JOURNAL OF COMPUTATIONAL PHYSICS 31, 313–334 (1979)

A Hybrid Vortex-ADI Solution for Flows of Low Viscosity*

ALEKSEI I. SHESTAKOV

Lawrence Livermore Laboratory, Livermore, California 94550

Received March 21, 1977; revised September 6, 1978

A hybrid numerical scheme is presented for use on two-dimensional, incompressible flows of low viscosity. The scheme divides the domain D into two regions, D_1 and D_2, and the Navier-Stokes equations are solved by different methods in each. In the region D_1, which is composed of a narrow strip adjoining the boundary, the grid-free vortex method is used. An alternating direction implicit (ADI) method is used in the interior domain D_2 which contains that part of D that lies a short distance away from the boundary. The hybrid method allows vorticity exchange between the two regions by partially merging and re-aligning them. Boundary conditions for the vorticity are automatically treated by the vortex method. Favorable numerical results have been obtained for the problem of re-circulating flow in a square cavity for Reynolds numbers of 400 and 1000.

I. INTRODUCTION

Much of the numerical work on the Navier–Stokes equations has been done by approximating the partial derivatives in the equations of motion by finite differences. Since the interesting phenomena occuring in fluids of low viscosity initially appear in regions of small area, it has been difficult to produce reliable results for problems with high Reynolds numbers.

The difficulty arises because the finite amount of space available in the computer restricts the number of grid points that can be used in the calculation. An effective upper bound is placed on R, the Reynolds number, since analysis implies that several grid points must be placed within the boundary layer whose thickness varies as $O(R^{-1/2})$. This problem is especially important in cases dealing with flows in wakes or separated flows when the initial boundary layer may not be visible to a coarse finite difference grid.

Another drawback of finite differences is that in areas near the boundaries sharp gradients give rise to large truncation errors which may swamp the original approximation. The problem may be further complicated by the truncation errors causing a numerical viscosity to form which is greater than the viscosity of interest [1]. It is often difficult to predict when difference methods go awry [2]. They can easily give

* Work performed under the auspices of the U.S. Department of Energy under Contract No. W-7405-Eng-48.

0021-9991/79/060313–22$02.00/0

incorrect solutions without signalling that further resolution is required. All of these drawbacks have led some researchers to construct other numerical schemes suitable for use in flows with high Reynolds numbers.

However, difference methods continue to enjoy a great popularity because of their usual reliability and the wide experience that people have with them. They can be successfully used in areas where greater resolution is needed by using a variable mesh. Another approach is to use coordinate scaling if it is expected that a greater variation will occur in one direction than in another. Some limitations of these remedies are that they are difficult to implement in areas other than those adjoining flat boundaries, and that one must have a priori knowledge where greater resolution is required.

A different approach is the grid-free vortex method proposed by Chorin [3]. The vortex scheme replaces the vorticity in the fluid and adjoining the boundary with vortex blobs of small but finite support. The equations of motion are solved by following the blobs throughout the fluid with a random motion added to model the viscous diffusion. The scheme has been successfully programmed both for exterior time dependent as well as interior steady-state flows [3, 4, 5]. However, experience has shown [5] that convergence is often slow. The increase in precision is obtained with the use of more vortex blobs which places a greater demand for storage. Since computation time is proportional to the square of the number of blobs, this gives rise to longer run times.

The following proposal attempts to avoid some of the limitations of the two methods. In areas adjoining the boundaries, the Navier–Stokes equations are solved by means of the vortex method. Away from the body, where viscous effects are small, the equations are solved by a well known alternate direction implicit (ADI) scheme [6]. The hybrid method is tested on the problem of computing the steady-state flow inside a square cavity when one of its sides slides in its own plane with constant unit velocity (see Fig. 1).

In Section II and its subsections, the hybrid scheme is developed. The subsections are devoted to describing the various independent components of the method: the vortex-method, the finite difference method and the problems associated with the interfacing of the two. Section III gives a short synopsis of past work on the square cavity problem. Section IV describes the numerical performance of the proposed method while Section V is a discussion of results.

II. Hybrid Numerical Scheme

The equations of interest are the non-dimensionalized Navier–Stokes equations for two-dimensional, incompressible flow, written in the vorticity transport form:

$$\partial_t \xi + \mathbf{u} \cdot \nabla \xi = \frac{1}{R} \Delta \xi, \tag{1}$$

$$\operatorname{div} \mathbf{u} = 0, \tag{2}$$

where

$$\mathbf{u} = (u, v) \equiv \text{velocity vector},$$
$$R \equiv \text{Reynolds number},$$
$$\xi = \operatorname{curl} \mathbf{u} \equiv \text{vorticity}.$$

Since $\mathbf{u}$ is two-dimensional, ξ is treated as a scalar.

Consider, in particular, the problem of computing the steady-state velocity field inside a unit-square cavity (Fig. 1). Initially the fluid is assumed to be at rest. At $t = 0$ the top edge is made to slide in its own plane inducing a constant, tangential unit velocity along that side.

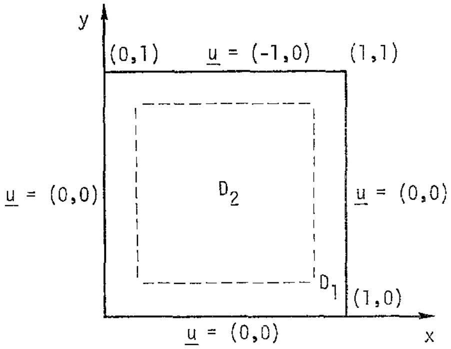


FIG. 1. Domain of interest and boundary conditions.

Let D denote the domain in which (1) and (2) are to be solved, and G denote the boundary of D. Express D as the union of two mutually disjoint subsets D_1 and D_2. Let D_2 represent that part of D which lies a distance δ away from G, and D_1 the remaining strip separating the boundary from D_2 (see Fig. 1).

If ξ^m represents the vorticity at the mth time step, express this as a sum of two functions,

$$\xi^m = \xi_1^m + \xi_2^m.$$

ξ_i^m represents the vorticity ξ^m restricted to the subset D_i, that is

$$\text{if} \quad (x, y) \in D_j, \quad \xi_i^m(x, y) = \begin{cases} 0 & \text{if } i \neq j \\ \xi^m(x, y), & \text{if } i = j. \end{cases} \tag{3}$$

The time advancement of ξ^m is done by two distinct methods. In the region D_1, the one adjacent to the boundary where sharp gradients occur, ξ^m $(= \xi_1^m)$ is represented by a collection of vortex blobs [3]. The field ξ_1^m is advanced by moving each vortex according to the inviscid equations, while the diffusion is accomplished in the mean by giving each a random displacement. In the interior region D_2, vorticity is represented by ξ_2^m, a function specified by its values on a computational grid. The equation of motion (1) is solved by finite difference approximations of the derivatives and using an ADI scheme to calculate the time advancement.

After the functions $\xi_1{}^m$ and $\xi_1{}^m$ are advanced forward, they may no longer satisfy (3). Some of the vortex blobs making up $\xi_1{}^m$ now lie in D_2 while the grid function $\xi_2{}^m$ has smeared over into D_1. At this stage a "clean up" is made. Blobs in D_2 are interpolated onto the ADI mesh. It will be shown that a judicious choice of the time step will allow vorticity originally in D_2 to travel no more than a specified distance which is chosen to be less than δ, the distance from D_2 to G. The vorticity which has been convected or has diffused from D_2 into D_1 is "remembered" on an "expanded" mesh. This vorticity is coagulated into new vortices; Eq. (3) is satisfied and the time step is completed.

