Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

P.W. Hemker, B. Koren

A non-linear multigrid method for the
steady Euler equations

Department of Numerical Mathematics Report NM-R8621 October

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

A Non-Linear Multigrid Method for the Steady Euler Equations

P.W. Hemker, B. Koren
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

Higher-order accurate Euler-flow solutions are presented for some airfoil test cases. Second-order accurate solutions are computed by an Iterative Defect Correction process. For two test cases even higher accuracy is obtained by the additional use of a τ-extrapolation technique. Finite volume Osher-type discretizations are applied throughout. Two interpolation schemes (one with and one without a flux limiter) are used for the computation of the second-order defect. In each Defect Correction cycle, the solution is computed by non-linear multigrid iteration, in which Collective Symmetric Gauss-Seidel relaxation is used as the smoothing procedure. The computational method does not require tuning of parameters. The solutions show a good resolution of discontinuities, and they are obtained at low computational costs. The rate of convergence seems to be grid-independent.

1980 Mathematics Subject Classification: 65N30, 76G15, 76H05.
Key Words and Phrases: steady Euler equations, second-order schemes, defect correction, multigrid methods.
Note: This work was supported by the Netherlands Technology Foundation (STW). It has been presented at the GAMM-Workshop on the Numerical Solution of Compressible Euler Flows, 10-13 June 1986, INRIA, Rocquencourt, France.

1. Introduction

Recently, a very efficient multigrid method has been developed for the solution of a robust, first-order accurate discretization of the Euler equations [7] . Two well-known drawbacks of first-order accurate discretizations of the Euler equations are: (i) their need for relatively fine grids in smooth flow regions, and (ii) their strong smearing of discontinuities that are not aligned with the grid. Second-order discretizations yield a strong improvement of both drawbacks. However, second-order discretizations are not solved with the same efficiency by the multigrid method. Moreover, with second-order discretizations stability problems easily arise and spurious non-monotonicity (wiggles) may be introduced. Motivated by the requirement of computational efficiency, Hemker [6] and Spekreijse [11] investigated an approximate solution method for stable, but apart from that, arbitrary second-order schemes. The method is based on the Defect Correction principle [2] . Recently, Koren [8] showed that these techniques are also feasible for the efficient computation of airfoil flows. For the present workshop, we basically use the same techniques as used in [8] . New is the use of a τ-extrapolation technique [3, 5] to further improve the accuracy of some solutions.

As flow problems for the Workshop we considered all problems as specified for the NACA0012- and Korn-airfoil (problem 2.1 - 2.5 and problem 3.1), and the last problem as specified for the bi-NACA0012-airfoil (problem 4.3). So, as the mandatory problems we selected the problems 2.3, 3.1 and 4.3.

In section 2, we briefly describe the basic discretization technique. In section 3, we discuss the solution method used: non-linear multigrid as an inner iteration for the solution of the elementary first-order system, and Iterative Defect Correction as an outer iteration for the approximate solution of the second-order system. In the sections 4 and 5, we describe the two second-order discretization

schemes that were used, and the τ-extrapolation technique. The main results of this paper are given in section 6. Conclusions are summarized in section 7.

2. Discretization

The non-isenthalpic, steady 2-D Euler equations can be written as

$$\frac{\partial f(q)}{\partial x} + \frac{\partial g(q)}{\partial y} = 0, \qquad \text{on } \Omega \subset \mathbb{R}^2 , \tag{2.1}$$

where

$$q = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho e \end{pmatrix}, \quad f = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho u(e + \frac{p}{\rho}) \end{pmatrix}, \quad g = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho v(e + \frac{p}{\rho}) \end{pmatrix}, \tag{2.2}$$

with for a perfect gas

$$e = \frac{1}{\gamma - 1}\frac{p}{\rho} + \frac{1}{2}(u^2 + v^2). \tag{2.3}$$

Here, ρ, u, v, p and γ denote density, velocity components in x- and y-direction, pressure and ratio of specific heats. We solve the non-isenthalpic Euler equations in their integral form

$$\int_{\delta\Omega^*} (fn_x + gn_y) ds = 0, \qquad \text{for all } \Omega^* \subset \Omega. \tag{2.4}$$

With $\delta\Omega^*$ we denote the boundary of Ω^*, and n_x and n_y are the components of the outward unit normal at $\delta\Omega^*$. A simple way to discretize (2.4) is to partition Ω into quadrilateral subregions Ω_{ij} (finite volumes), and to assume the fluxes $fn_x + gn_y$ to be constant along volume walls. This yields the following discretization

$$\sum_{k=1}^{4} \{f(q^l_{ij,k}, q^r_{ij,k}) n_{x_{ij,k}} + g(q^l_{ij,k}, q^r_{ij,k}) n_{y_{ij,k}}\} s_{ij,k} = 0, \quad \text{for all } ij . \tag{2.5}$$

Here the subscripts k refer to the kth wall of the quadrilateral volume Ω_{ij}, and the superscripts l and r to the left and right side of this wall respectively.

Because of the rotational invariance of the Euler equations, (2.5) may be further simplified to

$$\sum_{k=1}^{4} \{T^{-1}_{ij,k} f(T_{ij,k} q^l_{ij,k}, T_{ij,k} q^r_{ij,k}) s_{ij,k} = 0, \text{ for all } ij, \tag{2.6}$$

where $f(.,.)$ is the numerical flux function, and where $T_{ij,k}$ is a simple rotation matrix related to the wall $\partial\Omega_{ij,k}$, cf. [7]. The discrete solution is represented by $\{q_{ij}\}$, where q_{ij} is an approximation to the mean value of q over Ω_{ij}. The values $q^l_{ij,k}$ and $q^r_{ij,k}$ can be computed by (low degree) piecewise polynomial interpolation, using the values in one, two or three adjacent volumes, cf. [11].

For the numerical flux function $f(q^l, q^r)$, we approximate the solution of the 1-D Riemann problem for the two gas states q^l and q^r. For this approximation we have chosen the approximate Riemann solver as proposed by Osher [10]. The choice for Osher's scheme is motivated among others by: (i) its consistent treatment of non-reflecting boundary conditions, and, particularly, (ii) its suitability for Newton-type solution techniques.

3. Solution method

When we use the first-order discretization, the non-linear system (2.6) becomes

$$\sum_{k=1}^{4} T_{ij,k}^{-1} f(T_{ij,k} q_{ij}, T_{ij,k} q_{ij,k}) s_{ij,k} = 0, \text{ for all } ij, \tag{3.1}$$

in which $q_{ij,k}$ denotes the state in $\Omega_{ij,k}$, the kth neighbouring cell of Ω_{ij}.

To solve this system, as in [7, 9] , we consider non-linear point (=volume) relaxation methods, in which one or more local Newton steps are used for the collective relaxation of the 4 unknowns in each single volume. The most efficient relaxation is obtained by taking a large tolerance for the Newton iteration, so that in all but exceptional cases a single Newton step is sufficient. These relaxation methods are simple and robust, but they need an acceleration. For Collective Symmetric Gauss-Seidel point-relaxation, a suitable acceleration is found in multigrid. Multigrid in its non-linear form, the Full Approximation Scheme (FAS), appears to be efficient and robust. It is easily preceded by Full MultiGrid (FMG) to obtain good initial estimates [5] .

However, when we use a higher-order discretization and adopt this solution method, we lose efficiency. Gauss-Seidel is no longer an efficient smoother. Therefore, to approximately solve the higher-order discretized systems we use an Iterative Defect Correction (IDeC) process, cf. [2] . Denoting the system of equations resulting from a first- and second-order discretization by $N_h^1(q_h)=0$ and $N_h^2(q_h)=0$ respectively, IDeC applied to $N_h^2(q_h) = 0$ is written as

$$N_h^1(q_h^{(0)}) = 0, \tag{3.2a}$$

$$N_h^1(q_h^{(n+1)}) = N_h^1(q_h^{(n)}) - N_h^2(q_h^{(n)}), \qquad n = 0,1,2,... \tag{3.2b}$$

For the solution of $N_h^1(q_h)=r_h$, as it appears in both (3.2a) and (3.2b), we maintain the efficient multigrid method.

Although the second-order discretization only manifests itself via the right-hand side, the efficiency of this indirect solution method is amazingly good. For sufficiently smooth problems, both theory [5] and practice [6] show that already $q_h^{(1)}$, the first iterand in (3.2b), is second-order accurate. But also, if the problem is not smooth, in practice the first iterands yield already very good approximations [9] .

4. Second-order schemes

Standard second-order schemes, as the central or the fully one-sided upwind scheme, can be written as

$$q_{i+1/2,j}^{l} = q_{i,j} + \frac{1+\kappa}{4}(q_{i+1,j} - q_{i,j}) + \frac{1-\kappa}{4}(q_{i,j} - q_{i-1,j}), \tag{4.1a}$$

$$q_{i+1/2,j}^{r} = q_{i+1,j} + \frac{1+\kappa}{4}(q_{i,j} - q_{i+1,j}) + \frac{1-\kappa}{4}(q_{i+1,j} - q_{i+2,j}). \tag{4.1b}$$

Here $i+1/2$ denotes the wall separating volume i and $i+1$. For $\kappa=1$, $\kappa=1/3$, $\kappa=0$ and $\kappa=-1$, we find the central-, the upwind biased-, the Fromm-, and the fully one-sided upwind- schemes respectively. Similar relations hold for $q_{i,j+1/2}^{l}$ and $q_{i,j+1/2}^{r}$. Disadvantages of these schemes are: (i) that they cannot be applied in a consistent way in the neighbourhood of boundaries, and (ii) that they may yield solutions with wiggles.

As a scheme which is consistent near boundaries, in [6] we introduced the so-called superbox scheme. A superbox is defined as a set of 2×2 volumes. At the 4 inner walls of a superbox the central scheme is used, whereas at the 8 outer walls the fully one-sided upwind scheme is applied. The superbox scheme has second-order consistency for an arbitrary set of 2×2 volumes, but not for a single volume. Hence, the remaining first-order error components contain only high frequencies. Therefore, this error can be eliminated simply by computing states at the volume vertices as averages over neighbouring volumes. Like the above-mentioned κ-schemes, the superbox scheme may also yield solutions with wiggles [9] .

Second-order schemes exist which, by the use of flux limiters, yield solutions without spurious non-monotonicity. With $q^l_{i+1/2,j,k}$ and $q^r_{i+1/2,j,k}$ the k-th component ($k = 1,2,3,4$) of $q^l_{i+1/2,j}$ respectively $q^r_{i+1/2,j}$, these schemes can be written as

$$q^l_{i+1/2,j,k} = q_{i,j,k} + \tfrac{1}{2}\psi(R_{i,j,k})(q_{i,j,k} - q_{i-1,j,k}), \tag{4.2a}$$

$$q^r_{i+1/2,j,k} = q_{i+1,j,k} + \tfrac{1}{2}\psi(\frac{1}{R_{i+1,j,k}})(q_{i+1,j,k} - q_{i+2,j,k}), \tag{4.2b}$$

with ψ denoting the flux limiter, and with

$$R_{i,j,k} = \frac{q_{i+1,j,k} - q_{i,j,k}}{q_{i,j,k} - q_{i-1,j,k}} . \tag{4.3}$$

Examples of flux limiters have been proposed e.g. by Van Leer and Van Albada [1] . The Van Leer limiter is defined by

$$\psi(R) = \frac{R + |R|}{R + 1}, \tag{4.4}$$

and the Van Albada limiter by

$$\psi(R) = \frac{R^2 + R}{R^2 + 1}. \tag{4.5}$$

Because of the factor $R + 1$ in (4.5), the scheme (4.2) with the Van Albada limiter can be seen as a modification of the Fromm scheme. A general discussion of flux limiters can be found in [13, 12] .

