JOURNAL OF COMPUTATIONAL PHYSICS 105, 290–300 (1993)

A Self-Adaptive Domain Decomposition for the Viscous/Inviscid Coupling. I. Burgers Equation

RENZO ARINA

Dipartimento di Ingegneria Aerospaziale, Politecnico di Torino, 10129 Torino, Italy

AND

CLAUDIO CANUTO

Dipartimento di Matematica, Politecnico di Torino, 10129 Torino, Italy and Istituto di Analisi Numerica del C.N.R., 27100 Pavia, Italy

Received March 19, 1990; revised July 13, 1992

A new formulation of the viscous/inviscid coupling, termed χ-formulation, has been applied to the Burgers equation: the equation is modified in such a way that the viscous terms are neglected in dependence of their magnitude. We show that the modified χ-equation can be solved on a single domain at a cost comparable to the cost of solving the original equation, despite a nonlinearity being added. Furthermore, we consider a domain decomposition method, based on the χ-formulation, by splitting the original problem into an inviscid Burgers equation and a χ-viscous Burgers equation. The interface between the subdomains is automatically adjusted by the proposed method, yielding an optimal resolution of the boundary-layer structure.

1. INTRODUCTION

An important problem encountered in fluid dynamics is the accurate calculation of the basic properties, such as pressure, velocity, and temperature, of a given flow field. Treating the fluid as viscous and compressible, the equations to be solved are the Navier–Stokes equations. Even with the simplification of incompressibility, or the assumption of steady flow, the numerical solution of the full Navier–Stokes equations in domains of practical interest is still a difficult task. However, many problems of interest are characterized by high Reynolds numbers. In these cases the effects of viscosity are confined to relatively thin shear layers in the neighborhood of the surfaces bounding the flow and their wakes; outside these regions the inviscid Euler equations are applicable. A fruitful approach is to make separate calculations for the external inviscid flow and the viscous shear layers and to combine them to provide a composite solution to the problem. This approach, known as *viscous/inviscid interaction* [1], has given many important results, and it has reached an advanced state of development. Many efforts have been spent in organizing the two separate parts of the overall calculation to interact with each other in an iterative way so that convergence to the final solution is achieved. The most critical points of all these iterative techniques are their applicability in the presence of massive separation and their extension to three-dimensional flow fields.

A new formulation of the problem of the viscous/inviscid coupling has been recently proposed (see [2]), under the name of *χ-formulation*. The main idea is to replace the viscous terms in the Navier–Stokes equations by a nonlinear function of the viscous terms themselves; such a function, termed χ, coincides with the viscous terms if these are sufficiently large and is zero if the viscous terms are small. In other words, we neglect or retain the viscous terms by inspecting the size of the higher-order derivatives of the solution itself. This approach differs from the existing zonal methods, which are based on an "a priori" decomposition of the flow field into an inviscid part and a viscous one. The χ-formulation—which will be reviewed in some detail in Section 2—yields a self-adaptive splitting of the domain into "viscous" and "inviscid" regions; it guarantees a smooth transition of the solution between the viscous and inviscid zones; finally, it is supported by a rigorous mathematical theory, which accounts for the previous features (see [2, 3]). The χ-formulation can also handle the self-adaptive coupling of other mathematical models, such as the rotational/irrotational approximations of the Navier–Stokes equations (see [4]).

In the original paper by Brezzi *et al.* [2], clear evidence was given of the good accuracy of the χ-formulation in solving model problems with boundary and interior layers.

0021-9991/93 $5.00

However, the problem of finding an efficient implementation of the new formulation—with the aim of obtaining a self-adaptive domain decomposition—was left open.

The present paper is the first of a series, in which the numerical implementation of the χ-formulation is discussed. We address two main issues: (i) how to deal with the non-linearity introduced in the equation by the χ-function; (ii) how to use the information provided by the χ-function to establish an optimal viscous/inviscid domain decomposition. Concerning point (i), we observe that the Navier–Stokes equations are themselves non-linear; hence many solution techniques which work for the original equations can be extended to our χ-perturbed equations. For steady problems, implicit time-advancing techniques with linearization around the solution at the previous time step have proven successful in solving the fluid flow equations. We explore such an approach for the χ-formulation. Coming to point (ii), our goal is to solve the equations in χ-formulation only in a subdomain of the physical domain, while the inviscid equations are solved in the remaining region (the obvious assumption, here, is that solving the inviscid equations is less expensive than solving the original equations). By definition, the inviscid region can be detected by checking where the function χ is identically zero. If such a check is inserted into an iterative procedure, a self-adaptive domain decomposition is obtained, starting from a rough initial guess. During the iterative process, the subdomains are adjusted according to the indications of the χ-function; the grids in each subdomain are adapted to the new geometry, along the philosophy of the moving grid approach (see, e.g., [5]). The final decomposition (which consists of non-overlapping subdomains) is "optimal," in the sense that the inviscid equations are solved precisely where the viscous terms would be negligible, if the complete equations had been solved instead. Thus, the domain decomposition is tailored on the structure of the solution itself.

Any domain-decomposition poses the problem of the matching of the solutions at the interface between the subdomains. The matching conditions are easy to derive and implement if the equations are of the same type on both sides of the interface (e.g., Navier–Stokes/Navier–Stokes or Euler/Euler coupling). The situation becomes much more involved—both theoretically and computationally—if one attemps to match different types of equations (e.g., Euler/Navier–Stokes coupling) (for studies in this direction see, e.g., [6]). Our approach is such that on the interface the function χ is zero; i.e., the solution is inviscid there. Hence, we actually have to implement an Euler/Euler coupling, which is a much easier task.

In the present paper, we confine our investigations to the simple case of the Burgers equation, as a prototype of the balancing of viscous and inviscid effects in fluid flows. In subsequent papers, the Navier–Stokes equations will be considered. Preliminary results about the application of the χ-formulation to the Navier–Stokes equations written in conical coordinates, are reported in Ref. [7]. For the sake of clarity, the analysis of the adaptive algorithm is split into three parts. After recalling the main mathematical properties of the χ-formulation (Section 2), we propose and discuss a solution technique for the χ-equations in a single domain (Section 3); next, we study the viscous/inviscid coupling on a fixed domain decomposition (Section 4); finally, we investigate several strategies for adjusting the decomposition in an interactive process (Section 5).

2. THE χ-FORMULATION FOR THE VISCOUS/INVISCID COUPLING

In this section, we introduce our self-adaptive formulation of the viscous/inviscid coupling, using a very simple convection–diffusion problem as a model for more complex physical situations. Besides, we briefly recall some mathematical properties of our formulation. They will provide a theoretical justification to the approximation method described in the next section.

Suppose we want to solve the following convection-diffusion problem in the interval $(0, L)$:

$$\begin{aligned} &-\nu U_{xx} + aU_x + bU = g, \qquad 0 < x < L, \\ &U(0) = u_0, \qquad U(L) = u_L. \end{aligned} \tag{2.1}$$

Here, $\nu > 0$ is the (constant) viscosity coefficient, whereas the coefficients a, b and the data g are supposed to be smooth functions of x. It is known that, under the technical assumption that $-\frac{1}{2}a_x + b \geqslant 0$ for all $x \in (0, L)$, problem (2.1) has a unique solution, which is as smooth as allowed by the smoothness of a, b, and g.

