Numerical Heat Transfer, Part B, 41: 447–462, 2002

1040-7790/02 $12.00 + .00

A FIRST STEP IN APPLYING THE SINC COLLOCATION METHOD TO THE NONLINEAR NAVIER–STOKES EQUATIONS

Susheela Narasimhan
Department of Mechanical Engineering, University of Utah, Salt Lake City, Utah, USA

Joseph Majdalani
Department of Mechanical & Industrial Engineering, Marquette University, Milwaukee, Wisconsin, USA

Frank Stenger
Department of Computer Science, University of Utah, Salt Lake City, Utah, USA

Different numerical approaches have been proposed in the past to solve the Navier–Stokes equations. Conventional methods have often relied on finite-difference, finite-element, and boundary-element techniques. Multigrid methods have been recently introduced because they help to obtain a faster convergence rate of the error residual. A difficulty plaguing numerical methods today is the inability to treat singularities at or near boundaries. Such difficulties become even more pronounced when coupled with the need to handle semi-infinite and infinite domains. Sinc-based numerical algorithms have the advantage of handling singularities, boundary layers, and semi-infinite domains very effectively. In addition, they typically require fewer nodal points and are proven to provide an exponential convergence rate in solving linear differential equations. This study involves a first step in applying the Sinc-based algorithm to solve a nonlinear set of partial differential equations. The example we consider arises in the context of a driven-cavity flow in two space dimensions. As such, the steady and incompressible Navier–Stokes equations are solved by means of two-dimensional Sinc collocation in conjunction with the primitive variable method and a pressure correction algorithm based on artificial compressibility. Simulations are also carried out using forward differences, central differences, and a commercial code. Results are compared with one another and with the Sinc-collocation approximation. It is found that the error in the Sinc-collocation approximation outperforms other solutions, especially near the singular corners of the cavity.

Received 26 January 2001; accepted 11 September 2001.
Address correspondence to Prof. Joseph Majdalani, Department of Mechanical & Industrial Engineering, Marquette University, 1515 W. Wisconsin Ave., Milwaukee, WI 53233, USA. E-mail: maji@mu.edu

1. INTRODUCTION

The purpose of this article is to present a novel application of the Sinc collocation method to the nonlinear Navier–Stokes equations in two space dimensions. The Sinc collocation method is a spectral decomposition technique whose computational algorithm resembles trigonometric interpolation by Fourier series. By comparison to traditional finite-difference, finite-element, and boundary-element methods, the Sinc collocation approach has been shown to be more suitable in handling singularities and semi-infinite domains. Furthermore, the residual error entailed in the Sinc collocation approach is known to exhibit an exponential convergence rate, even in the presence of singularities. When these features are added to the requirement by a Sinc collocation algorithm for fewer nodal points in the solution domain, this spectral technique becomes an attractive alternative which, in some cases, can become superior to traditional multigrid and modern high-speed computational methods.

Since its inception by Stenger [1], the notion of a Sinc expansion has appeared in diverse physical settings including interpolations, integrations, and solutions of both ordinary and partial differential equations [2–4]. For example, in performing accurate interpolations of discrete signals, geometric transforms, and test measurements, the Sinc approximation has been used extensively by Jeng [5], by Schanze [6], and by Kober, Unser, and Yaroslavsky [7]. The latter have demonstrated that Sinc interpolation methods can significantly outperform conventional methods of nearest neighbor.

In evaluating the homogeneous Lamé equations, Stenger [8] has presented two integral formulations through Sinc convolution. In evaluating Cauchy-type integral equations, Bialecki and Keast [9] have shown that a numerical method based on a truncated sum of Sinc functions could yield excellent approximations for analytic functions with endpoint singularities. In fact, Sinc methods have been found to be particularly useful in the treatment of ordinary and partial differential equations with singularities. This is evidenced throughout the work of Bowers and Lund [10]; therein, the Galerkin method has been successfully used in conjunction with Sinc functions to approximate the solution of the Poisson problem. In the same context, Lewis, Lund, and Bowers [11] have applied the space-time Sinc-Galerkin method for the numerical solution of the parabolic class of partial differential equations in one spatial dimension. Later, in Smith et al. [12], the Sinc-Galerkin method has been extended to handle fourth-order ordinary differential equations. Even at this high order, the consistency of the method in exhibiting an exponential convergence rate could be shown. In a subsequent study, Smith, Bowers, and Lund [13] applied the Sinc-Galerkin method to several examples involving fourth-order time-dependent partial differential equations in both space and time.

