JOURNAL OF COMPUTATIONAL PHYSICS 102, 114–127 (1992)

Adaptive Spectral Methods with Application to Mixing Layer Computations

H. GUILLARD

INRIA, Centre de Sophia-Antipolis, 2004 Av des lucioles, 06565 Valbonne, France

AND

J. M. MALÉ AND R. PEYRET

Laboratoir de Mathématiques, CNRS UA-168, Université de Nice, Parc Valrose, 06034 Nice, France and INRIA, Centre de Sophia-Antipolis, 2004 Av des lucioles, 06565 Valbonne, France

Received June 7, 1990; revised October 17, 1991

This paper reports some experiments on the use of adaptive Chebyshev pseudospectral methods for compressible mixing layer computations. Different functionals measuring the optimality of the polynomial approximation are discussed and compared. In particular, we address the problem of the practical computation of the various functionals. The utility of the self-adaptive method is then demonstrated by some examples from compressible mixing layer calculations. © 1992 Academic Press, Inc.

I. INTRODUCTION

Spectral methods are among the most powerful methods to solve partial differential equations with smooth solutions. However, for problems whose solutions exhibit large derivatives, such as those arising, for instance, in fluid mechanics or combustion studies, these methods are difficult to apply when the large gradients are not localized near the boundaries, and specialized techniques like domain-decomposition or change of variables are required (see, e.g., [12, 21]). In [13, 4], an adaptive spectral method was introduced to deal with such problems. The principle of the method is to write the equations in a computational domain through a parametrized mapping. This change of variables is then adapted to the computed solution by minimizing a certain functional of the solution. In [13, 4], motivated by some theoretical estimates of the convergence rate of the polynomial approximation, this functional was chosen as the weighted Sobolev norm in H^2_ω. In these papers, numerical results of some relevant combustion problems were presented showing the effectiveness of the method. In [3], an alternate functional based on the minimization of the leading term of the interpolation error was introduced. These techniques prove to be very efficient in combustion studies and beautiful results including the computation of cellular flames [3] and pulsating solutions in solid fuel combustion [6, 5] were obtained with them. Some recent studies [1] also show that these methods can help in computing the solutions of (almost) discontinuous problems described by hyperbolic equations.

In this paper, we report some numerical experiments on the use of adaptive Chebyshev techniques in the computation of compressible mixing layers. Spectral Chebyshev calculations of mixing layers for compressible as well as incompressible fluids are difficult because large gradients are located in the middle part of the computational domain. A standard spectral procedure cannot be utilized and other techniques must be employed, namely domain-decomposition or adequate mapping that expands the region of large derivatives. This latter technique is considered in the present paper which compares and discusses the use of different functionals whose minimisation defines the coordinate transform: the weighted Sobolev H^2_ω norms proposed in [13, 4], the non-weighted H^2 norm, as well as the functional proposed in [3] are tested. This latter choice deserves some comments and in particular we show that this functional is in fact related to the derivation in the Fourier space of the trigonometric polynomial associated to the Chebyshev approximation of the function under consideration. The results confirm the utility of the adaptive techniques and show that, at least in the present case, the choice of a specific functional is not too critical. However, we found that the actual computation of the functional is critical and must be carefully done. In particular, it is often useful to apply some filtering to the quantity to be minimized.

The outlines of the paper are as follows: In Section II, we make precise the principle of the adaptive method and

0021-9991/92 $5.00

discuss the use of different functionals. Section III deals with the practical computation of the functionals. Finally, in Section IV, we illustrate the use of these methods by some relevant results from compressible mixing layer computations.

II. THE ADAPTIVE SPECTRAL METHOD

We are interested in solving evolution equations of the type:

$$\frac{\partial u}{\partial t} + Au = 0, \qquad x_1 \leqslant x \leqslant x_2 \tag{1}$$

where A is some spatial differential operator. It is known that when the solution of (1) exhibits rapid spatial changes, spectral methods suffer from the appearance of oscillations that can destroy the accuracy and stability of the computation. In order to improve the capability of these methods to compute such solutions, we introduce a mapping $x = f(\mathbf{a}, \xi)$, where f is a known function and $\mathbf{a}$ is a parameter vector to be determined. Equation (1) is then transformed by this mapping to obtain another evolution equation,

$$\frac{\partial \bar{u}}{\partial t} + \bar{A}\bar{u} = 0, \qquad -1 \leqslant \xi \leqslant +1, \tag{2}$$

that will be solved by a standard Chebyshev spectral method (for instance, a collocation method). We call the x-space the physical space while the ξ-space is called the computational space. Let us assume that by some means, the value of the parameter $\mathbf{a}$ has been chosen for the time levels t^j, $j = 0, ..., n$, and that we have obtained the approximate solutions of (1) and (2) for these time levels. Thus at $t = t^n$, the approximate solution is available as a finite sum of Chebyshev polynomials in the computational space,

$$\bar{u}^n(\xi) = \sum_{k=0}^{K} a_k T_k(\xi) \tag{3}$$

and, in terms of transformed Chebyshev polynomials in the physical space,

$$u^n(x) = \bar{u}^n(f^{-1}(\mathbf{a}, x)) = \sum_{k=0}^{K} a_k T_k(f^{-1}(\mathbf{a}, x)). \tag{4}$$

We then have to compute the approximate solution at $t = t^{n+1}$. If the solution is rapidly evolving, it is likely that there will be a time index n where the coordinate transform have to be changed in order to follow the behavior of the solution. For this, we introduce a new parameter $\mathbf{b}$ and a new computational space whose independent variable is denoted by η (see Fig. 1) and consider the function $\hat{u}$ defined by

$$\hat{u}^n(\eta) = u^n(f(\mathbf{b}, \eta)) = \bar{u}^n(f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta))). \tag{5}$$

For each η, the value of $\hat{u}$ can be computed by the following expression:

$$\hat{u}^n(\eta) = \sum_{k=0}^{K} a_k T_k(f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta))). \tag{6}$$

(Note that, in general, $\hat{u}$ is not a polynomial except if $\mathbf{a} = \mathbf{b}$). The principle of the adaptive method is then to choose $\mathbf{b}$ in such a way that $\hat{u}$ would have a nice representation in the Chebyshev basis $\{T_k(\eta)\}$, $k = 0, ..., K$. Then $\hat{u}$ will be replaced by its Chebyshev interpolant and the computation will proceed in the new coordinate system until a new adaptation is found to be necessary. The measure of the rate of convergence of the Chebyshev expansion of $\hat{u}$ is done by considering a certain functional $J(\hat{u})$. It is clear that the efficiency of the adaptive method depends directly on the choice of this functional. There are clearly a lot of possibilities. An obvious choice is to define this functional by an expression of the form

$$(J(\hat{u}))^2 = \sum_{k=0}^{K} d_k \hat{a}_k^2, \tag{7}$$

where $\hat{a}_k$ are the computed Chebyshev coefficients of $\hat{u}$ and d_k is a positive increasing function of k. Such a functional gives more and more weight to the coefficients $\hat{a}_k$ as k increases and then tends to be small for functions with fast convergence rate. Actually, the functional proposed in [3] is of the form (7) with d_k a monomial in k. We recall here the definition of this functional which was originally introduced by a minimisation of the leading term of the projection error

$$|u|_{j,\cos} = \left[\int_{-1}^{+1} (L^j u)^2 \, \omega(s) \, ds \right]^{1/2}, \tag{8}$$

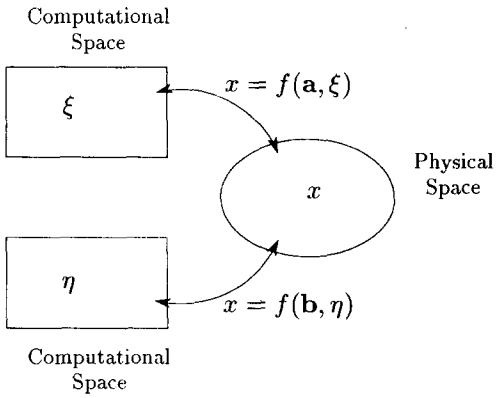

FIG. 1. Coordinate transforms.

