

## A Parallel Algorithm for Nonlinear Convection-Diffusion Equations

Jeffrey S. Scroggs\*

**Abstract.** A parallel algorithm for the efficient solution of nonlinear time dependent convection diffusion equations with small parameter on the diffusion term will be presented. The method is based on a physically motivated domain decomposition that is dictated by singular perturbation analysis. The analysis is used to determine regions where certain reduced equations may be solved in place of the full equation. Parallelism is evident at two levels. Domain decomposition and pipelining provides parallelism at the highest level, and within each domain there is ample opportunity to exploit parallelism. The method is suitable for the solution of problems arising in the simulation of fluid dynamics. Experimental results for a nonlinear equation in two-dimensions will be presented.

**1. INTRODUCTION.** In this paper, an algorithm that is appropriate for solving nonlinear convection-diffusion equations in multiple dimensions is presented and demonstrated. The method is appropriate for equations that exhibit strong nonlinearities, such as the equations that arise when modeling fluids. The techniques presented here are particularly well-suited in situations that require resolution of one or more of the small scales. Such situations arise, for example in hypersonic fluid dynamics and combustion problems where the chemistry depends on the viscous profiles. The method is an asymptotics-induced numerical method suitable for parallel processors which represent the state of the art in scientific computers, and is an extension of the algorithm presented in [1, 2] to two dimensions. The contents of this paper concentrate on a description of the algorithm and computational results. Asymptotic analysis gives qualitative and quantitative information as  $\epsilon \downarrow 0$ , where  $\epsilon$  is the coefficient of the diffusion term. The numerical method presented here exploits the analysis to determine an accurate solution for small positive  $\epsilon$ .

\* ICASE, NASA Langley Research Center, Hampton, Virginia. Supported by NASA Contract No. NAS1-18605.

The method incorporates an iterative technique to solve a related hyperbolic equation. The iteration was shown to converge and demonstrated in [3]. A linearized version of the original problem is solved in each step of the iteration. Computational experiments show that in just a few steps of the iteration, the solution to the nonlinear equation may be obtained.

The decomposition into domains is accomplished using a symbiosis of numerics and asymptotics. The asymptotic analysis identifies the regions where diffusion is negligible. The identification of subdomain boundaries is accomplished during the computation—no *a priori* knowledge of the shock location is assumed. In these regions, it is sufficient to solve a reduced equation. Solving this reduced equation can significantly reduce the work in the numerical method, and/or increase the potential for parallelism. The numerics provides a means of solution in the subdomains, and also a feedback mechanism. As a feedback mechanism, the numerical scheme can expose regions of unexpected behavior, confirming or correcting the asymptotics-induced subdomain boundaries. This decomposition permits the use of locally refined meshes, allowing the concentration of computational effort in the regions where it is needed most. Since the computational requirements are reduced, both the domain decomposition and the use of the reduced equation are preconditionings for this problem.

The problem is presented in Section 2. Asymptotic analysis specific to this problem is discussed in Section 3. The iteration and method for detection of the subdomain boundary is discussed in Section 4. The numerical schemes used in the method are presented in Section 5. The method is stated in algorithmic form in Section 6. In Section 7 computational results are presented. The particular numerical methods discussed herein are not new; however, their combination to form this method is.

**2. PROBLEM.** The example here will be taken from Computational Fluid Dynamics (CFD) in the transonic regime. The gasdynamic equations, including viscous effects, are used as a model in these settings. Except for very simple geometries and boundary conditions there is no analytic solution to these gasdynamic equations, and a numerical solution is difficult to obtain. For these reasons new algorithms are usually developed and tested on a more tractable canonical equation. The nonlinear parabolic equation,

$$(1) \quad P[u] := u_t + uu_x + \sigma u_y - \epsilon \Delta u = 0,$$

is such a canonical equation and will be the focus of this paper. This equation contains many of the properties that make the gasdynamic equations difficult to solve; namely, it is capable of modeling rapid variations such as shocks and boundary layers.

When the equation is nondimensionalized [4], the diffusion coefficient  $\epsilon$  is inversely proportional to the Reynolds number. Based on free-stream conditions in transonic flow, the Reynolds number for this problem is large. Thus,  $\epsilon$  is a small parameter in this setting. Asymptotic analysis exploits the smallness of the positive parameter  $\epsilon$  and involves study of the solution as  $\epsilon$  tends to zero ( $\epsilon \downarrow 0$ ).