A. *Vortex method*

This section briefly lists the pertinent equations of the vortex scheme [3].

Express $\xi_1{}^m$ as a collection of vortex blobs,

$$\xi_1{}^m(\mathbf{r}) = \sum_j \zeta_j \xi_0(\mathbf{r} - \mathbf{r}_j{}^m).$$

where

$$\xi_0(\mathbf{r}) = \begin{cases} (2\pi\sigma r)^{-1}, & r < 0 \\ 0, & r \geqslant \sigma, \end{cases}$$

$r^2 = |\mathbf{r}|^2$, and $\mathbf{r}_j{}^m = (x_j{}^m, y_j{}^m)$ gives the location of the jth blob at the mth time step. The symbol σ denotes the "cut-off length" [3].

The function $\xi_1{}^m$ generates a stream function $\psi_1{}^m$ and a velocity $\mathbf{u}_1{}^m$:

$$\psi_1{}^m(\mathbf{r}) = \sum \zeta_j \psi_0(\mathbf{r} - \mathbf{r}_j{}^m); \tag{4}$$

$$\mathbf{u}_1{}^m(x, y) = \sum_j \zeta_j \mathbf{u}_0(x - x_j{}^m, y - y_j{}^m), \tag{5}$$

where

$$\mathbf{u}_0(x, y) = (\partial_y \psi_0, \partial_x \psi_0) = \begin{cases} (-y, x)/2\pi r^2, & r \geqslant \sigma \\ (-y, x)/2\pi\sigma r, & r < \sigma. \end{cases}$$

ψ_0 is made continuous by the addition of a constant. Since $\mathbf{u}_0$ does not satisfy any boundary conditions neither does $\mathbf{u}_1$.

Before advancing $\xi_1{}^m$, the function $\mathbf{u}^m$ must be evaluated, where $\mathbf{u}^m$ denotes the velocity due to all the vorticity in D. The velocity induced by the interior vorticity, $\xi_2{}^m$, is denoted by $\mathbf{u}_2{}^m$ and satisfies

$$\operatorname{curl} \mathbf{u}_2{}^m = \xi_2{}^m \qquad \text{in } D,$$

$$\mathbf{u}_2{}^m \cdot \mathbf{n} = -\mathbf{u}_1{}^m \cdot \mathbf{n} \qquad \text{at } G,$$

when $\mathbf{n}$ is the unit normal vector along G. In the present context, $\mathbf{u}_2{}^m$ is constructed with the aid of another stream function $\psi_2{}^m$. One solves the potential equation:

$$\Delta\psi_2{}^m = -\xi_2{}^m \qquad \text{in } D \tag{6}$$

with the boundary values,

$$\psi_2{}^m = -\psi_1{}^m \qquad \text{along } G,$$

where $\psi_1{}^m$ is given by (4). Then

$$\mathbf{u}_2{}^m = (\partial_y\psi_2{}^m, -\partial_x\psi_2{}^m).$$

Along and near the boundary the source term in (6) is zero. This avoids the usual difficulties that finite difference methods have in resolving sharp gradients of the vorticity for large R.

In the computer program, (6) is solved by finite differences with the aid of a fast Poisson solver [7]. The computed grid values are then approximated by a bi-quadratic spline function similar to that given by Buneman [8]. The spline is defined by spline coefficients which are stored over the old function values. These coefficients are computed by solving a linear system which arises when the spline is made to return the original grid values of ψ_2 when the spline is evaluated at a grid point. Additional requirements require knowledge of the normal derivatives at the boundary. Approximations to these derivatives are obtained by discretizing (6) at the boundary and using a "mirror image point". This approximation is correct to first order in the mesh size. An alternative would be to use backward differences. The advantage of the spline is that its derivatives becomes continuous functions which approximate $\mathbf{u}_2$ and can be evaluated anywhere inside D_1.

The definition

$$\mathbf{u}^m = \mathbf{u}_1{}^m + \mathbf{u}_2{}^m$$

generates a velocity which has no normal component at G.

The function $\xi_1{}^m$ is advanced as in [3]. If k is the time step and $(x_i{}^m, y_i{}^m)$ the position of the ith blob, then

$$(x_i^{m+1}, y_i^{m+1}) = (x_i{}^m, y_i{}^m) + k\mathbf{u}^m(x_i{}^m, y_i{}^m) + \boldsymbol{\eta} \tag{7}$$

Equation (5) is used to evaluate $\mathbf{u}_1{}^m$, with the proviso that there be defined a small $\rho < \sigma$ such that

$$\mathbf{u}_0(r) = 0, \qquad \text{for} \quad r \leqslant \rho.$$

This ρ is necessary in order to avoid a "self force" of the blobs. The random variable $\boldsymbol{\eta} = (\eta_1, \eta_2)$ has a Gaussian distribution with mean zero and variance $2k/R$, and solves the diffusion equation: $\xi_t = R^{-1}\Delta\xi$ in the mean.

The tangential boundary component is cancelled as in [3]. The vorticity at the boundary is approximated by the formula $\zeta = -\partial_y u$ and this amount is generated in a thin boundary layer along G to perform the cancellation. The length of G is broken up into segments of length h; this thin layer is combined into new blobs, and is allowed to diffuse by means of a random displacement. This process also defines the "cut-off" length $\sigma = h/2\pi$. The completion of the time advancement of $\xi_1{}^m$ creates a new field $\xi_1^{m+1,*}$ which has spread beyond the region D_1. It is surprising that although the above approximation is inaccurate in the presence of separation or in corner regions, the vortex method can describe those flow configurations (see [3] and below).

B. *Finite Difference Scheme*

In the interior region, and ADI scheme is used to advance $\xi_2{}^m$. Let $\zeta_{i,j}^m$ represent the grid values of $\xi_2{}^m$. The ADI scheme alternatively solves

$$2(\zeta^{m+1/2} - \zeta^m) = k[-u^m D_x \zeta^{m+1/2} - v^m D_y \zeta^m + R^{-1}(D_{xx}\zeta^{m+1/2} + D_{yy}\zeta^m)], \quad \text{and} \tag{8a}$$

$$2(\zeta^{m+1} - \zeta^{m+1/2}) = k[-u^m D_x \zeta^{m+1/2} - v^m D_y \zeta^{m+1} + R^{-1}(D_{xx}\zeta^{m+1/2} + D_{yy}\zeta^{m+1})]. \tag{8b}$$

Unnecessary subscripts have been suppressed; $D_x\zeta$, $D_{xx}\zeta$ etc. represent the usual divided differences of ζ.

Before (8) is implemented the velocity $\mathbf{u}^m$ must be tabulated at each grid point where (8) is used. The function $\mathbf{u}_2{}^m$ is calculated by taking divided differences of the solution of (6). The contribution due to vortex blobs, $\mathbf{u}_1{}^m$, is not computed by means of (5), since this would involve a large amount of computer time. Instead one solves

$$\Delta\psi_1 = 0 \quad \text{in} \quad D_2 \tag{9}$$

with boundary values on D_2 computed with (5). The source term here is null because of (3). Equation (9) can be solved by any standard method such as cyclic reduction, if D_2 is rectangular. Divived differences of the computed values later yield $\mathbf{u}_1{}^m$ at the grid points.