where $\omega(s)=(1-s^2)^{-1/2}$ is the Chebyshev weight and L is the operator defined by $Lu=\sqrt{1-s^2}\,du/ds$. The reason for our notation $|u|_{j,\cos}$ will appear below. To prove our claim that this functional is indeed of the form (7), we first remark that the operator L is the derivation in the Fourier space of the even function $\tilde{u}$ defined on $[-\pi,\pi]$ by $\tilde{u}(\theta)=u(\cos\theta)$. More precisely, using the change of variables formula, we obtain

$$\int_{-1}^{+1}(L^j u)^2\,\omega(s)\,ds=\int_0^{\pi}\left(\frac{d^j\tilde{u}}{d\theta^j}\right)^2 d\theta \tag{9}$$

and therefore $|u|^2_{j,\cos}$ is exactly equal to $(\pi/2)\sum_{k=0}^{\infty} c_k k^{2j} a_k^2$, where a_k are the Chebyshev coefficients of u, and c_k has the usual meaning, $c_k=1$ for $k>0$ and $c_k=2$ for $k=0$. This remark also shows that this functional provides a rigorous bound of the projection error in the L^2_ω norm. Actually we have:

PROPOSITION II.1. *For any real σ and any function v such that $|v|_{\sigma,\cos}<+\infty$, the inequality*

$$\|v-P_K v\|_{0,\omega}\leqslant K^{-\sigma}\,|v|_{\sigma,\cos} \tag{10}$$

holds, where $\|v\|_{0,\omega}$ is the weighted L^2 norm, $\|v\|^2_{0,\omega}=\int_{-1}^{+1}|v(s)|^2\,\omega(s)\,ds$, and P_K is the L^2_ω projection in the space of polynomials of degree K.

Proof. The proof of this statement follows immediately from the proof of Theorem 2.1 in [10]. We have

$$\begin{aligned}\|v-P_K v\|^2_{0,\omega}&=|v-P_K v|^2_{0,\cos}\\&=\frac{\pi}{2}\sum_{k=K+1}^{\infty}c_k a_k^2\\&=\frac{\pi}{2}\sum_{k=K+1}^{\infty}k^{-2\sigma}c_k k^{2\sigma}a_k^2\\&\leqslant\frac{\pi}{2K^{2\sigma}}\sum_{k=K+1}^{\infty}c_k k^{2\sigma}a_k^2\\&\leqslant\frac{1}{K^{2\sigma}}\,|u|^2_{\sigma,\cos}.\end{aligned}$$

∎

Therefore, the functional $|v|_{\sigma,\cos}$ provides a rigourous bound of the projection error in the L^2_ω norm. Because one obviously has $|v|_{\sigma,\cos}\leqslant\|v\|_{\sigma,\omega}$, with $\|v\|_{\sigma,\omega}$, the norm of the Sobolev space H^σ_ω, defined when $\sigma\in\mathbb{N}$ by $\|v\|^2_{\sigma,\omega}=\sum_{j=0}^{\sigma}\|d^j v/ds^j\|^2_{0,\omega}$, this bound is, in some sense, tighter that the bound obtained by replacing $|v|_{\sigma,\cos}$ by $\|v\|_{\sigma,\omega}$ as done in [10]. This explains the comment (c), p. 431 in [3].

We now turn to the description of the functionals originally considered in [13, 4]. Using Proposition II.1 and the preceding remark, it is clear that the H^σ_ω norm can also be chosen as a functional bounding the projection error in L^2_ω. Moreover, this norm is also a bound of the projection error in higher order norms as expressed in the following estimate [10]:

PROPOSITION II.2. *For any σ and μ such that $1\leqslant\mu\leqslant\sigma$, there exists a constant C such that*

$$\|v-P_K v\|_{\mu,\omega}\leqslant C(\sigma)\,K^{2\mu-\sigma-1/2}\,\|v\|_{\sigma,\omega},\qquad\forall v\in H^\sigma_\omega. \tag{11}$$

Therefore in [13, 4], the measure of the rapidity of the convergence of the Chebyshev expansion of $\hat{u}$ was taken as the functional:

$$J^2_\omega(\hat{u})=\|\hat{u}\|_{2,\omega}. \tag{12}$$

Many other possible choices can be deduced from (11). In fact, (11) defines a family of error bounds and the actual error is the smallest of the right-hand side of (11). Therefore alternate definitions of the functional can be obtained by using for instance $\sigma=1$ or $\sigma=3$ in (12) instead of $\sigma=2$. These latter choices were also tested in [13, 4], where it was shown that $\sigma=1$ does not give good results, while the results obtained with $\sigma=3$ were not significantly different from the ones obtained with $\sigma=2$. Also, in the estimate (11), we observe that $\|v\|_{\sigma,\omega}$ can be replaced by $\|dv/ds\|_{\sigma-1,\omega}$ which can be used to define another family of functionals (in practice, not very different from the previous ones). These last functionals are effectively used in the present calculations, but for reasons of simplicity, the notation $\|v\|_{\sigma,\omega}$ will be conserved.

We also mention that, for neither of the two families of functionals we have discussed, the function that realizes the minimum of the functionals is a polynomial of degree one. This seems surprising because one could expect this function to be the simplest one represented by a polynomial expansion. On the contrary it can be shown that the function that realizes the minimum of the $|\cdot|_{j,\cos}$ functionals is $u(x)=a+b\arccos x$ which does not even belong to H^1_ω and whose Chebyshev coefficients are decreasing only like k^{-2}. A functional whose minimum is attained on a polynomial of degree 1 is defined by the non-weighted H^2 norm. Therefore, we have also tested this functional in our experiments.

Finally, it must be noted that the adaptive methods are based on the minimization of some estimate of the interpolation error. The actual error of the numerical solution of a differential problem also depends on the stiffness of the mapping. Hence a too severe mapping could alter the accuracy of the approximation as mentioned in [2, 13]. Some theoretical results on this point can be inferred from [18], while numerical experiments on the convergence of mapped Chebyshev methods are reported in [7].

III. PRACTICAL IMPLEMENTATION OF THE ADAPTIVE METHOD

A crucial point in the above algorithm is the actual computation of the functionals $J(\hat{u})$. Even, if the value of the function $\hat{u}(\eta)$ is known for each η by formula (6), the exact computation of $J(\hat{u})$ is impossible and, in practice, we have to use a quadrature formula to evaluate this integral. We describe below three ways to perform this evaluation.

METHOD A. In this method, we replace $\hat{u}$ by its interpolated polynomial $I_K(\hat{u})$. Thus, in principle, the computation of the different functionals require only the use of a Chebyshev transform (for the $|\cdot|_{\sigma,\cos}$ semi-norm) or of a Gauss–Lobatto integration formula (for the $\|\cdot\|_{\sigma,\omega}$ norm). However, this requires us to compute the value of $\hat{u}$ at the collocation points $\eta_k = \cos(k\pi/K)$, $k = 0, ..., K$. Going back to formula (6), we see that this implies the computation of $f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta_k))$ for $k = 0, ..., K$, and then the computation of the matrix $T_j(f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta_k)))$. Because $f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta_k))$ is not a collocation point in the η-space, this matrix must be computed using the formula $T_j(s) = \cos(j \arccos(s))$, defining the Chebyshev polynomials. In other words, each computation of the functional implies projection of u in η-computational space. This process can be rather costly because of the numerous uses of the arc cos function.