In his 1997 review of Sinc methods, Stenger [14] presented a novel procedure, based on Sinc convolution matrices, for solving the Poisson problem. This innovative method has been successfully applied by Stenger and O'Reilly [15] in three types of medical applications involving optimal controls, reconstruction of X-ray tomography, and inversion of ultrasonic tomography.

For a second-order, two-point boundary-value problem with multiple domains, Morlet, Lybeck, and Bowers [16, 17] have combined the Sinc collocation domain decomposition method with the Schwarz alternating technique to overcome the problem's singularities. In their work, the exponential convergence property was proven for a problem with subdomains. In the same context, the Poisson equation was solved using domain decomposition by Lybeck and Bowers [18, 19].

For the purpose of achieving higher-order precision, the Sinc function approximation has also been used in heat transfer problems by Narasimhan, Chen, and Stenger [20, 21], Lippke [22], and others. In [20], the two-dimensional, steady-state heat conduction problems in both a square and a semi-infinite rectangular cavity were considered. For the square geometry, the Sinc approximation was shown to outperform both the finite-difference and multigrid methods uniformly across the computational domain.

In an effort to solve by Sinc collocation the initial-boundary-value problems for the nonlinear evolution equations in one and two space dimensions, recent interest in applying Sinc methods to nonlinear problems has received favor in the work of Bellomo, Ridolfi, and co-workers [23–25]. Recent studies by Bowers and co-workers [26] have also applied the Sinc technique to the modeling of biofilms and wind-driven currents. The current study constitutes one such example whose main purpose is to determine a viable algorithm for applying the Sinc method to the set of nonlinear Navier–Stokes equations (NSE). To the authors' knowledge, such an attempt represents a first step toward better understanding the manner in which Sinc methods could be effectively extended to full solutions of the NSE system. To illustrate the process, the algorithm will be applied to the cavity-driven problem. For comparison purposes, the problem will also be solved with finite differences (both forward and central) and using a known commercial code [27]. Unlike Bellomo et al. [23–25], who applied the Sinc collocation on an equispaced domain, we shall follow conventional Sinc practices of clustering more Sinc points near edges in order to handle singularities more effectively.

2. METHODOLOGY

In seeking a solution for the incompressible NSE system, it may be safe to say that the two most popular strategies consist of using either the vorticity stream-function approach or the primitive-variable approach. In this study, the primitive-variable approach will be adopted.

2.1. The Primitive-Variable Approach

Using the asterisk to denote dimensional variables, the two-dimensional incompressible Navier–Stokes equations can be written as

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \tag{1}$$

$$\frac{\partial u^*}{\partial t^*} + u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} = -\frac{1}{\rho}\frac{\partial p^*}{\partial x^*} + \nu\left(\frac{\partial^2 u^*}{\partial x^{*2}} + \frac{\partial^2 u^*}{\partial y^{*2}}\right) \tag{2}$$

$$\frac{\partial v^*}{\partial t^*} + u^* \frac{\partial v^*}{\partial x^*} + v^* \frac{\partial v^*}{\partial y^*} = -\frac{1}{\rho}\frac{\partial p^*}{\partial y^*} + \nu\left(\frac{\partial^2 v^*}{\partial x^{*2}} + \frac{\partial^2 v^*}{\partial y^{*2}}\right) \tag{3}$$

where x^*, y^*, t^*, u^*, v^*, and p^* represent the streamwise and cross-streamwise space coordinates, time, the streamwise, and cross-streamwise velocities, and pressure, respectively. The density and kinematic viscosities are given by ρ and ν. The foregoing set can be made dimensionless via

$$x = \frac{x^*}{L_{\text{ref}}} \qquad y = \frac{y^*}{L_{\text{ref}}} \qquad t = \frac{t^* V_{\text{ref}}}{L_{\text{ref}}} \qquad u = \frac{u^*}{V_{\text{ref}}}$$
$$v = \frac{v^*}{V_{\text{ref}}} \qquad p = \frac{p^*}{\rho V_{\text{ref}}^2} \qquad \text{Re} = \frac{V_{\text{ref}} L_{\text{ref}}}{\nu} \tag{4}$$