C. *Boundary Conditions for the Difference Scheme*

The ADI scheme (8) is used in its unaltered form at points making up the boundary of D_2. The implicit nature of the ADI method necessitates specifying values for $\zeta_1^{m+1/2}$, ζ_2^{m+1} at grid points lying outside D_2. These values are obtained by allowing the computational mesh covering D_2 to "expand" one mesh width each half time step.

Equation (8a) is solved successively by rows. Because only derivatives in the x-direction are discretized implicitly, each row of the mesh generates a trigiagonal system of linear equations. Grid values of $\xi_2^{m+1/2}$ lying two or more mesh widths away from

D_2 are set to zero. This assures that the number of unknowns matches the number of equations and restricts vorticity in D_2 from traveling more than one mesh width in a time duration of $k/2$.

Vorticity from D_2 can travel out in only two ways, by convection or by diffusion. It can be assured that ξ_2 is not convected beyond one mesh width when k satisfies a CFL type limitation,

$$\| \mathbf{u} \|_{\max} \frac{k}{2} \leqslant d. \tag{10}$$

Diffusion is equivalent to random motion. According to Chebyshev's theorem [9], realizations of gaussianly distributed random variables will rarely be greater in magnitude than three standard deviations. To be more precise [10, 11], the probability that a random displacement is greater than λ standard deviations is $\exp(-\lambda^2/2)$. This gives a value of 0.011 when $\lambda = 3$, and 1.5×10^{-8} when $\lambda = 6$. During a time step of duration k, vorticity will diffuse with a standard deviation of $(2k/R)^{1/2}$. Hence, in a time step of length $k/2$, ξ_2 will rarely travel beyond one mesh width if

$$3(k/R)^{1/2} \leqslant d. \tag{11}$$

The solution of (8a) is the function $\xi_2^{m+1/2}$ which has spread one mesh width beyond D_2. Equation (8b) is solved similarly. The resulting tridiagonal systems are solved successively by columns. The function ξ_2^{m+1} is set to zero for grid points lying more than two mesh widths away. This is justified by the same reasoning as above. The result of these two half steps generates a function $\xi_2^{m+1,*}$ which has spread over two mesh widths into the region D_1.

D. *Domain Interaction*

1. ξ_1 flow from D_1 into D_2. The time advancement step (7) scatters the vorticity ξ_1 throughout D. As in [3], blobs displaced outside D are discarced; however, vorticity carried by blobs that have entered into D_2 must be interpolated onto the finite difference mesh in order to satisfy (3). The replacement of vortex blobs by grid values is done by linear interpolation. Two considerations arise. First, no vorticity should be created or destroyed. Secondly, the process of interpolation should not disturb the fluid, that is, there should only be a small difference between the velocities induced by the blob before and after the interpolation.

In the numerical example, the following scheme was used. Assume that the vortex to be interpolated,

$$\xi_1^{m+1}(\mathbf{r}) = \zeta_1 \xi_0(\mathbf{r} - \mathbf{r}_1^{m+1}), \qquad \text{where} \quad \iint_D \xi_1^{m+1}(\mathbf{r})\, d\mathbf{r} = \zeta_1,$$

lies inside one mesh square (see Fig. 2). Horizontal and vertical lines are drawn through the vortex center, $\mathbf{r}^{m+1}$, and the ratios of areas of the resulting rectangles to the area of the mesh square are computed,

$$A_1 = (d - x)(d - y)/d^2, \qquad A_2 = x(d - y)/d^2, \qquad \text{etc.}$$

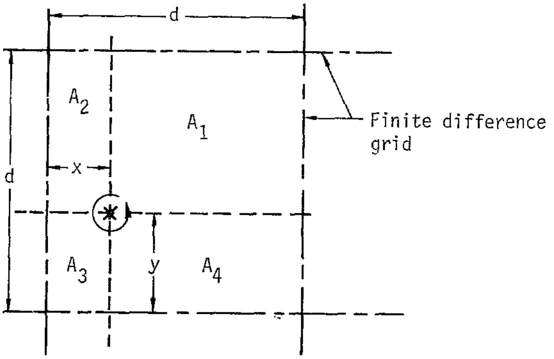


FIG. 2. Area weighing to interpolate vortex blob onto finite difference grid.

The A_j's are used to compute the new grid vorticities,

$$\zeta_{i,j}^{m+1} = \zeta_1 A_1/d^2, \qquad \zeta_{i+1,j} = \zeta_1 A_2/d^2, \qquad \text{etc.}$$

Using the trapezoidal rule for integration, it can be easily seen that no vorticity was created or destroyed by the interpolation.

The above interpolating scheme is equivalent to assuming that the vortex blobs have a square base and that the basic blob stream function has the form

$$\xi_0(x, y) = \begin{cases} \dfrac{1}{d^4}(d - |x|)(d - |y|), & |x|, |y| < d \\ 0, & \text{otherwise.} \end{cases}$$

To perform the interpolation with this blob, one simply evaluates the blob function at the grid points.

It is interesting to note that the above method of interpolating is equivalent to Peskin's delta function technique [12]. However, the support of the discrete delta function used above is 4 mesh squares instead of 16 as proposed by Peskin.

Before using the above linear delta function some experimentation was done with other candidates. One such candidate was Peskin's trigonometric delta function:

$$\delta_{ij}(x, y) = \alpha g(x - x_j)\, g(y - y_i)$$

$$g(x) = \begin{cases} 1 + \cos\dfrac{\pi x}{d}, & |x| < d \\ 0, & \text{otherwise,} \end{cases}$$

$$\alpha = \text{constant to insure that} \int \delta(x, y) = 1.$$

This particular interpolating scheme was tested against the scheme described above. The test was done by interpolating vortex blobs at several random places in D and checking which version produced a velocity field that compared better with the field induced by the vortex blob. The linear scheme compared more favorably and was therefore chosen as the candidate for the computer program.

The subject of interpolating vorticity, or equivalently, charges, onto a computational mesh is an interesting topic in itself. Other choices may be easily devised that use one mesh point (nearest grid point method) or three, or more. The interested reader is advised to consult Eastwood and Hockney [13] for details.

2. ξ_2 flow from D_2 into D_1. The completion of the ADI step results in an "expanded" mesh with vorticity ξ_2 deposited on mesh points outside D_2. This vorticity must then be rearranged into the form of vortex blobs to satisfy (3). A simple method, albeit expensive, is to replace every nonzero vorticity value of ξ_2 outside D_2 with a blob centered on a mesh point. This new blob would have the strength $\zeta_e = d^2\zeta_{i,j}$. Unfortunately, this procedure generates too many new vortex blobs per time step.

In order to minimize the number of newly reconstructed vortex blobs, the following simple scheme was devised.

Assume that only the four neighboring values $\zeta_{i,j}$, $\zeta_{i+1,j}$, $\zeta_{i+1,j+1}$ and $\zeta_{i,j+1}$ need need to be reconverted into a vortex blob. The total volume of vorticity that needs to be transferred is $\zeta_e = (\zeta_{i,j} + \zeta_{i+1,j} + \zeta_{i+1,j+1} + \zeta_{i,j+1})\, d^2$. These four grid values are combined into one blob centered at the center of mass of the four vorticity values.

Define $|\zeta_e| \equiv |\zeta_{i,j}| + |\zeta_{i+1,j}| + |\zeta_{i+1,j+1}| + |\zeta_{i,j+1}|$. Then the center of the blob (x_e, y_e) is

$$\begin{aligned} x_e &= x_i + d(|\zeta_{i+1,j+1}| + |\zeta_{i+1,j}|)/|\zeta_e|, \\ y_e &= y_j + d(|\zeta_{i,j+1}| + |\zeta_{i+1,j+1}|)/|\zeta_e|, \end{aligned} \tag{12}$$

where (x_i, y_j) are the coordinates of the point whose grid vorticity value is $\zeta_{i,j}$. This scheme places the center of the new blob inside the mesh square.