Both the Van Leer and the Van Albada limiter cannot be used near boundaries. Near boundaries we have to use schemes without limiter, which may introduce some small wiggles. Concerning both limiters, we prefer the Van Albada one, because at the upstream side of shock waves ($R \gg 1$), it renders the fully one-sided upwind scheme, which is a natural scheme in those regions.

5. Tau-extrapolation

In a multigrid environment, where solutions on more grids are available, it is natural to consider approaches as τ-extrapolation [3, 5] to further improve the accuracy of a solution. Tau-extrapolation relies on the existence of an asymptotic expansion of the truncation error for $h \to 0$, and -globally- no a-priori information about the validity of such an assumption is available. It is known that τ-extrapolation is feasible for finding a more accurate solution if the solution is sufficiently smooth.

Let the non-linear equation

$$N_h(q_h) = r_h , \tag{5.1}$$

with $q_h \in X_h$, $r_h \in Y_h$, be a discretization of

$$N(q) = r , \tag{5.2}$$

where $q \in X$, $r \in Y$. Let the discretization be such that $r_h = \bar{R}_h r$ and let the operator $N_h : X_h \to Y_h$ satisfy

$$N_h(R_h q) = r_h + \tau_h(q) , \tag{5.3}$$

$$\tau_h(q) = h^p \bar{R}_h \tau(q) + \mathcal{O}(h^{\tilde{p}}), \quad \text{for } h \to 0 , \quad \tilde{p} > p , \tag{5.4}$$

where $R_h : X \to X_h$ and $\bar{R}_h : Y \to Y_h$ are restrictions (linear surjections) and where $\tau(q)$ is independent of h. The latter assumption means that $\tau_h(q)$, the local truncation error for the solution q, satisfies an asymptotic expansion.

Further, assume that we have a sequence of nested discretizations for $h = 2^{-k} h_0$, $k = 1,2,...$; this means that, for h and $2h$, restrictions $R_{2h,h} : X_h \to X_{2h}$ and $\bar{R}_{2h,h} : Y_h \to Y_{2h}$ exist such that

$R_{2h,h}R_h = R_{2h}$ and $\bar{R}_{2h,h}\bar{R}_h = \bar{R}_{2h}$.
Then for the relative truncation error, defined by

$$\tau_{2h,h} := N_{2h}\, R_{2h,h} - \bar{R}_{2h,h}\, N_h\,, \tag{5.5}$$

we easily derive

$$\tau_{2h,h}\, R_h = \tau_{2h} - \bar{R}_{2h,h}\, \tau_h\,. \tag{5.6}$$

With the procedure as described in section 2 we construct a sequence of nested discretizations for the Euler equations. Under the assumption (5.4) we can compute $\tau_{2h,h}(q_h)$ and apply τ-extrapolation to improve the accuracy of our solution.
Combining (5.3), (5.4) and (5.6), and assuming $r_h = 0$, we see

$$\begin{aligned} h^p\, \bar{R}_{2h}\, \tau &= \frac{1}{2^p - 1}\, \tau_{2h,h}\, R_h + \mathcal{O}(h^{\tilde{p}}) \\ &= \bar{R}_{2h,h}\, \tau_h + \mathcal{O}(h^{\tilde{p}}) \\ &= 2^{-p}\, \tau_{2h} + \mathcal{O}(h^{\tilde{p}})\,, \end{aligned} \tag{5.7}$$

and hence

$$\begin{aligned} N_h(R_h q) &= h^p\, \bar{R}_h\, \tau(q) + \mathcal{O}(h^{\tilde{p}}) \\ &= h^p\, \bar{P}_{h,2h} \bar{R}_{2h}\, \tau(q) + h^p\, (I_h - \bar{P}_{h,2h}\bar{R}_{2h,h})\bar{R}_h\, \tau(q) + \mathcal{O}(h^{\tilde{p}}) \\ &= \frac{1}{2^p - 1}\bar{P}_{h,2h}\, \tau_{2h,h}(R_h q) + (I_h - \bar{P}_{h,2h}\bar{R}_{2h,h})\, \tau_h(q) + \mathcal{O}(h^{\tilde{p}}). \end{aligned} \tag{5.8}$$

We conclude that for $\bar{P}_{h,2h}$ and $\tau(q)$ such that $(I_h - \bar{P}_{h,2h}\bar{R}_{2h,h})\bar{R}_h\tau(q) = \mathcal{O}(h^{\tilde{p}-p})$ we obtain a discretization scheme consistent of $\mathcal{O}(h^{\tilde{p}})$ if we solve for q_h the equation

$$N_h(q_h) = \frac{1}{2^p - 1}\, \bar{P}_{h,2h}\, \tau_{2h,h}(q_h)\,. \tag{5.9}$$

We see that, with the accuracy restriction, we still have some freedom in the choice of $\bar{P}_{2h,h}$.

Since for a given $\tilde{q}_h \in X_h$ the $\tau_{2h,h}(\tilde{q}_h)$ is easily evaluated from (5.5), the solution of (5.9) is readily found by a Defect Correction iteration. We apply τ-extrapolation in combination with the superbox discretization $N_h^2(q_h) = 0$. The IDeC-process is now written as

$$N_h^1(q_h^{(n+1)}) = N_h^1(q_h^{(n)}) - N_h^2(q_h^{(n)}) + \tfrac{1}{3}\, \bar{P}_{h,2h}\, \tau_{2h,h}(q_h^{(n)}), \quad n = 0,1,2,\ldots\,. \tag{5.10}$$

In the initial step ($n = 0$) the right-hand side is taken equal to zero, so that (approximately) the first-order system is solved. In the next step(s) ($n = 1,\ldots,n_0$) the last term in the right-hand side of (5.10) is replaced by zero. Later in the iteration ($n > n_0$), equation (5.10) is applied in full. As (5.10) converges, we find an approximate solution $q_h = R_h\tilde{q}$, that satisfies

$$N_h^2(R_h\tilde{q}) = \bar{P}_{h,2h}\, \bar{R}_{2h}\, \tau_h(\tilde{q}) + \mathcal{O}(h^{\tilde{p}})\,, \quad \tilde{p} > 2\,. \tag{5.11}$$

6. Results

In this section we present the results for the airfoil flows mentioned in section 1. For all computations we used grids with the outer boundary at a distance of at least 10 chord lengths away from the airfoil. In all cases, we imposed the unperturbed flow at the outer boundary. (We did not overspecify the boundary conditions. I.e. at e.g. a subsonic outer boundary, we did not impose more than 3 boundary conditions at the inflow part and not more than 1 boundary condition at the outflow part.

For proper upwind discretizations, overspecification is another good possibility, which gives often faster convergence.) For all flows considered, the Kutta condition was satisfied without taking any additional precaution.

In [8] we investigated the iterative solution method for the non-linear systems $N_h^1(q_h) = r_h$, as they appear in (3.2) and (5.10). Experimentation with the non-linear multigrid (FAS) showed that the straightforward use of V-cycles with a single symmetric pre- and post-relaxation per level gives a very good convergence rate. For all results presented in [8] as well as in this paper, we used a coarsest grid with 8 volumes tangentially to each airfoil and 2 volumes normally. We notice that this is extremely coarse, if seen in the light of suspicion raised against the possibilities of a multigrid method by ERIKSSON & RIZZI [4].

In [8] it is further shown (i) that for shock-free airfoil flows, the superbox scheme yields solutions which very closely approach the exact zero-drag, and (ii) that for airfoil flows with shock(s) the upwind scheme supplied with the Van Albada limiter yields solutions with hardly any spurious non-monotonicity. Based on this knowledge, for the present problems, we used the superbox scheme for all flows which were expected to be shock-free and the upwind scheme with the Van Albada limiter for all flows which were expected to have shocks.

As solution-strategy we used for all flow problems: 10 IDeC-cycles with a single FAS-cycle per IDeC-cycle.

For the flow problems as specified for the NACA0012-airfoil (problem 2.1-2.5) we used as the finest grid a 128×32 O-type grid (fig. 1), with the outer boundary at an approximate distance from the airfoil of 100 chord lengths. As the coarsest grid we used an 8×2-grid, which yields a 5-level multigrid algorithm. For the first two problems (2.1 and 2.2) we used the superbox scheme. For the remaining problems (2.3-2.5) we used the upwind scheme with the Van Albada limiter. We did not apply τ-extrapolation to any of the NACA0012-problems.

The results as obtained for problem 2.1 are given in fig. 2a and 3. For the convergence history we present for this problem (as well as for all other problems) a graph of the residual ratio $\sum_{ij} ||(N_h^2(q_h^{(n)}))_{ij}|| \;/\; \sum_{ij} ||(N_h^2(q_h^{(0)}))_{ij}||$ versus the number of IDeC-cycles. Here n denotes the n-th iterand in (3.2) or (5.10) and $||.||$ the sum of absolute values of residual components. It should be emphasized that the figures show the convergence to the solution of the equations $N_h^2(q_h) = 0$ or $N_h^2(q_h) = \tau_h$, which -as mentioned before- we do not want to solve exactly, but only up to truncation error accuracy by a few iterations of (3.2) or (5.10). A few of these iterations are sufficient to obtain higher-order accuracy. The occurrence of a new right-hand side in each IDeC-cycle inhibits to show this efficient convergence to a higher-order accurate solution by means of a residual history. In this way, the mandatory figures give no relevant information. They only show that finally convergence to the fixed point of the iteration (3.2b) or (5.10) can be attained. Fig. 2a shows that this is the case for problem 2.1 indeed. (As a starting point for all convergence histories we take a first-order accurate approximation, $q_h^{(0)}$.) More representative for the computational rate are e.g. the convergence histories of the lift and drag. For the various airfoil flows considered here, as well as for those in [8], we observed a (practical) convergence of both lift and drag in (on an average) 5 IDeC-cycles. As values of the lift, drag and moment (with respect to the quarter-chord point) we found for problem 2.1 (after the 10th IDeC-cycle): $c_l = 0.3291$, $c_d = 0.0007$ and $c_m = -0.0021$. In fig. 3 distributions are given of M, c_p and $\Delta\Sigma = (p/p_\infty)(\rho/\rho_\infty)^{-\gamma} - 1$. The maximum value of $\Delta\Sigma$ found is 0.0130 at the airfoil's nose.

For problem 2.2 we failed to find a steady solution. We succeeded only in performing a single FAS-cycle for the first-order discretization $N_h^1(q_h) = 0$. Since this problem was expected to yield an unsteady solution, and since our solution method cannot compute unsteady flows, we consider this result satisfactory.