The ensuing nondimensional NSE system becomes

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{5}$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{1}{\text{Re}}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \tag{6}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{\partial p}{\partial y} + \frac{1}{\text{Re}}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \tag{7}$$

Different methods have been followed in the past for the solution of the incompressible NSE system expressed in primitive-variable form. One method involves introducing an artificial compressibility term into the continuity equation. This artificial compressibility term is then used as a pressure-correction factor that will eventually vanish when the steady state is reached. Another method involves considering a separate Poisson equation for pressure in lieu of the continuity equation. The artificial compressibility artifact will be employed here in unison with the Sinc collocation method.

2.2. Sinc Collocation in One Dimension

The Sinc collocation is similar to the Fourier spectral technique for approximating functions and derivatives. Before applying the approach to the NSE system, it may be helpful to illustrate the procedure with a simple example in one space dimension. The purpose of the example is to explain how functions can be approximated with Sinc collocation. On that account, we consider the cubic polynomial

$$f(x) = 2x^2 + x - 3x^3 \qquad 0 \leq x \leq 1 \tag{8}$$

Clearly, $f(x)$ vanishes at both ends of the interval. In order to approximate this function, we invoke, for a function extending over an interval $a \leq x \leq b$, the logarithmic transformation [3]

$$\phi(x) = \log\left(\frac{x - a}{b - x}\right) = \log\left(\frac{x}{1 - x}\right) \tag{9}$$

Subsequently, the Sinc points are defined by

$$x_k = \frac{e^{kh}}{1 + e^{kh}} \qquad h = \sqrt{\frac{\pi d}{\beta N}} \tag{10}$$

where $d = \pi$, $\beta = 1$, and N is the number of Sinc points (left and right) that will be used in the $f(x)$ approximation [28]. Based on Eqs. (9)–(10), the collocation expansion can be expressed as

$$f(x) \simeq \sum_{k=-N}^{k=N} C_k S[\phi(x), kh] \tag{11}$$

where the Sinc function $S[\phi(x), kh]$ is given by

$$S[\phi(x), kh] = \frac{\sin(\{\pi[\phi(x) - kh]\}/h)}{\pi[\phi(x) - kh]/h} \qquad k = -N, -N+1, \ldots, -1, 0, 1, \ldots, N-1, N \tag{12}$$

When the Sinc function is approximated at the $(2N + 1)$ nodal points, one may express the results in matrix form. Using $[I]$ to represent the identity matrix, one may write

$$[I]_{(2N+1)(2N+1)}[C_k]_{(2N+1)(1)} = [f(x_k)]_{(2N+1)(1)} \tag{13}$$

At the outset, the $2N + 1$ collocation constants $[C_k]$ can be evaluated from the function at the nodal points. Once these constants are determined, Eq. (11) can be used to evaluate the function at any intermediate point. Figure 1a compares true and approximate values obtained with $N = 10$. In the interest of clarity, the corresponding absolute error in the Sinc approximation is calculated and tabulated in Table 1 at several values of N. Note that the error drops rapidly as N is increased. However, for $N \geq 160$, one notices a flattening in the error. This is due to inevitable limitations in machine precision leading to the accumulation of round-off errors.

By virtue of Eqs. (11), (12), and (9), Sinc expansions always vanish at the endpoints. As such, the scheme needs to be modified for functions not exhibiting this property. This notion will be illustrated by considering a polynomial that has nonzero values at the ends. For simplicity, let us consider

$$f(x) = 2x^2 + x - 3x^3 + 3 \tag{14}$$

which has a value of 3 at $x = 0, 1$. To overcome this difficulty, the Sinc approximation of Eq. (14) must be augmented by two splines at either ends. This can be accomplished by setting