It should be noted that if the four grid values $\zeta_{i,j}$, $\zeta_{i+1,j}$, etc. were originally all of the same sign, (12) is the inverse of the linear interpolation scheme considered in the previous section.

Many variations of the above are possible. In the numerical problem, the vorticity ξ_2 which was to be reconverted into vortex blobs was deposited on two concentric square rings enclosing the region D_2. The computer program replaced these values with a collection of blobs lying at least one and less than two mesh widths away from D_2. For example, consider the four grid values: $\zeta_{i,j}$, $\zeta_{i+1,j}$, $\zeta_{i+1,j+1}$, $\zeta_{i,j+1}$; the above scheme was used to reconvert the vorticity, $\frac{1}{2}\zeta_{i,j}$, $\frac{1}{2}\zeta_{i+1,j}$, etc. into one blob lying inside a mesh square. The remaining $\frac{1}{2}\zeta_{i,j}$ vorticity is used in a neighboring cell. In the corners, a variation is used to insure that all the vorticity on the expanded mesh is reassigned into blob form.

III. Circulating Flow in a Square Cavity

Although the hybrid method is designed for use on time dependent problems, it was tested in an asymptotic time sense to compute a steady-state flow. The problem of interest is to determine the velocity field inside a square cavity when one of the sides slides in its own plane with constant unit velocity. The resulting flow has many interesting features which make it suitable for testing different numerical schemes. There are also some experimental results [14] with which the numerical work can be compared.

The essential features of the flow are well documented in the literature (Refs. [15–21]). The cited numerical work proceeds somewhat differently than here as no time dependence is assumed and the steady-state flow is computed by iteration. The earliest work by Kawaguti [15] produced good results for small R ($R \leqslant 64$), but signalled the difficulty in obtaining large R solutions by failing to converge for $R = 128$. By changing the iteration procedure, Burggraf was able to obtain solutions for R as high as 400. It was then discovered that convergence for larger R could be obtained by using variations of upwind differencing [17–20].

The use of upwind differences reduces the accuracy of the difference scheme to first order, but makes the linear algebraic equations easier to solve because the diagonal terms are increased in magnitude. Greenspan constructed an iterative scheme which converged for R as high as 10^5 [17]. However, he was primarily interested in obtaining a convergent numerical method, and did not display any numerical results other than contour plots of vorticity and stream function. Greenspan's results for a mesh size of 1/20 do not show any counterrotating vortices and his use of the finer mesh size of 1/40 is felt to be too crude to adequately resolve the thin boundary layer at the high Reynolds numbers.

Other users of upwind differences were Runchal et al. [18] and Bozeman and Dalton [19]. Runchal's group obtained convergent solutions for R as high as 10^4 using a nonuniform grid. However, this work also only shows contour plots and provides no numerical values. Bozeman and Dalton study both types of schemes, central differences as well as one-sided, and they use either the convective or the divergence form of the equations for the non-linear terms. Their results show that at larger R ($= 1000$), central differences of the non-linear equations produce linear algebraic equations which fail to converge. Problems occur even for one-sided differences. For $R = 1000$, discretizing the convective form of the non-linear terms produced an obviously erroneous result, instead of one central vortex, there appeared two large vortices of opposite rotation, one lying above the other.

A recent numerical study of the cavity flow problem was undertaken by Nallasamy and Prasad [20]. Their numerical technique [21] also used one sided differences, but allowed them to obtain results for R as high as 50,000. They were able to get good resolution for mesh sizes as small as 1/60, and could observe the variation of the secondary vortices with Reynolds number.

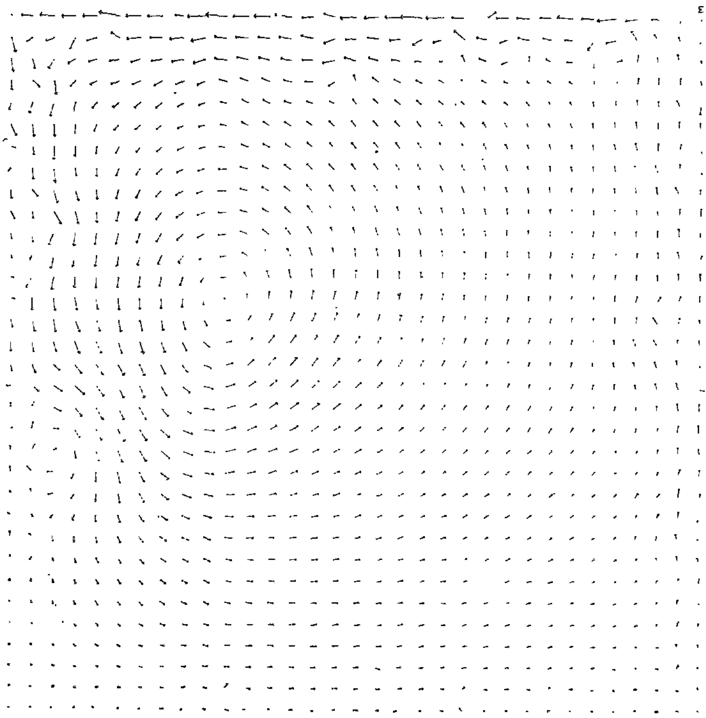
An alternative to finite differences was presented by Rubin and Khosla [22]. Their time-dependent approach is based on polynomial spline interpolation. The benefits of these methods is their higher order, hence they can produce good results with very few mesh points.

IV. Numerical Results

The hybrid method was tested on the square cavity problem. Initially the fluid was at rest and the time dependent equations were used to obtain the steady-state solution. Numerical results are displayed for the following sets of parameters:

R	Mesh size		Time step	Dist(G, D_2)
	Interior	Boundary		
400	1/32	1/20	0.05	3/32
1000	1/32	1/30	0.07	3/32
	1/64	1/30	0.1	4/64

The pictorial output is of the form of Figs. 3–8. The boundaries of the cavity coin-



Figs. 3–8. Velocity plots. Direction of arrows varies as direction of u. Length of arrows is proportional to $| u |^{1/2}$. $R = 1000$, $k = 0.1$, $d = 1/64$, time step 53. Example of creation of negative vorticity by primary vortex.

cide with the outside arrows. Each arrow represents the magnitude and direction of the velocity at the tail.

Most of the computation was done for the case $R = 1000$ and therefore those results will be described first. There is little difference between results using an interior mesh size $d = 1/32$, or $d = 1/64$. Figures 3–5 are from runs with $d = 1/64$, while Figs. 6–7 are with $d = 1/32$.

In both cases, as the run begins, strong vortices of positive sign are shed from the sliding edge to induce a tangential velocity equal to 1. After several steps these blobs are interpolated onto the edge of D_2 where they begin to form a large vortex. This central vortex moves toward the cavity center and spirals into its equilibrium position.

As the large vortex spirals in, it generates a tangential velocity component, $\mathbf{u} \cdot \mathbf{s}$, which is positive along the four walls. To compensate for this, blobs of negative [redacted] stationary corner $(x, y) = (0, 0)$.