For problem 2.3 our results are given in fig. 2b and 4. Clearly visible in all graphs showing some solution component, is the good capturing of both shock waves. Further, the iso-Mach-number and iso-entropy distribution (fig. 4a and 4c) clearly show the good capturing of the slip line leaving the

airfoil's tail, whereas the iso-pressure distribution (fig. 4b) shows the perfect smoothness of the pressure across the slip line (up to the airfoil's tail). Spreading of all three discontinuities in radial direction is only due to the grid enlargement in this direction. The iso-Mach-number and surface pressure distribution agree perfectly with the results obtained by Schmidt & Jameson [14] on a 320×64 (!) O-type grid. As values for the lift, drag and moment we found for this problem: $c_l = 0.3565$, $c_d = 0.0582$ and $c_m = -0.1209$ (again after the 10th IDeC-cycle). As maximum value for $\Delta\Sigma$ we found 0.0088 at the airfoil's nose.

For both problem 2.4 and 2.5, we performed 3 computations. First, we solved both problems for $\alpha = 0$ (unperturbed). Next, we solved them by starting the solution process with some positive $\alpha = \epsilon << 1$ rad., and by changing this value (discontinuously) to $\alpha = 0$ after the 2nd IDeC-cycle. Finally, we solved both problems by starting with $\alpha = -\epsilon$, and by replacing it by $\alpha = 0$ (again) after the 2nd IDeC-cycle. We did this for several values of ϵ. Both for 2.4 and 2.5 we obtained for all computations the same symmetric solution (fig. 5 and 6). We did not observe any hysteresis. For c_d we found 0.0145 for problem 2.4, and 0.0206 for problem 2.5. As maximum value for $\Delta\Sigma$ at the airfoil's nose we found 0.0103 for both problems.

For the problem specified for the Korn-airfoil (problem 3.1) we computed: (i) a second-order accurate solution using a 128×32 O-type grid (fig. 7a) and the superbox scheme, and (ii) a more accurate solution using a 64×16 (!) O-type grid (fig. 7b), the superbox scheme and τ-extrapolation. In both cases we did not use the airfoil geometry specified by the Workshop-organisation but a (slightly) more detailed geometry provided by the NLR. For both computations we took the outer boundary at an approximate distance from the airfoil of 100 chord lengths, and used as coarsest grid an 8×2-grid. This yielded a 5-level resp. 4-level algorithm. For the first computation we used again 10 IDeC-cycles with 1 FAS-cycle per IDeC-cycle. For the latter we used first 5 IDeC-cycles with the last term in the right-hand side of (5.10) equal to zero, and next 5 IDeC-cycles with the full right-hand side. Also here we used only a single FAS-cycle per IDeC-cycle. The results obtained for both computations are given in fig. 8, 9 and 10. In most graphs showing some solution component, the less severe potential-flow break-down as obtained with τ-extrapolation is clearly visible. As values for c_l we found 0.5831 for the solution obtained without τ-extrapolation on the 128×32 grid, and 0.5872 (!) for the solution obtained with τ-extrapolation on the 64×16-grid. As values for c_d and c_m we found for the latter solution: 0.0036 respectively -0.1527. As extrema for $\Delta\Sigma$ at the airfoil's nose, we found 0.0167 for the first solution and -0.0274 for the latter.

For the bi-NACA0012-airfoil we only considered problem 4.3. As second-order scheme we used the upwind scheme with Van Albada limiter. As grid we used the 64×24 single-domain grid as shown in fig. 11. A 4-level multigrid algorithm was used with a coarsest grid of 8×3 cells. In fig. 12a and 13a we show results for 10 IDeC-cycles without τ-extrapolation. The Mach-number distributions in fig. 13a do not show the expected transonic internal flow. The same flow was recomputed with one cycle *without*, followed by 9 cycles *with* τ-extrapolation. The results obtained are presented in fig. 13b-13d. Here, the Mach-number distributions (fig. 13b) do show a transonic internal flow. The rather large difference between the internal Mach-number distribution as obtained without and with τ-extrapolation is due to its large sensitivity to the Mach-number distribution just upstream of the bi-airfoil. A spurious feature in both iso-Mach-number distributions is the boundary-layer-like behaviour along the airfoil surfaces. This is related to the relatively large spurious entropy generation at the airfoil noses. Cause of this is the non-smoothness of the grid at these points. For the solution obtained with τ-extrapolation we found as values for the lift, drag and moment: for the lower airfoil $c_l = 1.1233$, $c_d = 0.0936$ and $c_m = -0.0624$; for the upper airfoil $c_l = 0.0006$, $c_d = -0.0329$ and $c_m = 0.0972$. Remarkable is the applicability of the τ-extrapolation technique despite of the non-smoothness in the grid and the presence of a shock wave.

For the airfoil flows computed, we needed an average of 5 IDeC-cycles to drive the lift coefficients to within 0.5% of their final value. On the single-pipe CDC Cyber 205 (on which we performed all our computations), this took for a 128×32-grid $\sim$ 100 sec (i.e. $\sim$ 5 msec per volume per FAS-cycle) in

scalar mode, and $\sim$ 50 sec in vector mode. In scalar mode, we obtain the same computational rate per cell, independent of the gridsize. The convergence rates of both FAS (inner iteration) and IDeC (outer iteration) appear to be independent of the number of volumes on the finest grid (grid-independence). This implies that the computational effort to find a solution is directly proportional to the number of cells in the grid.

We did not extensively tune our code for use on vector computers since the algorithm is not particularly well suited for vectorization. However, for large computations, where all data cannot be kept in core, the small number of iteration cycles required (5 IDeC-cycles on an average) results in a small number of out-of-core data transports. For most Euler codes this is significantly more. Since IO-times rather than CPU-times may be the bottleneck in large scale computations on vector computers, we consider this feature as another advantage of the multigrid method used.

7. Conclusions

To compute airfoil flows, we used two second-order accurate schemes in combination with Iterative Defect Correction (IDeC) and non-linear multigrid (FAS). As second-order schemes we used a scheme without a flux limiter (the superbox scheme) and a scheme with such a limiter (the upwind scheme supplied with the Van Albada limiter). For 2 out of the 7 flow problems considered, we applied a τ-extrapolation technique to further improve the accuracy.

The superbox scheme was found to be a suitable second-order scheme for the computation of flows without shock(s), whereas the scheme with the Van Albada limiter was better suited for flows with shock(s).

Concerning the application of τ-extrapolation, it appeared that it enables us to find an accurate solution already on a really coarse grid.

For a second-order accurate multigrid computation of airfoil flows with the steady Euler equations, IDeC is found to be an efficient method. It appeared that for the solution (up to truncation error) of the discretized non-isenthalpic equations, it is sufficient to perform only a few IDeC-cycles, with only 1 FAS-cycle per IDeC-cycle. Given the grid-independence of both FAS and IDeC, an extension to 3-D seems feasible.

It is an important property of the present computational method that it is completely parameter-free: it does not require any tuning of parameters.

Acknowledgement

The authors want to thank W.J. Boerstoel and A. Kassies from the Dutch National Aerospace Laboratory, NLR, for their kind providing of O-type grids for the NACA0012- and Korn- airfoil.

References

1. G.D. Van Albada, B. Van Leer, and W.W. Roberts (1982). A comparative study of computational methods in cosmic gasdynamics, *Astron. Astrophys.*, 108, 76-84.
2. K. Böhmer, P. Hemker, and H.J. Stetter (1984). The defect correction approach, *Computing Suppl.*, 5, 1-32.
3. A. Brandt (1982). Guide to multigrid development, in *Multigrid Methods*, 220-312, ed. W. Hackbusch and U. Trottenberg, Springer Verlag.
4. L.E. Eriksson and A. Rizzi (1985). Computer-aided analysis of the convergence to steady state of discrete approximations to the Euler equations, *J. Comp. Phys.*, 57, 90-128.
5. W. Hackbusch (1985). *Multigrid Methods and Applications*, 4, Springer Verlag, Berlin, Heidelberg.
6. P.W. Hemker (1985). *Defect correction and higher order schemes for the multigrid solution of the steady Euler equations*, CWI Report NM-R8523, To appear in Proceedings Multigrid Conference, Cologne, Oct. 1985.
7. P.W. Hemker and S.P. Spekreijse (1985). *Multiple grid and Osher's scheme for the efficient solution*

of the steady Euler equations, CWI Report NM-R8507, To appear in Appl. Num. Math. 1986.

8. B. KOREN (1986). *Evaluation of second order schemes and defect correction for the multigrid computation of airfoil flows with the steady Euler equations*, CWI Report NM-R8616.
9. B. KOREN (1986). *Euler flow solutions for a transonic windtunnel section*, CWI Report NM-R8601.
10. S. OSHER (1981). Numerical solution of singular perturbation problems and hyperbolic systems of conservation laws, in *Analytical and Numerical Approaches to Asymptotic problems in Analysis*, ed. O. Axelsson, L.S. Frank and A. van der Sluis, North Holland Publ. Comp..
11. S.P. SPEKREIJSE (1985). *Second order accurate upwind solutions of the 2-D steady state Euler equations by the use of a defect correction method*, CWI-report NM-R8520.
12. S.P. SPEKREIJSE (1986). *Multigrid solution of monotone second-order discretizations of hyperbolic conservation laws*, CWI-report NM-R8611.
13. P.K. SWEBY (1984). High resolution schemes using flux limiters for hyperbolic conservation laws, *SIAM J.Numer.Anal.*, 21, 995-1011.
14. H. VIVIAND (1985). Numerical solutions of two-dimensional reference test cases, in *Test Cases for Inviscid Flow Field Methods*, ed. H. Yoshihara and P. Sacher, AGARD, AGARD Advisory Report No.211.

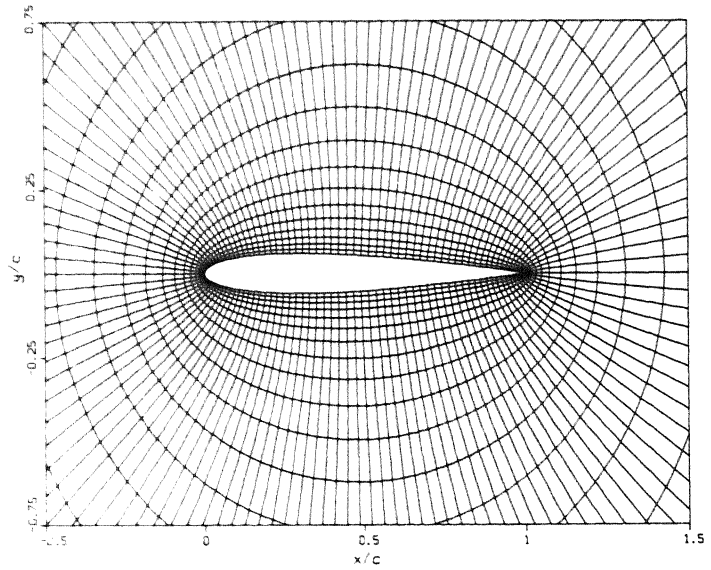
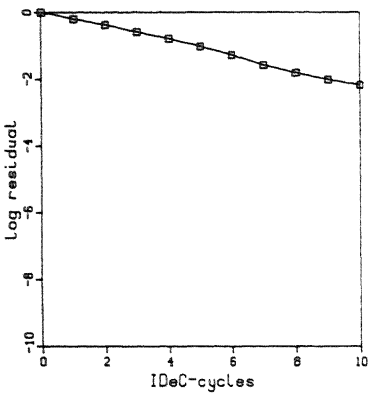
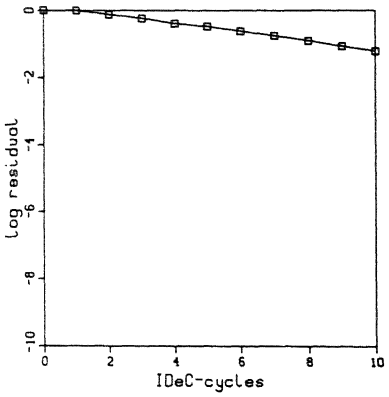


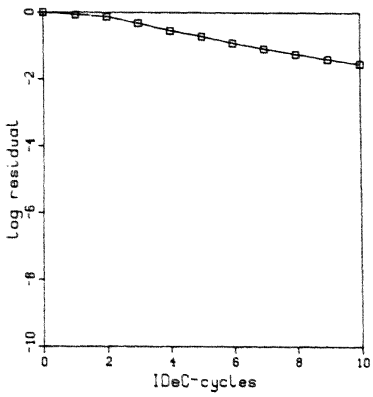
Fig. 1: 128×32-grid NACA0012-airfoil.