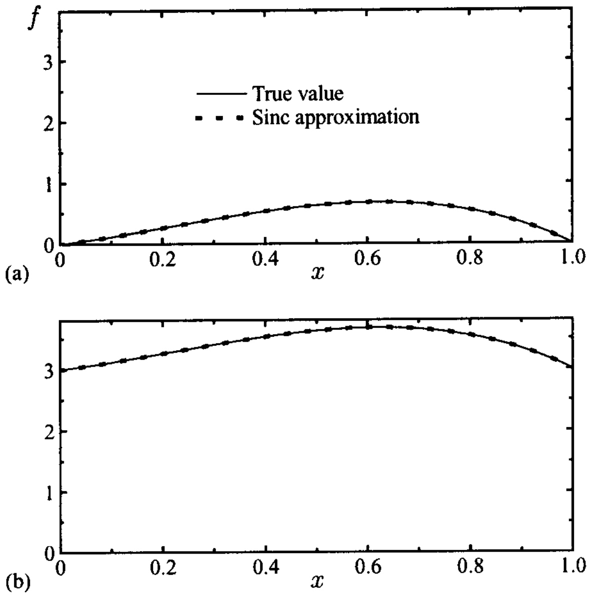

Figure 1. Sinc collocation approximation for a function that is (*a*) zero and (*b*) nonzero at the endpoints.

Table 1. Error in the Sinc approximation for the test functions given in Figure 1 using increasing values of N. Errors are identical for both cases (*a*) and (*b*) in Figure 1

$x\backslash N$	5	10	20	40	80	160	320	640
0.00	0	0	0	0	0	0	0	0
0.05	2.4E-3	8.7E-5	1.3E-5	2.7E-8	5.6E-11	6.1E-16	1.6E-16	1.5E-16
0.10	9.5E-3	5.7E-4	1.1E-5	1.4E-7	5.4E-11	4.4E-16	2.1E-16	1.5E-16
0.15	7.2E-3	7.3E-4	3.0E-5	1.7E-7	1.6E-11	8.3E-17	3.3E-16	1.7E-16
0.20	4.8E-4	1.0E-3	2.7E-6	1.1E-7	1.5E-11	1.6E-15	2.8E-16	6.1E-16
0.25	7.5E-3	3.7E-4	3.3E-5	1.2E-7	3.8E-11	1.1E-15	1.7E-16	6.1E-16
0.30	1.1E-2	5.2E-4	2.1E-5	1.6E-7	9.6E-11	1.9E-15	6.1E-16	1.7E-16
0.35	1.2E-2	1.1E-3	1.3E-5	1.4E-7	6.9E-11	1.8E-15	3.9E-16	4.4E-16
0.40	9.4E-3	1.1E-3	3.5E-5	1.1E-7	4.8E-11	1.4E-15	2.2E-16	2.2E-16
0.45	5.1E-3	6.9E-4	2.8E-5	1.9E-7	1.0E-10	5.6E-16	1.2E-15	0
0.50	0	0	0	0	0	0	0	0
0.55	4.6E-3	6.5E-4	2.6E-5	1.9E-7	9.8E-11	7.8E-16	1.1E-16	0
0.60	7.8E-3	9.9E-4	3.2E-5	1.0E-7	4.6E-11	7.8E-16	8.9E-16	1.1E-16
0.65	8.9E-3	8.9E-4	1.1E-5	1.3E-7	6.4E-11	1.3E-15	1.1E-16	5.6E-16
0.70	7.6E-3	3.9E-4	1.7E-5	1.4E-7	8.7E-11	8.9E-16	5.6E-16	8.9E-16
0.75	4.4E-3	2.6E-4	2.6E-5	9.7E-8	3.3E-11	2.2E-15	1.1E-15	1.1E-16
0.80	2.3E-4	6.4E-4	1.9E-6	8.7E-8	1.3E-11	2.0E-15	7.8E-16	1.1E-15
0.85	2.8E-3	3.9E-4	2.0E-5	1.2E-7	1.3E-11	0	3.3E-16	1.1E-16
0.90	2.6E-3	2.5E-4	6.0E-6	9.8E-8	4.1E-11	3.3E-16	3.9E-16	7.2E-16
0.95	2.5E-4	2.6E-5	5.7E-6	1.6E-8	3.9E-11	4.2E-16	7.2E-16	5.6E-16
1.00	0	0	0	0	0	0	0	0

$$f(x) \simeq C_{-N-1}x + \sum_{k=-N}^{k=N} C_k S[\phi(x), kh] + C_{N+1}(1-x) \tag{15}$$