METHOD B. An alternative is to transform the integral $J(\hat{u})$ in the current ξ-coordinate system before computing it. For clarity of the exposure, we shall describe below the computation of the H^1_ω functional, but the discussion is not restricted to the use of this norm. First, using the change of variable theorem, we obtain

$$\int_{-1}^{1} (\hat{u}_\eta(\eta))^2 \omega(\eta)\, d\eta = \int_{-1}^{1} \frac{(\bar{u}_\xi(\xi))^2}{\phi_\xi(\xi)} \omega(\phi(\xi))\, d\xi,$$

where ϕ is the one-to-one mapping between the ξ and η coordinate systems defined by $\eta = \phi(\xi) = f^{-1}(\mathbf{b}, f(\mathbf{a}, \xi))$. Because $\bar{u}_\xi(\xi)$ is known at the Gauss–Lobatto points, this integral can easily be evaluated by

$$\int_{-1}^{1} (\hat{u}_\eta(\eta))^2 \omega(\eta)\, d\eta \simeq \sum_{k=0}^{K} c_k \frac{(\bar{u}_\xi(\xi_k))^2}{\phi_\xi(\xi_k)} \frac{\omega(\phi(\xi_k))}{\omega(\xi_k)},$$

where $c_k = 1$ for $k \in \{1 \cdots K-1\}$, and $c_0 = c_K = 2$. The computation of the other integrals appearing in the various functionals used in this paper can be performed in the same way. Note that Methods A and B are not equivalent because interpolation and integration do not commute. Figure 2 compares the results obtained by the two methods in the computation of $J^2_\omega(\hat{u}))$ as a function of b, where $\hat{u}(\eta) = u(f(b, \eta))$ with $u(x) = \frac{1}{2}\tanh(2x)$ and the coordinate transform $f(b, \eta)$ is defined by

$$x = f(b, \eta) = L[(1-b)\eta^3 + b\eta], \qquad -1 \leqslant \eta \leqslant 1, \tag{13}$$

where L the length of the half-domain is $L = 7.53$. In this test, we assume that the function $\bar{u}(\xi)$ is initially defined at the collocation points of the ξ-coordinate system by $\bar{u}(\xi) = u(f(a, \xi))$ with $a = 0.15$. With a number of collocation points equal to 72, the two computations lead to almost the same results. When 36 collocations points are used, it can be seen that the results are, again, practically identical in the vicinity of the optimal parameter. Method B appears to be more accurate than Method A; moreover, Method B is less expensive than Method A by almost a factor of 2 because it avoids the projection of u in η-computational space.

METHOD C. A third way to compute the functionals can be devised as explained now. Again, for the clarity of the exposure, we shall only consider the minimisation of the H^1_ω

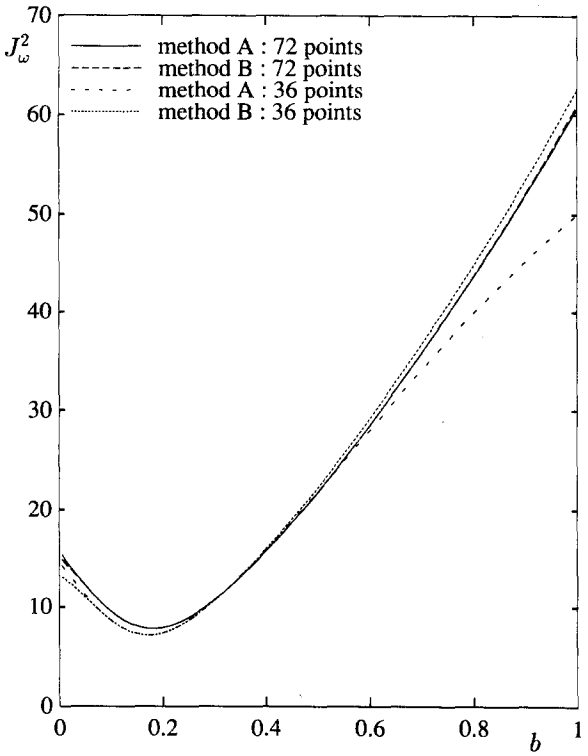

FIG. 2. Comparison of methods *A* and *B* for computing the functional J^2_ω.

norm. Instead of replacing $\hat{u}$ by $I_K(\hat{u})$ in $J^1_\omega(\hat{u})$, we could use definition (5) and the chain rule to obtain the equality

$$J^1_\omega(\hat{u}))^2 = \int_{-1}^{1} \left| \bar{u}_\xi(f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta))) \times \frac{f'(\mathbf{b}, \eta)}{f'(\mathbf{a}, f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta)))} \right|^2 \omega(\eta)\, d\eta, \quad (14)$$

where f' is the derivative of f with respect to its second argument. This expression may be approximated by

$$(J^1_\omega(\hat{u}))^2 \simeq \sum_{k=0}^{K} c_k \left| \bar{u}_\xi(f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta_k))) \times \frac{f'(\mathbf{b}, \eta_k)}{f'(\mathbf{a}, f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta_k)))} \right|^2. \quad (15)$$

This expression differs from $(J^1_\omega(I_K(\hat{u})))^2$ because derivation and interpolation do not commute. Because, in general, at two successive time indexes we have $\mathbf{b} \simeq \mathbf{a}$, one may be tempted to simplify (15) by doing the identifications:

$$\bar{u}(f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta_k))) \simeq \bar{u}(\eta_k); \qquad \frac{\bar{u}_\xi(f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta_k)))}{f'(\mathbf{a}, f^{-1}(\mathbf{a}, f(\mathbf{b}, \eta_k)))} \simeq \frac{\bar{u}_\xi(\eta_k)}{f'(\mathbf{a}, \eta_k)} = u_x(f(\mathbf{a}, \eta_k)). \quad (16)$$

In other words, the value of the above quantities are attached to the index k and not to their position in physical space. Similar simplifications are almost always used in adaptive techniques with finite difference methods. If such a simplification is done, (15) simply reduces to

$$(J^1_\omega(\hat{u}))^2 \simeq \sum_{k=0}^{K} c_k\, |u_x(x_k)|^2 \cdot [f'(\mathbf{b}, \eta_k)]^2, \quad \text{with} \quad x_k = f(\mathbf{a}, \eta_k), \quad (17)$$

and as in Method B, we avoid the projection of the u function in η-computational space. Our experience with mixing layer computations, as well as with the Burgers' problem treated in [13], is that the use of this simplification cannot be done without the inclusion of a penalty term in the definition of the functional, to prevent the jacobian of the change of variable to become too large. For example, in the mixing layer computations (see Section IV) when using this simplification, the adaptive algorithm always yields the maximum clustering of the collocation points in the vicinity of the x-axis, although the width of the layer is continuously increasing. On the contrary, the complete computation of the functionals $\|\hat{u}\|_{2,\omega}$, $|\hat{u}|_{2,\cos}$, or $\|\hat{u}\|_2$ by Method A or B leads to a correct behavior of the change of variable that nicely follows the evolution of the width of the mixing layer, as will be seen in Section IV. Moreover, we never need to include a penalty term in the functionals. Therefore, we do recommend the complete computation of the functionals by Method A or B and especially by Method B; that would be preferred in terms of CPU cost.

In the computation of $J(\hat{u})$, we also found it useful to apply a smooth filter (here, a raised-cosine filter has been used) to the function $\bar{u}$. We emphasize that this filtering only serves to define the new mapping and is not used to compute the unknowns at subsequent times. The utility of smoothing the argument of the functional can be understood as follows: It is clear from formula (6) that if the function $\bar{u}$ is not very well behaved, then the function $\hat{u}(\mathbf{b})$ will also be ill-behaved for certain values of the parameter $\mathbf{b}$. In particular, for these values the computation of the functional $J(\mathbf{b}) = J(\hat{u}(\mathbf{b}))$ may be very inaccurate, leading, for example, to the existence of several local minima. As a consequence, the minimisation algorithm may be unsuccessful or converge to a bad value. But, it is precisely when $\bar{u}$ becomes ill-behaved, that a new change of variable is computed. This situation then is likely to occur. Therefore, to avoid this type of problem, it is advantageous to use as an argument of the functional a smooth function obtained by filtering $\bar{u}$ in the y-direction.

As an illustration, we consider in Fig. 3 the behavior of the parameter a defining the mapping used in the mixing layer computations (see Eq. (26), $L_y = 25$). This computation was done with 41 points in the x-direction and 81 in the y-direction which is barely enough. It is apparent that when no filtering is done (dotted curve) that the value of the parameter oscillates around an increasing mean value. As explained above, this results from a lack of accuracy in the computation of the functional leading to the existence of several local minima.

