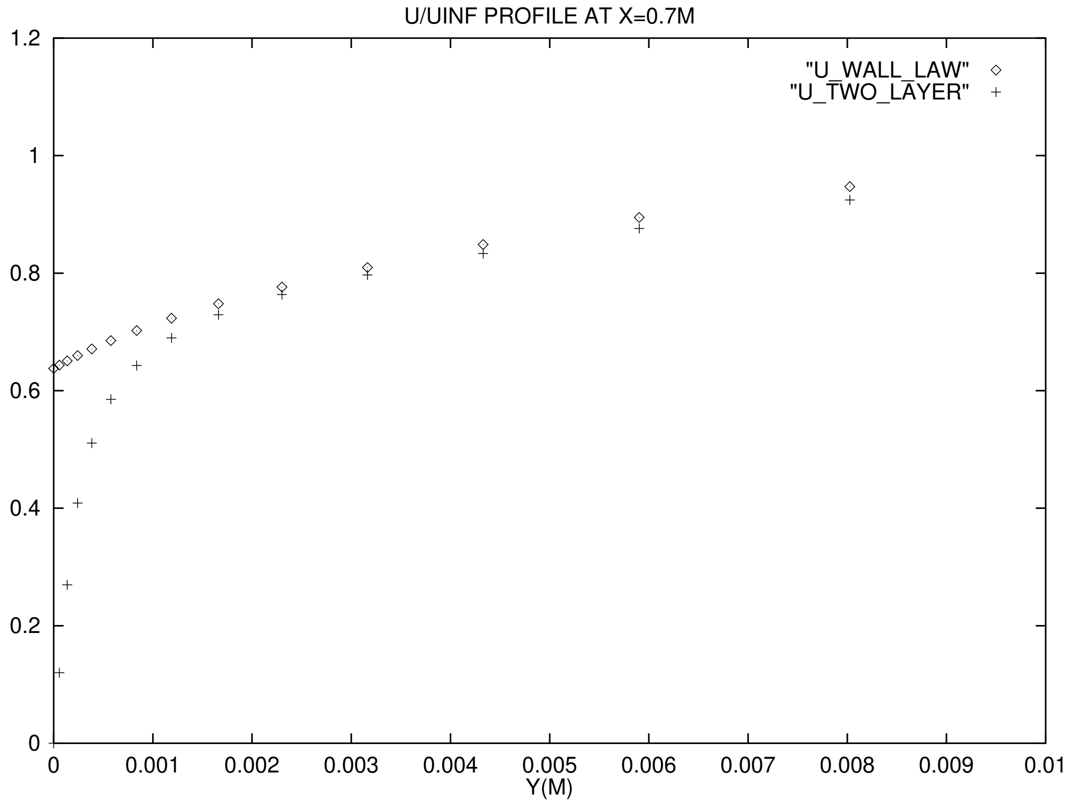


Figure 11: Flat Plate Boundary Layer, Velocity profile, computed with wall-laws and theoretical values.

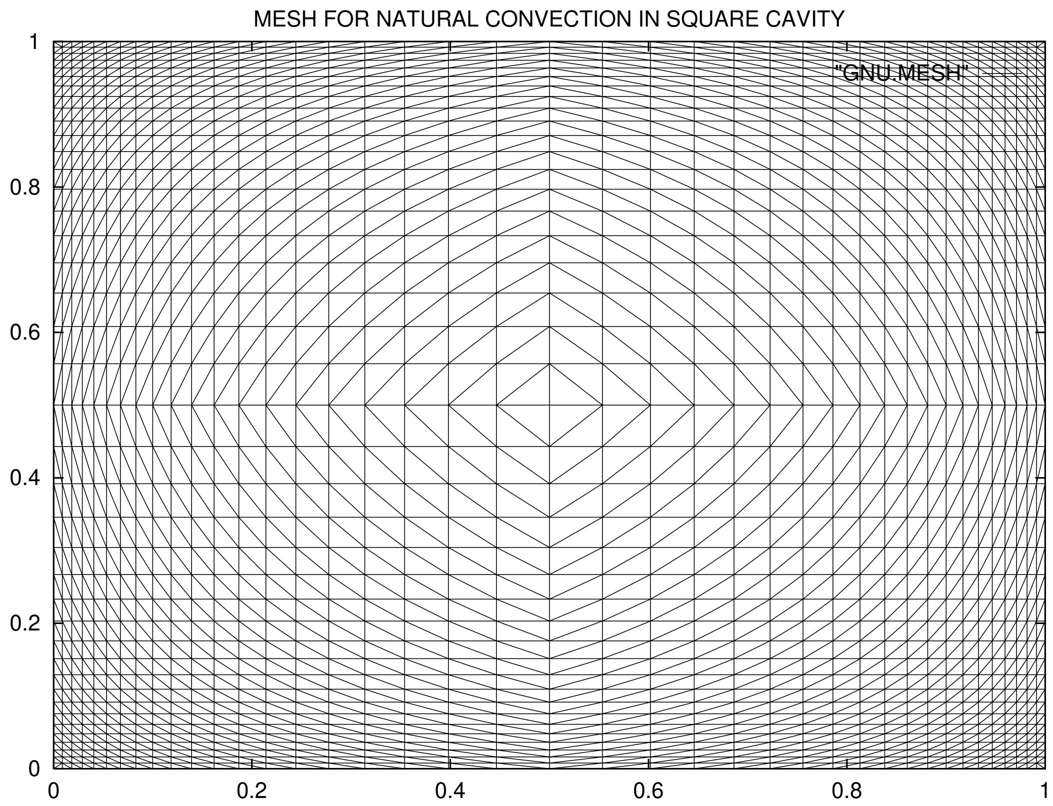


Figure 12: Natural Convection in Square Cavity: The mesh.

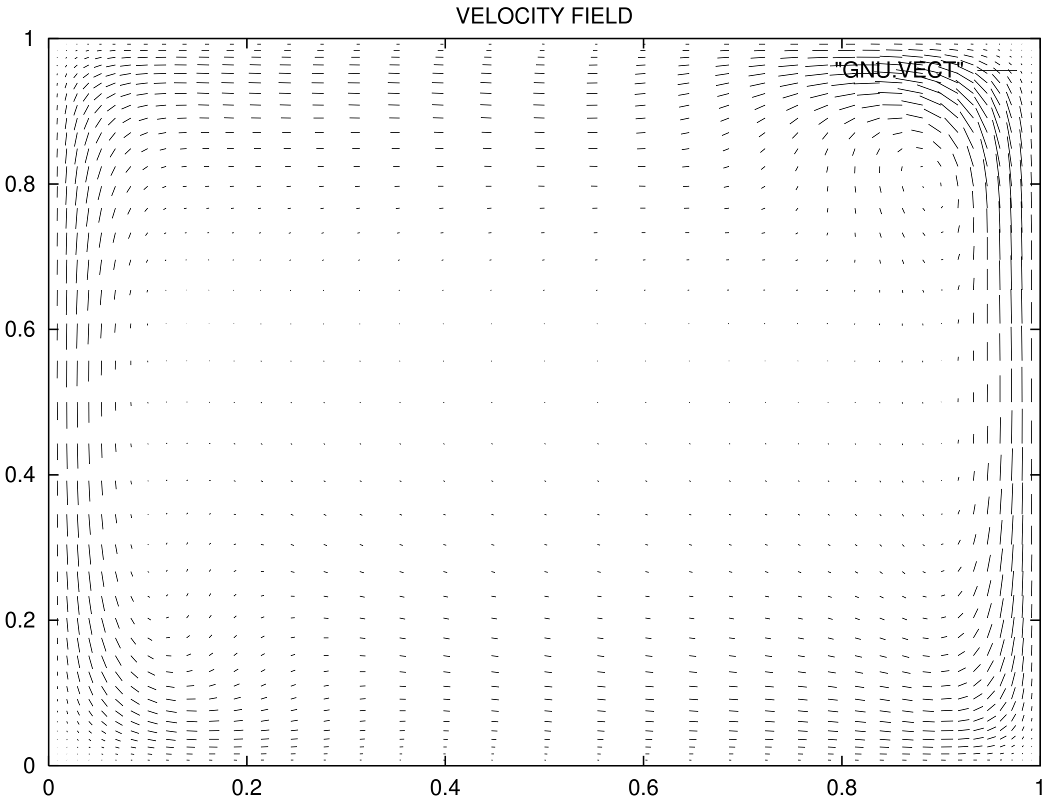


Figure 13: Natural Convection in Square Cavity: $Ra = 6.10^5$, Velocity field.

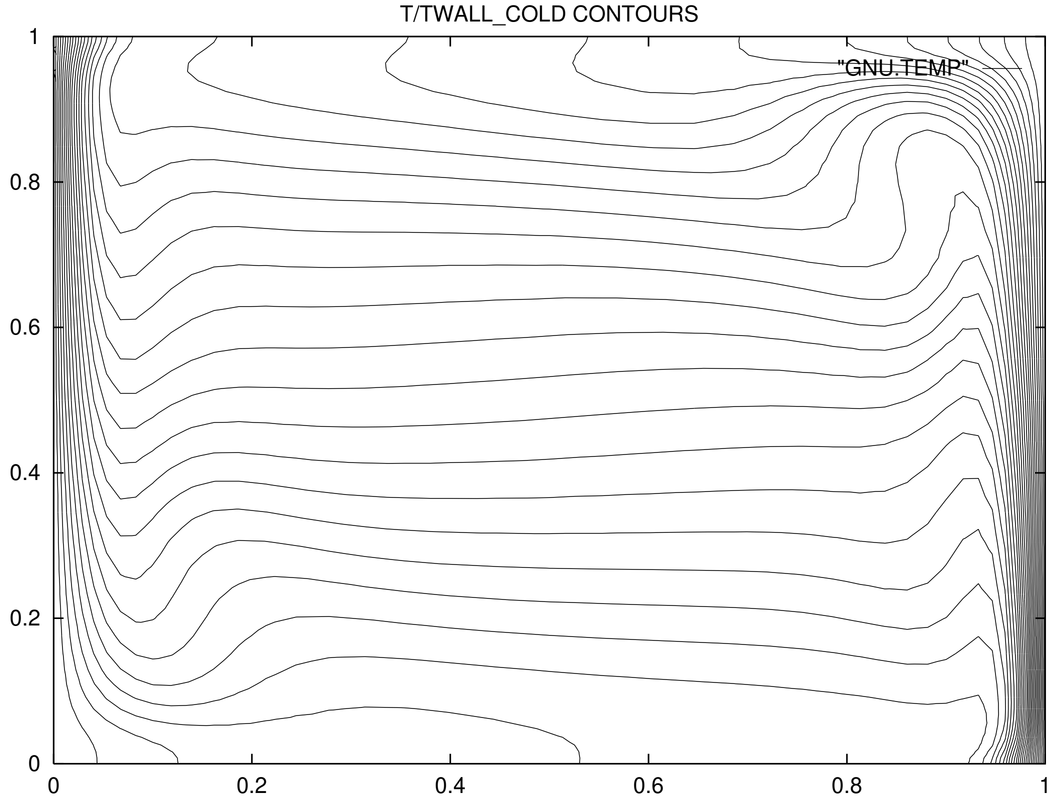


Figure 14: Natural Convection in Square Cavity: $Ra = 6.10^5$, Temperature field.