As the Sinc function $S[kh, \phi(x)]$ goes to zero at the boundaries, the constants C_{-N-1} and C_{N+1} can be readily evaluated to be 3 in order to ensure that the function itself equals 3 at $x = 0, 1$. After finding C_{-N-1} and C_{N+1}, the regular collocation constants C_k can be evaluated as before. A comparison between the Sinc approximation and the true function is given in Figure 1*b*. The error in this approximation is found to be identical to that given in Table 1. Clearly, the inclusion of splines does not seem to degrade the Sinc approximation.

3. THE DRIVEN-CAVITY PROBLEM

The Sinc collocation scheme described previously is now applied to the NSE system in two space dimensions. The physical setting considered is that of the driven-cavity problem. The corresponding governing equations are given by Eqs. (5)–(7). These will be solved using the primitive-variable method with artificial compressibility.

In this study, the NSE solution was first attempted using the vorticity-stream function approach. As usual, the vorticity-stream function approach involved calculating second-order derivatives on the boundary. To that end, the original Sinc function had to be divided by the derivative of the logarithmic transformation variable that accompanied the Sinc formulation [28]. This operation caused the solution matrices to become ill-conditioned. For this reason, the vorticity-stream function approach was no longer pursued.

3.1. Domain and Variables

As usual, the bottom of the cavity is located at $y = 0$, $0 \le x \le 1$, and the velocity is constant and equal to the reference velocity V_{ref} along $y = 1$, $0 \le x \le 1$. The dimensionless velocity boundary conditions at $y = 1$ are hence $u(x, 1) = 1$ and $v(x, 1) = 0$. The vertical walls are rigid at both $x = 0$ and $x = 1$, $0 \le y < 1$. Due to the singularities at $(0, 1)$ and $(1, 1)$, the approximations for u and v can be written as sums of Sinc expansions and splines at the endpoints:

$$\begin{aligned} u(x, y) \simeq & \sum_{k_1=-N}^{N} \sum_{k_2=-N}^{N} C_{k_1,k_2} S[\phi(x), k_1 h] S[\phi(y), k_2 h] + x + (1-x) \\ & + \sum_{k_1=-N}^{N} C_{k_1} S[\phi(x), k_1 h] + x \sum_{k_2=-N}^{N} C_{k_2} S[\phi(y), k_2 h] \\ & + (1-x) \sum_{k_2=-N}^{N} C_{k_2} S[\phi(y), k_2 h] \end{aligned} \tag{16}$$

$$v(x, y) \simeq \sum_{k_1=-N}^{N} \sum_{k_2=-N}^{N} C_{k_1,k_2} S[\phi(x), k_1 h] S[\phi(y), k_2 h] \tag{17}$$

In like fashion, the pressure can be approximated by

$$
\begin{aligned}
p \simeq & \sum_{k_1=-N}^{N} \sum_{k_2=-N}^{N} C_{k_1,k_2} S[\phi(x), k_1 h] S[\phi(y), k_2 h] + y \sum_{k_1=-N}^{N} C_{k_1} S[\phi(x), k_1 h] \\
& + (1-y) \sum_{k_1=-N}^{N} C_{k_1} S[\phi(x), k_1 h] + x \sum_{k_2=-N}^{N} C_{k_2} S[\phi(y), k_2 h] \\
& + (1-x) \sum_{k_2=-N}^{N} C_{k_2} S[\phi(y), k_2 h]
\end{aligned}
\tag{18}
$$

These approximations can be used in conjunction with a pressure-correction scheme to develop the computational algorithm.