On the contrary, when a filtering is applied, the functional is convex with a unique minimum, giving a smooth behavior of the change of variable. We also note that filtering the argument of the functional has a beneficial influence on the number of changes of mapping that are performed. Because a change of mapping is costly (one has to interpolate in the new coordinate system the whole fields of the different unknowns), it is advantageous to perform this change of mapping as seldom as possible. In the present computations, a change of mapping is done only if the new computed value of the parameter a differs by more than 1% from its current value. In the previous case, 31 changes of mapping have been done between $t = 0$ and $t = 20$ when the filtering was applied, while in the absence of filtering, the number of performed changes of mapping was 44.

In addition to the adaptive algorithm described above, the functional can also be used to monitor the number of collocation points as in [3] by requiring the error to be lower than a prescribed level. For instance, in the mixing layer computations presented below, we define two func-

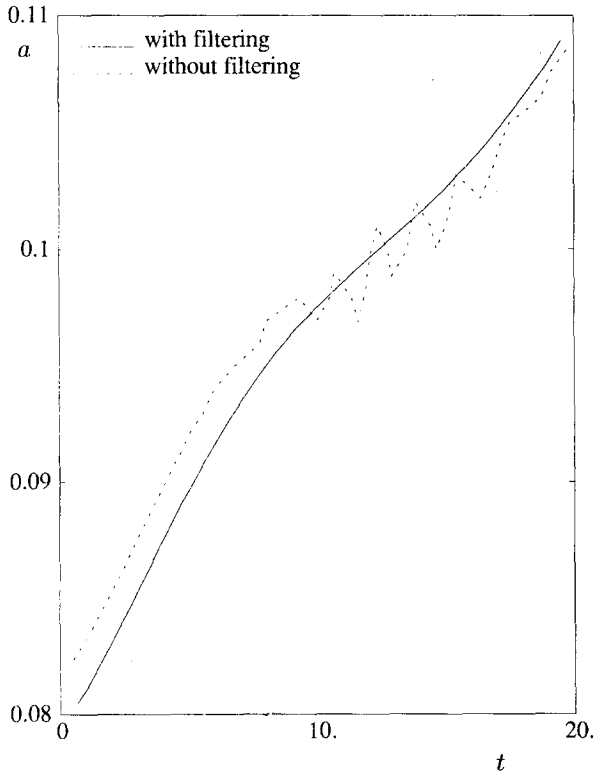

FIG. 3. Evolution of parameter a versus time: influence of filtering.

tionals I and J in the x and y directions, respectively, to control the number of collocation points in each space direction. This is done by enforcing the following conditions:

$$\frac{I}{N_x^2} < \varepsilon_x, \qquad \frac{J}{N_y^2} < \varepsilon_y.$$

Now we summarize the complete adaptive algorithm. It begins after the computation of the solution W^n at the nth time-cycle and of the functionals I and J respectively in the x- and y-directions; a is the current value of the parameter and J_{ref} is the minimum of the functional at the last adaptation, when a was chosen. The adaptation procedure is as follows:

I—if $\{J(a) - J_{\text{ref}}/J_{\text{ref}} > \alpha\}$
  find $\min_b J(b)$.
  if $\{\|b-a\|/\|a\| > \beta\}$
    project W^n in the new computational space using formula (6)
    if $\{J(b)/N_y^2 > \varepsilon_y\}$
      increase N_y up to the next value admissible by the FFT subroutine ($2^p 3^q 5^r$)
    else, go to step II
  else, go to step II
endif

II—if $\{I(b)/N_x^2 > \varepsilon_x\}$
  increase N_x in the same way as for N_y
else, end of adaptation
endif

In the mixing layer computations reported below, the parameters ε_x, ε_y, α, and β have the values: $\varepsilon_x = \varepsilon_y = 0.005$, $\alpha = \beta = 0.01$. The minimization of the functional was obtained by the golden section search.

IV. COMPUTATION OF TEMPORALLY GROWING COMPRESSIBLE MIXING LAYERS

Interest in computation of compressible mixing layers has been growing in the last few years, due to their application in Scramjet conception. Figure 4 is a sketch of the time-developing mixing layer considered in this paper. The basic initial state has constant pressure and total enthalpy. The vertical component of velocity is zero while the horizontal component has an inflection point and is unstable to small disturbances. A small perturbation (random or sinusoïdal) is added to this basic state. We refer to [8, 9, 11, 17, 16] for linear stability analyses of this problem. Instability occurs for any convective Mach number and Reynolds number; nevertheless, effects of compressibility and viscosity are stabilizing. Direct numerical simulations of this type of flows have been appearing for the last few years, using either Euler or Navier–Stokes equations. The numerical methods typically used are finite-difference schemes with or without TVD-type viscosity, except in [14, 24], in which high-order compact difference schemes are used. With respect to spectral methods, while analogous computations have been carried out for the incompressible problem [20, 23, 19, 15], results of direct simulations of compressible mixing layers using these methods have not yet been published. The difficulties are various and rather important: first, as for the incompressible mixing layer, the initial distribution of horizontal velocity shows very substantial gradients at the center of the domain. Moreover, when time increases, as the

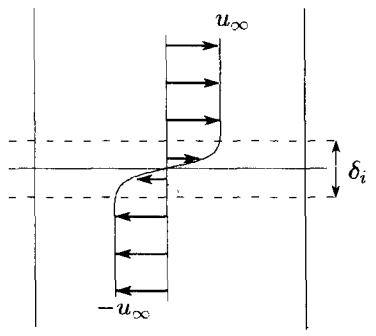

FIG. 4. Initial configuration.

layer thickens and rolls up, the stiff gradients move from their initial location and lose their privileged directions. In addition, specific problems appear in the compressible case: (a) when the convective Mach number is sufficiently large, vertical shocklets appear within the flow; (b) in supersonic cases (not considered here), additional unstable modes with slow decay in the free stream are present; (c) compressibility effects slow down the development of the instability, leading to an increase of integration time with respect to the incompressible case. For instance, with a Reynolds number equal to 1000, we found that the characteristic time for vortex pairing is about 95 for a convective Mach number of 0.3, while it is 110 for a convective Mach number of 0.5. The aim of this paper is to show the efficiency of the adaptive method in capturing the vertical gradients and no special treatment for resolving horizontal gradients has been developed; therefore the application field of the present method is restricted to shock-free configurations. Nevertheless, the method can correct a lack of accuracy in the x-direction by automatically increasing the number of collocation points.

The equations governing the flow are the two-dimensional, time-dependent Navier–Stokes equations for compressible fluids, which, in dimensionless form, are written

$$\begin{aligned}
&\rho_t + (\rho u)_x + (\rho v)_y = 0 \\
&(\rho u)_t + (\rho u^2 + p)_x + (\rho uv)_y \\
&\quad = \frac{1}{\mathrm{Re}}\left[(\tau_{11})_x + (\tau_{12})_y\right] \\
&(\rho v)_t + (\rho uv)_x + (\rho v^2 + p)_y \\
&\quad = \frac{1}{\mathrm{Re}}\left[(\tau_{12})_x + (\tau_{22})_y\right] \\
&p_t + \gamma p(u_x + v_y) + up_x + vp_y \\
&\quad = \frac{1}{\mathrm{Re}\, M_*^2 \,\mathrm{Pr}}\left[(kT_x)_x + (kT_y)_y\right] \\
&\qquad + \frac{\gamma - 1}{\mathrm{Re}}\left[\tau_{11}u_x + \tau_{12}(u_y + v_x) + \tau_{22}v_y\right]. \qquad (18)
\end{aligned}$$

In these equations, x and y are the spatial directions, t the time, and subscripts refer to partial derivatives, ρ is the density, T the temperature, p the pressure, u and v the horizontal and vertical components of the velocity, and γ is the ratio of specific heats. The viscosity μ and the thermal conductivity k are equal, either constant (equal to 1) or given by Sutherland's law; $\tau_{11} = 2\mu(2u_x - v_y)/3$, $\tau_{12} = \mu(u_y + v_x)$, and $\tau_{22} = 2\mu(2v_y - u_x)/3$ are the components of the viscous stress tensor. The dimensionless numbers appearing in (18) (Reynolds, Mach, and Prandtl numbers), are defined as

$$\begin{aligned}
\mathrm{Re} &= \frac{2u_\infty \delta_i \rho_\infty}{\mu_\infty}, \\
M_* &= \frac{2u_\infty}{\sqrt{\gamma \mathscr{R} T_\infty}} = 2M_\infty, \qquad (19) \\
\mathrm{Pr} &= \frac{C_p \mu_\infty}{k_\infty}.
\end{aligned}$$