Now, we want to replace problem (2.1) by a modified problem, in which the viscous term is neglected wherever it is "small". Precisely, we neglect it where the absolute value of U_{xx} falls below a certain threshold, say, less than $\delta > 0$. To this end, choosing a second parameter σ, $\sigma > 0$, let us introduce a continuous function $\chi = \chi_{\delta,\sigma} : \mathbb{R} \to \mathbb{R}$ such that

$$\chi(s) = \begin{cases} s & \text{if} \quad |s| \geqslant \delta + \sigma, \\ 0 & \text{if} \quad |s| \leqslant \delta, \\ \text{a strictly increasing} & \\ \text{function} & \text{if} \quad \delta < |s| < \delta + \sigma. \end{cases} \tag{2.2}$$

Throughout this paper we will invariably use a third-order polynomial in the transition region $\delta < |s| < \delta + \sigma$, defined by the condition that χ be continuously differentiable in $\mathbb{R}$ (see Fig. 1). This may be advisable in the numerical implementation of the χ-problem.

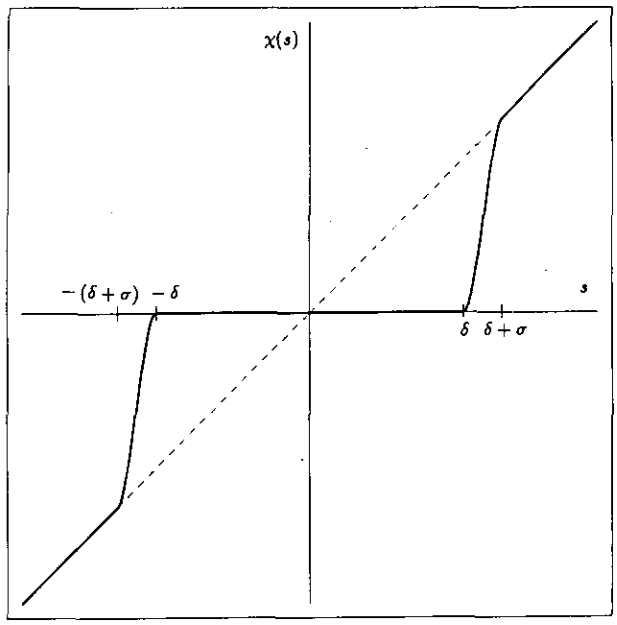

FIG. 1. The function $\chi(s)$ (Eq. (2.2) with third-order polynomial in the transition region), $\delta = 1.5$, $\sigma = 0.5$.

Then, let us consider the modified problem:

$$\begin{aligned} &-\nu\chi(u_{xx}) + au_x + bu = g, \qquad 0 < x < L, \\ &u(0) = u_0, \qquad u(L) = u_L. \end{aligned} \tag{2.3}$$

We define (2.3) to be the *χ-formulation* of problem (2.1). Note that $u = u_{\delta,\sigma}$ depends indeed upon the parameters δ and σ through the function $\chi_{\delta,\sigma}$. For the sake of simplicity, we drop the indices in our notation. The parameter σ in (2.2) has the only purpose of allowing χ to take on all the values between 0 and δ (or $-\delta$ and 0). In other words, the polynomial functions on the intervals $\delta < |s| < \delta + \sigma$ yield a continuous transition between the state $\chi = 0$ (the inviscid state) and the states $\chi = \pm(\delta + \sigma)$ (the fully viscous regime). We will see in the sequel that this is precisely the key feature of our formulation, which generates a smooth behavior of the solution u of (2.3) at the viscous/inviscid interface. Remarkably, the actual size of σ is virtually inessential in our mathematical considerations. Indeed, one can even take the limit as $\sigma \to 0$ in (2.3) and still retain a perfectly mathematically meaningful problem. We will briefly come back on this point later on. Furthermore, let us note that the particular form of the function χ in the transition region $\delta \leqslant |s| \leqslant \delta + \sigma$, does not affect the properties of the χ-solution, provided the requirement of strict monotonicity is satisfied.

It has been proved in [2] that problem (2.3) possesses a unique solution. Moreover, the solution u depends continuously upon the data in a suitable Sobolev norm [3], i.e., the χ-formulation leads to a well-posed problem in the sense of Hadamard. The mathematical analysis relies upon the two following properties of the function χ: (i) χ is a "maximal monotone graph," namely (in our situation) a non-decreasing function defined for all s; (ii) χ coincides with the linear function $\chi(s) = s$ for $|s|$ large enough. Among the mathematical properties of the solution of problem (2.3) which have been established in [2], let us recall here the two following ones:

$$\max_{0 \leqslant x \leqslant L} |U(x) - u(x)| \leqslant C\,\delta\nu; \tag{2.4}$$

u has bounded second derivative (in the sense of distributions) throughout the domain. (2.5)

Property (2.4) is an approximation result, which states that the solution of the χ-formulation deviates from the solution of the original problem by at most a factor proportional to the product $\delta\nu$. The estimate provides a practical guideline for the choice of the parameter δ as a function of the diffusion coefficient ν. Recall that in problems where the viscous/inviscid coupling is relevant, the physical viscosity is usually quite small. Hence, (2.4) indicates that δ is allowed to range in a fairly wide interval, without affecting the accuracy of the approximation. Statement (2.5) is a smoothness result. It implies that u is continuously differentiable all over the domain, in particular in the viscous/inviscid interface region. This property is the distinguished feature of the χ-formulation. Indeed, if we choose a priori the position of the viscous/inviscid interface and solve the full problem on one side of the interface and the inviscid problem ($\nu = 0$) on the other side, then we cannot expect more than C^0-continuity. With our formulation, the position of the interface is not fixed in advance, and it is free to adjust itself in order to produce the smoothest solution of (2.3). How this occurs can be understood by looking at the structure of the interface. The interval $(0, L)$ can be divided into three regions: the viscous region Ω_ν, where $|\chi(u_{xx})| \geqslant \delta + \sigma$ and the original equation is solved; the inviscid region Ω_I, where $\chi(u_{xx}) = 0$ and we solve the reduced equation without diffusion; finally, a transition region Ω_T, where $0 < |\chi(u_{xx})| < \delta + \sigma$ or, equivalently, $\delta < |u_{xx}| < \delta + \sigma$. Thus, the viscous/inviscid interface is not reduced to a point, but it is a region, where $|\chi(u_{xx})|$ takes on all the values between 0 and δ. This explains why χ has to be defined as a monotonic, continuous function on $\mathbb{R}$.

If we let $\sigma \to 0$ in (2.2), the graph of $\chi_{\delta,\sigma}$ tends to the graph of the multivalued function $\tilde{\chi}$: $\mathbb{R} \to$ subsets of $\mathbb{R}$

$$\tilde{\chi}(s) = \begin{cases} 0, & |s| < \delta, \\ [0, \delta], & s = \delta, \\ [-\delta, 0], & s = -\delta, \\ s, & |s| > \delta. \end{cases} \tag{2.6}$$

The corresponding problem has to be formulated in the following form:

$$\begin{aligned} &au_x + bu - g \in v\tilde{\chi}(u_{xx}), \qquad 0 < x < L, \\ &u(0) = u_0, \qquad u(L) = u_L. \end{aligned} \tag{2.7}$$

It means that the left-hand side of (2.7) is zero whenever $|u_{xx}| < \delta$, it equals vu_{xx} whenever $|u_{xx}| > \delta$, and it can take any value between 0 and $v\delta$ (resp., $-v\delta$ and 0) whenever $u_{xx} = +\delta$ (resp. $-\delta$). By the results in [2], (2.7) is again a well-posed problem. In this case the viscous/inviscid interface is such that u_{xx} takes the constant value δ (or $-\delta$) there (i.e., u is a piece of parabola), whereas $|\chi(u_{xx})|$ varies between 0 and δ in a continuous way.