3.2. Modified Pressure-Correction Scheme

The nonlinear convective terms in Eqs. (6) and (7) can be linearized by using the velocities stored during a preceding iteration. For example, in order to evaluate $u\partial u/\partial x$, one may use $u^{n-1}\,\partial u^n/\partial x$, where u^n represents the velocity at the current iteration step n. In this study, the steady-state Navier–Stokes equations are repeatedly solved until the modified continuity equation is satisfied. The modified pressure-correction algorithm requires one to perform the following steps:

1. Initialize the velocities and pressure u^0, v^0, and p^0 in the entire domain.
2. Calculate the pressure gradients $-\partial p/\partial x$, and $\partial p/\partial y$ using the Sinc collocation expression (18).
3. Update the velocities by using the Sinc collocation equations for u and v given by Eqs. (16)–(17) and by solving the steady-state Navier–Stokes equations in the primitive variable form:

$$
\frac{1}{\text{Re}}\left(\frac{\partial^2 u^n}{\partial x^2} + \frac{\partial^2 u^n}{\partial y^2}\right) - u^{n-1}\frac{\partial u^n}{\partial x} - v^{n-1}\frac{\partial u^n}{\partial y} = \frac{\partial p^n}{\partial x}
\tag{19}
$$

$$
\frac{1}{\text{Re}}\left(\frac{\partial^2 v^n}{\partial x^2} + \frac{\partial^2 v^n}{\partial y^2}\right) - u^{n-1}\frac{\partial v^n}{\partial x} - v^{n-1}\frac{\partial v^n}{\partial y} = \frac{\partial p^n}{\partial y}
\tag{20}
$$

   Note that the convective terms are linearized by using the most recently stored values u^{n-1} and v^{n-1}.
4. Obtain the velocity gradients $\partial u/\partial x$ and $\partial v/\partial y$ from the current velocity field.
5. Define the artificial compressibility $D = (\partial u/\partial x) + (\partial v/\partial y)$.
6. Calculate the pressure correction term $p_{\text{corr}} = -\lambda D$, where λ is a small number.
7. Update the pressure field by using $p^n = p^{n-1} + p_{\text{corr}}$.
8. Repeat steps 1–7 until satisfied. This condition will typically occur when D becomes so small that the continuity equation becomes virtually satisfied and when both velocity and pressure fields would have reached their steady-state values.

3.3. Special Treatment at the Boundaries

The pressure correction near the endpoints requires evaluating the derivatives $\partial u/\partial x$ and $\partial v/\partial y$. These, in turn, require evaluating the derivatives of the Sinc functions at the boundaries. This is rendered difficult by the fact that the derivatives of the Sinc functions can lead to a numerical overflow near the boundaries. One reason can be attributed to the tight grid spacing near the boundaries, where Sinc points become clustered in a geometric fashion.

A variety of plausible approaches were tried in this study in order to improve the convergence history for the NSE system. The first approach was based on a revised definition of the Sinc function within the series. This revised definition consists of dividing the original Sinc function by the first derivative of the logarithmic transformation function raised to the power of the order of the derivative to be approximated [28]. For example, if we were to approximate a second-order derivative, we would have to raise the first derivative of the transformation function (ϕ') to the second power. The revised definition would read, in that case,

$$f(x) \simeq \sum_{k=-N}^{N} \frac{f(kh)S[\phi(x), kh]}{(\phi'(x))^2} \tag{21}$$

The main disadvantage of this approach lies in the ill-conditioning of the solution matrix stemming from Eq. (13). In fact, the condition number of the resulting matrix becomes enormous. This, of course, defeats practical attempts to make progress toward a solution. Unfortunately, a more convenient technique to handle derivatives at boundaries has yet to be developed. It is hoped that future research with Sinc methods will be successful in devising a scheme that is capable of overcoming similar difficulties.

The second approach that was attempted consisted of approximating each endpoint derivative by its adjacent value. The latter could be determined from the closest nodal point near the boundary. This approximation was possible here because of the fine-grid resolution near the boundaries, where derivatives changed very slowly. In the driven-cavity problem, this approach led to a diverging solution. As a result, it was abandoned.

The third approach we used was to calculate the derivatives on the boundaries using finite differences. For example, boundary gradients of u and v were evaluated using first-order operators such as

$$\frac{\partial u}{\partial x} \simeq \frac{u_{i,j} - u_{i-1,j}}{\Delta x} \tag{22}$$

$$\frac{\partial v}{\partial y} \simeq \frac{v_{i,j} - v_{i,j-1}}{\Delta y} \tag{23}$$

These approximations worked very well and led to a rapidly converging solution.