The method will be described and demonstrated by solving (1) on the spatial

domain

$$(2) \quad \Pi := \{(x, y) | 0 \leq x \leq 1, 0 \leq y \leq 1\},$$

where the temporal variable is restricted to  $0 \leq t < T$ . Thus the entire computational domain is

$$(3) \quad D := \{(x, t) | 0 \leq x \leq 1, 0 \leq y \leq 1, 0 < t \leq T\}.$$

The solution satisfies

$$(4) \quad u(0, x, y) = \gamma(x, y),$$

for  $(x, y) \in \Pi$ , and the boundary condition

$$(5) \quad u(t, x, y) = \alpha(t, x, y),$$

for a suitable portion of  $\partial D$  (namely, the inflow portion). The solution to (1) is assumed to be uniquely defined.

This equation will be used as a model for shocks. When modeling shocks there are (at least) two sets of appropriate scales—the scales associated with the original variables  $(x, t)$ , and the scales appropriate in a small neighborhood of the shock. These are discussed in the next section.

**3. ASYMPTOTIC ANALYSIS.** Many problems of scientific interest have multiple scales. These problems are characterized by the presence of distinguishable physical mechanisms, each associated with a temporal or spatial gauge or scale. Efficient and accurate domain decomposition methods may be accomplished by identifying the various scales and exploiting their relationships.

The most easily tractable multiple-scale problems are those in which there are only a small number of widely separated groups of scales and the motion on the fastest scales has little influence on the smooth part of the solution. An identifying feature of this class is the presence of local regions in which the solution undergoes rapid variation. Such regions are called boundary or internal layers, when located in the neighborhood of a boundary or in the interior of the domain, respectively. These are the problems that are most natural for multitasking because it is easy to decompose the domain according to the regions of different local behavior. The method presented here is appropriate for this class of multiple-scale problems.

Asymptotic analysis provides analytic tools to identify and utilize the multiple scales. The relative importance of any two physical processes in a given domain may be measured by the ratio of the corresponding scales; thus, the various scales may be ranked by a set of dimensionless parameters, the ratios of scales. When the ratio of two scales is a large or a small number, then it often happens that one of the competing mechanisms is dominant in most of the domain. For example, in laminar duct flow with large Reynolds number the effects of viscosity may be ignored except in a neighborhood of the shock and boundary layers. The scales of the various competing processes (and, therefore, the relative magnitudes of the dimensionless parameters) usually change as the phenomenon evolves.

The scales for Equation (1) are for the convection terms  $uu_x$  and  $\sigma u_y$ , and for the diffusion term  $\epsilon \Delta u$ . Competition between convection and diffusion is crucial to the understanding of fluid flow, and the determining which of these is dominant can be made by examining the relationship between their scales. When modeling transonic flow, except in regions of rapid variation such as in shocks and boundary-layers, convection dominates diffusion. Asymptotic analysis is used to exploit these physical properties, providing the theoretical basis for a domain decomposition. The analysis identifies the following two types of subdomains: regions where the solution is smooth, where a reduced equation may be solved; and regions of rapid variations, such as in a neighborhood of a shock, where the full equation must be solved.

Since the behavior of  $u$  as  $\epsilon \downarrow 0$  is of interest, it is natural to first study the solution of the reduced equation

$$(6) \quad P_0[U] := U_t + UU_x + \sigma U_y = 0,$$

obtained by setting  $\epsilon = 0$  in Equation (1). Weak solutions  $U$  are sought for (6) with data (4-5). In order that  $U$  be uniquely defined, it is necessary to impose an entropy condition [5]. Suppose that  $U$  has a single shock. That is, suppose  $U$  is the solution to (6) subject to (4-5) that is discontinuous only along a curve  $(t, x, y) = (t, \Gamma_1(t), \Gamma_2(t))$ . For small  $\epsilon$ , this curve lies in the shock-layer region of the solution to the full problem. The size of this region tends to zero as  $\epsilon \downarrow 0$ . Analytic methods for choosing  $\Gamma = (\Gamma_1, \Gamma_2)$  are discussed by Whitham [6], Kevorkian and Cole [7], and others. Since  $\Gamma$  is not needed for the computations, methods for choosing  $\Gamma$  will not be discussed here.