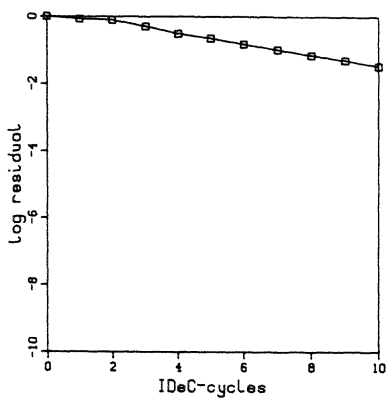
a. $M_\infty = 0.63$, $\alpha = 2^o$ (problem 2.1).



b. $M_\infty = 0.85$, $\alpha = 1^o$ (problem 2.3).

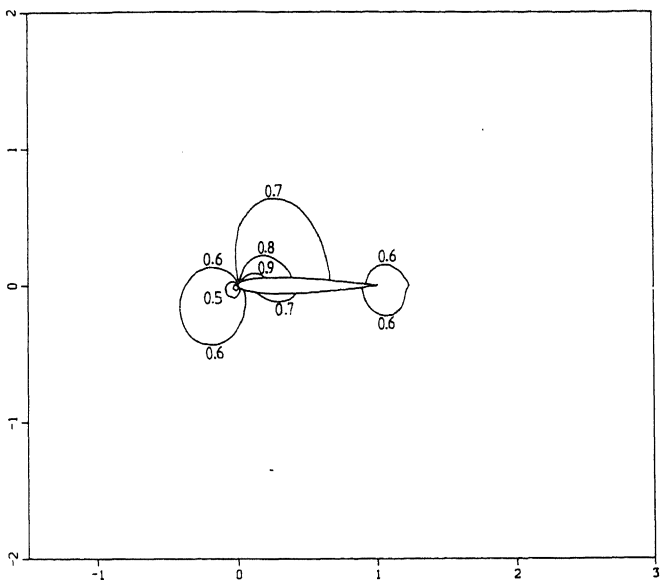


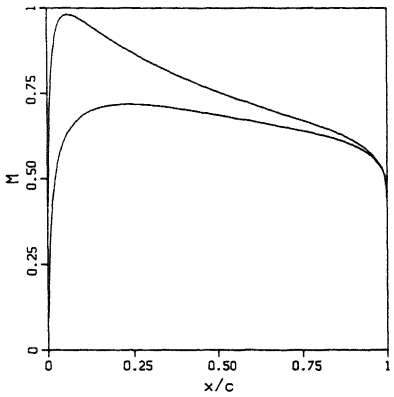
c. $M_\infty = 0.81$, $\alpha = 0^o$ (problem 2.4).



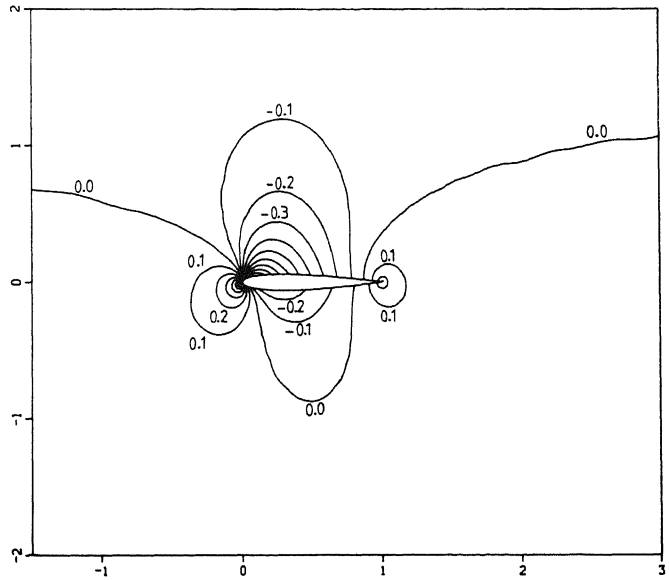
d. $M_\infty = 0.82$, $\alpha = 0^o$ (problem 2.5).

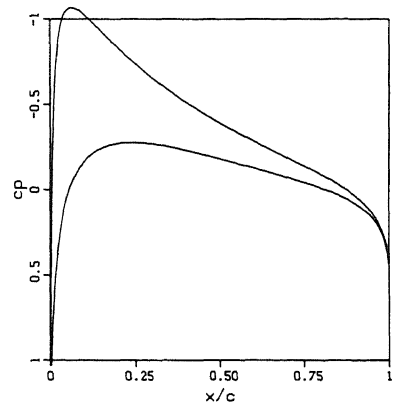
Fig. 2: Convergence histories NACA0012-airfoil.



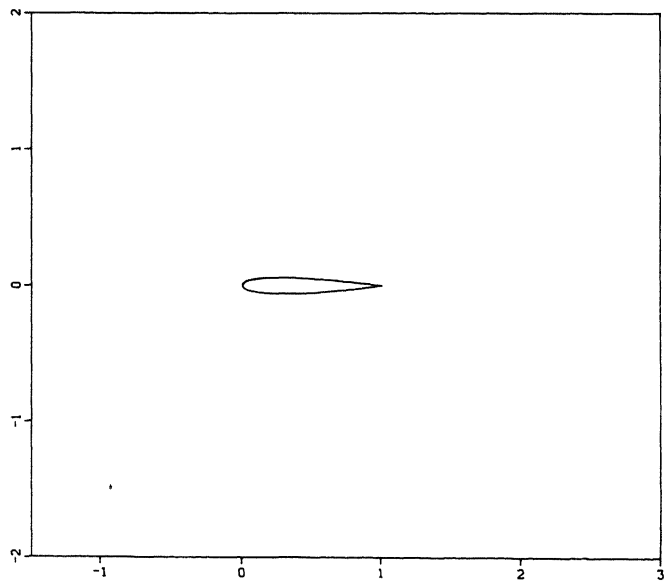


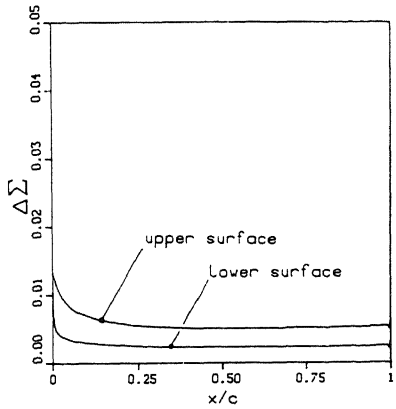
a. Mach number distributions.





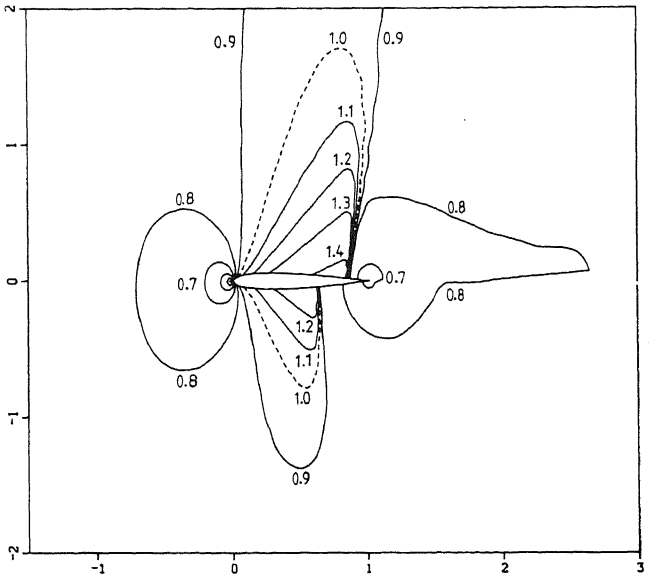
b. Pressure distributions.

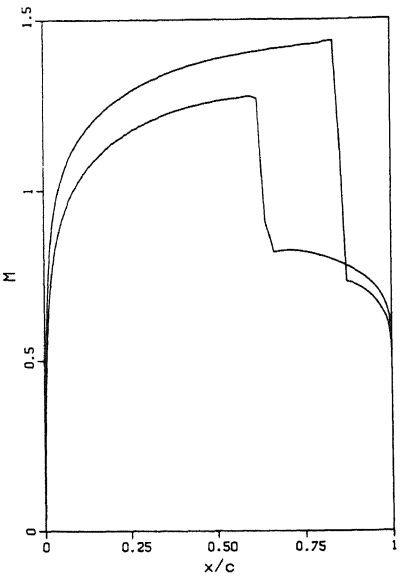




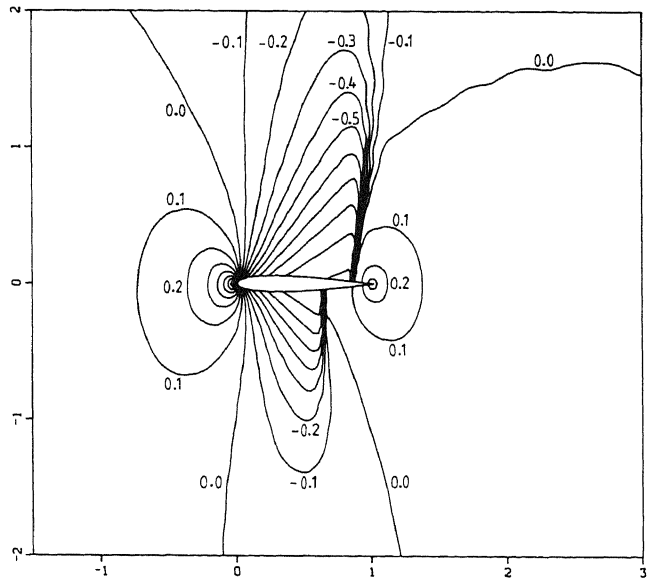
c. Entropy distributions.

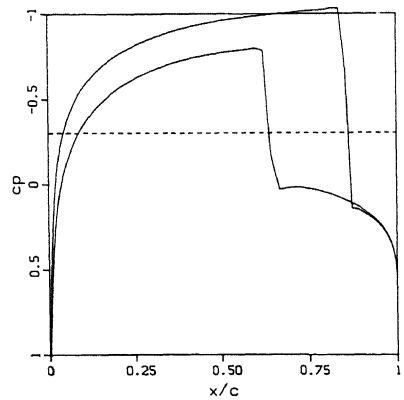
Fig. 3: Solution NACA0012-airfoil at $M_\infty = 0.63$, $\alpha = 2^o$ (problem 2.1).