In these expressions, C_p is the specific heat at constant pressure and $\mathscr{R}$ is the perfect gas constant. The non-dimensionalisation has been performed using, the initial vorticity thickness δ_i as the length unit, the velocity difference across the layer $2u_\infty$ as the velocity unit, and T_∞, ρ_∞, μ_∞, and k_∞ the temperature, density, viscosity, and thermal conductivity in the free stream as units. Finally, the Navier–Stokes equations are completed by the perfect gas equation of state which, in dimensionless form, is written

$$p = \frac{1}{\gamma M_*^2}\rho T. \qquad (20)$$

The initial conditions are composed of a basic state upon which is superimposed a small perturbation. The basic horizontal velocity profile is hyperbolic tangent, and there is no vertical velocity; the pressure is constant and the temperature is calculated from boundary-layer approximation with $\mathrm{Pr} = 1$ (in other terms, the total enthalpy of the basic flow is constant):

$$\begin{aligned}
u &= \frac{1}{2}\tanh(2y), \qquad v = 0, \\
p &= \frac{1}{\gamma M_*^2} = \frac{1}{4\gamma M_\infty^2} \qquad (21) \\
T &= 1 + \frac{\gamma - 1}{2} M_\infty^2 (1 - (2u)^2).
\end{aligned}$$

The perturbation is either a small random function of x and y lying in the vicinity of $y = 0$, or a small deterministic perturbation of the form

$$\begin{aligned}
u' &= \frac{\varepsilon y \lambda}{20\pi} \sin\left(\frac{2\pi x}{\lambda}\right) \exp\left(-\frac{1}{10}y^2\right) \\
v' &= \frac{\varepsilon}{2} \cos\left(\frac{2\pi x}{\lambda}\right) \exp\left(-\frac{1}{10}y^2\right), \qquad (22)
\end{aligned}$$

where ε represents the amplitude of the perturbation and λ is the non-dimensional wave length. The perturbation component on the horizontal velocity ensures the total pertur-

where $\min_i$ denotes the minimum over all the collocation points, u is the velocity, and c is the sound speed. In order to take into account the momentum and energy equations, the constant σ is defined by

$$\sigma = \max\left(\frac{4}{3K_1}, \frac{1}{\mathrm{Pr}\, M_\infty^2 K_2}\right).$$

The constants K_0, K_1, and K_2 depend on the type of boundary conditions associated with the velocity and the temperature. Following the heuristic way proposed by [22], this criterion is extrapolated to the two-dimensional case. Then it is modified to take into account the presence of the mapping (13), $y = f(a, \eta)$, and we finally obtain the rather complicated expression

$$\Delta t = \min_{(i,j)} \left(C \Big/ \left(\frac{2\,|u_{i,j}|}{L_x K_0} N_x^2 + \frac{\chi_{i,j} N_y^2}{K_0} + \frac{c_{i,j}}{K_0} \sqrt{\frac{4N_x^4}{L_x^2} + \frac{N_y^4}{f'(a,\eta_j)^2}} + \sigma \frac{1}{\mathrm{Re}\, \rho_{i,j}} \left(\frac{4N_x^4}{L_x^2} + \frac{N_y^4}{f'(a,\eta_j)^2} \right) \right) \right) \tag{30}$$

where

$$\chi_{i,j} = \frac{|v_{i,j}|}{|f'(a,\eta_j)|} + \frac{4}{3\mathrm{Re}\, \rho_{i,j}} \frac{|f''(a,\eta_j)|}{|(f'(a,\eta_j))^3|},$$

N_x and N_y being the degree of the polynomial approximation in the x- and y-directions, respectively.

This criterion gives a useful guideline although it is not completely satisfactory; first, the constants have been determined for the modified Euler scheme, rather than the three-level Runge–Kutta scheme. Then, the local linearization while conserving the behavior in N^2 or N^4 can lead to an underestimation of the allowable time step. Furthermore, the neglecting of the crossed derivatives is of questionable validity as the flow becomes strongly two-dimensional. Finally, the periodicity in the x-direction has not been taken into account. These features lead to an indetermination of the constant C. The calculations reported below have been performed with $K_0 = 17$, $K_1 = 41$, $K_2 = 142$, and $C = 10$.

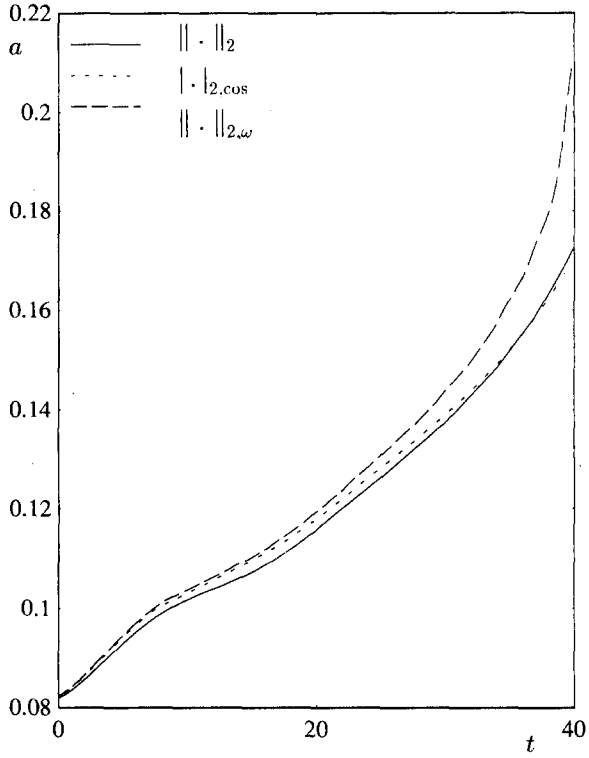

FIG. 5. Compared evolution of a for the three functionals.

Let us also note that with the values of the physical parameters (Mach and Reynolds numbers) and numerical ones (numbers of collocation points) considered here, the convective and diffusive contributions are of the same magnitude. In other words, the maximum allowable time step is practically the one governed by first-order derivative terms, so that an implicit discretization of the diffusive terms (avoiding the N^4 term) would not improve the efficiency of the algorithm.

Now we report some numerical experiments concerning mixing layer computations. Table I gives a complete list of the calculations reported here with the values of the significant parameters. All the experiments have been done with $\gamma = 1.4$.

We first compare the behavior of the three functionals

TABLE I

List of Runs

Run	M_∞	Re	Pr	μ	L_x	L_y	Perturbation	ε	λ
A	0.8	400	1.0	Variable	20	25	Sinusoïdal	0.1	20
B	0.8	400	1.0	Constant	20	25	Sinusoïdal	0.1	20
C	0.8	400	1.0	Constant	20	10	Sinusoïdal	0.1	20
D	0.6	400	1.0	Constant	20	10	Sinusoïdal	0.1	20
E	0.8	400	1.0	Constant	20	50	Sinusoïdal	0.1	20
F	0.6	400	1.0	Constant	20	50	Sinusoïdal	0.1	20
G	0.4	400	1.0	Constant	20	50	Sinusoïdal	0.1	20
H	0.5	1000	0.7	Variable	15.82	7.91	Random	0.005	7.91

TABLE II

Comparison of the Extrema of ω and ρ at $t=40$, for the Three Different Functionals

	$\|\cdot\|_{2,\omega}$	$\|\cdot\|_2$	$\|\cdot\|_{2,\cos}$
$\omega_{\min}$	-0.27455	-0.27622	-0.27456
$\omega_{\max}$	5.4751×10^{-3}	5.2829×10^{-3}	5.8574×10^{-3}
$\rho_{\min}$	0.65798	0.65814	0.65802
$\rho_{\max}$	1.2014	1.2015	1.2016

discussed in Section II, namely the H^2_ω and H^2 norms and the functional $|\cdot|_{2,\cos}$. The test case is the run (A).