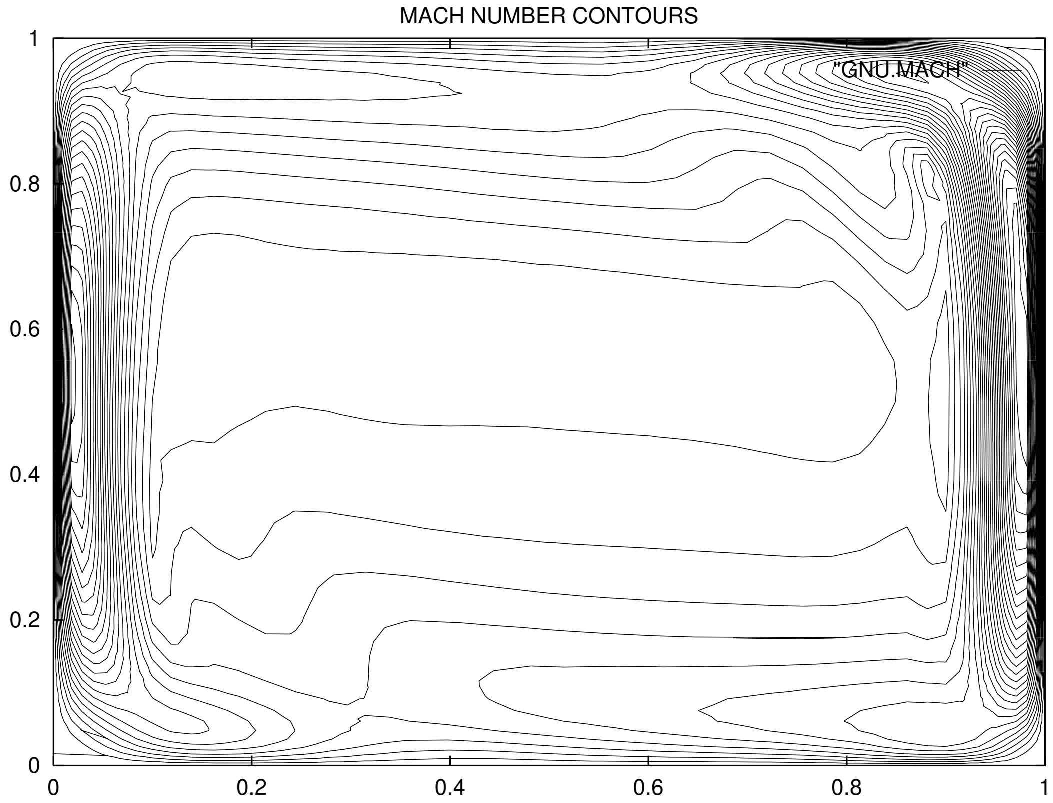


Figure 15: Natural Convection in Square Cavity: $Ra = 6.10^5$, Mach number contours.

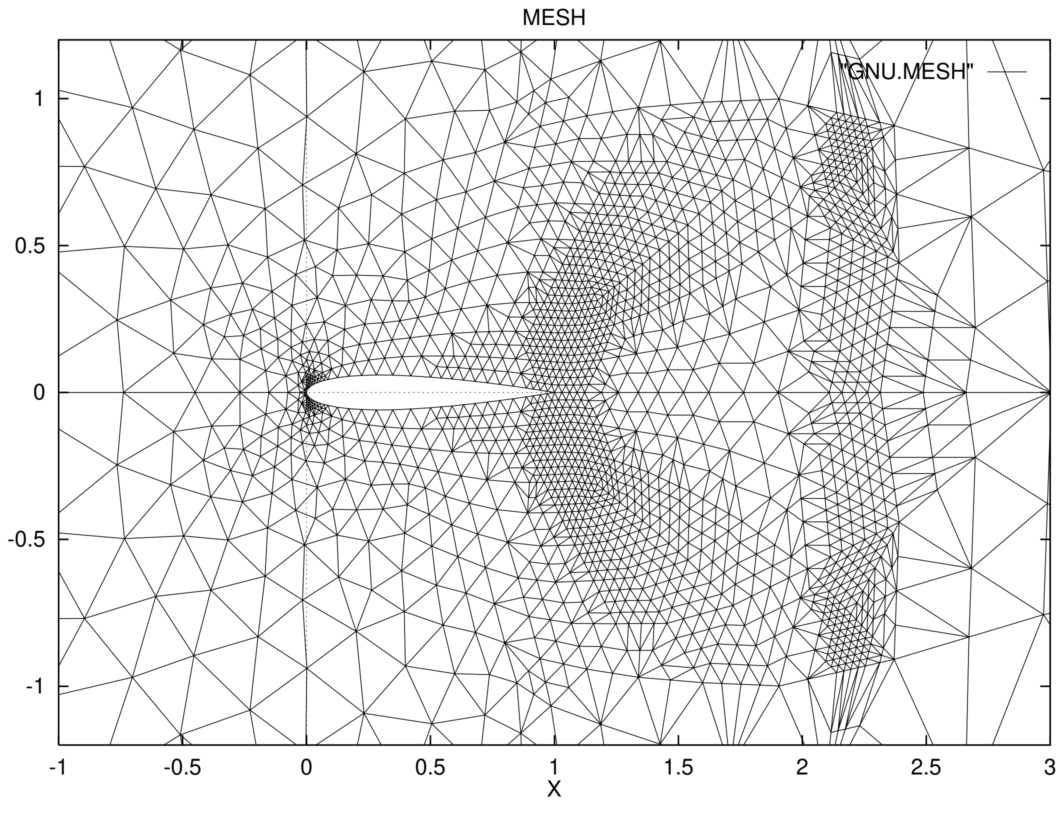


Figure 16: NACA 0012, The Fish Tail: $M_\infty = 0.95$, Euler Computation, The mesh (partial view)

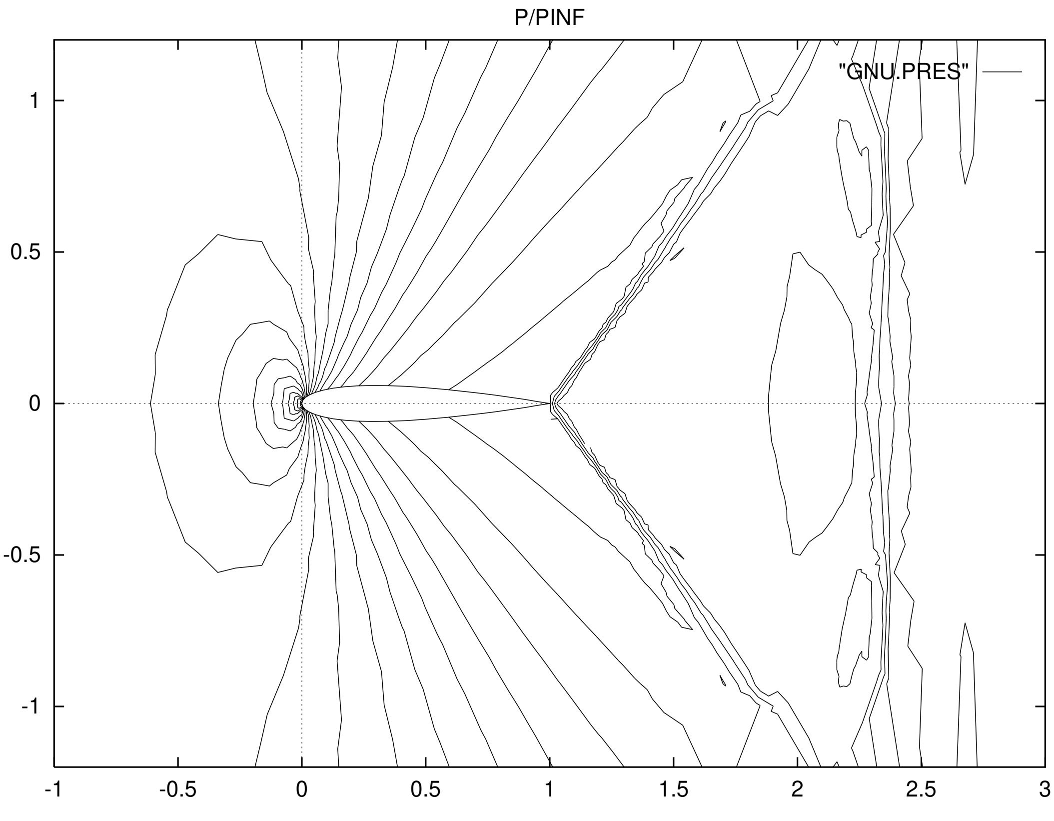


Figure 17: NACA 0012, The Fish Tail: $M_\infty = 0.95$, Euler Computation, iso-pressure lines (partial view)