This flow of negative vorticity is responsible for the peculiar loop that the stream lines make near the upstream corner of the sliging edge. (Figs. 5, 7). A strong blob of positive sign is generated at the sliding edge to satisfy the condition $\mathbf{u} \cdot \mathbf{s} = 1$

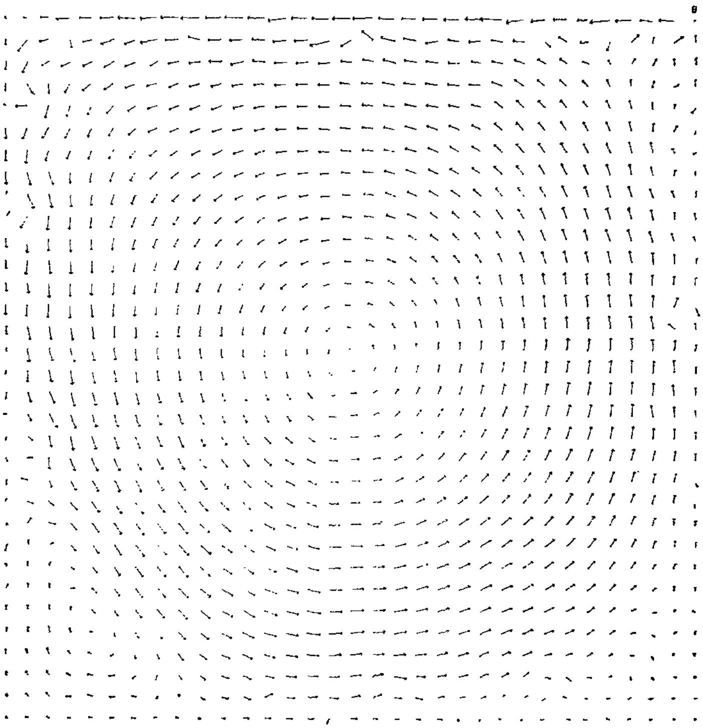


FIG. 4. $R = 1000$, $k = 0.1$, $d = 1/64$, time step 238.

TABLE I

Comparison for Runs with Different Time Steps and Different Mesh Widths

R	d	h	k	Number of vortices	Time/iteration (seconds)
400	1/32	1/20	0.1	930	8.5
			0.08	1060	10.5
			0.05	1440	16.5
			0.04	1650	20
1000	1/32	1/30	0.07	1800	25
		1/20	0.07	1500	18
		1/20	0.1	1220	12.5
		1/20	0.2	750	7
	1/64	1/30	0.1	1550	25

d = interior mesh width

h = boundary discretization length

k = time interval

Number of vortices = approximate stable number of vortices in D_1

Time/iteration = running time to complete 1 time step

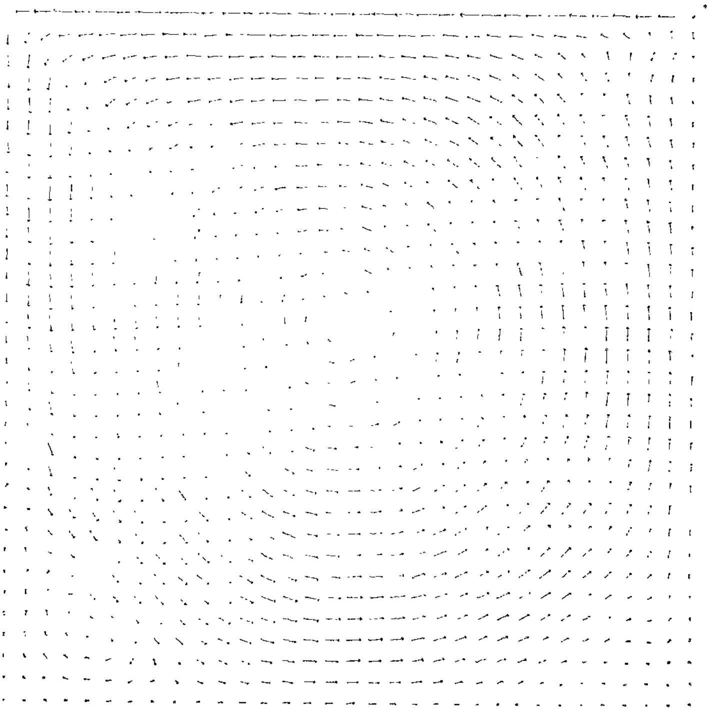


FIG. 5. $R = 1000$, $k = 0.1$, $d = 1/64$. Average over time steps 201–260.

and counteract the influence of the accumulated negative vorticity. As this blob diffuses into the fluid it generates the counterclockwise rotation in the corner.

After several time steps the number of blobs present inside the domain D_1 stabilizes. A balance is reached between the number of blobs generated each time step and the number lost, either by interpolation onto D_2 or lost through the boundaries. This number has almost a linear dependence on the time step k (see Table I). Decreasing k, increases the number of vortices. Apparently each vortex has a finite life span in D_1 ; decreasing the time step allows it to exist for more time steps. Since the number of vortices that are generated per step remains the same, more will exist. The computer program alters this by allowing vortices to merge when they overlap. At the time that the blobs need to be moved, it is necessary to calculate their mutual separation to calculate $\mathbf{u}_1$. If that distance is less than a predetermined amount. the two vortices are merged and treated as one. The value used by the program is $\sigma/10$, where σ is the "cut-off length" (see Section II.A).

Computation was halted after a visual inspection of the velocity plots showed little change from one time step to another. For the case $R = 1000$, $d = 1/64$, 260 time steps were run with a time step $k = 0.1$ The boundary discretization length, h, was set at 1/30. At the time computation was halted, the center of the large vortex

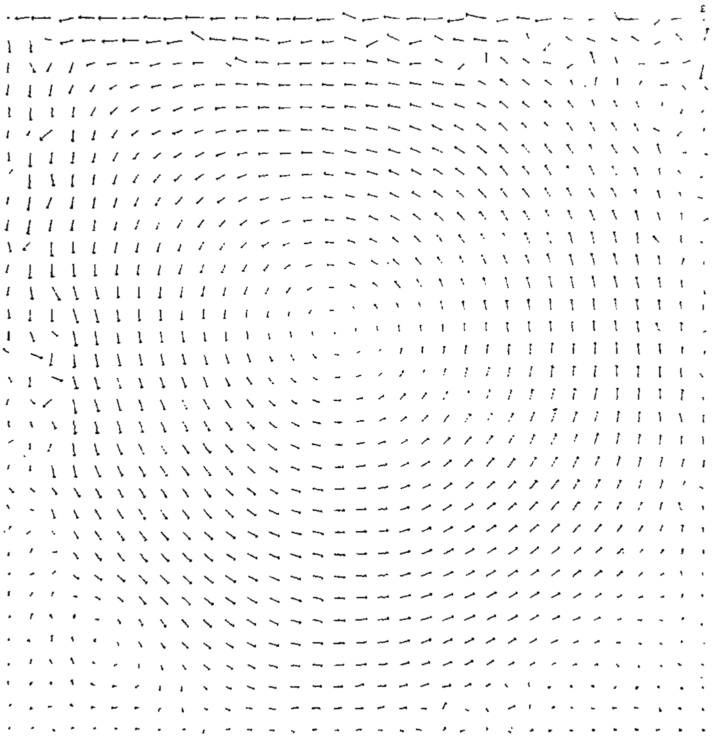


FIG. 6. $R = 1000$, $k = 0.07$, $d = 1/32$, time step 433.

moved in a decreasing spiral about an equilibrium point. The diameter of the last circular spiral was approximately 0.05, or $3d$. No variation of parameters was done for this case. Some experimentation with different values for h and k was done for the case $R = 1000$, $d = 1/32$ (see Table I). The output shown in Figs. 6, 7 used values of $k = 0.07$, $h = 1/30$. It should be noted that changing h, or k between time steps does not require any modification of the input vorticity fields. The new run uses a different value for σ, but the circulation of each vortex remains the same.