The temporal evolution of the parameter a computed with these three different functionals is shown in Fig. 5. It can be seen that, although the evolutions of a are similar for the three different cases, the value of a given by the functional of [3] is always the lowest, while the largest is given by the weighted Sobolev norm. This can be explained by considering that the $|\cdot|_{2,\cos}$ and H^2 functionals give less weight to the boundaries than the H^2_ω norm; thus, this latter functional tends to give a larger importance to rather small variations of the function appearing far from the middle of the domain. This effect becomes more pronounced toward the end of the computation, where the values of a given by the different functionals begin to differ substantially. The results obtained with the three functionals are almost undistinguishable when looking at the contour curves or at the profiles of the different quantities.

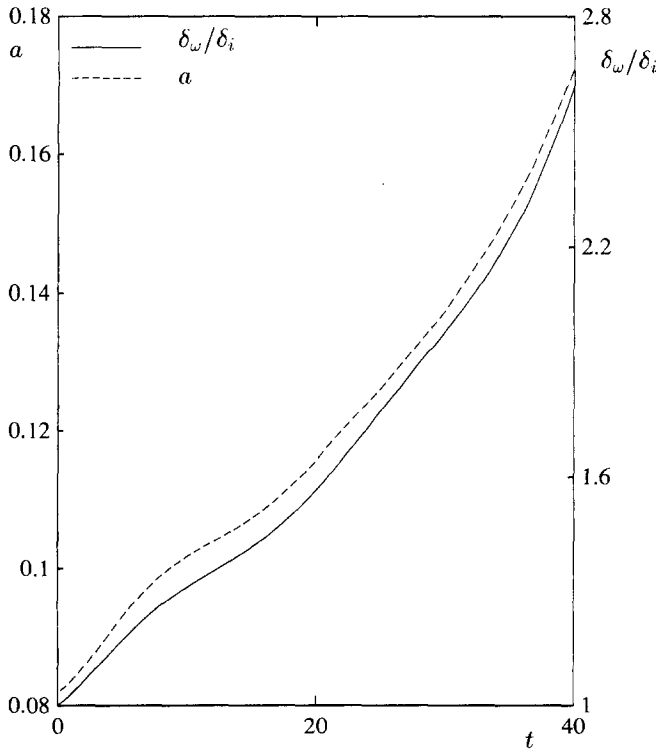

FIG. 6. Evolution of a and δ_ω/δ_i along time.

Table II presents a more precise comparison between the results obtained with the three different definitions of the functional. One can see that the relative difference on the maximum values of the vorticity field never exceeds 3% while the variation in the density field are even smaller. This might seem surprising, but the mapping sensitivity to the variations of the parameter a becomes weaker as a is increased; then, although the final values of the parameter a differ substantially for the three functionals, the location of the collocation points in the physical space are almost the same.

Let us define, in dimensional units, a quantity commonly used to measure the development of the shear layer instability, i.e., the vorticity thickness,

$$\delta_\omega = (u_{+\infty} - u_{-\infty}) \Big/ \left| \frac{\partial}{\partial y}\left(\frac{\overline{\rho u}}{\bar\rho}\right)\right|_{\max} \tag{31}$$

where $\bar\rho$ and $\overline{\rho u}$ are spatial averages in the x-direction. Figure 6 shows the evolution of the parameter a defining the mapping (13), together with the evolution of the vorticity thickness. The parallel evolution of this parameter and of the thickness of the mixing layer is remarkable and indicates that the adaptive procedure is effective in distributing the collocation points inside the zone of large derivatives while the layer is expanding. We recall here that when using Method C described in Section III, we were unable to obtain such a nice behavior of the coordinate transform.

The evolution of the mixing layer is visualized by instantaneous contours of the potential vorticity, ω/ρ, ω being the vorticity defined here by $\omega = (v_x - u_y)/2$. Figure 7 shows the isolines of ω/ρ in the central part of the domain between $y=-5$ and $y=5$ up to time $t=40$, for the case A. The adaptive method used in the presented calculation has employed the H^2 norm; at $t=0$, there were 61×101 points and the refinement procedure did not find it necessary to increase N_x or N_y up to $t=40$.

One can see how, in response to the perturbation introduced in the basic flow that the isolines bend and the layer thickens while time increases. In Figure 8 the evolution of the vorticity profiles at $x\simeq 2.93$ is shown. The amplitude of the initial sharp gradient at the center part of the domain decreases while the layer thickens. Also note that the regions of large second derivatives move from the origin and that local extrema appear at the outer part of the layer.

After $t=45$, regions of strong compression appear in the flow and the results are seriously contaminated by spurious oscillations due to the insufficient resolution of the x-direction. The adaptive procedure reacts to the appearance of these compressions by refining the mesh in this direction. However, even with a fast increase in the

number of collocation points (it has been doubled by the adaptive algorithm between $t=45$ and $t=60$), the results remain oscillatory and have to be filtered. Restricting ourselves to shock-free configurations, no special treatment for this case has been developed.

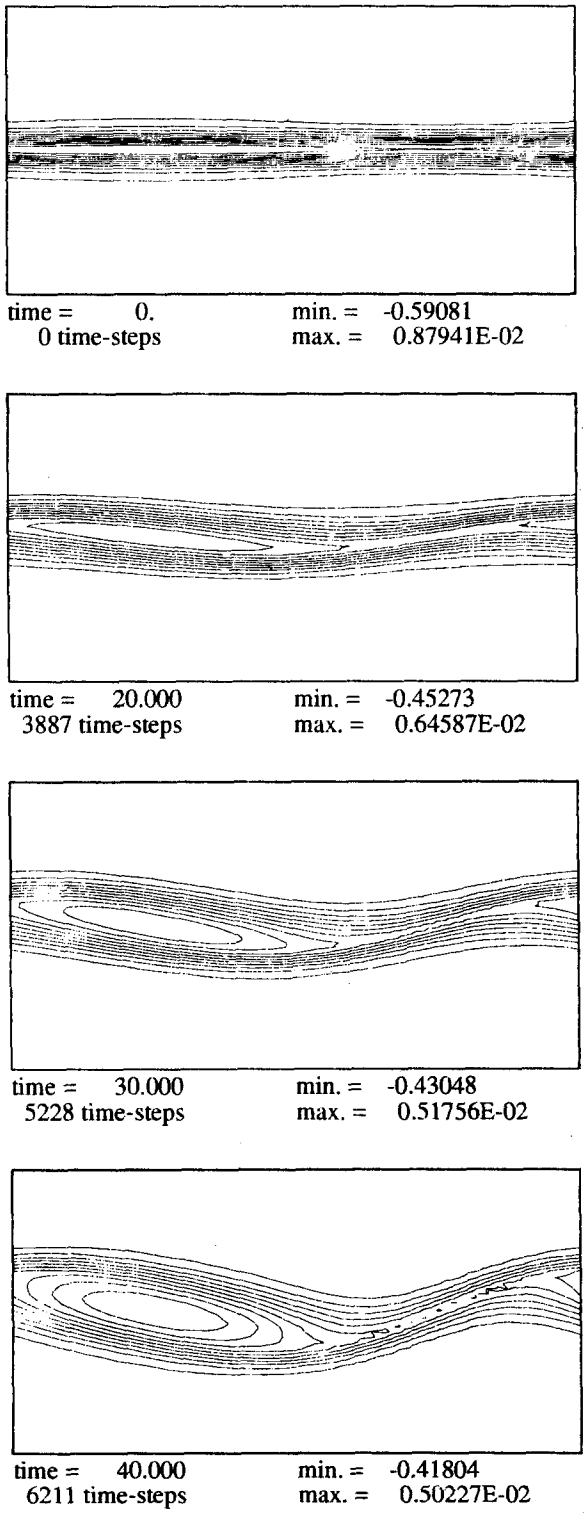

FIG. 7. Evolution of ω/ρ.