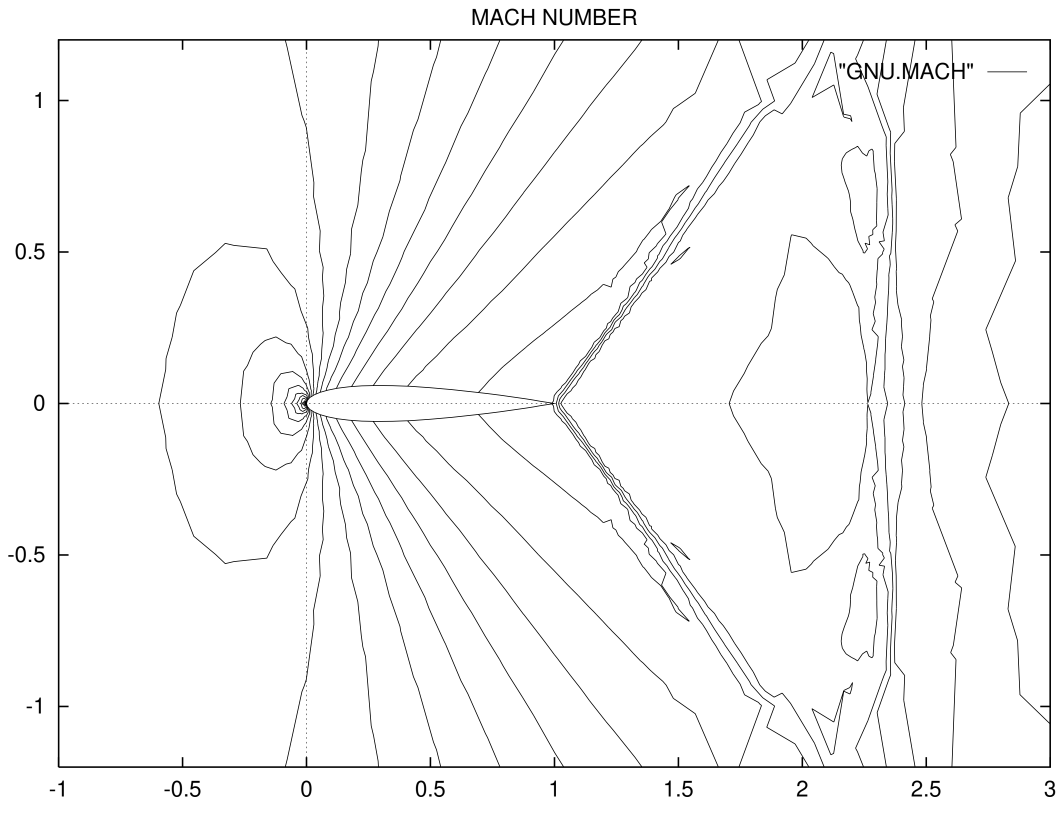


Figure 18: NACA 0012, The Fish Tail: $M_\infty = 0.95$, Euler Computation, iso-Mach lines (partial view)

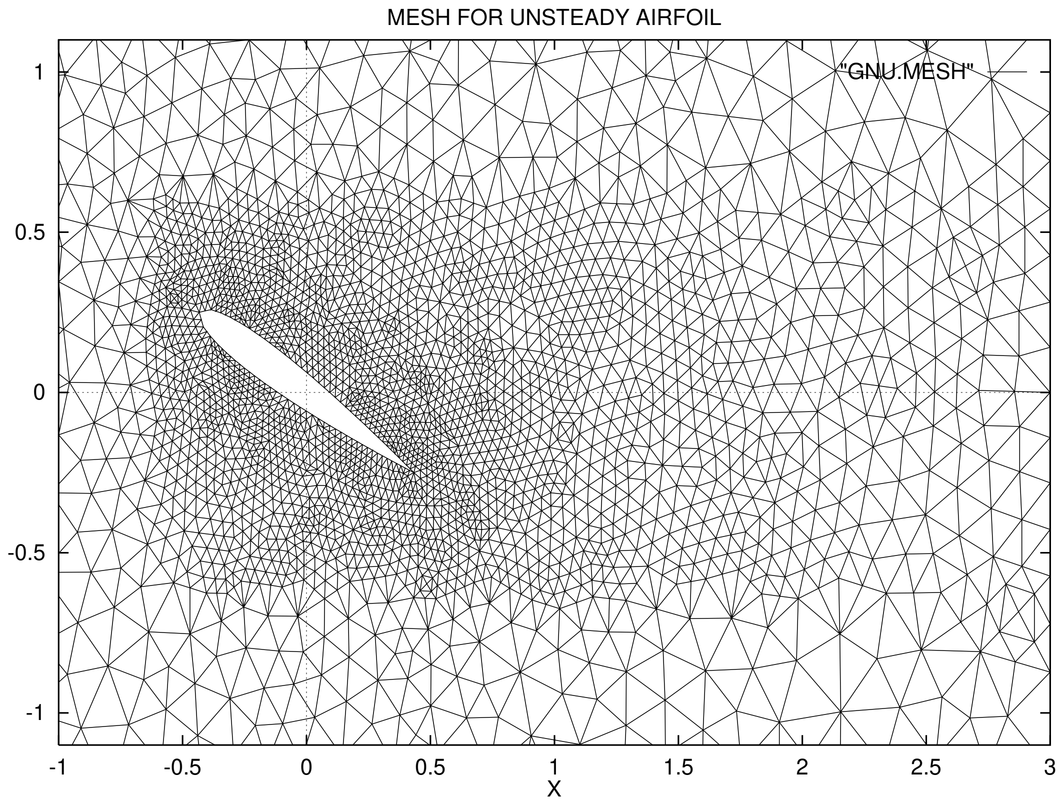


Figure 19: NACA 0012, Unsteady FLow: $M_\infty = 0.1$, $Re_\infty = 10^6$, Navier-Stokes Computation, The mesh (partial view)

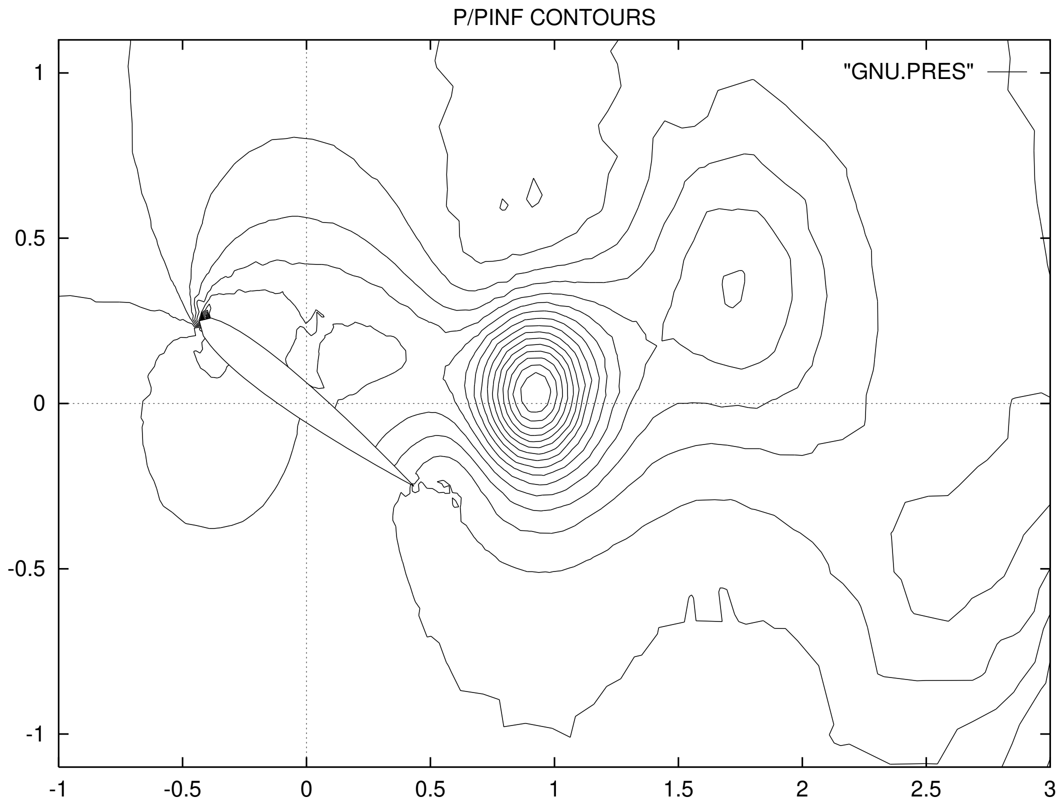


Figure 20: NACA 0012, Unsteady FLow: $M_\infty = 0.1$, $Re_\infty = 10^6$ Navier-Stokes Computation, iso-pressure lines (partial view)

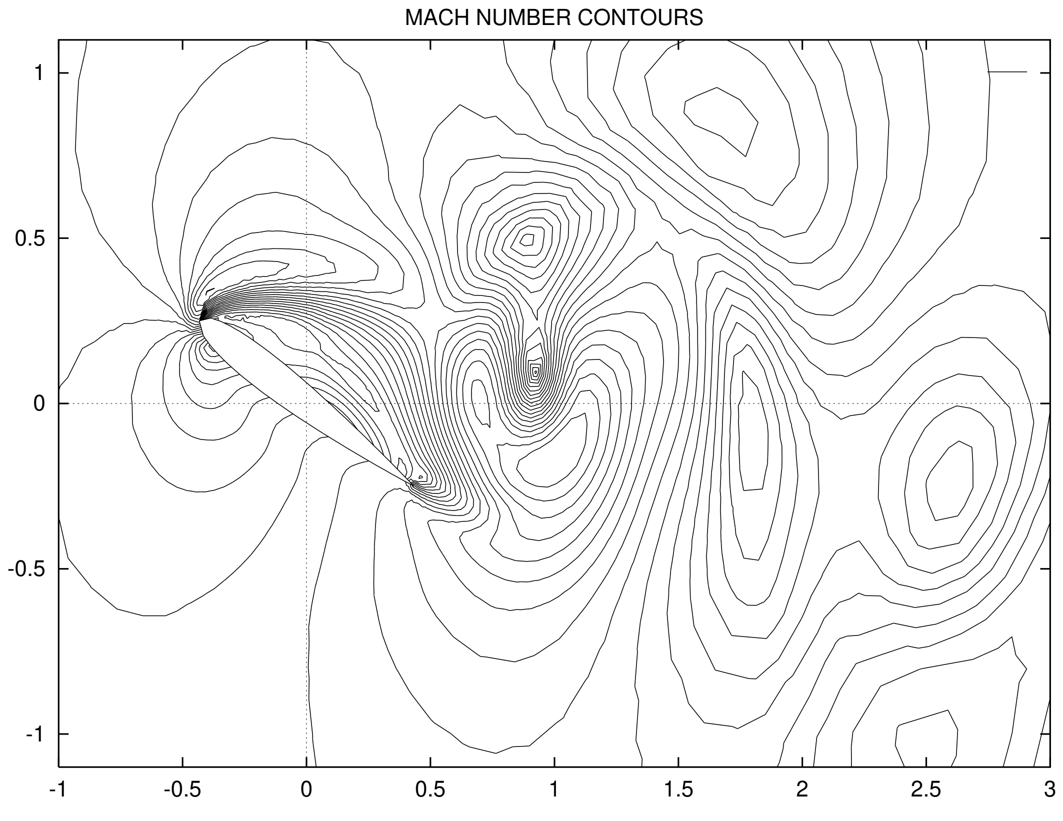


Figure 21: NACA 0012,Unsteady FLow: $M_\infty = 0.1$, $Re_\infty = 10^6$ Navier-Stokes Computation, iso-Mach lines (partial view)