Figures 4 and 6 are velocity plots after one time step for two different runs. Although these plots are representative, there is a variation of the field from one time step to the next. This variation occurs in the interior region, D_2 as well as in the exterior region. In D_2, the center of the large vortex moves in a decreasing spiral about its equilibrium point. In the exterior region, the vortex method produces observable differences in the velocity fields corresponding to different time steps. These differences are caused by the random component, present because of the manner in which the vortex blobs are diffused. The differences in the velocity fields are especially evident in the stationary corners. At times the counter-rotating vortices disappear only to be regenerated at later time steps. When these corner vortices disappear, the numerical method generates the necessary negative vorticity at the boundary and transports

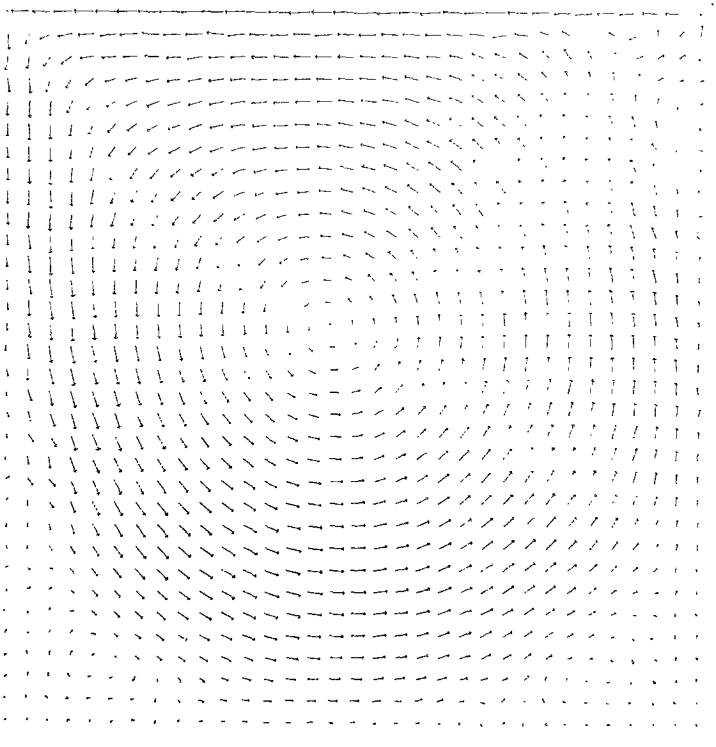


FIG. 7. $R = 1000$, $k = 0.07$, $d = 1/32$. Average over time steps 431–460.

it to the corners to reestablish the corner vortices. This disappearance and regeneration is more frequent for coarser mesh sizes, h and d.

For a steady-state problem of the type presented here it is possible to obtain details of the flow by averaging over several time steps. Figures 5, 7, and 8 represent the velocity fields after averaging. As is evident, most of the noise of previous plots is suppressed and the finer details of the flow become apparent. The velocity profiles shown in Figs. 9 and 10, and the calculation of the extent of the corner vortex were done using the averaged velocity fields plotted in Figs. 5, 7 and 8.

It is interesting to contrast the performance of the hybrid scheme to the use of only the voıtex method in the square cavity problem [5]. In the author's earlier work, the velocity fields at any one time step were unrecognizable and only after averaging over many time steps could any characteristic features be observed. However, even averaging over 320 time steps gave unsatisfactory results. The velocity profiles were not smooth and the stream lines had irregularities.

Besides the case $R = 1000$, the program was run for $R = 400$. The interior mesh size was kept constant at $d = 1/32$ and some experimentation was done with k and h (see Table I). The results depicted in Figs. 8, 9, and 10 use the average of 40 time steps. As in the case for $R = 1000$, varying k affected the total number of vortex blobs present in D_1, which also affected the duration of each time step (Table I).

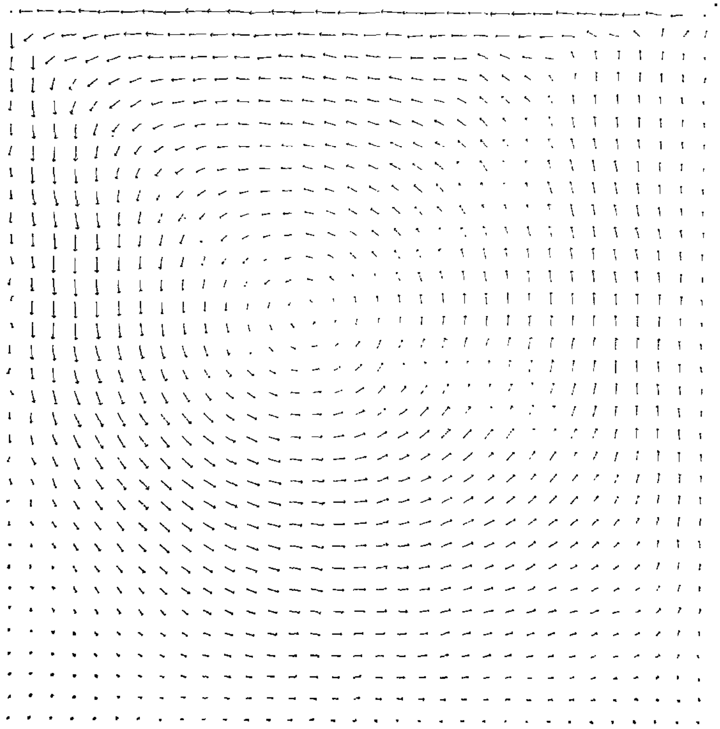


FIG. 8. $R = 400$, $k = 0.05$, $d = 1/32$. Average over time steps 331–370.

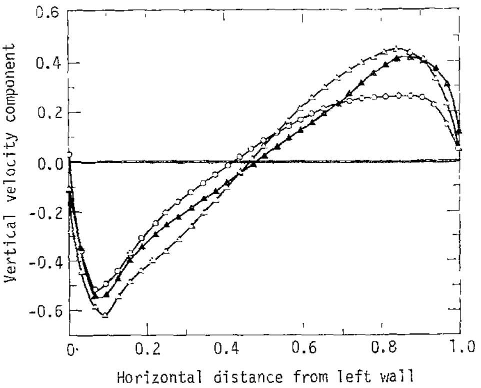


FIG. 9. Velocity profile of v-component along a horizontal line through the vortex center. ○, $R = 400$; △, $R = 1000$, $d = 1/32$; ▲, $R = 1000$, $d = 1/64$.