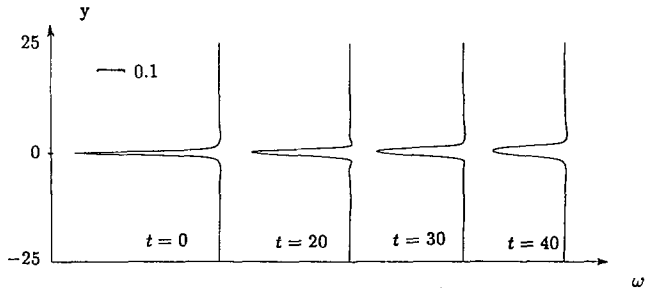

FIG. 8. Profiles of ω versus y.

On Figure 6, in solid line, the evolution of δ_ω/δ_i is plotted. The results are in very close agreement with those obtained in [25], using a second-order finite difference method. Their most accurate results using a 150×150 mesh yield at $t=40$, a value of $\delta_\omega/\delta_i\approx2.55$, while the present computations show $\delta_\omega/\delta_i\approx2.60$.

To investigate the influence of variable viscosity, in Table III we have summarized the differences between constant (case B) and variable (case A) viscosity. The extrema of vorticity and density show little changes (about 3%) between the two cases, and the vorticity thickness variation is less than 1%. Thus, using constant viscosity rather than Sutherland's law when temperature changes moderately is a valid approximation in such flows.

We now study the effects of the location of the y-boundaries on the development of the instability. In Fig. 9 we show the compared growth of the vorticity thickness in cases B and C that differ only by the location of the y-boundaries. To point out the influence of acoustic effects, the reference time here is taken as the "acoustic" time $\tilde{t}_{\text{ref}}=\delta_i/c_\infty$. Therefore, the relation between the nondimensional "acoustic" time and the nondimensional time t used previously is $\tilde{t}=t/2M_\infty$. We note that, approximately at $\tilde{t}=20$, the two quantities begin to differ substantially. This difference may be due to the reflexion of acoustic waves on the y-boundaries. As a matter of fact, considering only vertical propagation of sound waves, neglecting the vertical fluid velocity component v, and approximating the sound speed by its free stream value c_∞, it appears that a sound wave generated in the layer at $\tilde{t}=0$ reaches the outer boundaries $y=\pm L_y$ at time $\tilde{t}_l\approx L_y$. After reflexion, it comes back to the layer, reaching it at $\tilde{t}_l\approx 2L_y$. The test case C has the nearest y-boundaries ($L_y=10$), and their influence is seen as $\tilde{t}_l\approx20$.

TABLE III

Comparison of the Extrema of ρ and ω and of Vorticity Thickness at $t=40$, $M_\infty=0.8$, for Constant and Variable Viscosity

μ	$\omega_{\min}$	$\omega_{\max}$	$\rho_{\min}$	$\rho_{\max}$	δ_ω/δ_i
Constant	−0.28054	5.4359×10^{-3}	0.65159	1.2033	2.62
Variable	−0.27622	5.2829×10^{-3}	0.65814	1.2015	2.60

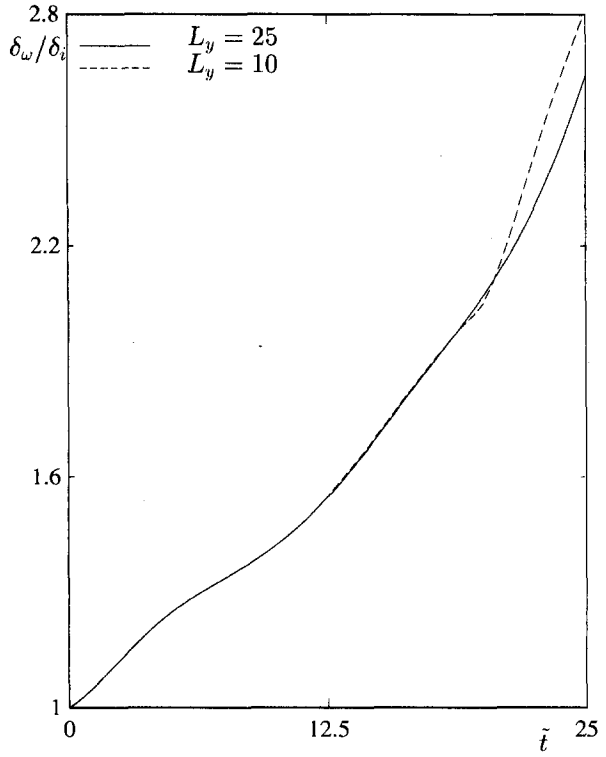

FIG. 9. Compared evolution of δ_ω/δ_i, for different locations of the y-boundaries, $M_\infty = 0.8$.

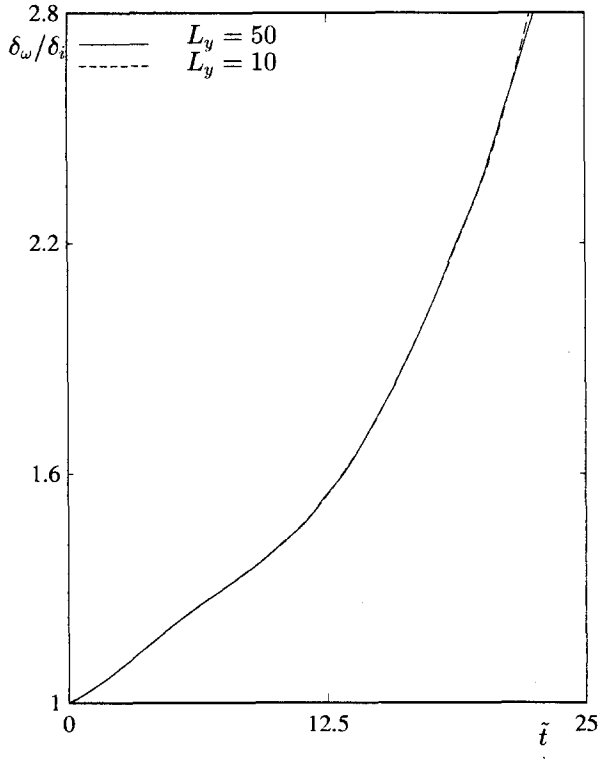

FIG. 10. Compared evolution of δ_ω/δ_i, for different locations of the y-boundaries, $M_\infty = 0.6$.

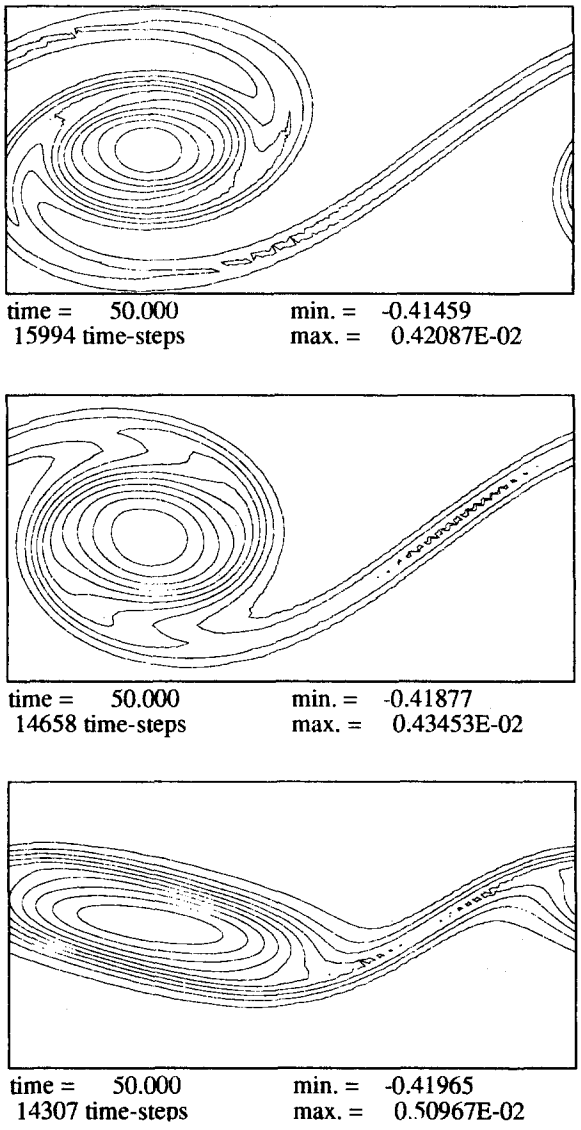

FIG. 11. Evolution of ω/ρ, for $M_\infty = 0.4$ (top), $M_\infty = 0.6$ (middle) and $M_\infty = 0.8$ (bottom).