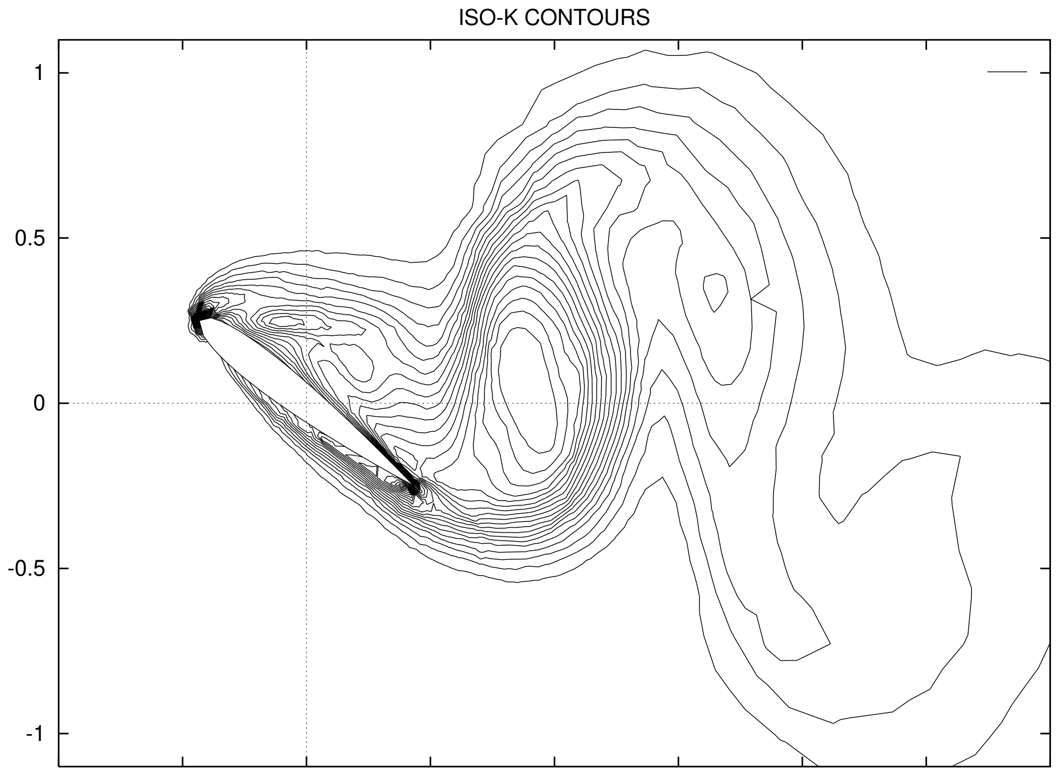


Figure 22: NACA 0012,Unsteady FLow: $M_\infty = 0.1$, $Re_\infty = 10^6$ Navier-Stokes Computation, iso-k lines (partial view)

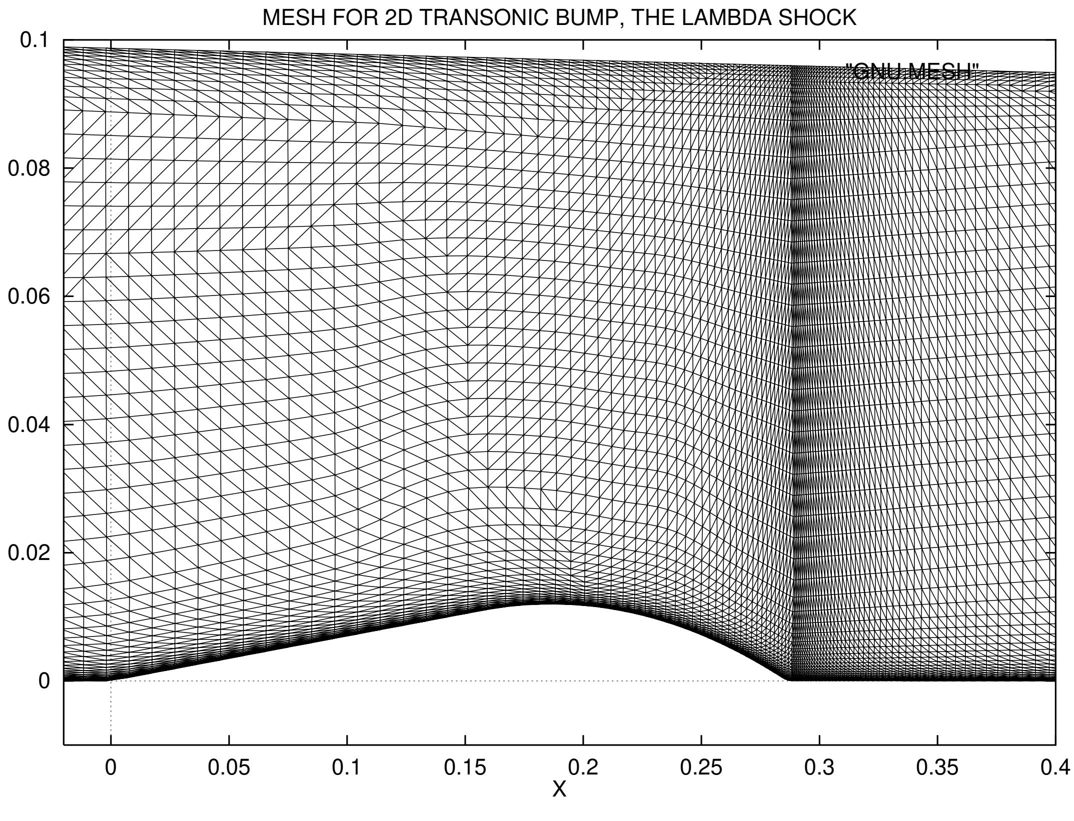


Figure 23: Transonic Bump Flow: $M_\infty = 0.635$, $Re_\infty = 1.3510^7$, Navier-Stokes Computation, The mesh (partial view)

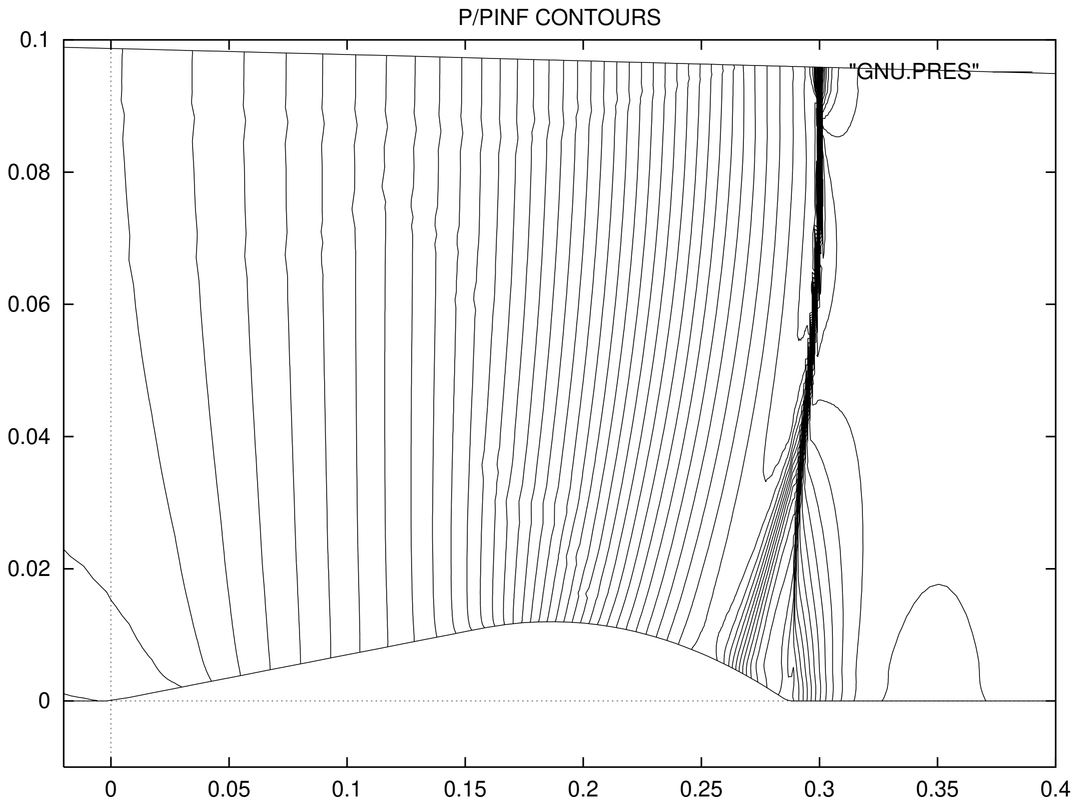


Figure 24: Transonic Bump Flow: $M_\infty = 0.635$, $Re_\infty = 1.3510^7$ Navier-Stokes Computation, iso-pressure lines (partial view)

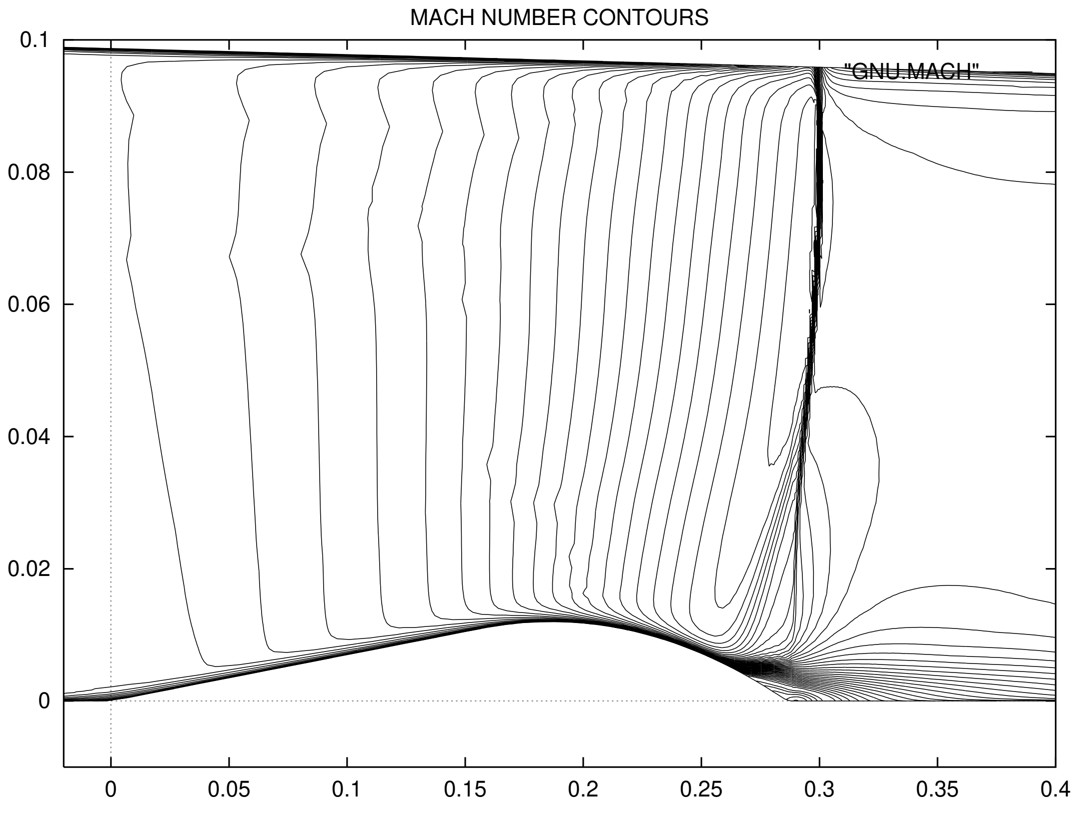


Figure 25: Transonic Bump Flow: $M_\infty = 0.635$, $Re_\infty = 1.3510^7$ Navier-Stokes Computation, iso-Mach lines (partial view)