The regions where  $U$  is a good approximation to  $u$  are defined by presenting functions which bound the difference  $U - u$ . These bounds are small except in an asymptotically small neighborhood of the shock. The bounds, based on Howes [8], are

$$(7) \quad |u - U| = O(\mu \exp[-d^2(x, t)/\epsilon^{1/2}]) + O(\epsilon)$$

when the derivatives of  $U$  are continuous across  $\Gamma$ , and

$$(8) \quad |u - U| = O(\mu \exp[-d^2/\epsilon^{1/2}]) + O(\epsilon^{1/2} \delta \exp[-d/\epsilon^{1/2}]) + O(\epsilon)$$

in the more general case when the derivatives of  $U$  are not continuous across  $\Gamma$ . Here  $d(t, x, y) =$  distance between  $(x, y)$  and  $(\Gamma_1, \Gamma_2)$  at time  $t$ , and  $\mu$  and  $\delta$  are upper bounds on the magnitude of the difference of the values of  $U$  and the magnitude of the difference of the normal derivative of  $U$  across  $\Gamma$ , respectively.

It is now reasonable to utilize the error bounds to make the definitions of the subdomains more precise. The internal layer is the following neighborhood of  $\Gamma$ :

$$(9) \quad D_{IL} = \{(x, y, t) | (x, y, t) \in D, [(x - \Gamma_1^{-1})^2 + (y - \Gamma_2^{-1})^2]^{1/2} \leq \Delta(t)\}.$$

Here  $\Delta(t) \leq K\eta(t)\epsilon^{1/4} \ln^{1/2} \epsilon$  is the width of the internal layer at time  $t$  ( $K$  is a constant independent of  $\epsilon$ ). The outer region is the complement of  $D_{IL}$  with respect to  $D$ , that is,

$$(10) \quad D_{OR} = \{(x, y, t) | (x, y, t) \in D, [(x - \Gamma_1^{-1})^2 + (y - \Gamma_2^{-1})^2]^{1/2} > \Delta(t)\}.$$

The upper bound on the size of the internal layer is based on the  $\exp[-d^2/\epsilon^{1/2}]$  term in the error bounds (7-8). The solution in the outer region is used to provide boundary data for the problem in the internal layer.

The bounds (7-8) motivate a preconditioning for the problem in  $D_{OR}$ . The bounds may be used to justify solving (6) in place of (1). In addition, relations (7-8) provide an error bound if diffusion (artificial or implicit in the numerical scheme) is incorporated into the solution process of either (6) or (1). Thus, the numerical method for  $D_{OR}$  may be chosen from the wide variety of methods designed for hyperbolic equations [9, 10, 11, 12].

Asymptotics identified two subdomains and provided preconditioners for the problems within the subdomains. The preconditioner for the full equation in  $D_{IL}$  is the use of the local scale  $\tilde{x} = x/\epsilon$ . This scale allows the diffusion to be modeled accurately, hence the grid is fine enough to resolve the shock. It is reasonable to use this scaling in the method, because computationally the internal-layer subdomain is of width  $O(\epsilon)$ . The preconditioning in the outer-region subdomain  $D_{OR}$  is to solve (6) in place of (1). Other asymptotic-induced preconditionings are possible. For examples of these, see [13, 14]. In the next section, the domain decomposition and preconditionings are combined with a functional iteration to form the computational method.

**4. DISCUSSION OF THE METHOD.** An iteration is formed by linearizing the reduced problem. Each step of the iteration requires the solution of (6) in the outer-region subdomain, and (1) in the neighborhood of a shock. Once the iteration has been described, the boundary detection scheme is presented. A method for handling the nonlinear behavior of solutions to Equation (6) is necessary to implement the domain decomposition. The method used here is an iterative method. This method provides feedback for the asymptotic-induced adaptive refinement. After the first iteration, a guess at the appropriate location of the refined regions can be made. The solution on the coarse mesh may be unreliable; thus, a refinement based on the coarse-mesh solution may result in errors in the location of the subdomain boundaries. The iteration coupled with the adaptive refinement would then allow the correction of the location of the refined region.

**4.1. Iteration.** In general, each step of the iteration requires the solution of a linear convection equation in the outer-region subdomain, followed by the solution of a nonlinear convection-diffusion equation in the internal layer. The convection equation

$$(11) \quad U_t^{k+1} + U^k U_x^{k+1} + \sigma U_y^{k+1} = 0$$

is formed by lagging the convection coefficient of (6).