For confirmation, this experiment has been repeated with $M_\infty = 0.6$ in test cases F and D, that differ only by the location of the y-boundary. In that case, the predicted time $\tilde{t}_l$ is also $\tilde{t}_l \approx 20$. As shown in Fig. 10, whose plots represent the compared evolutions of vorticity thickness versus the "acoustic" time in cases F and D, it is seen that the two curves separate at $\tilde{t} = \tilde{t}_l$ approximately. The separation of the curves for the cases at $M_\infty = 0.8$ is more perceptible than for the ones at $M_\infty = 0.6$, because an acoustic perturbation is of amplitude $O(M_\infty^2)$.

We conclude from these experiments that simulating an open subsonic flow with slip-wall conditions is valid, provided that these boundaries are located sufficiently far. When $M_\infty = 0.8$ and $L_y = 25$, for instance, the sound wave will have come back to the layer at $\tilde{t}_l \approx 50$. Thus, we can

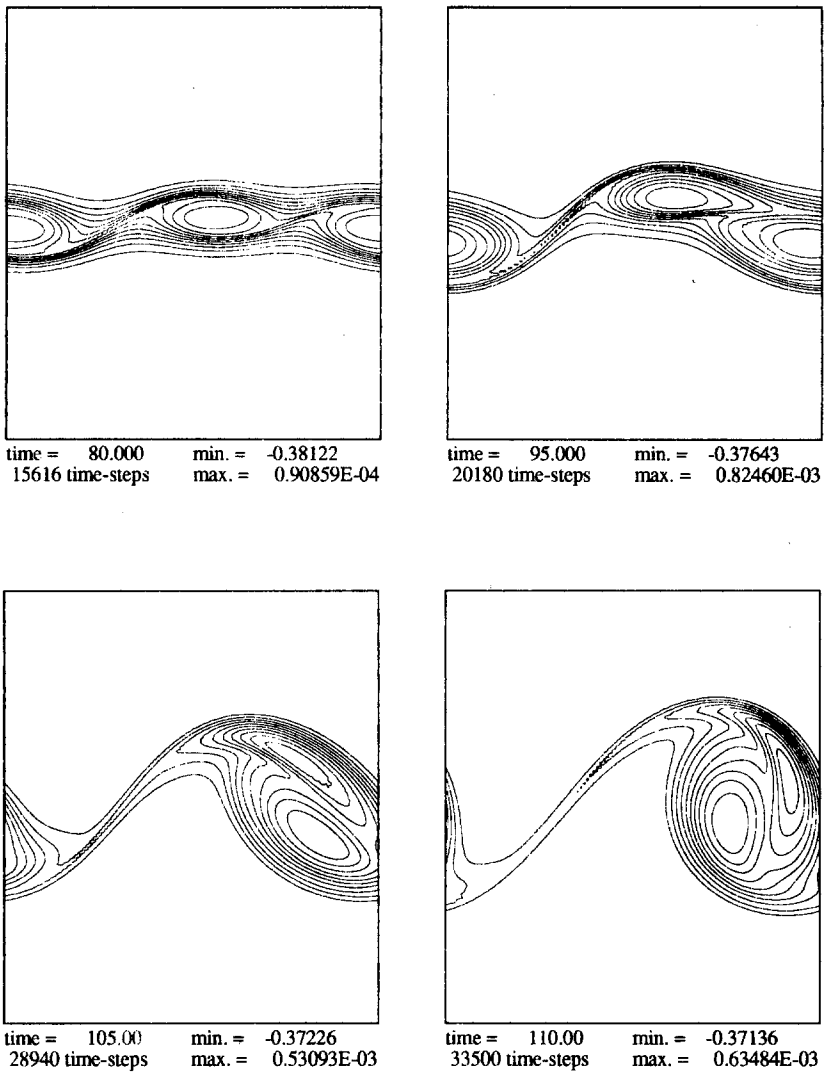

FIG. 12. $M_\infty = 0.5$, evolution of ω/ρ.

reasonably admit that at $t = 40$, corresponding to $\tilde{t} = 25$, our calculations have not suffered from acoustic reflexions on the y-boundaries.

In order to illustrate the effects of compressibility in mixing layer phenomena, we show in Fig. 11 the potential vorticity contours at $t = 50$ for cases E, F, and G. The three figures have been post-processed by application of a raised-cosine filter in the x-direction, to withdraw spurious oscillations in this direction. The case with $M_\infty = 0.4$, being the less compressible, is the one with the faster growth rate. On the contrary, the case $M_\infty = 0.8$ shows a very slow development. This is in agreement with the stabilizing effect of compressibility predicted by stability theory. Furthermore, it is interesting to note the change in the aspect of the vortex, becoming more elongated as M_∞ grows, is due to baroclinic and dilatational effects [24].

As a final illustration, we apply the method to a situation where a vortex pairing is occurring. The mixing layer is now excited by a random perturbation for a convective Mach number of 0.5 (test case H). The length of the box has been chosen to be two times the most amplified inviscid wavelength predicted by linear stability theory [8]. The evolution of the layer is depicted in Fig. 12. It shows the potential vorticity instantaneous contours at four different times. The pictures show the entire computational domain and thus, from the preceding discussion, it must be noted that the results are not free from the influence of the y-boundaries. At $t = 80$, in accordance with linear stability theory, two vortices have been created from the initial perturbation. As the vorticity is of the same sign in the layer, the two vortices begin to join at $t = 95$. Ten characteristic times later, they are rolling over each other. The pairing event is shown to be almost completed at $t = 110$.

The computation has begun with 81×81 points. The

number of points was increased automatically by the adaptive procedure described above and reached 129×129 between $t = 90$ and $t = 100$; this gave us a mesh of 181×129 at $t = 110$ by the end of the computation. The number of points in the x-direction is too large compared to the importance of the gradients. In this direction, there is no mesh adaptation, but only an enrichment monitored by the value of ε_x, which was likely taken too small in the present case. On the other hand, in the y-direction, where the adaptation is present, the number of points remains relatively small. The value of the parameter a of the mapping (26) was $a \simeq 0.17$ at initial time and has reached a value of $a \simeq 0.76$ at $t = 110$.

V. CONCLUSION

We have discussed the application of self-adaptive Chebyshev spectral methods to the computations of compressible mixing layers for subsonic convective Mach number. These methods are found to work well for this case, for different extents of the domain, allowing us to use a reasonable number of collocation points and accurately

Recherche. The principle of the second method of computation of the functionals described in Section III has been pointed out to us by one of the referees and we warmly thank him for his contribution.

REFERENCES

1. J. M. Augenbaum, *Appl. Numer. Math.* **5**, 459 (1990).
2. C. Basdevant *et al.*, *Comput. Fluids* **14**, 23 (1986).
3. A. Bayliss, D. Gottlieb, B. J. Matkowsky, and M. Minkoff, *J. Comput. Phys.* **81**, 421 (1989).
4. A. Bayliss and B. J. Matkowsky, *J. Comput. Phys.* **71**, 147 (1987).
5. A. Bayliss, B. J. Matkowsky, and M. Minkoff, *SIAM J. Appl. Math.* **49**, 1421 (1989).
6. A. Bayliss, B. J. Matkowsky, and M. Minkoff, *SIAM J. Appl. Math.* **49**, 1047 (1989).
7. A. Bayliss and E. Turkel, NASA Technical Memorandum ICOMP 90-24, 1990 (unpublished).
8. W. Blumen, *J. Fluid Mech.* **40**, 769 (1970).
9. W. Blumen, P. G. Drazin, and D. F. Billings, *J. Fluid Mech.* **71**, 305 (1975).
10. C. Canuto and A. Quarteroni, *Math. Comput.* **38**, 67 (1982).
11. V. D. Djordjevic and L. G. Redekopp, *Phys. Fluids* **31**, 3239 (1988).
12. U. Ehrenstein, H. Guillard, and R. Peyret, in *Proceedings of the 12th*