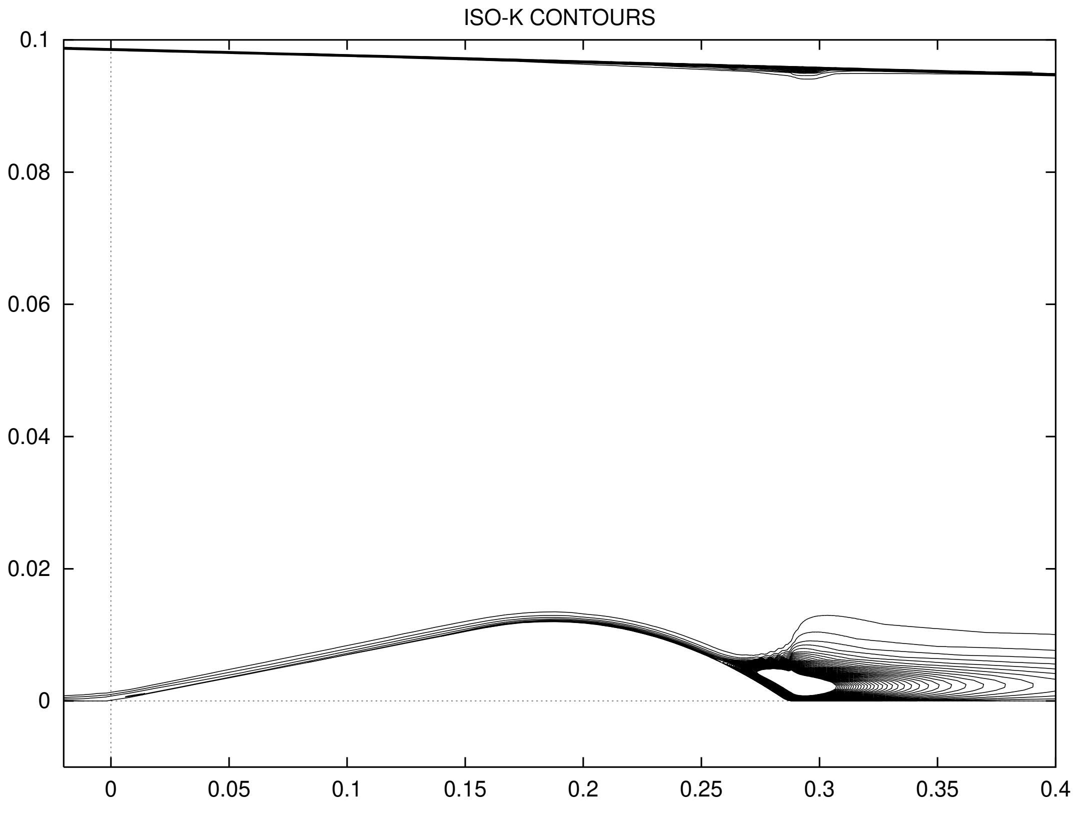


Figure 26: Transonic Bump Flow: $M_\infty = 0.635$, $Re_\infty = 1.3510^7$ Navier-Stokes Computation, iso-k lines (partial view)

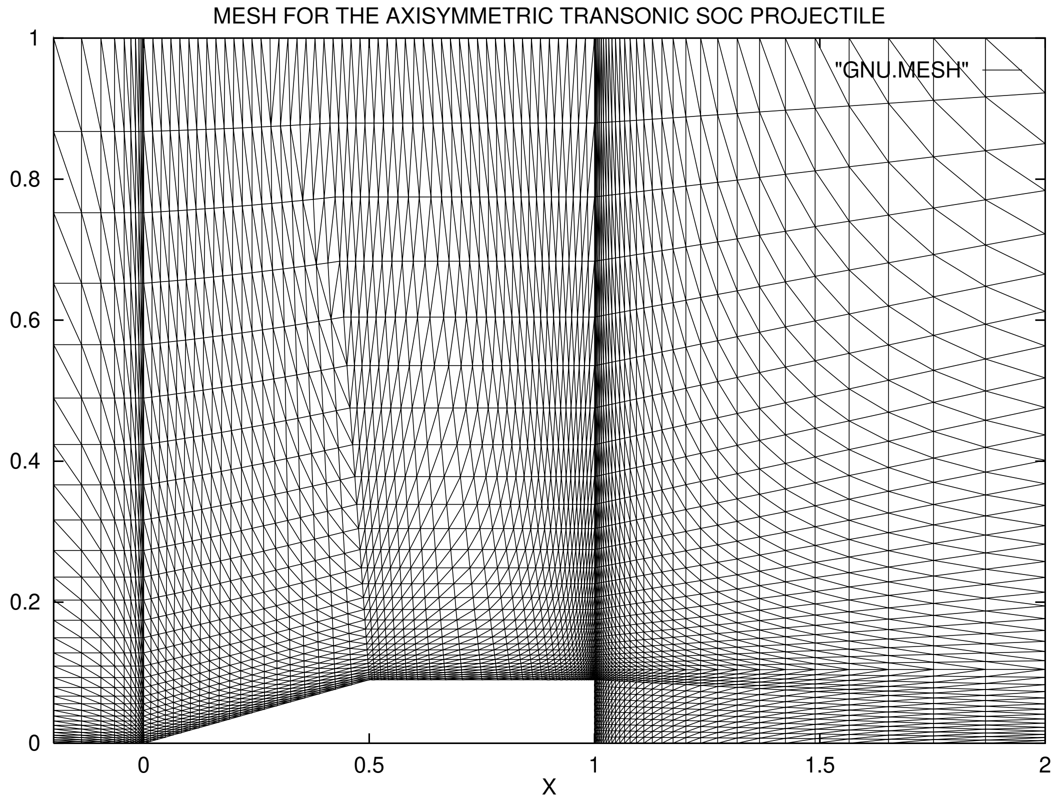


Figure 27: Transonic SOC projectile: $M_\infty = 0.96$, $Re_\infty = 10^6$, Navier-Stokes Computation, The mesh.

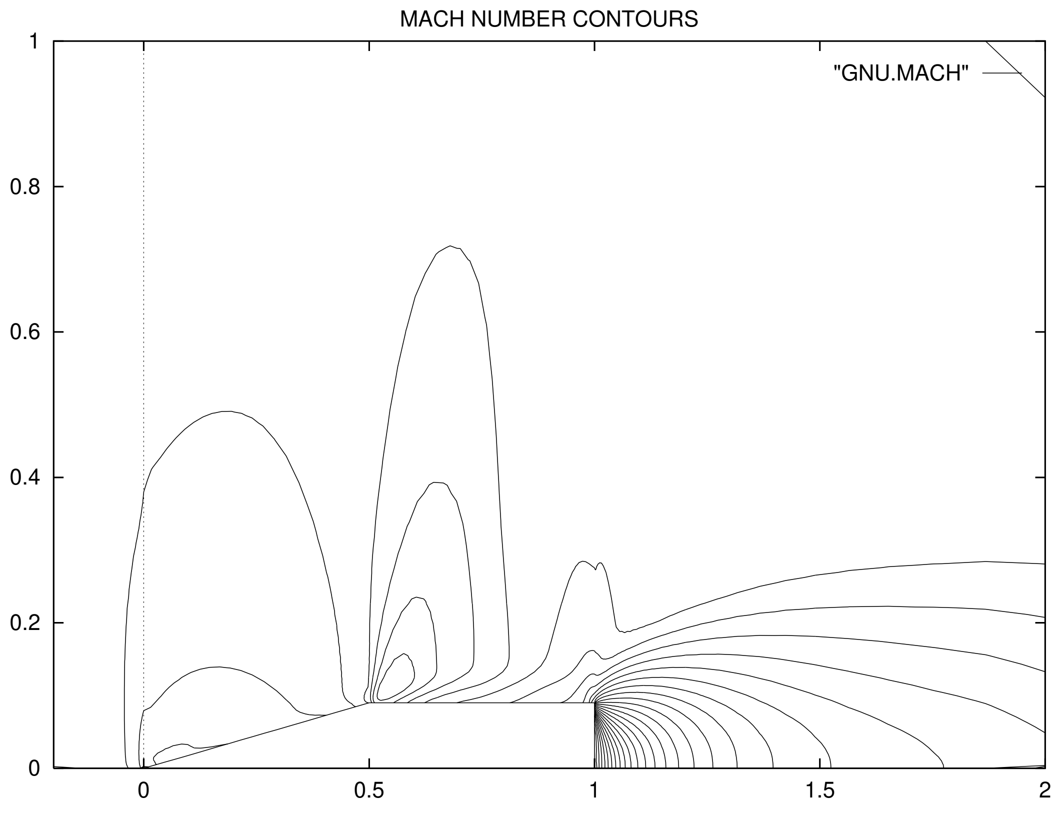


Figure 28: Transonic SOC projectile: $M_\infty = 0.96$, $Re_\infty = 10^6$, Navier-Stokes Computation, iso-Mach lines.