In the rest of this paper, we shall carry out our discussion of the χ-formulation on the non-linear model problem:

$$\begin{aligned} &-vU_{xx} + (f(U))_x = g, \qquad 0 < x < L, \\ &U(0) = U(L) = 0, \end{aligned} \tag{2.8}$$

where $f: \mathbb{R} \to \mathbb{R}$ is a smooth, convex function. We assume that the inviscid problem

$$\begin{aligned} &(f(U))_x = g, \qquad 0 < x < L, \\ &U(0) = 0, \end{aligned} \tag{2.9}$$

admits a smooth solution such that $f'(U) > 0$ for $0 \leqslant x \leqslant L$. In this case, the solution of (2.8) will exhibit a boundary-layer at $x = L$. The χ-formulation for problem (2.8) reads

$$\begin{aligned} &-v\chi(u_{xx}) + (f(u))_x = g, \qquad 0 < x < L, \\ &u(0) = u(L) = 0, \end{aligned} \tag{2.10}$$

where χ is again given by (2.2). This problem enjoys precisely the same mathematical properties described above as the linear problem (2.3) (see [3]). If the cutoff parameter δ is not exceedingly small, the solution of (2.10) will be such that $\chi(u_{xx})$ is zero in most of the interval $(0, L)$. It will be non-zero only in a neighborhood of $x = L$, where the boundary condition generates a boundary layer.

3. A SOLUTION TECHNIQUE OF THE χ-PROBLEM

In order to solve the non-linear χ-problem (2.10), a time-dependent technique can be used. The solution is viewed as the limit as $t \to \infty$ of the solution $u(x, t)$ of the following problem:

$$\begin{aligned} &u_t - v\chi(u_{xx}) + (f(u))_x = g, \qquad 0 < x < L,\ t > 0, \\ &u(0, t) = u(L, t) = 0, \qquad t > 0, \\ &u(x, 0) = u_0(x), \qquad 0 < x < L. \end{aligned} \tag{3.1}$$

Introducing a time step Δt and setting $u^n \simeq u(n\,\Delta t)$, the solution is advanced to $u^{n+1} \simeq u((n+1)\,\Delta t)$, by one step of the backward Euler scheme. Setting $\delta u^n = u^{n+1} - u^n$, we approximate (3.1) by

$$\begin{aligned} &\delta u^n + \Delta t\{-v\chi(u_{xx}^{n+1}) + (f(u^{n+1}))_x\} = 0, \qquad 0 < x < L, \\ &u^{n+1}(0) = u^{n+1}(L) = 0. \end{aligned} \tag{3.2}$$

Finally, we linearize the left-hand side of (3.2) around u^n. In doing this, we assume that the χ-function we are working with is everywhere continuously differentiable; this is the case for our choice of the χ-function (see Fig. 1). After linearization, (3.2) can be written in δ-form as

$$\begin{aligned} &\delta u^n + \Delta t\{-v\chi'(u_{xx}^n)(\delta u^n)_{xx} + (f'(u^n)\,\delta u^n)_x\} \\ &\qquad = -\Delta t\{-v\chi(u_{xx}^n) + (f(u^n))_x - g\}, \qquad 0 < x < L, \\ &\delta u^n(0) = \delta u^n(L) = 0. \end{aligned} \tag{3.3}$$

Obviously, the present approach is far from being efficient, since at each time iteration we retain the computational cost of solving a parabolic problem all over the domain, even where the χ-function is turned off. Indeed each iteration is as expensive as an iteration of the same time-dependent method applied to the original model problem (2.8), while it is likely that the addition of the nonlinearity χ leads to an increase of the number of iterations to converge to steady state. In the next sections, we will use a more clever strategy, namely, splitting the problem into a simpler hyperbolic problem in most of the domain and a χ-problem in a small region containing the boundary layer. However, the focus of the present section is twofold: (i) to investigate the convergence behavior of the iterative procedure (3.3) in the simplest case of spatial discretization; (ii) to give evidence of the good behavior of the χ-solution with respect to the exact solution of problem (2.8). In our investigations, we choose the Burgers equation as a model of convection–diffusion problem. Thus, $f(u) = \frac{1}{2}u^2$ in (2.10), the interval is (0, 1), and $g(x) = x$. Different values of v will be considered.

For the sake of simplicity, we use finite differences in space on a uniform mesh of $N+1$ grid points throughout the complete domain [0, 1]. In the next sections, we will work with a grid clustered in the boundary layer region. Diffusive terms are discretized by three-point centered differences, whereas convective terms are discretized by second-order upwind differences. The choice of N will depend on v, in order to guarantee a good resolution of the boundary layer structure.

As far as the choice of the parameter Δt is concerned, recall that we are interested in solving the steady problem (2.10), not in representing faithfully the time evolution of the solution of problem (3.1). Hence, as frequently occurs in

TABLE I

Accuracy of the Solution of Problem (2.10) for Different Values of σ

ν	N	δ	σ	$n \cdot it$	$\|u^{\chi}_{\delta,N} - u_N\|_2$	$\|u^{\chi}_{\delta,N} - u_N\|_\infty$	$\|u^{\chi}_{\delta,N} - u_{ex}\|_2$	$\|u^{\chi}_{\delta,N} - u_{ex}\|_\infty$
10^{-2}	200	1.0	2.0	18	$0.47998\ 10^{-9}$	$0.15820\ 10^{-3}$	$0.38913\ 10^{-7}$	$0.88042\ 10^{-2}$
		1.0	1.0	20	$0.27327\ 10^{-9}$	$0.11827\ 10^{-3}$	$0.39195\ 10^{-7}$	$0.87749\ 10^{-2}$
		1.0	0.5	19	$0.19384\ 10^{-9}$	$0.99016\ 10^{-4}$	$0.39334\ 10^{-7}$	$0.87607\ 10^{-2}$
		1.0	0.2	20	$0.14655\ 10^{-9}$	$0.85605\ 10^{-4}$	$0.39432\ 10^{-7}$	$0.87509\ 10^{-2}$

the computations of steady problems [8], we think of Δt as a "pseudo-time", which may act as an acceleration parameter of the iterative method (3.3). Whenever we are allowed to choose a very large Δt, we are essentially using Newton's method applied to problem (2.10), getting local quadratic convergence. However, in many cases, an arbitrarily large Δt cannot be chosen. The iterative method may converge to a "wrong" solution (for instance, a non-physical solution, in the case of the Navier–Stokes equations), or it may fail to converge. The latter situation was indeed encountered when we attempted to solve problem (2.10) by the plain Newton method. The non-linearity of the χ-function leads to the appearance of periodic cycles when δ and δ/σ exceed critical values. We found that convergence could still be achieved by choosing a sufficiently small Δt. However, the number of iterations was usually very large and quite sensitive to the choice of Δt. So, we resorted to a different, more effective strategy: keeping Δt large and backtracking along the direction defined by δu^n whenever a prescribed convergence criterion was not satisfied. More precisely, we adopted the automatic backtracking line search procedure as described in Section 6.3 of [9]. In our cases, we found that this backtracking algorithm is invoked only in a few initial iterations, with a negligible increase in computational cost. For the sake of completeness, we also implemented the trust-region method (see Section 6.4 in [9]), and we found that the convergence rate was comparable with that of the backtracking method, but with a larger cost. So, the results we are going to illustrate, were obtained by the backtracking procedure.