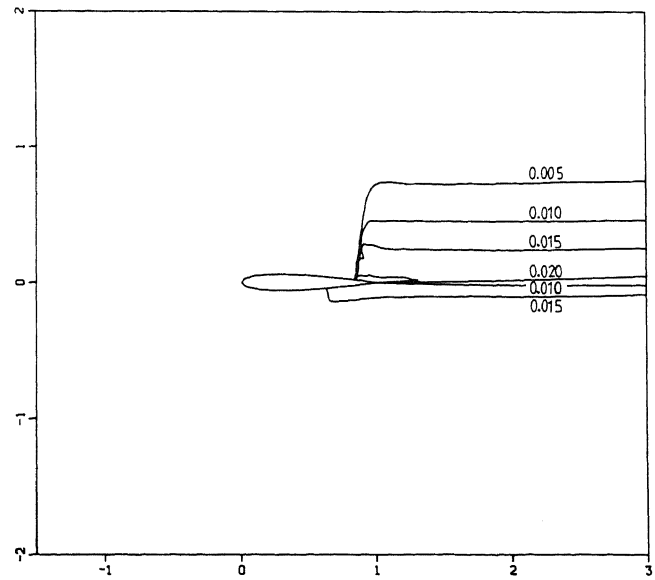


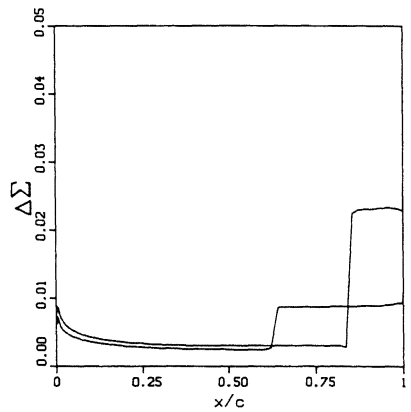
a. Mach number distributions.





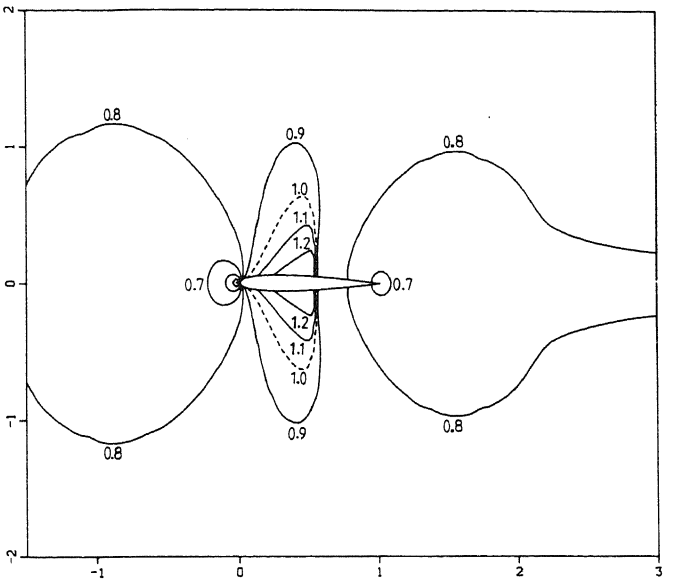
b. Pressure distributions.

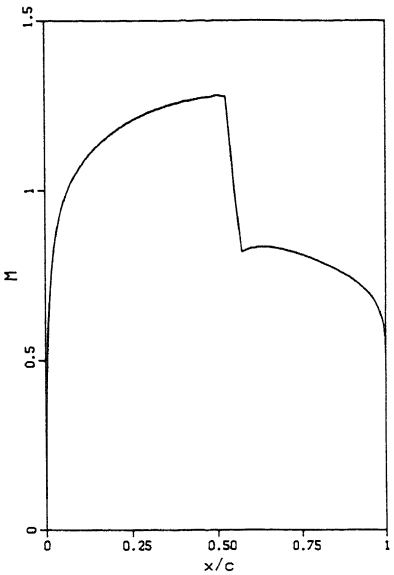




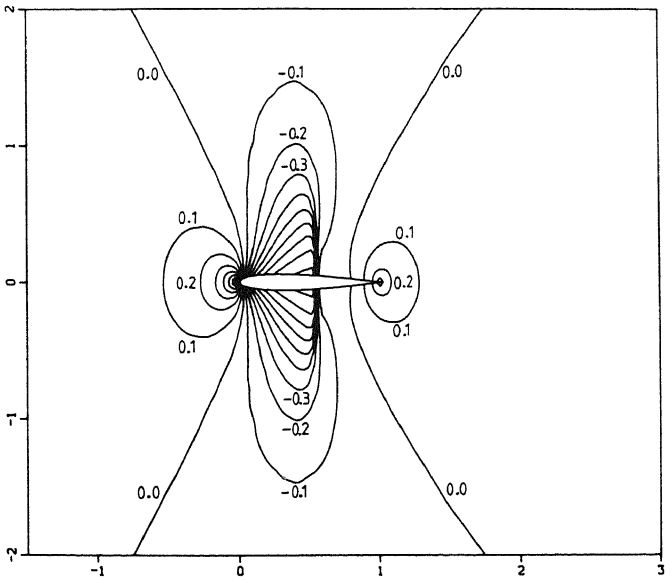
c. Entropy distributions.

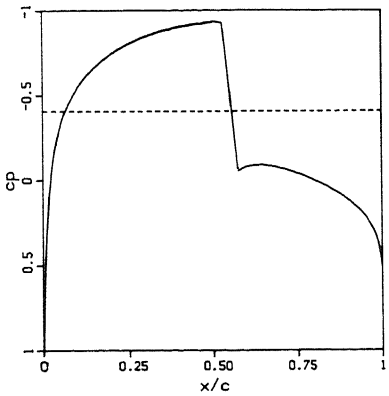
Fig. 4: Solution NACA0012-airfoil at $M_\infty = 0.85$, $\alpha = 1^o$ (problem 2.3).



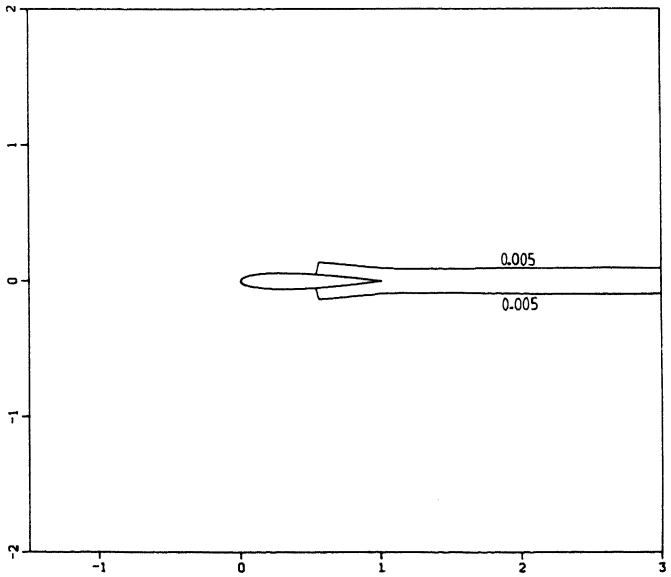


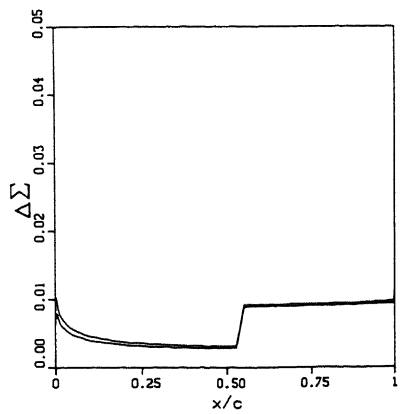
a. Mach number distributions.





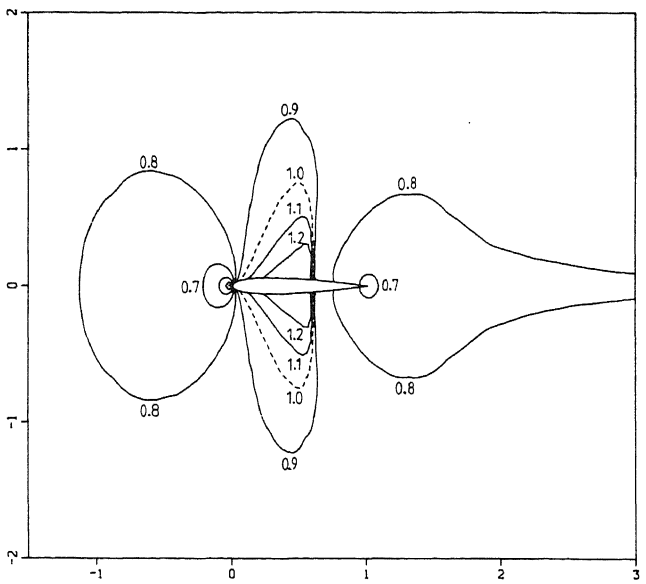
b. Pressure distributions.

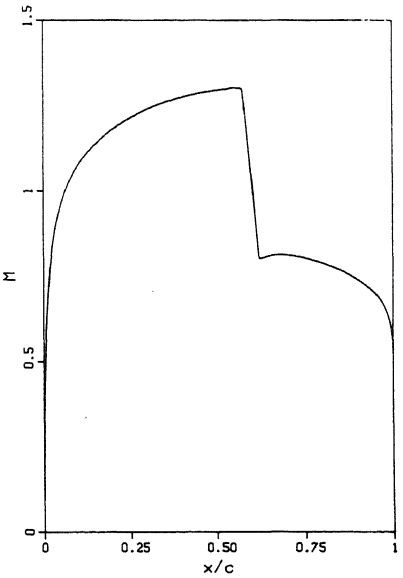




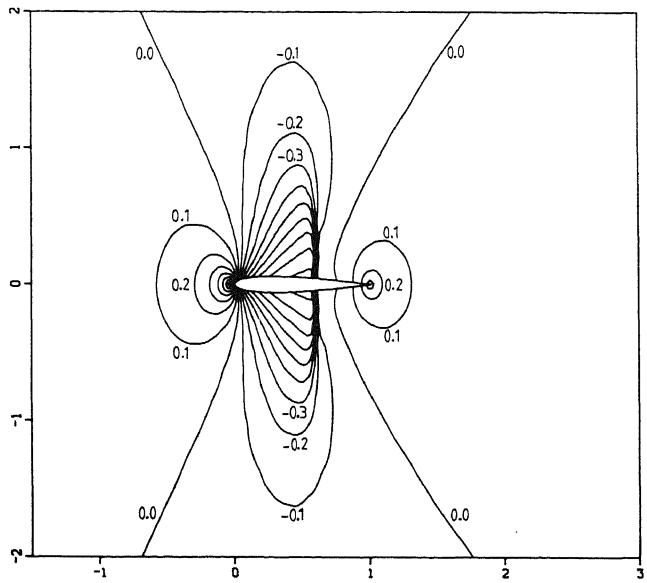
c. Entropy distributions.