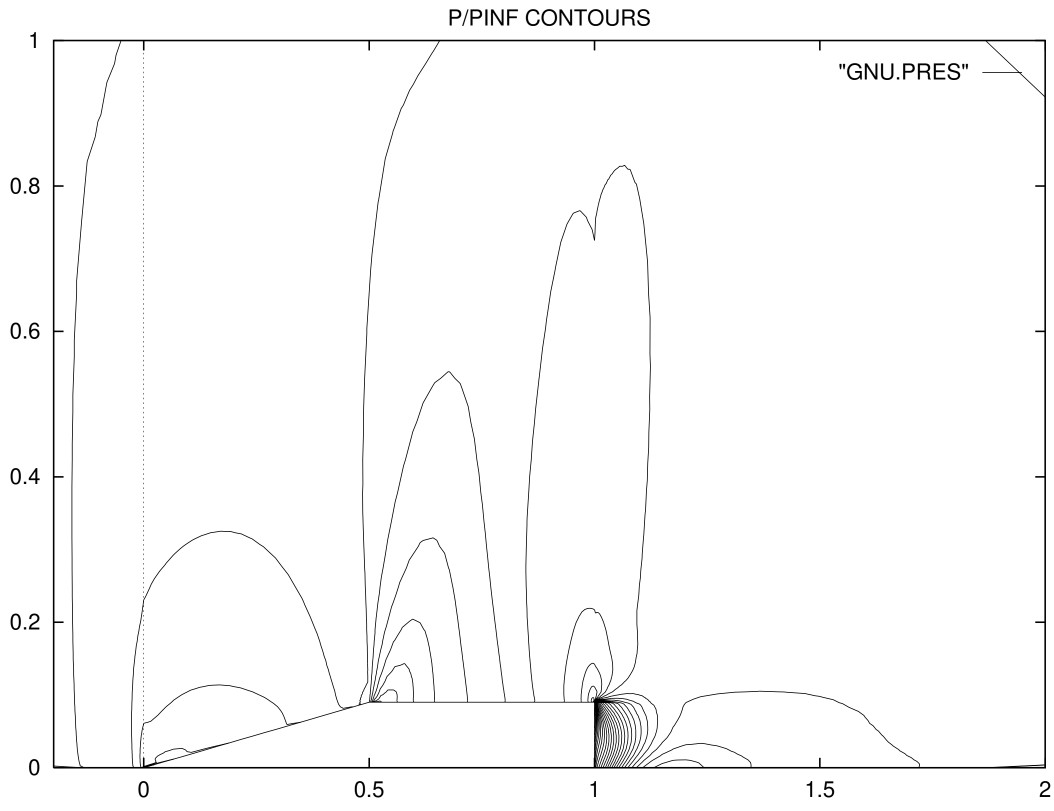


Figure 29: Transonic SOC projectile: $M_\infty = 0.96$, $Re_\infty = 10^6$, Navier-Stokes Computation, iso-pressure lines.

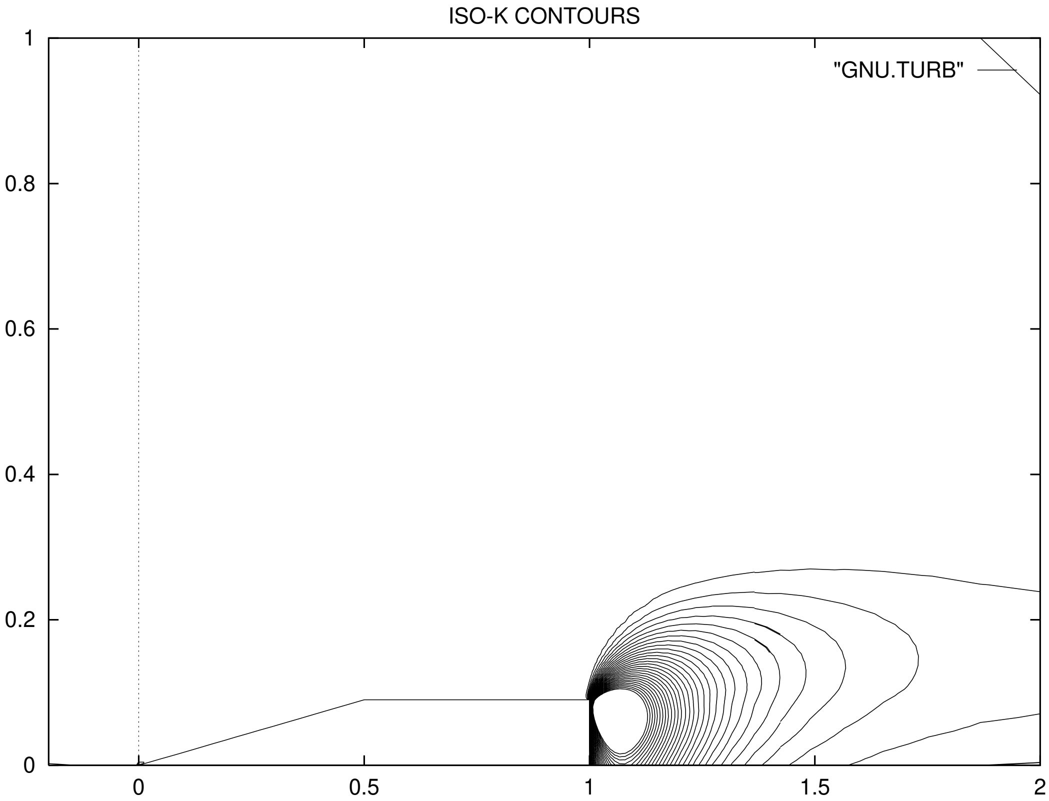


Figure 30: Transonic SOC projectile: $M_\infty = 0.96$, $Re_\infty = 10^6$, Navier-Stokes Computation, iso-k lines.

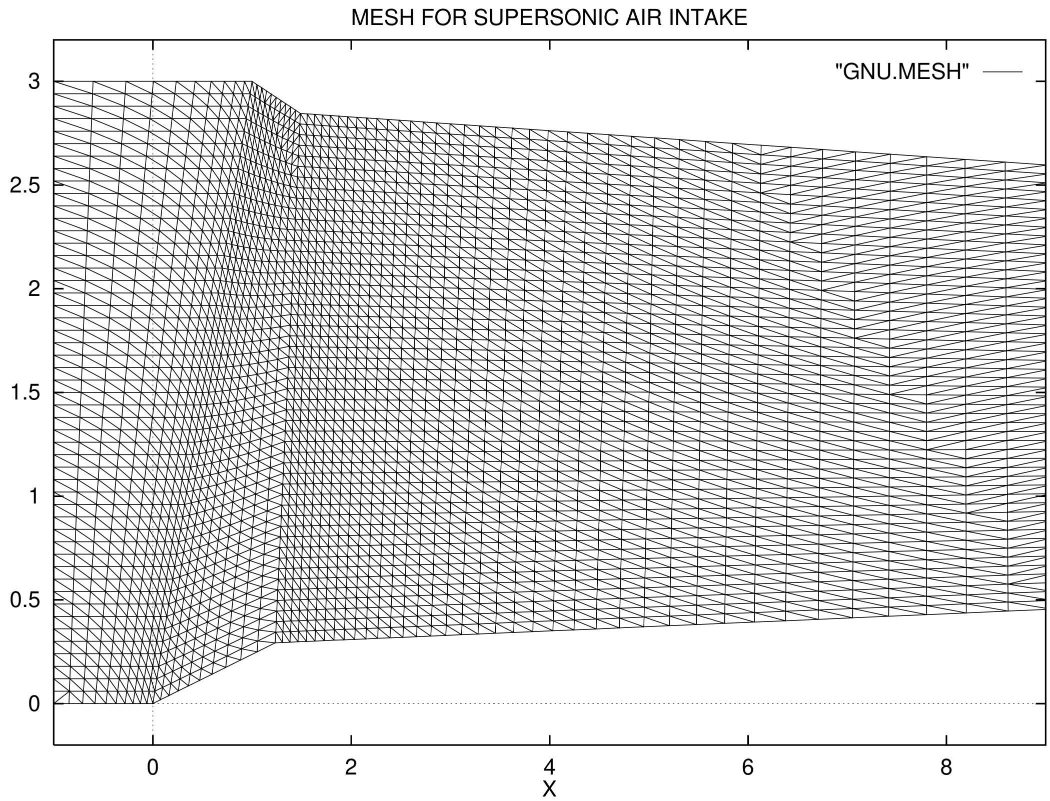


Figure 31: Axisymmetric Supersonic Air Intake: $M_{\infty} = 2.$, Navier-Stokes Computation, The mesh (partial view)

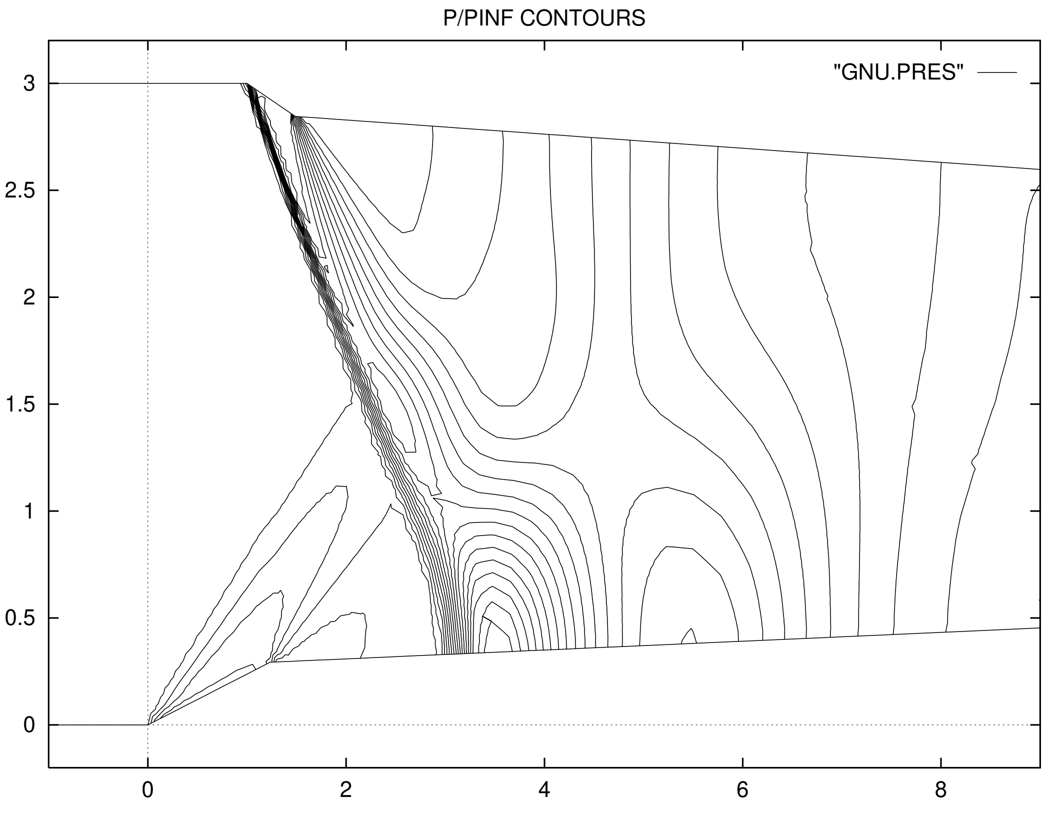


Figure 32: Axisymmetric Turbulent Supersonic Air Intake: $M_\infty = 2.$, $Re/m = 10^6$, Navier-Stokes Computation, pressure contours.