In Tables I and II, we report the results of several numerical tests concerning the efficiency of the iterative scheme and the accuracy of the χ-formulation for different values of δ and σ. Three values of the diffusion parameter ν have been considered: in each case, the number of space intervals N has been chosen in such a way that the boundary layer be correctly represented. The efficiency of the iterative method to solve (2.10) is measured by the number of iterations. All the results have been obtained with $\Delta t = 10^8 \max_{1 \le j \le N} (|u^n_j|/N)$ (i.e., the Courant number based on the hyperbolic part was 10^8). We report the number of iterations needed to drive $\|res^n\|_2$ below 1.10^{-13} starting from the initial guess $u_0(x) = (1-x)x$, where $\|\text{res}^n\|_2$ denote the discrete l^2-norm of the residual $\text{res}^n = -\nu\chi(u^n_{xx}) + (f(u^n))_x - g$ at the interior grid-points of the domain.

TABLE II

Accuracy of the Solution of Problem (2.10) for Different Values of δ (with Fixed Ratio δ/σ) and of the Diffusivity ν

ν	N	δ	σ	$n \cdot it$	$\|u^{\chi}_{\delta,N} - u_N\|_2$	$\|u^{\chi}_{\delta,N} - u_N\|_\infty$	$\|u^{\chi}_{\delta,N} - u_{ex}\|_2$	$\|u^{\chi}_{\delta,N} - u_{ex}\|_\infty$
10^{-2}	200	0.0	0.0	11	0.0	0.0	$0.40079\ 10^{-7}$	$0.86880\ 10^{-2}$
		1.0	0.5	19	$0.19384\ 10^{-9}$	$0.99016\ 10^{-4}$	$0.39334\ 10^{-7}$	$0.87607\ 10^{-2}$
		10.0	5.0	31	$0.15745\ 10^{-7}$	$0.95780\ 10^{-3}$	$0.35043\ 10^{-7}$	$0.93914\ 10^{-2}$
		10^2	50.0	30	$0.12647\ 10^{-5}$	$0.93838\ 10^{-2}$	$0.10494\ 10^{-6}$	$0.15600\ 10^{-1}$
		10^3	$5 \cdot 10^2$	36	$0.84447\ 10^{-4}$	$0.91584\ 10^{-1}$	$0.79356\ 10^{-5}$	$0.91155\ 10^{-1}$
10^{-4}	6,400	0.0	0.0	27	0.0	0.0	$0.22038\ 10^{-6}$	$0.72975\ 10^{-1}$
		10^2	50.0	15	$0.11808\ 10^{-14}$	$0.15687\ 10^{-5}$	$0.22037\ 10^{-6}$	$0.77191\ 10^{-1}$
		10^3	$5 \cdot 10^2$	16	$0.11407\ 10^{-12}$	$0.16291\ 10^{-4}$	$0.22028\ 10^{-6}$	$0.77177\ 10^{-1}$
		10^4	$5 \cdot 10^3$	15	$0.10617\ 10^{-10}$	$0.16661\ 10^{-3}$	$0.21936\ 10^{-6}$	$0.77033\ 10^{-1}$
10^{-6}	20,000	0.0	0.0	12	0.0	0.0	$0.19676\ 10^{-6}$	0.88932
		10^4	$5 \cdot 10^3$	10	$0.70988\ 10^{-15}$	$0.30951\ 10^{-5}$	$0.19676\ 10^{-6}$	0.88932
		10^5	$5 \cdot 10^4$	10	$0.70988\ 10^{-15}$	$0.30951\ 10^{-5}$	$0.19676\ 10^{-6}$	0.88932
		10^6	$5 \cdot 10^5$	10	$0.26396\ 10^{-11}$	$0.17583\ 10^{-3}$	$0.19676\ 10^{-6}$	0.88932

The accuracy of the solution of the χ-formulation is measured by the discrete l^2- or l^∞-norms of the difference between the solution $u^{\chi}_{\delta,N}$ of the finite-difference approximation to problem (2.10) using N grid intervals, and the solution u_N of the similar spatial approximation to the original Burgers equation (2.8).

Table I allows us to investigate the sensitiveness on the χ-problem of the parameter σ, which measures the steepness of the function $\chi(s)$ in the transition region between the viscous and inviscid states. As theoretically predicted, σ has a negligible influence both on the rate of convergence and on the final accuracy.

The sensitiveness of the χ-problem to the parameter δ (for fixed ratio δ/σ), and the diffusivity ν is illustrated by Table II. In order to evaluate the influence of the χ-nonlinearity on the overall cost of solving problem (2.10), we have applied the same iterative method to the solution of the viscous Burgers equation (2.8). The corresponding results appear in the rows $\delta = \sigma = 0$. The number of iterations is always moderate, and only weakly dependent on δ, until the cutoff level reaches the order of magnitude of the viscous term u_{xx} within the boundary layer. Note that the convergence behavior becomes better and better as the diffusivity parameter decreases. This is also true for the exact