Solutions to the reduced equation are poor approximations to the solution of the full equation in regions of large gradients, such as in the internal-layer subdomain. Thus, the full equation

$$(12) \quad U_t^{k+1} + U^{k+1} U_x^{k+1} + \sigma U_y^{k+1} - \epsilon \Delta U^{k+1} = 0$$

is solved in the internal layer at each iteration (for each  $k$ ). Dirichlet boundary data for the internal-layer subdomain is provided by the solution of (11) in the outer region. Other boundary conditions are possible [15].

The boundary of the internal-layer subdomain is allowed to change during the iterations. Denote the outer-region subdomain for iterate  $U^{k+1}$  by  $D_{OR}^k$ , and denote the complement of  $D_{OR}^k$  with respect to  $D$  by  $D_{IL}^k$ . That is,  $U^{k+1}$  is obtained by solving (11) in  $D_{OR}^k$ , then solving (12) in  $D_{IL}^k$ .

The temporal variable may need to be partitioned into several regions. For example, it is possible for the method to diverge if  $T$  is too large. In addition, the iteration requires that the solution be stored for the entire temporal region, possibly requiring too much memory. These problems are resolved by partitioning time into  $Q$  sections  $0 < T_0 < T_1 < \dots < T_Q = T$ . The iteration will be performed on the partition  $\Omega_q = (T_{q-1}, T_q] \times \Pi$  of the domain, using the solution at time  $t = T_{q-1}$  from the iteration on  $\Omega_{q-1}$  as the initial condition.

The theoretical basis for convergence of the iteration applied to this domain decomposition problem in one dimension is discussed in [2]. In addition, the iteration for systems of equations may be found in [3].

**4.2. Boundary Detection.** The location of the internal-layer subdomain is determined during the course of the iteration. This boundary detection scheme in a neighborhood of a shock is based on the size of the first partials of the solution with respect to the spatial variables. Both the physical and analytic motivation will be described here. The theoretical motivation of this technique is based on the error bounds derived for the one-dimensional case [2]. Implicit in this argument is that the computed solution be reliable.

Shocks form in regions of merging characteristics. (By merging, it is meant that the characteristics become asymptotically close). The boundary conditions imposed on the problem will be inflow conditions on both the  $x = 0$  and  $x = b$  boundaries; thus, the characteristics are traveling in the direction of increasing  $x$  from  $x = 0$ , and in the direction of decreasing  $x$  from  $x = b$ . These will merge somewhere inside  $D$ , forming a shock. The merging of the characteristics stabilizes the shock, and keeps it from dispersing. The solution to Equation (1) is constant along its characteristics; thus, where the characteristics merge, the solution will have large gradients. Based on this analysis, the subdomains used in the numerical method are

$$(13) \quad D_{OR}^k = \{(x, t) | (x, t) \in D, |u_x^k| + |u_y^k| \geq TOL\},$$

and,

$$(14) \quad D_{IL}^k = \{(x, t) | (x, t) \in D, |u_x^k| + |u_y^k| < TOL\}.$$

Heuristics, based on both accuracy and efficiency, are used to choose  $TOL$ . If  $TOL$  is too small, the internal-layer subdomain will be too small, and the data provided at the boundary of the internal-layer subdomain will have large perturbations as compared to the desired solution. Thus, accuracy will suffer when  $TOL$  is too small. If  $TOL$  is too large, the internal-layer subdomain will be too large, and the computational mesh will be refined in regions where the solution is smooth, creating excess work.

**5. COMPUTATIONAL DETAILS.** In this section the choice of the numerical schemes and some of the computer-science related issues are discussed. Both the numerical schemes and the choice of data structures allows the exploitation of parallelism.

The asymptotic analysis has provided a means to precondition the numerical problems. Because the sub-problems are well conditioned, the choice of numerical schemes may be made from a variety of standard methods. This is not usually the case. The class of problems for which the new algorithm is applicable are notoriously difficult to solve, and only a small number of schemes could be employed for their solution (prior to preconditioning). Since the sub-problems are well conditioned, numerical schemes used in the method presented here can be chosen based on criteria such as efficiency or the potential to exploit parallelism.

The solution in  $D_{OR}^k$  is obtained using a strictly upwind explicit finite difference scheme on a tensor-product grid (e.g. equally spaced in  $x$  and  $y$ ). Once a grid point has been identified as needing refinement by measuring the derivatives of the solution there, then each of the four grid rectangles adjacent to the grid point are included in the refined region surrounding the shock. These grid squares are refined based on the scaled coordinates; hence, the spacing within each of the coarse grid rectangles is  $\Delta x_{IL} = \epsilon \Delta x_{OR}$  and  $\Delta y_{IL} = \epsilon \Delta y_{OR}$  in the two spatial directions. Since the accuracy in solving in the internal-layer subdomain should be the same as in  $D_{IL}^k$ , the temporal variable is also stretched, so  $\Delta t_{IL} = \epsilon \Delta t_{OR}$ .