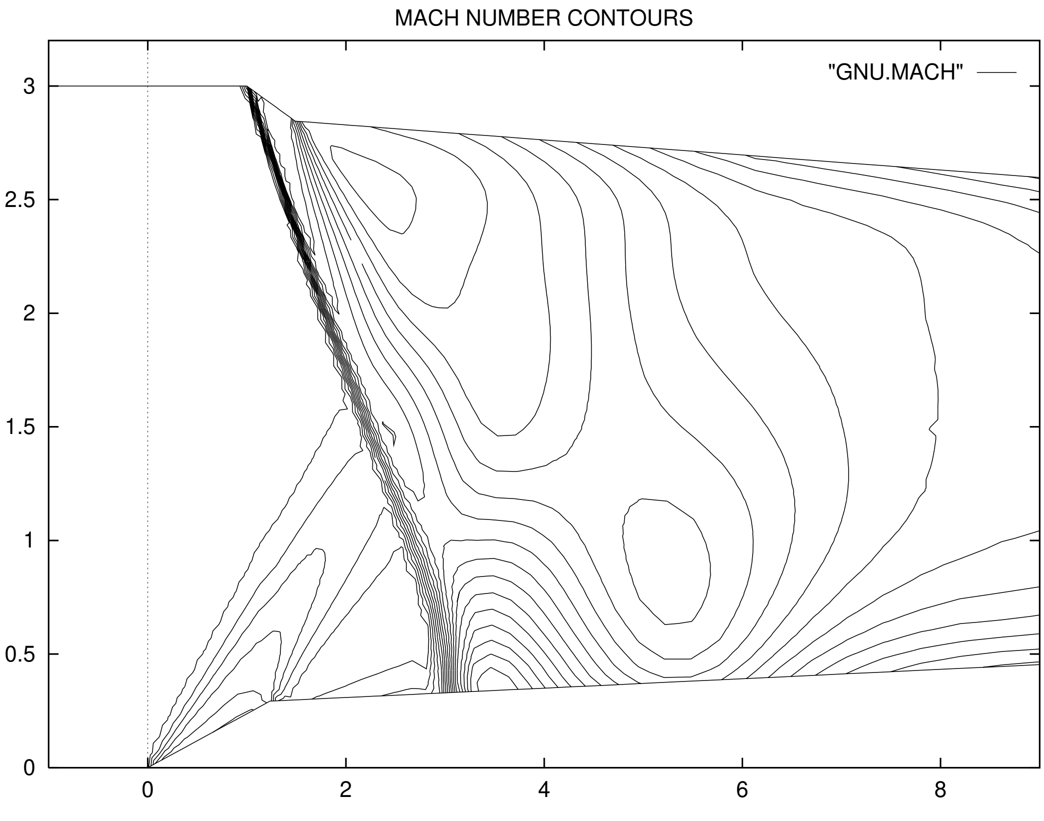


Figure 33: Axisymmetric Turbulent Supersonic Air Intake: $M_\infty = 2.$, $Re/m = 10^6$, Navier-Stokes Computation, Mach number contours.

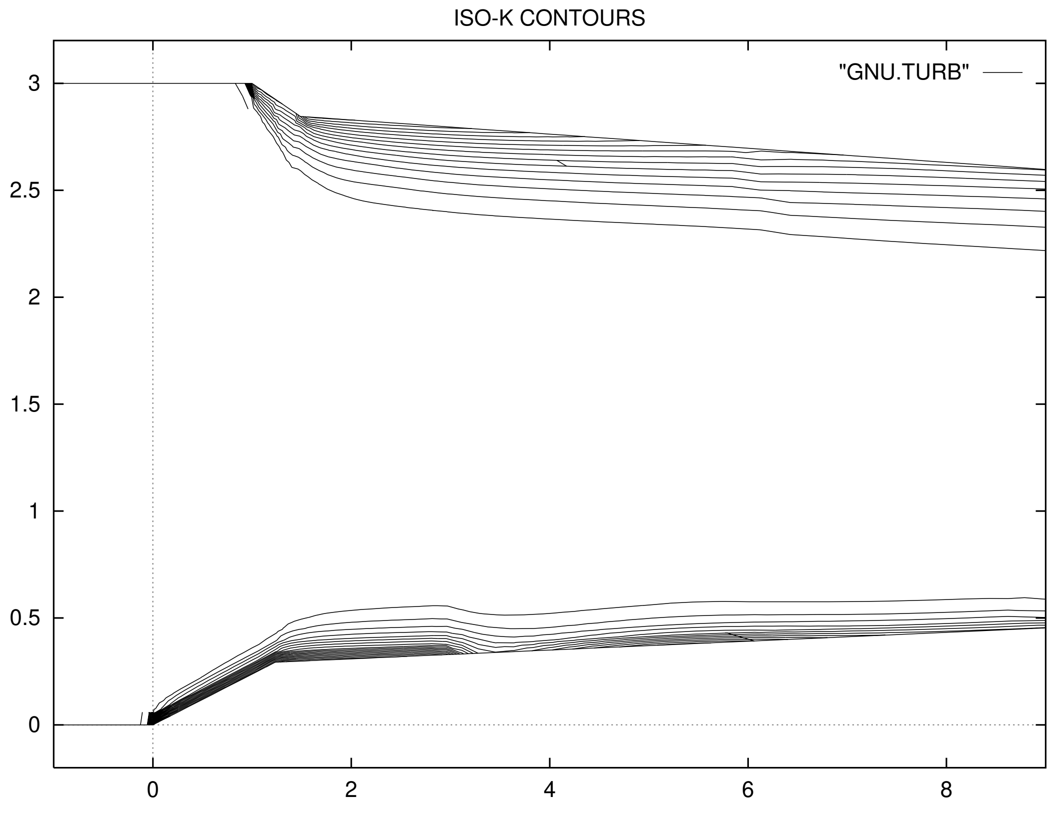


Figure 34: Axisymmetric Turbulent Supersonic Air Intake: $M_\infty = 2.$, $Re/m = 10^6$, Navier-Stokes Computation, Turbulent kinetic energy contours.

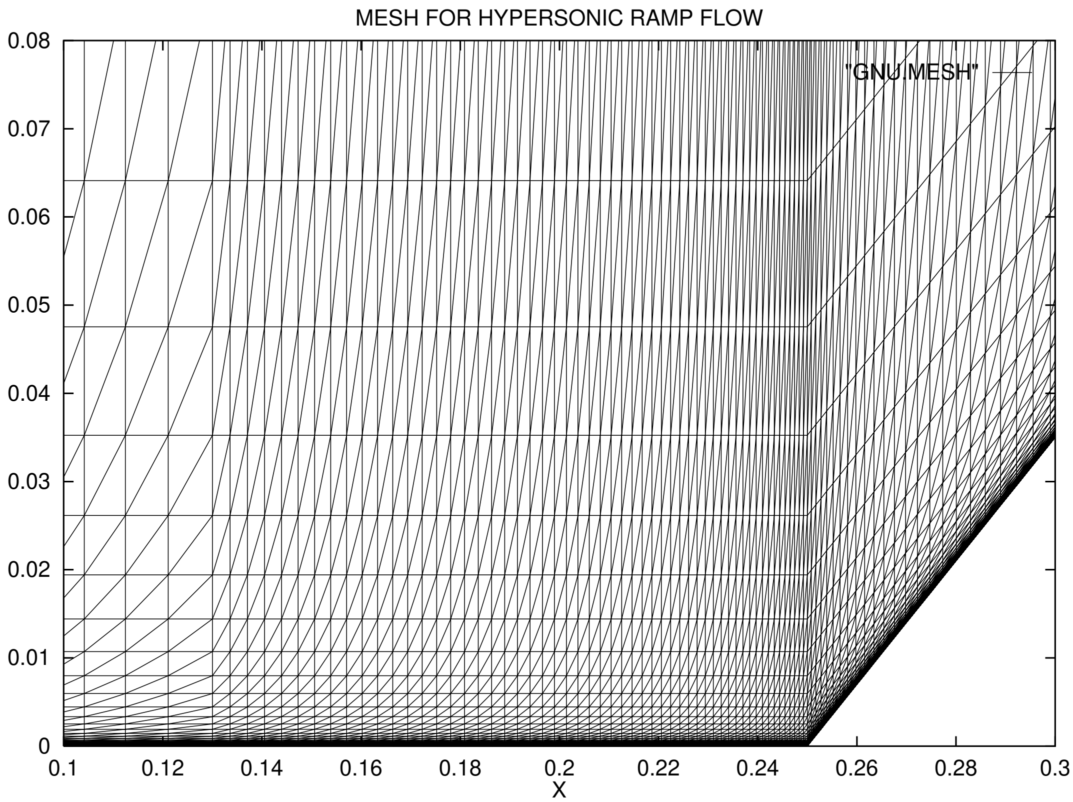


Figure 35: Hypersonic Turbulent Ramp Flow: $M_\infty = 5.$, $Re/m = 4.10^7$, Navier-Stokes Computation, partial view of the mesh.

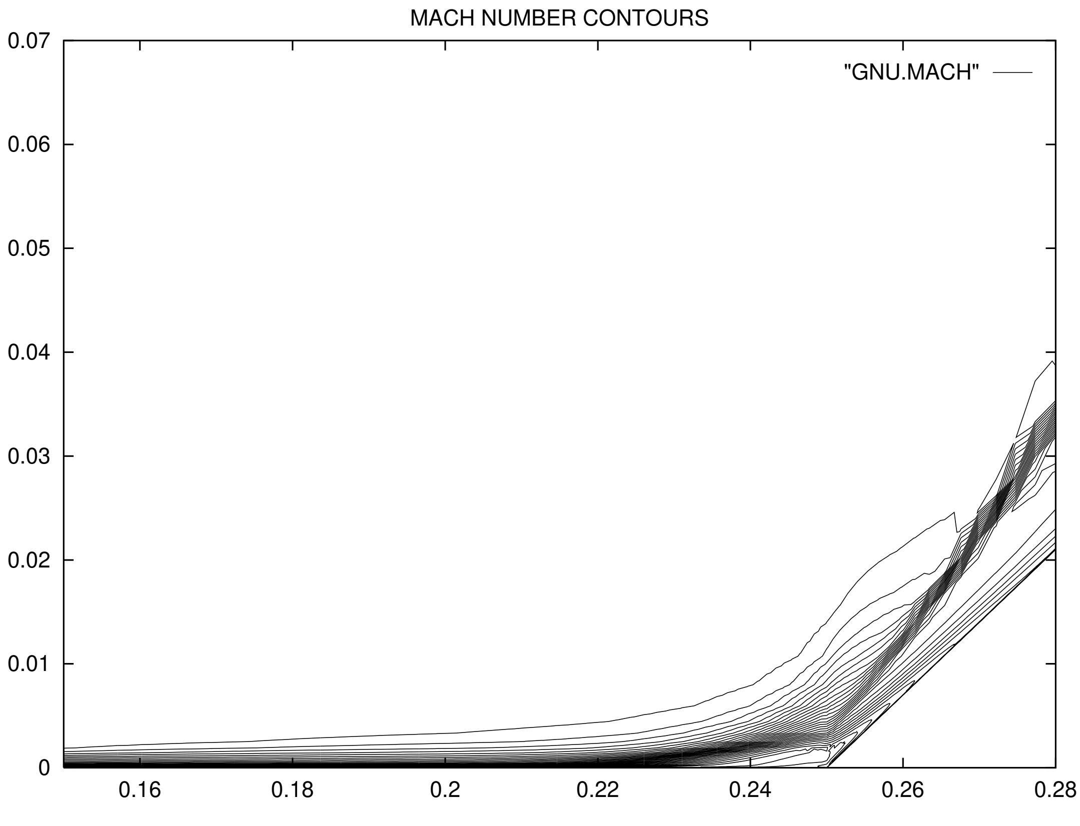


Figure 36: Hypersonic Turbulent Ramp Flow: $M_\infty = 5.$, $Re/m = 4.10^7$, Navier-Stokes Computation, Iso-Mach contours.