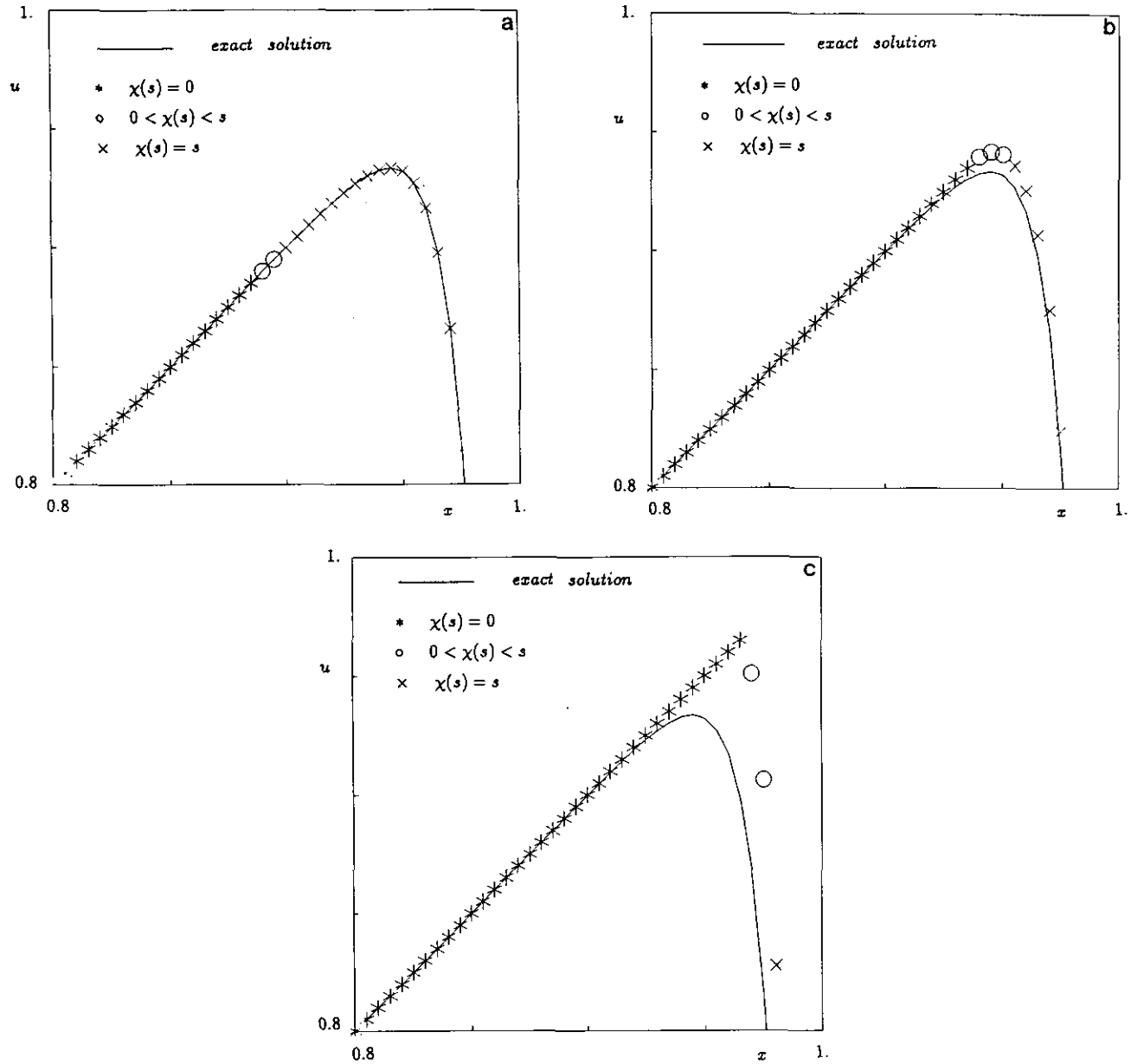

FIG. 2. Comparison between the solutions to the Burgers equation (solid line) and the χ-Burgers equation (symbols), $\nu = 10^{-2}$, $N = 200$: (a) $\delta = 1$, $\sigma = 0.5$; (b) $\delta = 10^2$, $\sigma = 50$; (c) $\delta = 10^3$, $\sigma = 5.10^2$; $* \chi(s) = 0$; $\circ\ 0 < \chi(s) < s$; $\times\ \chi(s) = s$.

Burgers problem, and it should be related to the backtracking strategy. Indeed, solving the exact Burgers equation without backtracking (i.e., pure Newton method) requires seven iterations for $\nu = 10^{-2}$, 11 iterations for $\nu = 10^{-4}$ and 14 iterations for $\nu = 10^{-6}$, i.e., the convergence rate exhibits an opposite trend. We conclude that for low-diffusion problems, the overhead of the χ-formulation becomes negligible with respect to the cost of solving the original problem. As far as accuracy is concerned, the behavior of the errors are in good agreement with estimate (2.4), which holds for the solutions of the finite-difference approximations to (2.8) and (2.10), too [3]. Furthermore, the solution of the χ-formulation of the Burgers equation is a close approximation to the exact solution for a wide interval of δ's.

The χ-solutions $u^{\chi}_{\delta,N}$, for $\nu = 10^{-2}$, $N = 200$, and different values of δ (with δ/σ constant), are graphically shown in Fig. 2 in the interval $[0.8, 1]$, together with the exact solution (solid line). Different symbols are used to represent the χ-solutions in the inviscid region Ω_I (stars), in the transition region Ω_T (circles), and in the viscous region Ω_V (crosses). Note that by increasing δ one moves the inviscid/viscous interface toward the boundary layer; however, the interface region is always across two-three grid points. Furthermore, except for the badly resolved case, 2c, corresponding to the values $\delta = 10^3$ and $\sigma = 5.10^2$, the matching at the viscous/inviscid interface appears to be quite smooth.

4. FIXED χ-VISCOUS/INVISCID DECOMPOSITION

The previous results show that the solution of our model problem is inviscid in most of the domain, except for a region near $x = L$. So, it is quite natural to split the domain into an inviscid region $[0, b]$ $(0 < b < L)$, where a cheaper hyperbolic problem is solved, and a χ-viscous region $[b, L]$ containing the boundary layer, where the χ-viscous problem is solved. In the rest of the paper we will investigate the effects of such a domain decomposition. First, in the present section, we will assume that the interface between the two regions is fixed and is determined in advance (information about its position can be obtained by overestimating the thickness of the boundary layer). Subsequently, in the next section we will introduce an automatic detection of the interface position, which will allow us to move it until an optimal position is reached.

With the adaptive technique in mind, it is convenient to reformulate our boundary-value problem in a reference domain, in such a way that the viscous and inviscid regions in physical space correspond to a fixed decomposition of the reference domain. To this end, we fix a family of coordinate mappings from the physical domain $[0, L]$ to the reference domain $[0, 1]$, depending upon a parameter β:

$$\Xi(x, \beta)\colon [0, L] \times [0, L] \to [0, 1]. \tag{4.1}$$

In the sequel, we choose

$$\Xi(x, \beta) = \begin{cases} \dfrac{x}{2\beta}, & 0 \leqslant x < \beta, \\ 1 - \dfrac{L - x}{2(L - \beta)}, & \beta < x \leqslant L, \end{cases} \tag{4.2}$$

which defines a linear coordinate mapping in each subdomain.

Let b denote the physical coordinate of the interface point. We select the coordinate transformation in the family (4.2) by setting

$$\xi(x) = \Xi(x, b). \tag{4.3}$$

Note that we invariably map the interface into the point of coordinate $\frac{1}{2}$ in the reference domain.

Problem (2.10) can be written in the new coordinates as

$$\begin{aligned} &-\nu\chi(\xi_x(\xi_x u_\xi)_\xi) + \xi_x(f(u))_\xi = g, \\ &\quad \text{for} \quad 0 < \xi < \tfrac{1}{2} \text{ or } \tfrac{1}{2} < \xi < 1, \\ &u(0) = u(1) = 0, \\ &+ \text{a suitable interface condition at } \xi = \tfrac{1}{2}. \end{aligned} \tag{4.4}$$

As pointed out in the introduction, it is preferable that the coupling between the inviscid region and the viscous one takes place at a point where the solution can be considered inviscid. Therefore, we assume that the interface point b has been chosen in such a way that $\chi(\xi_x(\xi_x u_\xi)_\xi) \equiv 0$ both in the interval $[0, \frac{1}{2})$ and in a right neighborhood of $\xi = \frac{1}{2}$. Thus, at the interface $\xi = \frac{1}{2}$ the proper matching condition is of hyperbolic type; i.e., only C^0-continuity has to be enforced. Hence, problem (4.4) can be made precise in the following domain decomposition problem:

$$\begin{aligned} \xi_x(f(u^I))_\xi &= g, \qquad 0 < \xi < \tfrac{1}{2}, \\ u^I(0) &= 0; \end{aligned} \tag{4.5.1}$$

$$\begin{aligned} -\nu\chi(\xi_x(\xi_x u^V_\xi)_\xi) + \xi_x(f(u^V))_\xi &= g, \qquad \tfrac{1}{2} < \xi < 1, \\ u^V(\tfrac{1}{2}) &= u^I(\tfrac{1}{2}), \\ u^V(1) &= 0, \end{aligned} \tag{4.5.2}$$

where u^I ($\equiv u$ in $[0, \frac{1}{2}]$) is the inviscid part of the solution, and u^V ($\equiv u$ in $[\frac{1}{2}, 1]$ is the χ-viscous part.