4. RESULTS AND DISCUSSION

The Sinc collocation method along with the modified pressure-correction algorithm and the finite-difference method for calculating derivatives near boundaries have given rise to a well-behaved solution algorithm. In this study, simulations were carried out at different Reynolds numbers. For brevity, they will be illustrated for $Re = 25$ and $\lambda = 0.0001$. Simulations were also carried out independently with a finite-difference algorithm using central differencing without upwinding as well as first-order upwinding for the nonlinear terms. For further reassurance, numerical results were also acquired from a commercially available computational fluid dynamics (CFD) package [27]. The computational meshes that were used are shown in Figure 2. Both finite-difference and Fluent codes utilized a uniform resolution of 100×100 steps. In the Sinc collocation, a value of $N = 10$ resulted in $(2N + 1) = 21$ steps in both x and y directions. As such, the total number of cells used by the Sinc collocation was approximately 4.4% of the cells considered by the other routines.

In Figure 3*a*, a comparison is presented showing the results of the Sinc collocation, Fluent, finite difference with central differencing, and finite difference with upwinding for the u profile along the vertical centerline of the cavity ($x = \frac{1}{2}$, $0 \leq y \leq 1$). Clearly, the agreement is satisfactory. In the same context, Figure 3*b* compares profiles for v along the horizontal centerline of the cavity ($y = \frac{1}{2}$, $0 \leq x \leq 1$). Here too, profiles seem to compare reasonably well except for some small discrepancies in magnitudes. Figures 4 and 5 give the iso-velocity contour plots of u and v within the entire cavity using all four numerical schemes. At the outset, favorable agreement is found between the Sinc approach and Fluent. By the same token, the least accuracy is realized with the finite-difference approach based on central differencing.

In addition to these plots, an error analysis that uses Fluent as a benchmark has indicated that the absolute errors in evaluating u and v increase as the singular corners at the top are approached (i.e., near $y = 1, x = 0, 1$). The error with the Sinc method was found to be the smallest. This may be attributed to the inherent capacity of a Sinc-generated grid to cope better with singularities at the top corners of the cavity, where more points are automatically distributed.

While 10,000 cells were employed in both finite-difference and CFD codes, the Sinc algorithm necessitated only 441 cells. Despite this 23:1 gain in spatial discretization, the Sinc matrices were dense and hence demanded longer computation time. The advantage of the Sinc approach in improving accuracy with fewer collocation points (than needed in the corresponding finite-difference or finite-element methods) appears to be offset by the dense matrices that become inevitable by virtue of the global approximation nature of the Sinc method. This problem becomes quite pronounced in the iterative solution of the nonlinear NSE system, where repeated matrix operations must precede the steady-state solution. This is one area where the use of parallel computing and message-passing interface (MPI) could be very helpful.

Another important functionality that has to be introduced within the Sinc function scheme is a better way to approximate derivatives at the boundaries. The traditional Sinc approach has relied on approximating the endpoint derivatives by using a modified definition of the Sinc function [3]. This has led, in our problem, to ill-conditioned matrices. To overcome this complication, a different approach,