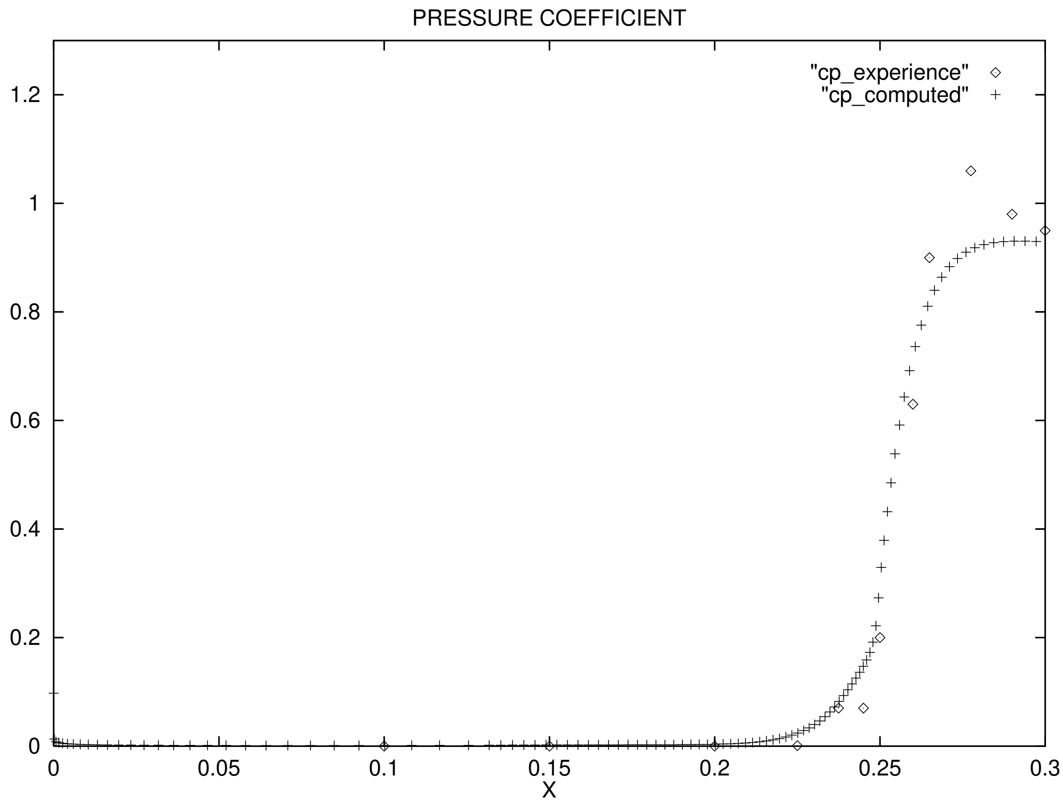


Figure 37: Hypersonic Turbulent Ramp Flow: $M_\infty = 5.$, $Re/m = 4.10^7$, Navier-Stokes Computation, experimental and computed pressure coefficient.

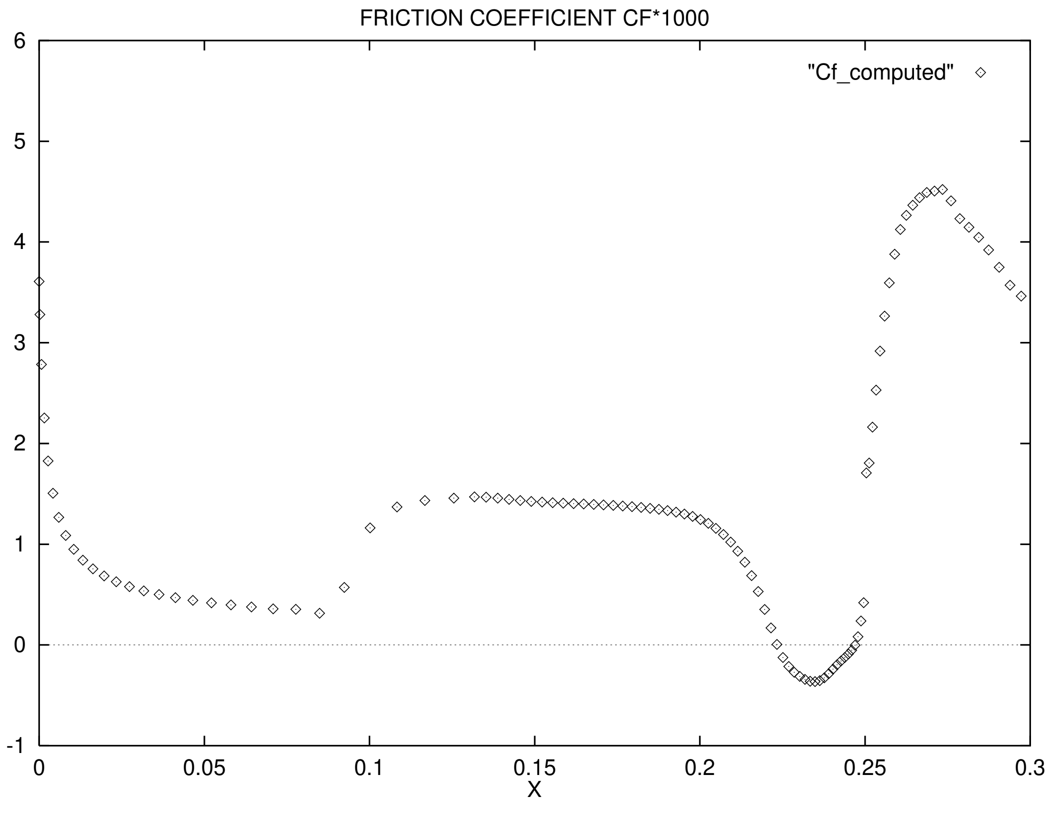


Figure 38: Hypersonic Turbulent Ramp Flow: $M_\infty = 5.$, $Re/m = 4.10^7$, Navier-Stokes Computation, computed skin friction coefficient.

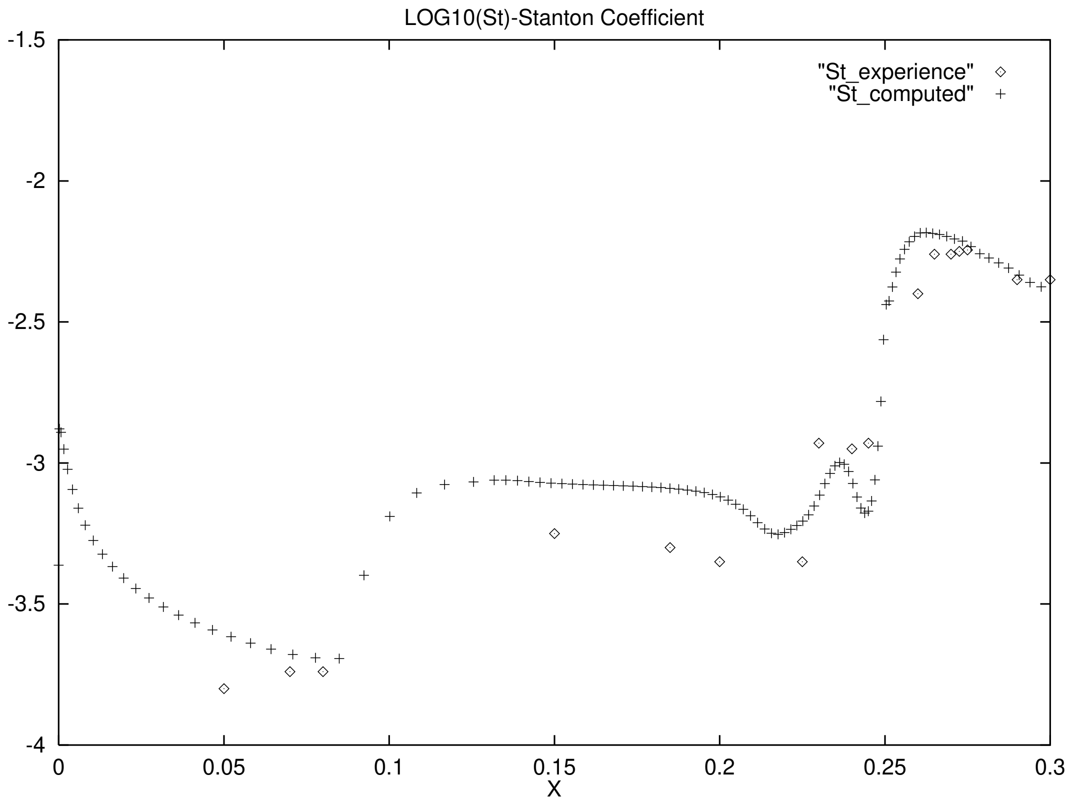


Figure 39: Hypersonic Turbulent Ramp Flow: $M_\infty = 5.$, $Re/m = 4.10^7$, Navier-Stokes Computation, experimental and computed heat flux coefficient (Stantom number).

9 Future Extensions : Release 2.0

At this time the second release of NSC2KE is under preparation. Our aim is to give the possibility to use different numerical techniques available for Fluid Dynamics Computations.

To improve the pre-processing step, the mesh definition will be possible using a language interpretor and an automatic mesh generator.

The release 1.0 proposes three different Finite-Volume-Galerkin schemes (Roe, Osher, Kinetic) for computing the Euler part of the Navier-Stokes equations. In the second release extensions to a new hybrid scheme and to SUPG techniques will be available.

Concerning turbulence modeling, the classical $k-\varepsilon$ model is available with wall-laws and two-layer techniques. NSC2KE 2.0 will also propose an RNG-based $k-\varepsilon$ model computing the flow up to the wall.

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 46 avenue Félix Viallet, 38031 GRENOBLE Cedex 1
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399