Let us consider the numerical approximation of such a problem. As far as the spatial discretization is concerned, the domain decomposition makes it easy to use a different grid in each subdomain. Precisely, we use $M + 1$ equally spaced points in the interval $[0, \frac{1}{2}]$, and $N + 1$ equally spaced points in the interval $[\frac{1}{2}, 1]$. The corresponding

distribution of grid points in the physical domain depends on the structure of the coordinate mapping (4.1). Let us point out that in each subdomain one could use different spatial discretization schemes, choosing among the most appropriate ones for the type of equation to be solved in the subdomain. For instance, an infinite-order spectral collocation method is appropriate in the χ-viscous region, where the solution is smooth but with several relevant physical features (think of the separation bubbles in multidimensional flows), whereas a low order finite volume method is normally sufficient in the external, inviscid region.

Let us now discuss how to solve efficiently the coupled problem (4.5.1)–(4.5.2). Two iterative strategies can be devised to this purpose. The first approach takes advantage on the specific structure of the hyperbolic operator in (4.5.1), which allows information to be propagated only from the left-hand side to the right-hand side of the domain. This situation is the scalar analog of a supersonic inviscid flow. Thus, first one solves problem (4.5.1), obtaining the exact interface value $u^I(\frac{1}{2})$; next, one solves problem (4.5.2). The hyperbolic problem can be solved in a cheap way by a space marching technique, whereas the χ-problem can be solved by the time-advancing technique presented in the previous section and discussed therein. The second approach is suitable for handling more general hyperbolic problems, where information propagates in both directions, as in the case of transonic flows. A time-advancing technique as the one discussed in Section 3 is applied to both problems (4.5.1) and (4.5.2). The inviscid solution u^I is advanced by one time step on the left domain; next the current value of $u^I(\frac{1}{2})$ is passed to the right domain, where the χ-viscous solution is then advanced by one time step. The procedure is repeated until a steady state is reached. So, during the iterative method one simultaneously enforces the differential equation at the interior points of each domain and the matching condition at the interface. In the sequel, we will explore the performances of such an iterative procedure.

We investigated two cases, corresponding to the values $\nu = 10^{-4}$ and $\nu = 10^{-6}$. We chose $b = 0.995$ in the former case and $b = 0.99995$ in the latter one. The number of grid intervals in the inviscid region was fixed to $M = 5$. The χ-parameters were $\delta = \sigma = 1$ in all runs. Table III shows the convergence behavior of the iterative method, and reports the accuracy of the decomposition as functions of the number N of the grid intervals in the χ-region. We report the number of iterations needed to drive the l^2-norm of the overall residual in (0, 1) below $1 \cdot 10^{-13}$ and the norms of the errors of the χ-solution with a very good approximation of the exact solution computed with 20,000 points in the boundary layer. The behavior of the iterative procedure is quite similar to what we found in the previous section. Conversely, a comparable accuracy is obtained with a much smaller number of grid points, provided the interface is placed as close as possible to the boundary-layer edge. In these tests, the position of the interface is determined a priori, but this may be difficult to do in more severe situations. This remark suggests developing an automatic strategy for finding an optimal position of the interface.

TABLE III

Number of Iterations and Accuracy of the Fixed χ-Viscous/Inviscid Decomposition (Problems (4.5.1)–(4.5.2)), with $M = 5$, $\delta = \sigma = 1$, for Different Values of Grid Intervals N in the χ-Viscous Region

ν	N	$n \cdot it$	$\|u^\chi_{\delta,N} - u_{ex}\|_2$	$\|u^\chi_{\delta,N} - u_{ex}\|_\infty$
10^{-4}	2^4	13	$0.57113\ 10^{-5}$	0.25196
	2^5	13	$0.48088\ 10^{-6}$	$0.72974\ 10^{-1}$
	2^6	13	$0.16819\ 10^{-7}$	$0.11864\ 10^{-1}$
	2^7	14	$0.21624\ 10^{-8}$	$0.62926\ 10^{-2}$
10^{-6}	2^4	18	$0.57382\ 10^{-8}$	0.25769
	2^5	17	$0.49161\ 10^{-9}$	$0.77336\ 10^{-1}$
	2^6	16	$0.24291\ 10^{-10}$	$0.18071\ 10^{-1}$
	2^7	15	$0.18679\ 10^{-11}$	$0.68390\ 10^{-2}$

5. SELF-ADAPTIVE χ-VISCOUS/INVISCID DECOMPOSITION

We now describe an automatic strategy of domain decomposition, which leads to an optimal position of the interface point b. Our strategy is based on the two following heuristic requirements: (i) the solution must be inviscid (in the sense that $\chi(u_{xx}) \equiv 0$) at the interface point; (ii) the χ-viscous region, which contains the boundary layer, must be as small as possible. Requirement (i) guarantees an inviscid (although smooth) matching at the interface; requirement (ii) guarantees that the grid points belonging to the χ-viscous domain are really clustered within the boundary-layer region, thus yielding an optimal resolution. The previous conditions can be satisfied by requiring that, in the χ-viscous region, the function $\chi(u_{xx})$ be identically zero at a fixed, small number of consecutive grid points, starting from the interface. More precisely, denote by x_i $(i = 0, ..., M + N)$ the grid points in the physical domain, and recall that the nodes $\{x_i,\ i = 0, ..., M\}$ belong to the inviscid region $[0, b]$, whereas the nodes $\{x_i,\ i = M, ..., M + N\}$ belong to the χ-viscous region $[b, L]$ (thus, $x_M = b$). Let N_0 $(0 \leqslant N_0 \ll N)$ be a fixed integer. We require the following property to be satisfied.

Interface Position Criterion. If u is the global solution of problems (4.5.1)–(4.5.2), then

- $\chi(u_{xx}) = 0$ at the grid points $\{x_i,\ i = 0, ..., M + N_0\}$.
- $\chi(u_{xx}) \neq 0$ at the grid point x_{M+N_0+1}.

Note that in our model problem we expect $\chi(u_{xx})$ to be actually nonzero at all the grid points x_i, $i \geqslant M + N_0 + 1$. As far as the choice of N_0 is concerned, let us observe that $N_0 = 0$ is allowed because it matches requirements (i) and (ii) mentioned above. However, on the discrete level, it is advisable to choose $N_0 > 0$ in order to have a "safety" region around the interface, in which the solution is inviscid. Typically, N_0 will be a small percentage of N, depending upon the spatial discretization scheme used in the viscous region.