The sub-problem in the internal-layer subdomain requires the solution of a parabolic PDE subject to boundary data provided by the solution in the outer region. The computational domain in  $D_{IL}$  has an irregular boundary, but is composed of many non-overlapping rectangular regions, each of which is composed of a tensor product grid. The mesh for these rectangles has been scaled; therefore, there are no large gradients in the solution on the refined mesh; hence, the computations are not sensitive to the particular difference scheme used to solve the partial differential equation. An explicit finite difference method was chosen to solve the equation. The scheme used was a combination of explicit strictly upwind discretization for the convection terms with a centered discretization for the diffusion in  $D_{IL}^k$ . Other methods could be employed to obtain the solution in the subdomains [16, 17].

The data structures for  $D_{IL}^k$  are a rectangle adjacency list plus the rectangular regions with some buffer regions. By including a buffer region of data from adjacent refined rectangles, the number of times a synchronization must occur is reduced while increasing the amount of data communicated at each synchronization and increasing the amount of redundant computations. The domain changes between iterations. Using a list of refined rectangles allows for simple creation and deletion of sections of the refined domain. Bi-linear interpolation is used to initialize refined regions. Values computed on the refined mesh are injected into the coarse mesh.

**6. OUTLINE OF THE ALGORITHM.** The stopping criteria is based on the norm of the difference between iterates

$$(15) \quad \kappa_q^k = \|u^k - u^{k-1}\|_{T_q} = \sum_{x \in P} |u^k(T_q, s) - u^{k-1}(T_q, s)|,$$

where the sum is over the set  $P$  of the spatial values of the discretized solution at time  $t = T_q$ . When the norm is less than some user specified tolerance, then the iteration is assumed to have converged. The algorithm requires an initial guess which must satisfy the boundary conditions (4-5). For these experiments, the initial guess is simply taken as the solution of the beginning of the temporal region

$$u^0(t, x, y) = u(T_{q-1}, x, y)$$

for  $T_{q-1} < t < T_q$ . As a summary, the numerical method is outlined in the algorithm below.

- I. Initialize.
  - A. Set temporal partition counter to  $q = 1$ .
  - B. Apply initial data (5) to the solution.
- II. Determine the solution on temporal partition  $q$ .
  - A. Determine initial guess  $u^0$  for  $\Omega_q$ .
  - B. Initialize iteration counter  $k = 1$ .
  - C. Solve Equation (11) to obtain  $u^k$  in  $D_{OR}^k$ .
  - D. Determine  $\partial D_{IL}^k$  (see (13)).
    1. Where  $\partial D_{IL}^k$  has receded, remove refined rectangles.
    2. Where  $\partial D_{IL}^k$  has advanced,
      - add refined rectangles with initial data from bi-linear interpolation.
  - E. Solve Equation (12) to obtain  $u^k$  in  $D_{IL}^k$ .
  - F. Inject values obtained in  $D_{IL}^k$  into coarse mesh.
  - G. Compute the norm  $\kappa_q^k$  according to Equation (15).
    1. If  $\kappa^k$  is too large, increment  $k$  and go to Step C.
    2. Else move to next temporal partition.
      - a. Restart. Solution at  $t = T_q$  is initial condition for  $\Omega_{q+1}$ .
      - b. Increment  $q$  and goto Step II.

**Algorithm 1** Iteration with Restart.

**7. EXPERIMENTS.** The experiment demonstrating the method is to solve the 2-dimensional extension of Burgers Equation

$$u_t + uu_x + \sigma u_y - \epsilon \Delta u = 0,$$

on the unit square in space for  $0 < t \leq 0.25$ . The boundary and initial conditions,

$$(16) \quad u(t, x, y) = 2 - 4x - 2xy + 2y,$$

were chosen so that the solution is initially smooth, and so that a shock develops at some finite time that is not aligned with the grid.

The results of computations on an Ardent Titan are presented in Figures 1-5. The solution is computed using the adaptive refinement, and projected onto the coarse mesh for visualization. The coarse-mesh squares inside the refinement are elevated



and shaded in the refined region figures. (There is no refined region at time  $t = .10$ ). These figures show that the refined region changes shape as the solution is marched in time. The refined region also changes shape between iterations (however, this is not shown). Comparisons of the solution with and without the adaptive refinement demonstrate that both the shock location and the profile of the shock are modified by the refinement.