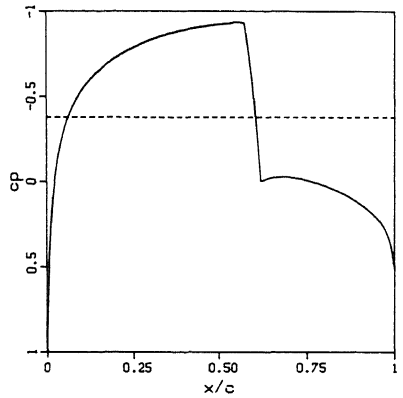
Fig. 5: Solution NACA0012-airfoil at $M_\infty = 0.81$, $\alpha = 0^o$ (problem 2.4).



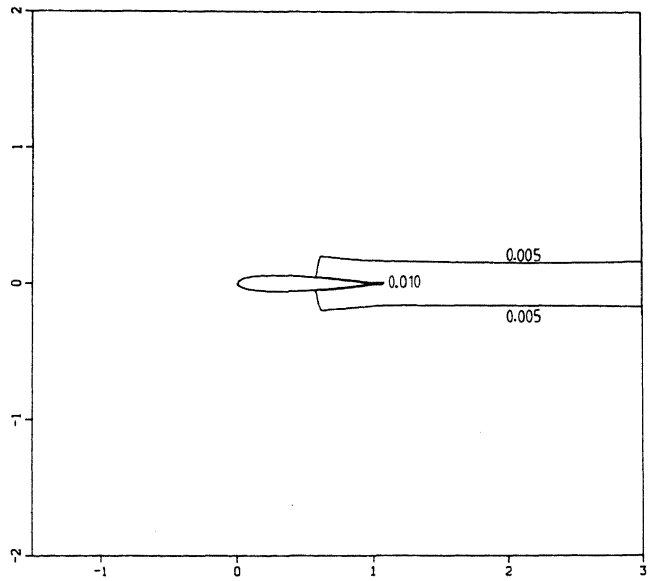


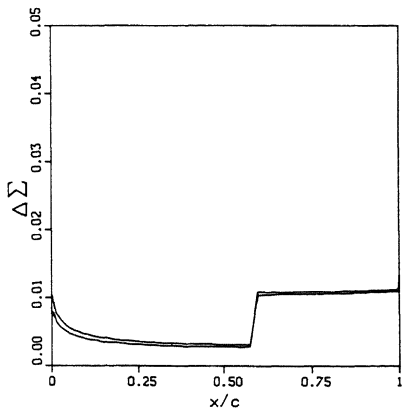
a. Mach number distributions.





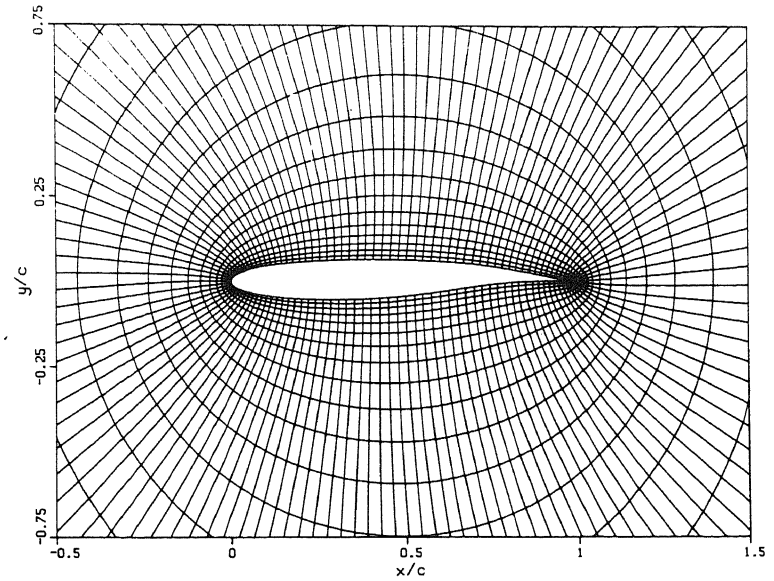
b. Pressure distributions.



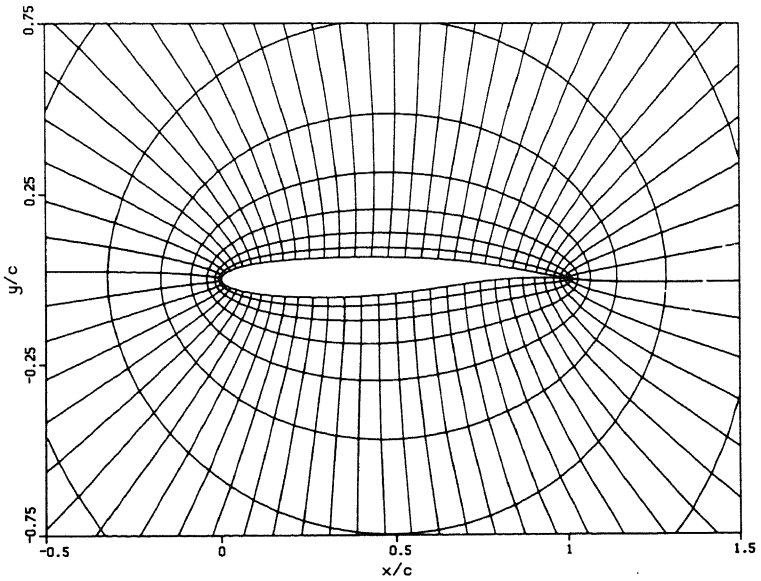


c. Entropy distributions.

Fig. 6: Solution NACA0012-airfoil at $M_\infty = 0.82$, $\alpha = 0^o$ (problem 2.5).

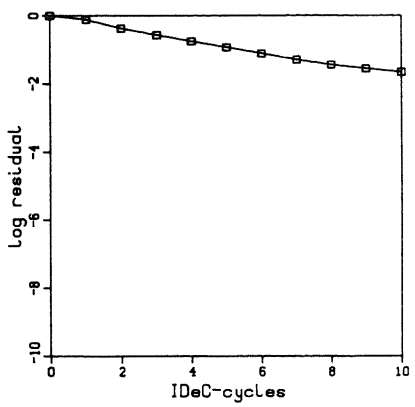


a. 128×32-grid.

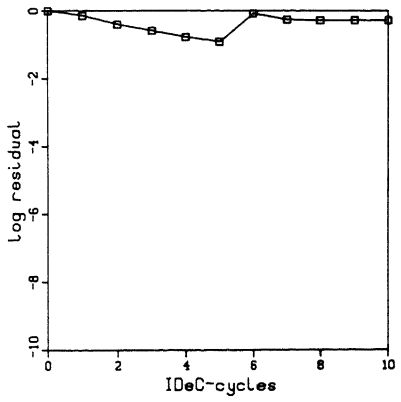


b. 64×16-grid.

Fig. 7: Grids Korn-airfoil.

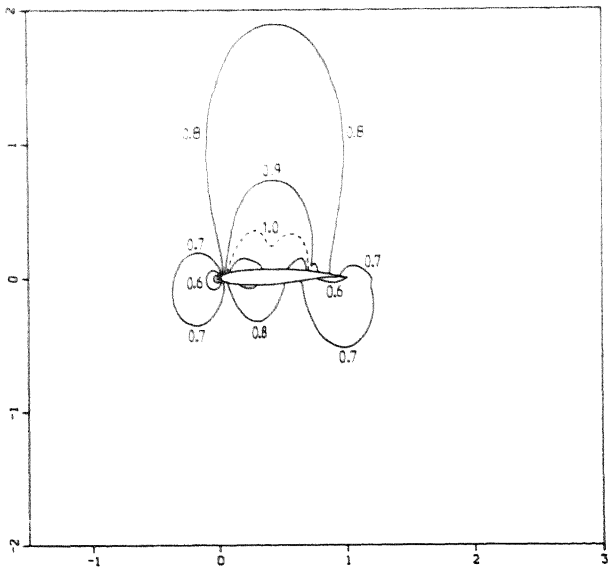


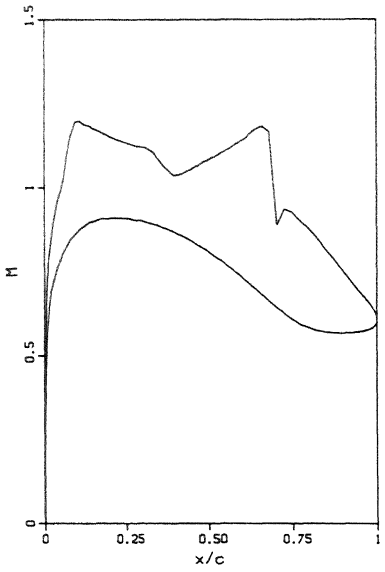
a. *Without* τ-extrapolation.



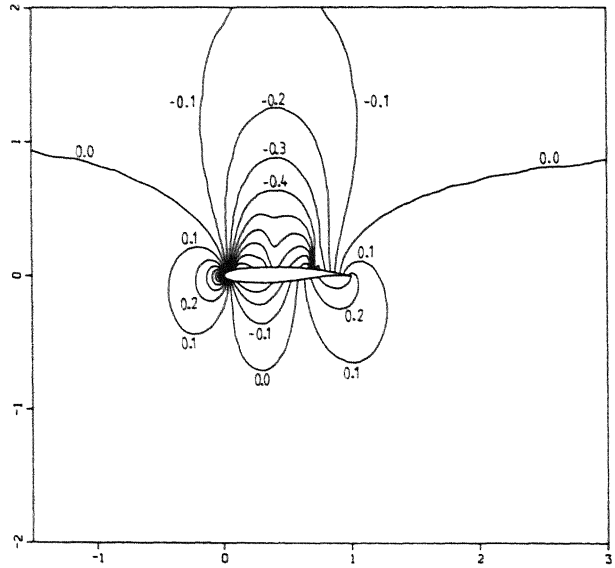
b. *With* τ-extrapolation.

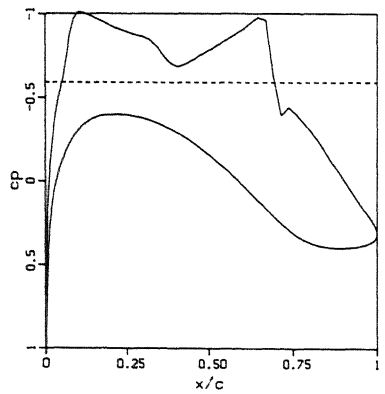
Fig. 8: Convergence histories Korn-airfoil at $M_\infty = 0.75$, $\alpha = 0^o$ (problem 3.1).