The above criterion can be enforced through the time-advancing iterative procedure described in the previous sections. Let $u^{(\text{old})}$ be a sufficiently accurate approximation of the solution to (4.5.1)–(4.5.2), corresponding to a certain interface position $b^{(\text{old})}$. The function $\chi(u_{xx}^{(\text{old})})$ is computed at all grid points in each subdomain (the second derivative is approximated consistently with the numerical scheme used at each grid point). Let us define the integer i^* by

$$i^* + 1 = \min\{i = 1, \ldots, M + N - 1 \mid \chi(u_{xx}^{\text{old}}) \neq 0 \text{ at } x_i\}, \quad (5.1)$$

and set $x^* = x_{i^*}$. Thus, $\chi(u_{xx}^{(\text{old})}) \equiv 0$ at all the grid points less or equal to x^*, whereas $\chi(u_{xx}^{(\text{old})}) \neq 0$ at the first grid point larger than x^*. Therefore, according to the interface position criterion, we choose the new interface position $b^{(\text{new})}$ in such a way that $x_{M+N_0}^{(\text{new})} = x^*$. Since $x_{M+N_0}^{(\text{new})}$ corresponds to the abscissa $\xi = \frac{1}{2}(1 + N_0/N)$ in the reference domain, the interface $b^{(\text{new})}$ satisfies the relation

$$\Xi(x^*, b^{(\text{new})}) = \frac{1}{2}\left(1 + \frac{N_0}{N}\right), \quad (5.2)$$

which can be uniquely solved for $b^{(\text{new})}$, provided $x^* \geqslant x_{\min}^* = (N_0/N)L$ (see Fig. 3). The new value of the interface defines the new coordinate mapping $\xi = \Xi(x, b^{(\text{new})})$, hence, a new grid distribution.

In order to continue the time iterations on the new grid, one has to transfer the values of $u^{(\text{old})}$ on the new grid. This can be accomplished either (i) by *linear interpolation* of the solution on each subinterval of the old grid, or (ii) by introducing a *time-dependent metric*, $\xi(x, t) = \Xi(x, b(t))$, which takes into account the grid displacement within the time-advancing scheme in a natural way (see, e.g., [Ref. 10]). For instance, the differential equation (4.5.2) may be replaced by

$$u_t + \xi_t u_\xi - \nu\chi(\xi_x(\xi_x u_\xi)_\xi) + \xi_x(f(u))_\xi = g,$$
$$\tfrac{1}{2} < \xi < 1, \quad t > 0. \quad (5.3)$$

In our numerical tests, we found the adaptive domain decomposition based on linear interpolation to be superior to the time-dependent metric approach in terms of robustness and convergence speed.

Our numerical tests (see below) indicate that the major displacements of the interface occur during the initial iterations. Subsequently, only minor adjustements of the interface take place. Therefore, in order to optimize the efficiency of the self-adaptive domain decomposition technique, it is advisable to avoid the update of the interface position whenever the displacement $\Delta b = |b^{(\text{new})} - b^{(\text{old})}|$ computed by the Interface Position Criterion, falls below a threshold $\Delta b_{\min}$.

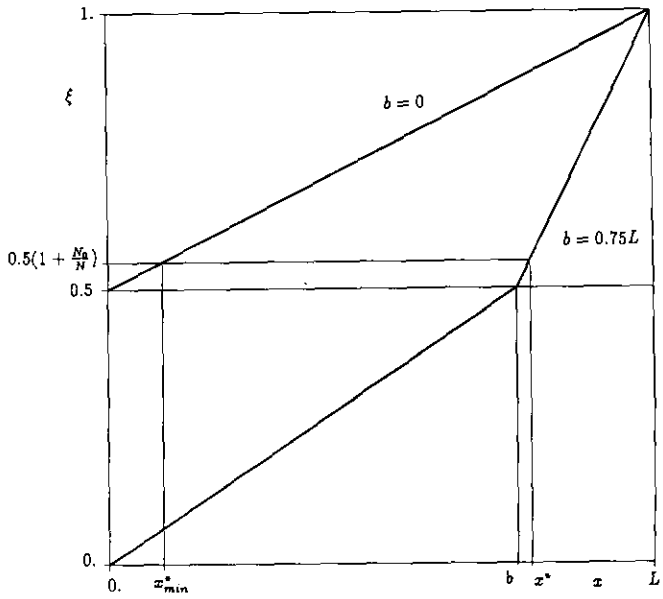

FIG. 3. Piecewise linear coordinate mapping; interface position criterion ($N_0 = N/10$).

We applied the self-adaptive domain decomposition method with linear interpolation, to the solution of the same Burgers problem considered in the previous section. Hereafter we report the results corresponding to the more severe case, namely $\nu = 10^{-6}$. The χ-parameters were $M = 5$, $\delta = \sigma = 1$, and $N_0 = 5$ for all cases. In Table IV we report the number of iterations and the accuracy for different values of grid intervals in the χ-viscous region. In all cases, except the last one, the initial position of the interface was $b_0 = 0.5$; this

TABLE IV

Number of Iterations and Accuracy of the Self-Adaptive Domain Decomposition, for Different Values of Grid Intervals in the χ-Viscous Region, with $\nu = 10^{-6}$, $\delta = \sigma = 1$, $M = 5$, $N_0 = 5$

N	$n \cdot it$	b	$\|u_{\delta,N}^{\chi} - u_{\text{ex}}\|_2$	$\|u_{\delta,N}^{\chi} - u_{\text{ex}}\|_\infty$
2^5	26	0.99986	$0.85209\ 10^{-8}$	0.30688
	54	0.999958	$0.23308\ 10^{-9}$	$0.55750\ 10^{-1}$
2^6	24	0.99982	$0.45781\ 10^{-8}$	0.23242
	53	0.999967	$0.46172\ 10^{-11}$	$0.95375\ 10^{-2}$
2^7	25	0.99988	$0.50897\ 10^{-10}$	$0.26540\ 10^{-1}$
	92	0.999970	$0.34716\ 10^{-12}$	$0.29886\ 10^{-2}$
$2^7(*)$	39	0.999977	$0.13303\ 10^{-12}$	$0.18490\ 10^{-2}$

choice represents a very rough overestimate of the boundary-layer thickness. Conversely, the last row (*) correspond to the choice $b_0 = 0.9999995$; i.e., the boundary-layer thickness was underestimated by two orders of magnitude. The value of b was determined by enforcing the Interface Position Criterion. For each value of N, the upper row corresponds to the stopping criterion $\Delta b_{\min} = 10^{-4}$, while the lower row has been obtained with $\Delta b_{\min} = 10^{-7}$. This last value has also been adopted for the case $N = 2^7$ (*). Figures 4a and 4b, show the convergence history of the l^2 norm of the residual $\|\text{res}\|_2$, and the corresponding displacement of the interface b for the case $N = 2^7$. Figure 4b demonstrates that after the first iterations, an essentially uncontrolled interface adjustment only produces oscillations of the residual, without a significant improvement in the accuracy. Conversely, in Fig. 4a, we see how a mild limitation on the interface displacements leads to a good compromise between accuracy and efficiency. Table IV shows that the choice $\Delta b_{\min} = 10^{-4}$ gives comparable results to those obtained in the previous section, both for convergence and accuracy. The last row of Table IV demonstrates that our self-adaptive technique works nicely also if we start from a severe underestimate of the boundary-layer thickness.

Finally, in Fig. 5 we compare the solution of the adaptive domain-decomposition problem, for $N = 2^7$ and $\Delta b_{\min} = 10^{-7}$ (symbols), with the exact solution to the Burgers equation (solid line) (the small oscillations in the solid line are due to the single-precision subroutines of our GKS graphic package). Note that the inviscid part of the χ-solution in the χ-region is formed by $N_0 = 5$ points, as required by the interface condition. The accuracy of the χ-solution is quite good. For the sake of comparison, we tried to run the same problem on the self-adaptive ODE collocation code COLSYS, available through Netlib (see, e.g., [11]), but we did not obtain convergence with as many as 7000 grid intervals. If we consider the less severe situation of the diffusion parameter $\nu = 10^{-4}$, then COLSYS converges in 31 iterations with a final grid of 640 intervals, whereas the present technique converges (with the same accuracy) in 28 iterations with $N = 64$ grid intervals. Thus our results in one dimension look encouraging for the extension to multidimensional problems.