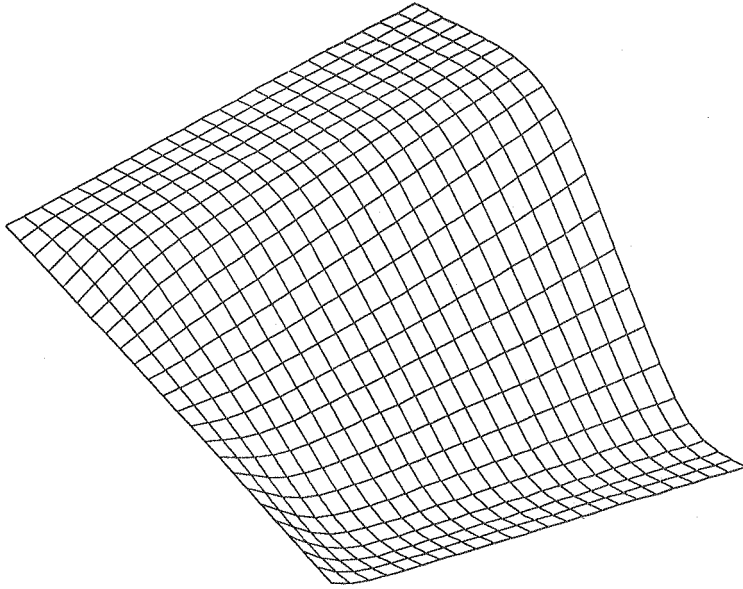


FIG. 1. Solution at  $t = .10$ .

This implementation is meant as a demonstration of the viability of the method; thus, the parallelism has not been exploited. Comparison of the computed solution to an analytic solution, and testing the efficiency of the method in a parallel computing environment are currently being researched.

**8. CONCLUDING REMARKS.** Parallelism may be exploited at several levels in the implementation of Algorithm 1. Domain decomposition provides large-grain parallelism. Medium-grain parallelism may be exploited by solving on each of the data structures in  $D_{IL}$  asynchronously. The domain decomposition is independent of the choice of numerical schemes for the subdomains; thus, schemes may be chosen which are a source of smaller-grain parallelism.

In this paper, asymptotics and numerics have been blended to form a new computational method suitable for a variety of difficult simulations arising in fluid dynamics and chemistry. The method has potential to exploit a large amount of parallelism and provides high accuracy. Asymptotic analysis provided a theoretical basis for the domain decomposition, identifying two types of subdomains: smooth outer regions, and an internal-layer subdomain with a shock.