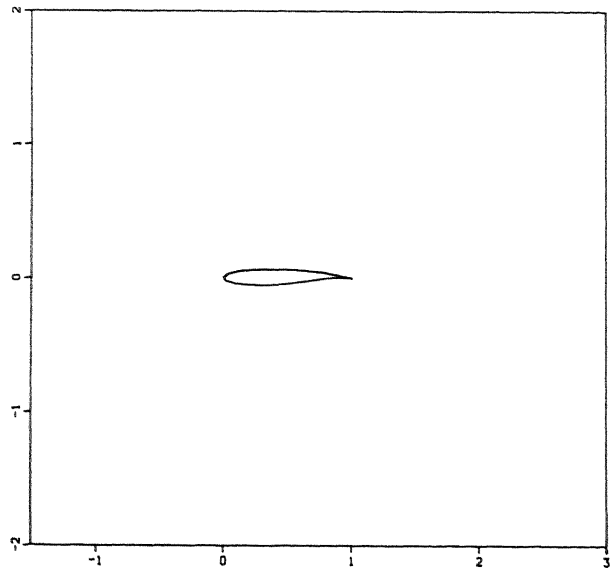


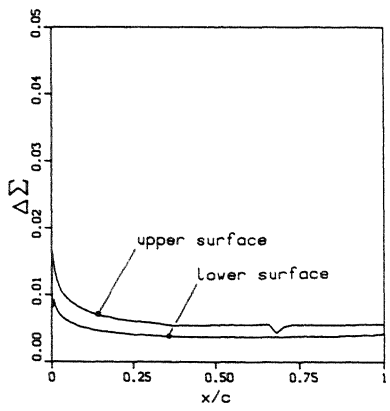
a. Mach number distributions.





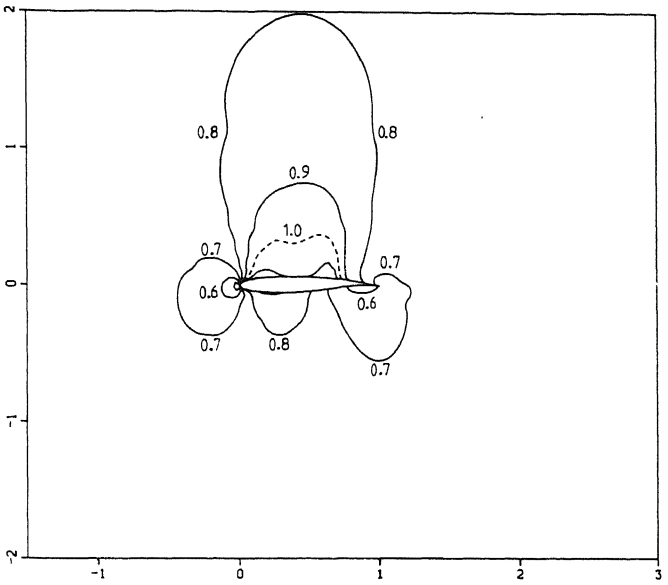
b. Pressure distributions.

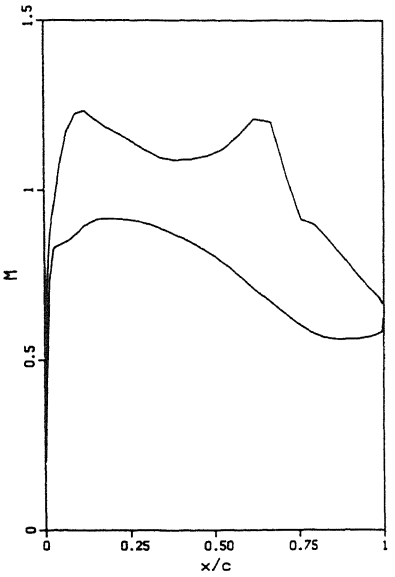




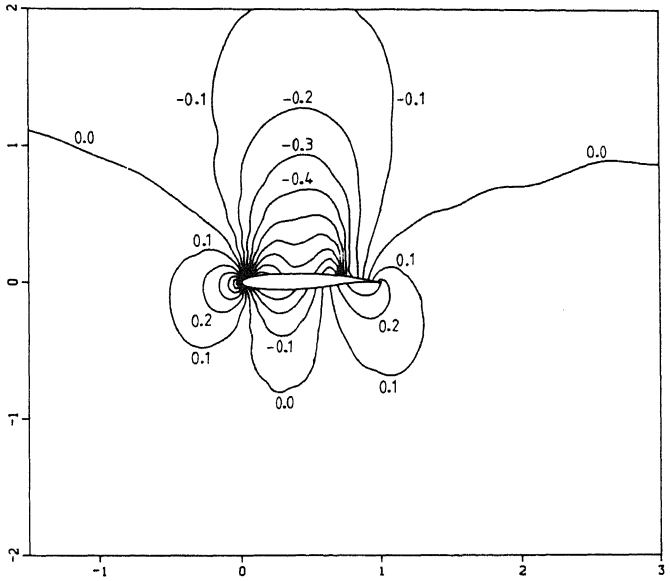
c. Entropy distributions.

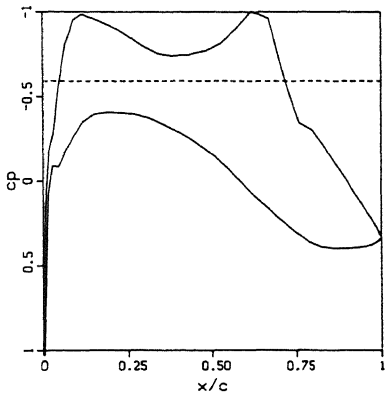
Fig. 9: Solution Korn-airfoil at $M_\infty = 0.75$, $\alpha = 0^o$ (problem 3.1); 128×32-grid, *without* τ-extrapolation.



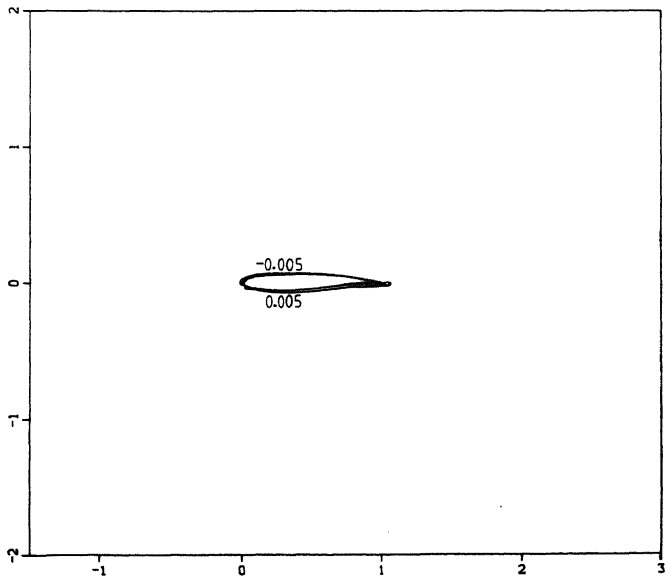


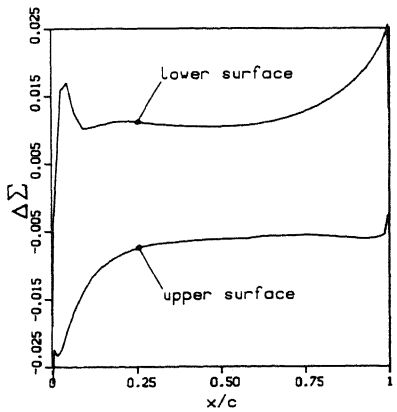
a. Mach number distributions.





b. Pressure distributions.





c. Entropy distributions.

Fig. 10: Solution Korn-airfoil at $M_\infty = 0.75$, $\alpha = 0^o$ (problem 3.1); 64×16-grid, *with* τ-extrapolation.

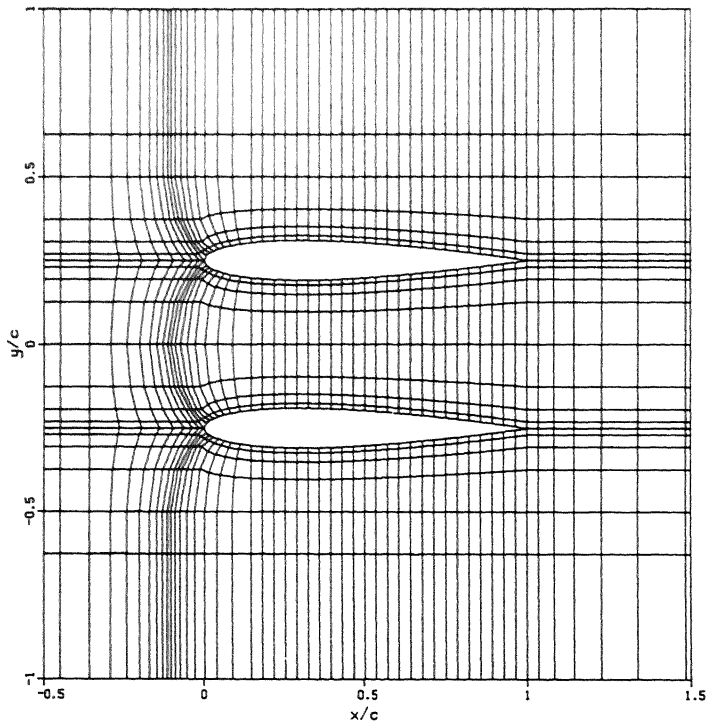
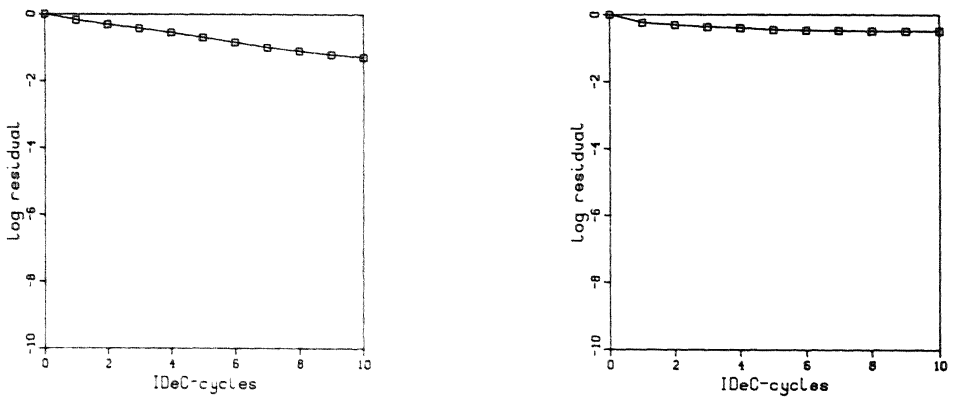


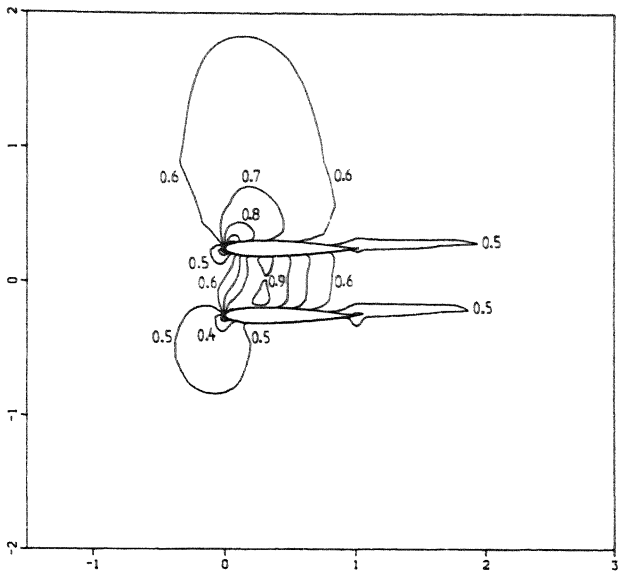
Fig. 11: 64×24-grid bi-NACA0012-airfoil.