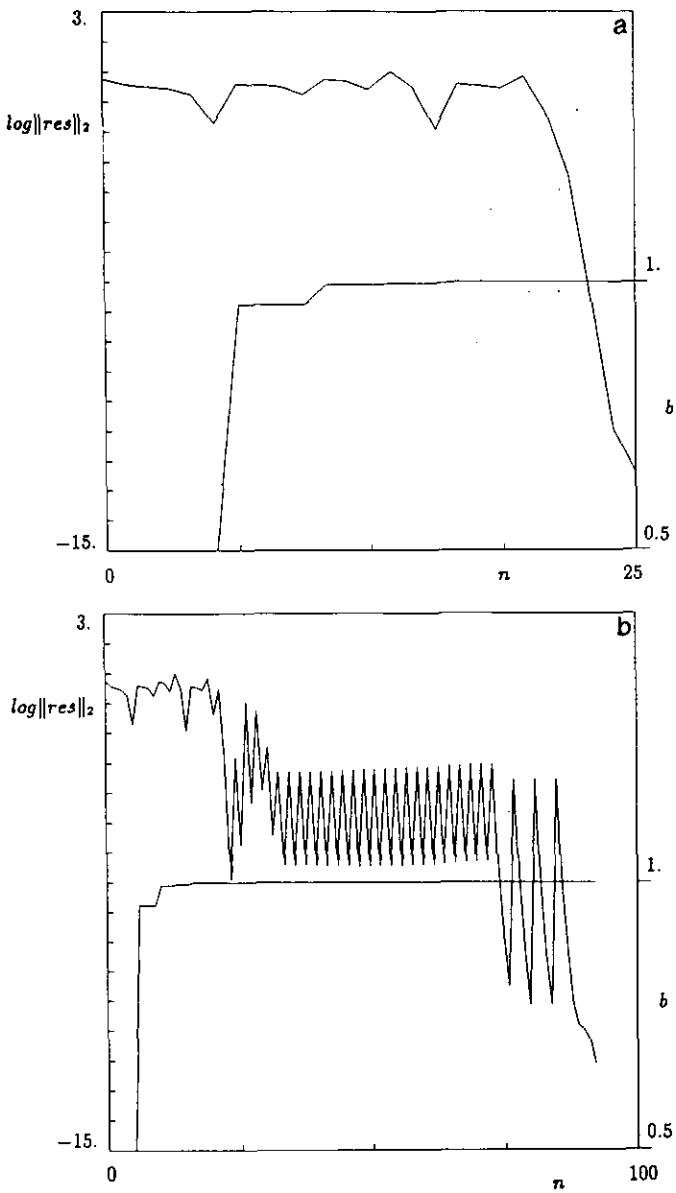

FIG. 4. Convergence histories ($\log \|\text{res}\|_2$) and interface histories (b): (a) $\Delta b_{\min} = 10^{-4}$; (b) $\Delta b_{\min} = 10^{-7}$.

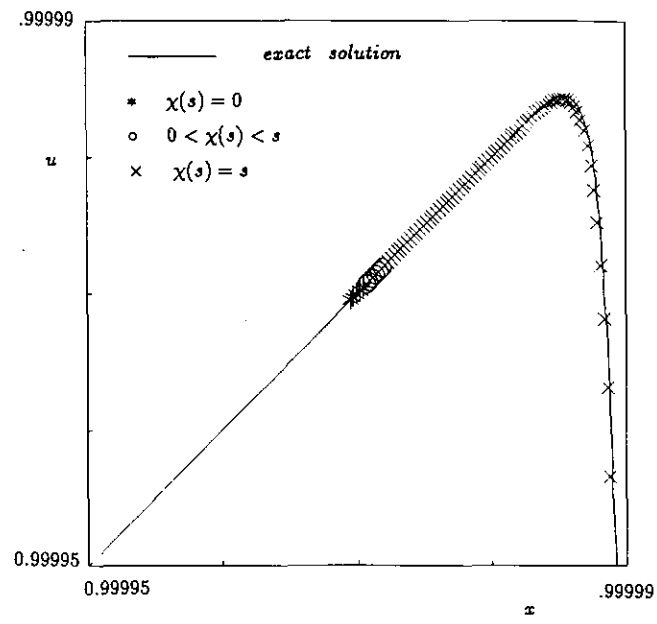

FIG. 5. Comparison between the exact solution to the Burgers equation (solid line) and the solution of the χ-viscous/inviscid adaptive method (symbols), for $\nu = 10^{-6}$: * $\chi(s) = 0$; O $0 < \chi(s) < s$; × $\chi(s) = s$.

6. CONCLUSIONS

A recently proposed formulation of the viscous/inviscid coupling, termed χ-formulation has been applied to the solution of the Burgers equation. In such a formulation, the equation is modified in such a way that the viscous terms are neglected in dependence of their magnitude. This yields a natural decomposition of the domain into a viscous region and an inviscid one. We have shown that the modified χ-equation can be solved on a single domain at a cost comparable to the cost of solving the original equation, despite the addition of a nonlinearity. Furthermore, we have considered a domain decomposition method, based on the χ-formulation, by splitting the original problem into an inviscid Burgers equation and a χ-viscous Burgers equation; an efficient iterative algorithm which alternates the solution on the subdomains is investigated. An interface position criterion is proposed to adjust the domain decomposition according to the indications of the χ-formulation, in such a way that the viscous/inviscid interface is automatically placed near the boundary-layer edge. This yields an optimal resolution of the boundary-layer structure.

ACKNOWLEDGMENT

Valuable discussions with Franco Brezzi during the course of this work are gratefully acknowledged.

REFERENCES

1. R. C. Lock and B. R. Williams, *Prog. Aerospace Sci.* **24**, 51 (1987).
2. F. Brezzi, C. Canuto, and A. Russo, *Comput. Methods Appl. Mech. Eng.* **73**, 317 (1989).
3. A. Rizza, *Calcolo* **27** (3/4), 219 (1990).
4. Y. Achdou and O. Pironneau, *C.R. Acad. Sci. Paris* **188** (1991); Report, Math. Dept., University of Houston.
5. J. G. Verwer, J. G. Blom, and J. M. Sanz-Serna, *J. Comput. Phys.* **82**, 454 (1989).
6. F. Gastaldi, A. Quarteroni, and G. Sacchi Landriani, in *Proceedings, Int. Conference on Domain Decomposition Methods, Houston, USA, 1989*, edited by R. Glowinski *et al.*
7. R. Arina and C. Canuto, in *Proceedings, 12th Int. Conference on Numerical Methods in Fluid Dynamics, Oxford, UK, 1990*, edited by K. Morton.
8. C. Hirsch, *Numerical Computation of Internal and External Flows*, Vol. 1 (Wiley, New York, 1988).
9. J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (Prentice–Hall, Englewood Cliffs, NJ, 1983).
10. P. R. Eiseman, *Comput. Methods Appl. Mech. Eng.* **64** (1/3), 321 (1987).
11. U. M. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations* (Prentice–Hall, Englewood Cliffs, NJ, 1988).