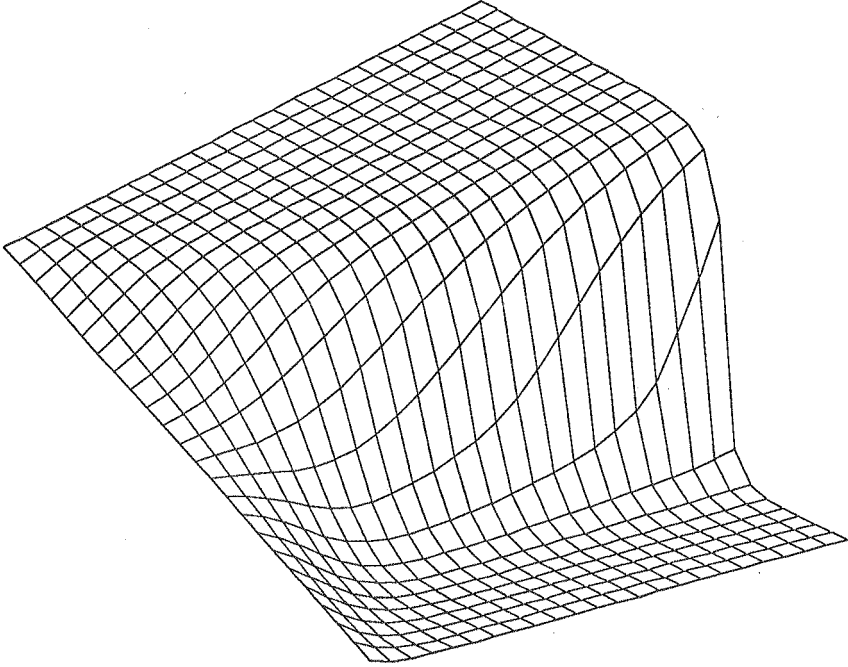


FIG. 2. *Solution at  $t = .20$ .*

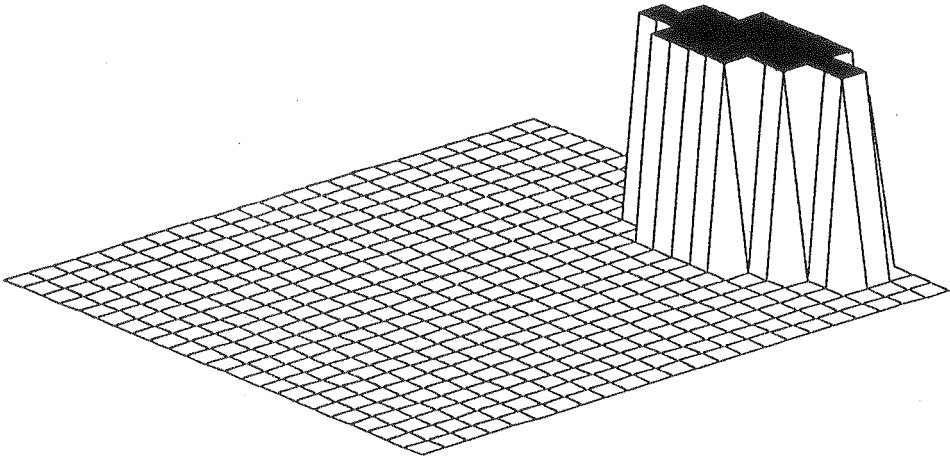


FIG. 3. *Refined Region at  $t = .20$ .*

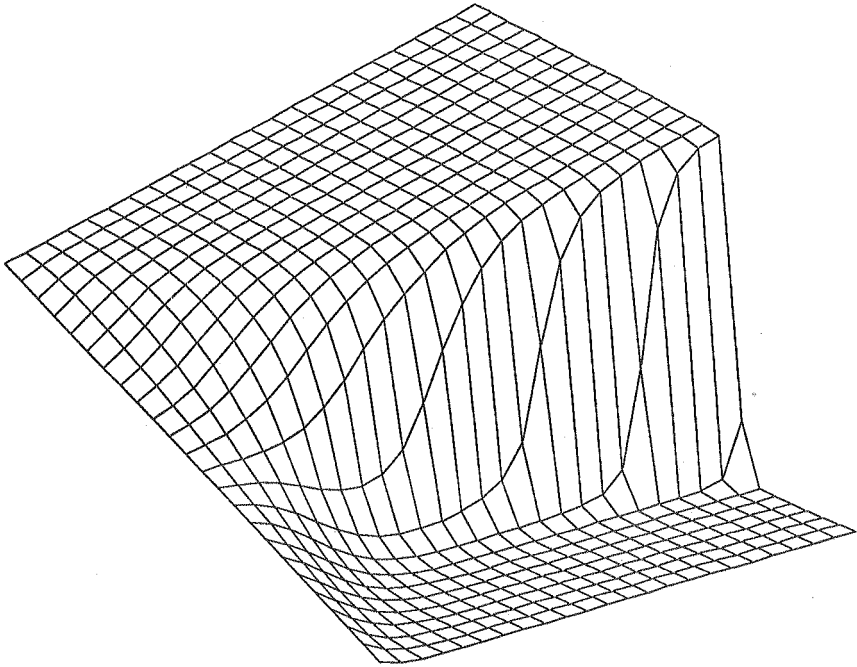


FIG. 4. *Solution at  $t = .25$ .*

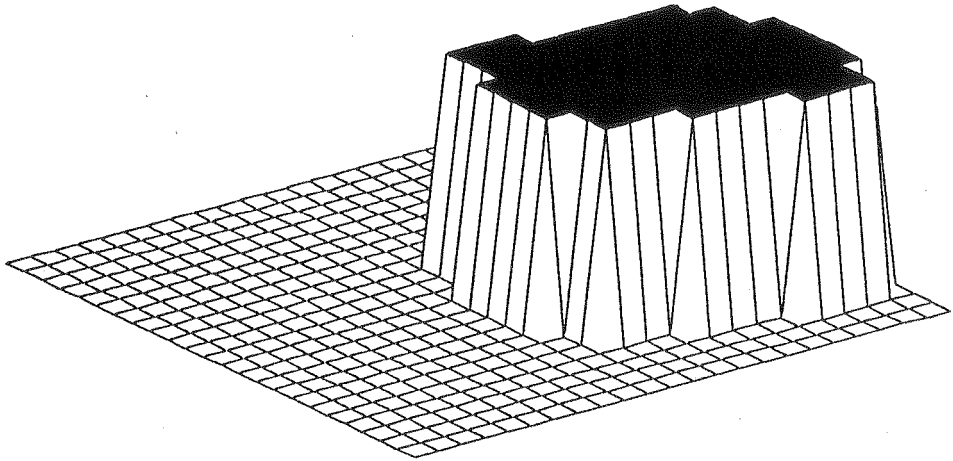


FIG. 5. *Refined Region at  $t = .25$ .*

## REFERENCES

- [1] J. S. SCROGGS AND D. C. SORENSEN, *An asymptotic induced numerical method for the convection-diffusion-reaction equation*, in Lecture Abstracts in Pure and Applied Mathematics, J. Diaz, ed., Marcel Decker, 1989.
- [2] J. S. SCROGGS, *A physically motivated domain decomposition for singularly perturbed equations*, ICASE Report 88-62, ICASE, NASA Langley Research Center, Hampton, Virginia 23665-5225, October 1989.
- [3] ———, *An iterative method for systems of nonlinear hyperbolic equations*, ICASE Report 89-62, ICASE, NASA Langley Research Center, Hampton, Virginia 23665-5225, 1989.
- [4] R. C. Y. CHIN, G. W. HEDSTROM, J. R. MCGRAW, AND F. A. HOWES, *Parallel computation of multiple-scale problems*, in New Computing Environments: Parallel, Vector, and Systolic, A. Wouk, ed., SIAM, Philadelphia, 1986, pp. 136-153.
- [5] P. D. LAX, *Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves*, no. 11, SIAM, Philadelphia, Penn 19103, 1973.
- [6] G. B. WHITHAM, *Linear and Nonlinear Waves*, John Wiley and Sons, New York, 1974.
- [7] J. KEVORKIAN AND J. D. COLE, *Perturbation Methods in Applied Mathematics*, Springer-Verlag, New York, 1981.
- [8] F. A. HOWES, *Multi-dimensional initial-boundary value problems with strong nonlinearities*, Arch. for Rat. Mech. Anal., 91 (1986), pp. 153-168.