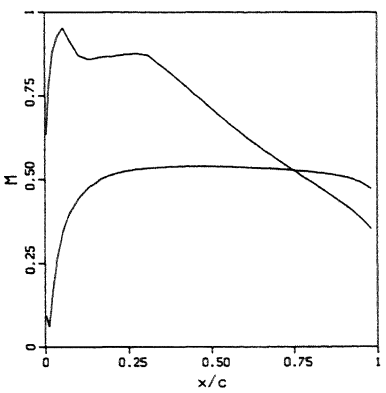


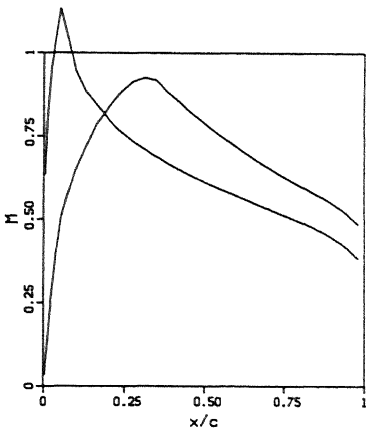
a. *Without* τ-extrapolation.

b. *With* τ-extrapolation.

Fig. 12: Convergence histories bi-NACA0012-airfoil at $M_\infty = 0.55$, $\alpha = 6^o$ (problem 4.3).



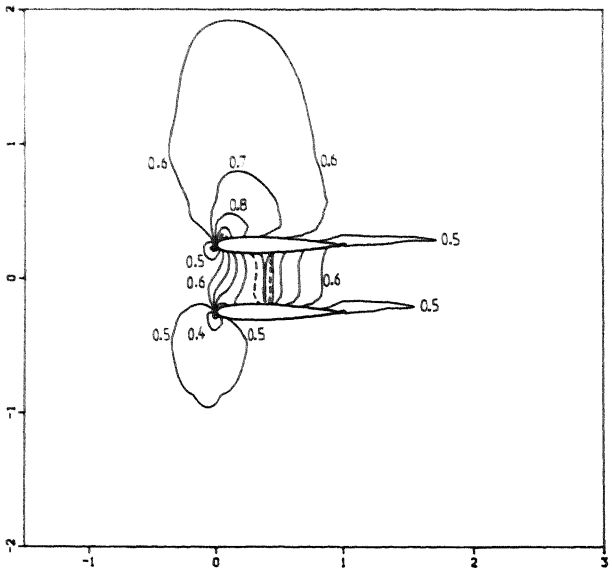


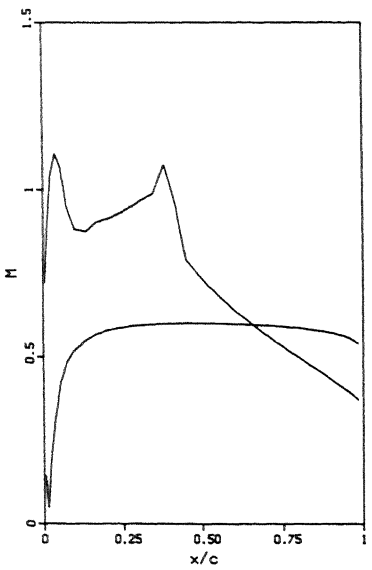


Lower airfoil.

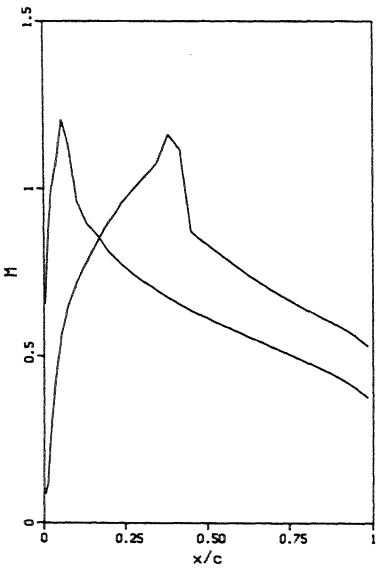
Upper airfoil.

Fig. 13a: Mach number distributions bi-NACA0012-airfoil at $M_\infty = 0.55$, $\alpha = 6^o$ (problem 4.3); *without* τ-extrapolation.



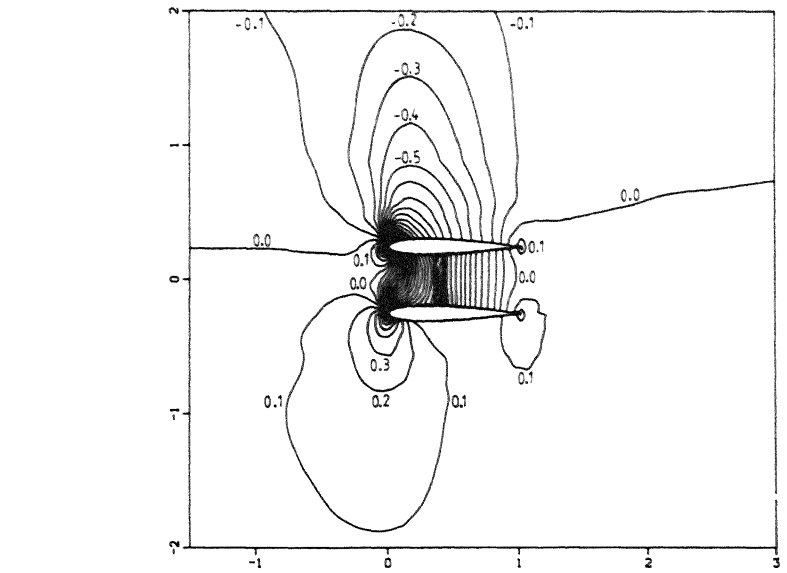


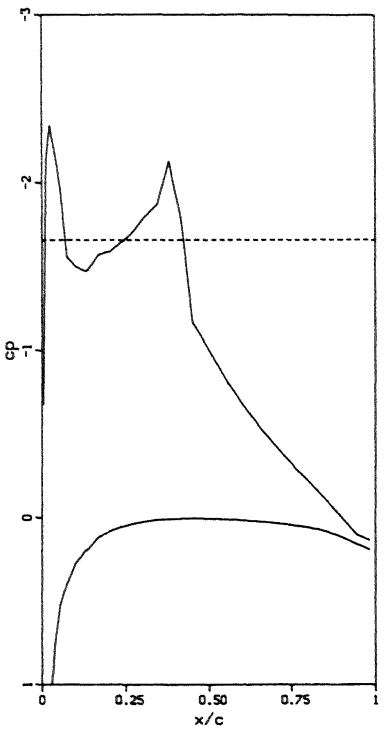
Lower airfoil.

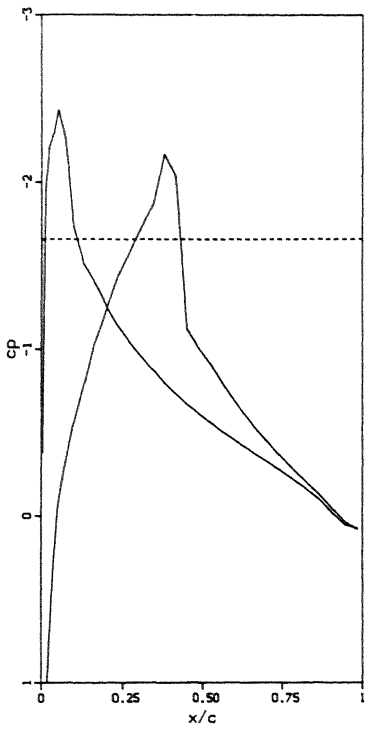


Upper airfoil.

Fig. 13b: Mach number distributions bi-NACA0012-airfoil at $M_\infty = 0.55$, $\alpha = 6^o$ (problem 4.3); *with* τ-extrapolation.



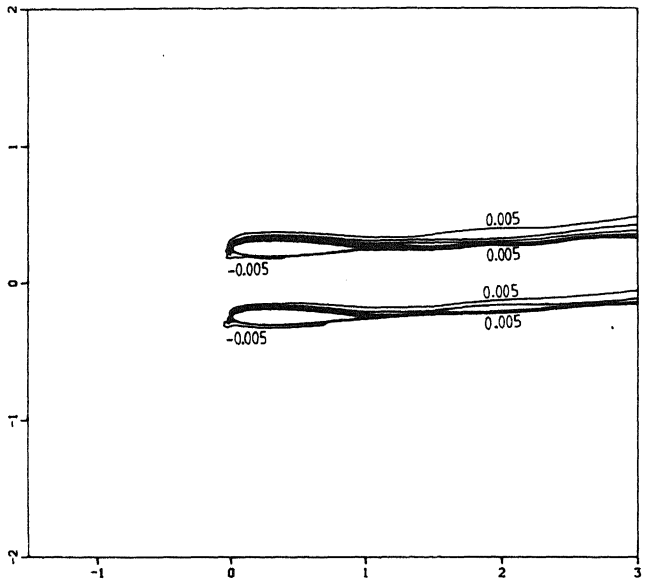


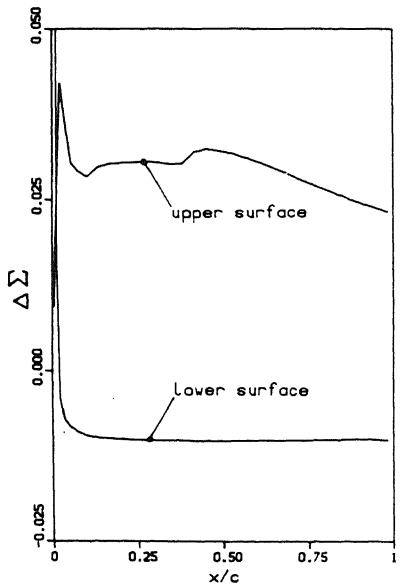


Lower airfoil.

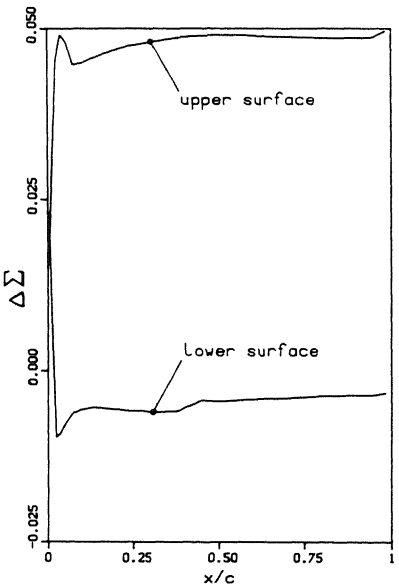
Upper airfoil.

Fig. 13c: Pressure distributions bi-NACA0012-airfoil at $M_\infty = 0.55$, $\alpha = 6^o$ (problem 4.3); *with* τ-extrapolation.





Lower airfoil.



Upper airfoil.

Fig. 13d: Entropy distributions bi-NACA0012-airfoil at $M_\infty = 0.55$, $\alpha = 6^o$ (problem 4.3); *with* τ-extrapolation.