Figures 10, 11 and 12 compare the results of this study with previous work. Figure 10 compares the u-velocity profiles with results of Burggraf [16], and Bozeman and

[redacted]

present results compare favorably with Burggraf's for $R = 400$, but disagree with those of BD at $R = 1000$. No explanation can be given for the latter discrepancy. The profile of BD has a smaller slope and turns back to zero earlier as it approaches the lower stationary wall. In the limit of inviscid flow in the center, the slope of the pro-

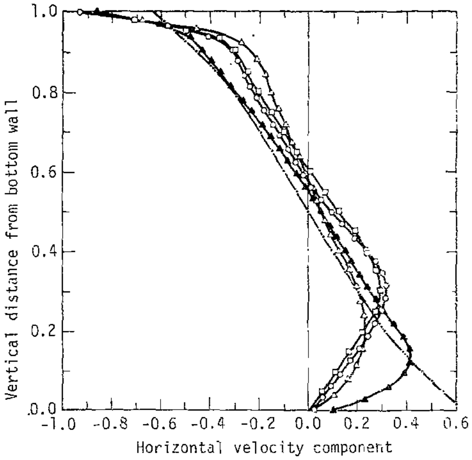


FIG. 10. Comparison of velocity profiles of u-component along a vertical line through the vortex center. ○ $R = 400$, $d = 1/32$ (this study); □, $R = 400$, Burggraf [16]; ▲, $R = 1000$, $d = 1/64$ (this study); △, $R = 1000$, Bozeman and Dalton [19]; – · – $R \rightarrow \infty$, Burggraf [16].

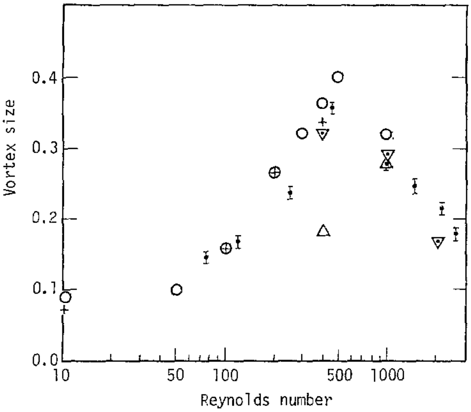


FIG. 11. Variation in size of upstream corner vortex vs. Reynolds number. Ī Experimental results, Pan and Acrivos [14]; +, Burgraf [16]; ○, Bozeman and Dalton [19]; ▽, Nallasamy and Prasad [20]; △, this study.

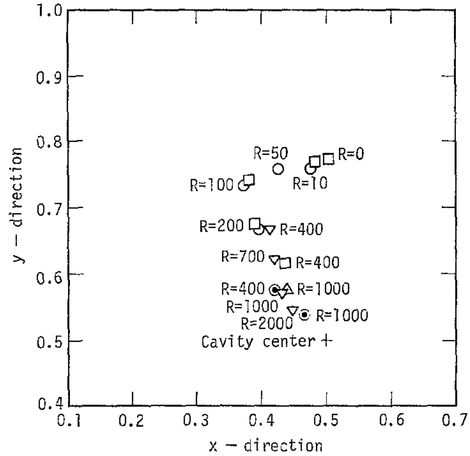


FIG. 12. Effect of Reynolds number on the location of the center of the primary vortex. □, Burggraf [16]; ○, Bozeman and Dalton, central differences [19]; △, Bozeman and Dalton, upwind differences [19]; V, Nallasamy and Prasad [20]; ⊙, this study.

file is equal to the negative of one half the vorticity in the center. The slope of BD implies a central vorticity of approximately 1.42 which is in disagreement with the value 1.83 given by Nallasamy and Prasad (NP) [20]. The present results yield values of approximately 2.1.

Figure 11 compares the spread of the upstream stationary vortex with other work. Results for the case $R = 10^3$ compare favorably with others, however, the case of $R = 400$ is disappointingly low. It is thought that the interior mesh size of 1/32 is too crude to adequately represent the flow.

Figure 12 plots the position of the center of the main vortex as it varies with R. The results of the present work bring the vortex center closer to the center of the cavity than previous studies.

Table I also includes approximate running times for the program once the number of vortices in D_1 had stabilized. The program was written in unsophisticated LRLTRAN, the Livermore Laboratory version of FORTRAN, and was run on the CDC 7600 R and S machines. The program was compiled by the CHAT compiler and the timing was done with the aid of the subroutine OØTIM. It should be noted that the times given in Table I give the total time needed to complete one entire time step. These times include such extraneous work as plotting the velocity field every time step.

V. Discussion of Results

The preceeding was a presentation of a numerical method for use in fluid flow problems with high Reynolds numbers. It is a hybrid combining a well known finite difference scheme with the newer vortex method [3]. The method has yielded accurate results on the square cavity flow problem, an example of steady circulating flow.

As a hybrid method, it must simultaneously satisfy the requirements imposed on each individual scheme, as well as to any other restrictions imposed by the combination process. It is well known that the ADI method is unconditionally stable when applied to linear problems. However, the author is unaware of stability conditions for the vortex scheme relating the time step k and the boundary discretization length h. Hence, one is forced to experiment.

One confusing aspect of the vortex method is the apparent lack of satisfaction of the tangential boundary condition, $\mathbf{u} \cdot \mathbf{s} = 0$. The velocity $\mathbf{u}$ does not include the necessary amount of vorticity along the boundary to cancel $\mathbf{u} \cdot \mathbf{s}$. In fact, $\mathbf{u}$ represents the velocity outside the boundary layer and is thus useful to spot unexpected features such as back flow or separation.

Although the numerical boundary layer can be evaluated at $t = t_m$, it has a global effect only at $t = t_{m+1}$ when it has been subdivided into vortex blobs and allowed to diffuse. Roughly half of the blobs are immediately lost as they cross the boundary and are thus discarded from the computation. The other half travel too far with the current choice of parameters. The structure of the blobs inside the cut-off length, σ $(= h/2\pi)$, was designed to have them exert a constant tangential velocity on G as they diffused normally into the fluid. However, the diffusive component is a random step in time, and in a time duration of k, the blobs experience a random push with a variance of $2k/R$. The parameters used imply that those blobs diffusing into the fluid travel beyond σ after they have been created. Thus, although at the time of its creation the numerical boundary layer contained the correct amount of vorticity to cancel $\mathbf{u} \cdot \mathbf{s}$ at G, after diffusing in the form of blobs it may not be strong enough, and the tangential boundary condition must be satisfied by the creation of another layer.

It may then be inquired if it is better to choose a smaller k, or larger σ, i.e. larger h, to obtain better results. Smaller time steps mean more vortices, hence longer running times. Another consideration was noted by Chorin [3]. The convective component in the vortex scheme is $O(k)$, while the diffusive component depends on the standard deviation of the random walk and is thus $O(k^{1/2})$. Hence, as k is decreased, the diffusive component may dominate, or at least exert a greater influence, over the convective component making the flow look more random. On the other hand, increasing σ, hence h, means the boundary layer is subdivided into fewer blobs and is, therefore, a worse approximation. The entire intention is defeated, the tangential condition may be better satisfied, but only at fewer points along the boundary.

It should be asked whether it is fair to demand fine detail from the vortex scheme near G. Fine detail is unavailable to finite difference methods anyway, unless one uses prohibitively small mesh sizes, or as proposed by Dorodnicyn [1] discretizes different approximations (boundary layer, ideal-fluid, etc.) in their corresponding areas. The method described here can be made more accurate by simultaneously decreasing k and d at the expense of greater demands of storage and computer time. The improvement in the satisfaction of the normal boundary condition can be easily judged by an examination of $\mathbf{u} \cdot \mathbf{n}$ along G. Although $\mathbf{u}$ along G represents flow outside the numerical boundary layer, it should be parallel to G. However, improvements in the satisfaction of the tangential boundary condition may be harder to judge and remains an open problem.