- [9] A. HARTEN, P. D. LAX, AND B. VAN LEER, *On upstream differencing and Godunov-type schemes for hyperbolic conservation laws*, SIAM Review, 25 (1983), pp. 35-61.
- [10] B. VAN LEER, *On the relation between the upwind-differencing schemes of Godunov, Engquist-Osher and Roe*, SIAM J. Sci. Stat. Comp., 5 (1984), pp. 1-20.
- [11] H. C. YEE AND A. HARTEN, *Implicit TVD schemes for hyperbolic conservation laws in curvilinear coordinates*, AIAA Journal, 25 (1987), pp. 266-274.
- [12] R. D. RICHTMYER AND K. W. MORTON, *Difference Methods for Initial-Value Problems*, Interscience Publisher, John Wiley and Sons, New York, NY, 1967.
- [13] M. GARBEY, *Quasilinear hyperbolic-hyperbolic singular perturbation problem: Study of a shock layer*, Mathematical Methods in the Applied Sciences, 11 (1989), pp. 237-252.
- [14] G. W. HEDSTROM AND F. A. HOWES, *A domain decomposition method for a convection diffusion equation with turning points*, in Domain Decomposition Methods, T. F. Chan, R. Glowinski, J. Periaux, and O. B. Widlund, eds., Philadelphia, PA, 1989, SIAM, pp. 38-46.
- [15] FABIO GASTALDI AND ALFIO QUATERONI, *On the coupling of hyperbolic and parabolic systems: Analytic and numerical approach*, ICASE Report 88-55, ICASE, NASA Langley Research Center, Hampton, Virginia 23665-5225, October 1988.
- [16] D. B. GANNON, *Self adaptive methods for parabolic partial differential equations*, Tech. Rep. Ph.D. Thesis, University of Illinois, Urbana-Champaign, 1980.
- [17] D. HOFF AND JOEL SMOLLER, *Error bounds for finite-difference approximations for a class of nonlinear parabolic systems*, Math. Comp., 45 (1985), pp. 35-49.