The domain interaction imposes no profound considerations or restrictions. As presented here, no restriction arises from the flow of ξ_1 into D_2 and only two criteria should be adhered to in ıegards to flow of ξ_2 into D_1. First, the method assumes that vorticity cannot be convected in a normal direction much beyond one mesh size in a time duration of $k/2$. Secondly, the standard deviation of the random walk should not be so large as to allow too much vorticity to diffuse beyond one mesh size in a time $k/2$. Therefore, it is not permissible to decrease d without simultaneously decreasing k.

Another consideration is what choice to make for δ, the distance from the interior domain, D_2, to the boundary, G. It is advantageous to expand D_2 as much as possible to minimize the number of vortices. However, one must allow room for the mesh expansion. The runs for $R = 100$, $d = 1/64$, used $\delta = 4d$. Larger values of δ were tried with no noticeable effect on the velocities.

The runs for $R = 1000$, $d = 1/32$ were done to check whether a drastic difference could be observed by changing the mesh size. The results were not as sharp as those with $d = 1/64$, but were enlightening in that they signalled that further resolution was required from the mesh. The coarse mesh produced a slight mismatching between the two subregions in the u-velocity profile near the sliding edge. This gave evidence that the truncation error of the ADI method in the interior regions dominated the flow. The coarse mesh is also responsible for the different slopes of the v-velocity profiles for the two values of d, 1/32 and 1/64 (see Fig. 9). The slope of these lines is directly related to the amount of vorticity present in the interior and the difference between the two slopes gives further evidence to the artificial effects of the coarse grid on the solution.

The runs for $R = 400$ were done using an interior mesh $d = 1/32$ to check how the method performed on a problem with smaller R. The overall results were encouraging. The u and v velocity profiles are given in Figs. 9 and 10 and the u-profile is compared with Burggraf's numerical results (Fig. 10). By comparing the u-profiles of $R = 1000$ with those of $R = 400$, it is evident how the wall affects the interior solution for the lower Reynolds number. When $R = 400$, the spin of the large central core is slowed down earlier as the bottom wall is approached. The viscous effects of the stationary wall have a braking influence on the rotation in the center. The location of the center of the central vortex is graphed in Fig. 12 and it lies further away from the cavity center than the one for $R = 1000$, $d = 1/64$.

The major disappointments for the $R = 400$ runs were with the inability to consistently see the small scale phenomena, such as the counter-rotating vortices in the corners. As in the $R = 1000$, $d = 1/32$ case, the upstream corner vortex would appear as it is composed of vortex blobs of negative sign which were pushed into the corners, but at later time steps would disappear as these escaped through G. Since in the case $R = 1000$, $d = 1/64$, the corner vortices did not exhibit such a behavior, it is conjectured that the computational grid is responsible. A grid that is too coarse cannot correctly interpret this small-scale motion.

Running the program for $R = 2000$ proved costly. The program was written with the expectation that it would not be necessary to use an interior mesh size larger than $d = 1/64$. However, this proved not to be the case. A total of 380 time steps were run experimenting with various h and k's. Using $k = 0.08$ and $h = 1/30$ caused instability; the interior vortex began to accelerate and large amounts of vorticity were introduced at each time step. Decreasing k to 0.05 removed the instability but a distinct grid effect could be seen. Even using $k = 0.05$, $h = 1/45$ could not remove the obvious difficulties. There were large interior velocitites and a drastic mismatching of u, v-profiles from the domain D_2 with D_1. It is believed that a finer interior mesh must be used to get better results. It could then be said that the limitation of the proposed method lies not in the vortex method, nor in the combination process, but in the difference scheme itself. Even with the absence of areas of sharp gradients, the difference method carries an inherent truncation error which exerts an increasing influence on problems of larger Reynolds numbers.

It is the author's conjecture that the early breakdown of difference methods occurs from the inability to correctly interpret areas of sharp derivatives which usually occur near boundaries. Therefore use of a scheme of the type proposed may allow one to solve problems with somewhat larger R, but eventually a limit in R will be reached when the truncation error of the difference method will dominate.

It should be noted that little thought was given to the choice of the finite difference method for the interior region. The ADI method presented here was chosen because of its second order accuracy. Although methods using central differences do not introduce the artificial diffusion present with upwind differences, they lead to linear equations that are more difficult to solve (see the comparison made by Ref. [19]). Thus, another difference scheme may make the proposed hybrid method more applicable. However, in any combination process it is important to allow vorticity transfer from one region to the other. A version of the expanding scheme proposed here should be adaptable to other methods.

The basic idea should be applicable in a wide class of flows. The domain of interest is subdivided into two subdomains, one bounded near the boundaries, and one away from the boundaries, possibly unbounded, depending on the problem. In the domain near the boundary the equations of motion are solved by the vortex method. In the other domain a suitable finite-difference method, or another method of a non-random type, can be chosen, with care taken if this domain is unbounded.

Acknowledgments

The author wishes to thank Professor Chorin for the unselfish aid and encouragement that was extended him during the course that this work was being done. Thanks also go out to Drs. Killeen, Alder and Chang of the Lawrence Livermore Laboratory for extending the generous support and facilities of the laboratory without which this work could not have been accomplished.

References

1. A. A. Dorodnicyn, *in* "Proceedings of IIIrd International Conference on Numerical Methods in Fluid Mechanics," Lecture Notes in Physics, pp. 1–11, Springer-Verlag, New York/Berlin, 1973.
2. D. G. Fox, J. R. Herring, R. H. Kraichnan, and S. A. Orszag, *J. Fluid Mech.* **66** (1974), 417.
3. A. J. Chorin, *J. Fluid Mech.* **57** (1973), 785.
4. B. Davari, "Numerical Study of Viscous Flows Past a Circular Cylinder: Application of Chorin's Method," Lawrence Berkeley Laboratory Publication LBL-2480, Berkeley, Calif., 1974.
5. A. I. Shestakov, "Vortex Blobs in a Square Cavity, Application of Chorin's Method," Lawrence Berkeley Laboratory Publication LBL-3393, Berkeley, Calif., 1974.
6. R. D. Richtmyer and K. W. Morton, "Difference Methods for Initial Value Problems," Interscience, New York, 1967.
7. B. L. Buzbee, G. H. Golub, and C. W. Nielson, *SIAM J. Numer. Anal.* 7 (1970), 627.
8. O. Buneman, *J. Comput. Phys.* **11** (1973), 250.
9. J. Lamperti, "Probability," Benjamin, New York, 1966.
10. W. L. Deemer and D. F. Votaw, *Ann. Math. Statist.* **26** (1955), 498.
11. D. Margolies, private communication, 1978.
12. C. Peskin, *J. Comput. Phys.* **10** (1972), 252.
13. J. W. Eastwood and R. W. Hockney, *J. Comput. Phys.* **16** (1974), 342.
14. A. Acrivos and F. Pan, *J. Fluid Mech.* **28** (1967), 643.
15. M. Kawaguti, *J. Phys. Soc. Japan* **16** (1961), 2307.
16. O. Burggraf, *J. Fluid Mech.* **24** (1966), 113.
17. D. Greenspan, *Comput. J.* **12** (1969), 88.
18. A. K. Runchal, D. B. Spalding, and M. Wolfshtein, *Phys. Fluids* **12**, 12; *Supplement II* (1969), II.21.
19. J. D. Bozeman and C. Dalton, *J. Comput. Phys.* **12** (1973), 348.
20. M. Nallasamy and K. K. Prasad, *J. Fluid Mech.* **79**, 2 (1977), 391.
21. M. Nallasamy and K. K. Prasad, *J. Comput. Phys.* **15** (1974), 429.
22. S. G. Rubin and P. K. Khosla, *J. Comput. Phys.* **24** (1977), 217.