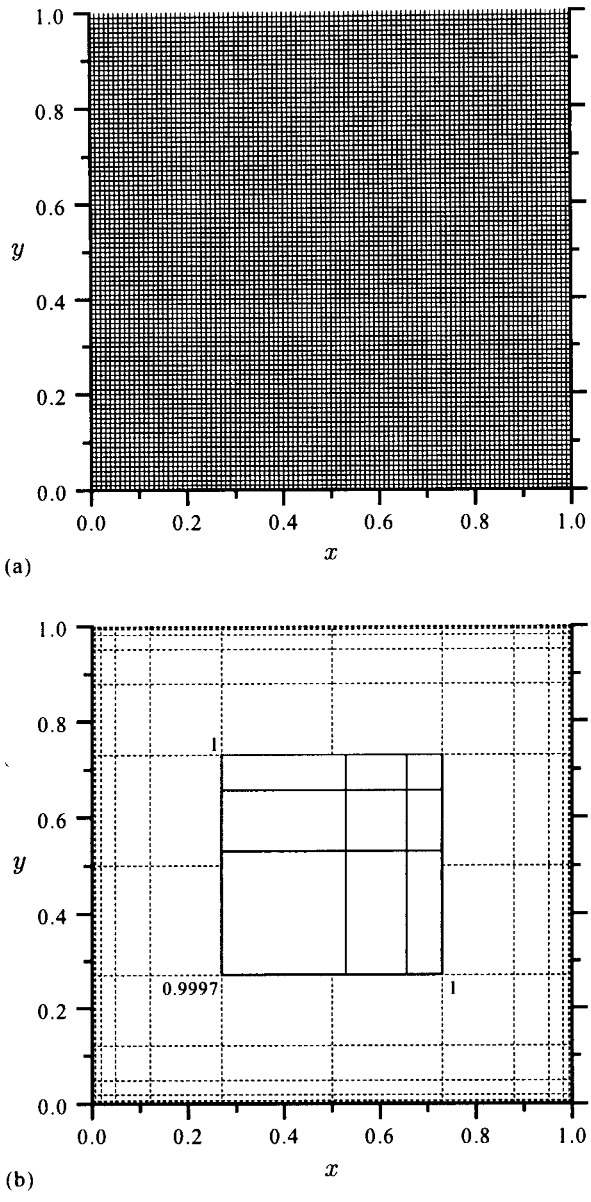

Figure 2. Grid resolution inside the square cavity using (*a*) 101 × 101 nodal points in both finite-difference and Fluent computations, and (*b*) 23 × 23 nodes in the Sinc collocation. The inset in (*b*) magnifies the geometric grid resolution near the upper right corner.

namely, one that was based on finite differencing, had to be resorted to. It is hoped that a better way of approximating the derivative will be later developed in order to extend the application of Sinc collocations to more complex engineering problems.

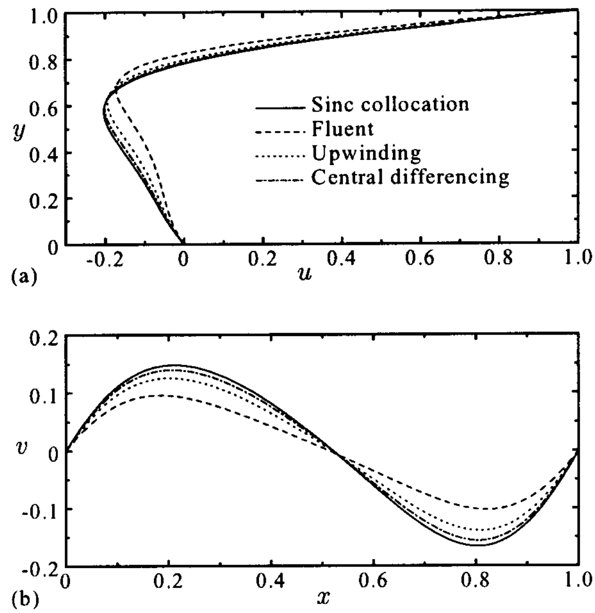

Figure 3. Comparison of velocity profiles for (*a*) *u* along the cavity's vertical centerline and (*b*) *v* along the cavity's horizontal centerline.

Suggestions for future developments include methods to transform the Navier–Stokes equations into integral equations (which can be more easily solved). An attempt could also be made to assign certain weights to the coefficients within the Sinc expansion. These weighing or relaxation factors could be related to the neighboring velocities in a manner to introduce artificial upwinding into the Sinc collocation method. The fact remains that Sinc collocation is a global spectral approximation method and not a localized pointwise approximation. Unlike finite-difference or finite-element methods, where localized approximations are inherent, upwinding remains, at present, more difficult to accommodate into a Sinc collocation scheme. As such, it needs to be carefully addressed.

5. CONCLUSIONS

In this article, the Sinc collocation expansion was applied to the two-dimensional Navier–Stokes equations to solve the driven-cavity flow problem. The primitive-variable method was used in conjunction with a modified pressure-correction algorithm based on artificial compressibility. Calculations of the velocity and pressure distributions were repeated until the mass balance was satisfied and the velocity profiles no longer changed. The flow profiles obtained from Sinc collocation were compared with the results obtained from central differences, forward differences, and a commercial CFD code. Comparisons indicated that the profiles agreed well with each other except for some discrepancies near the left- and

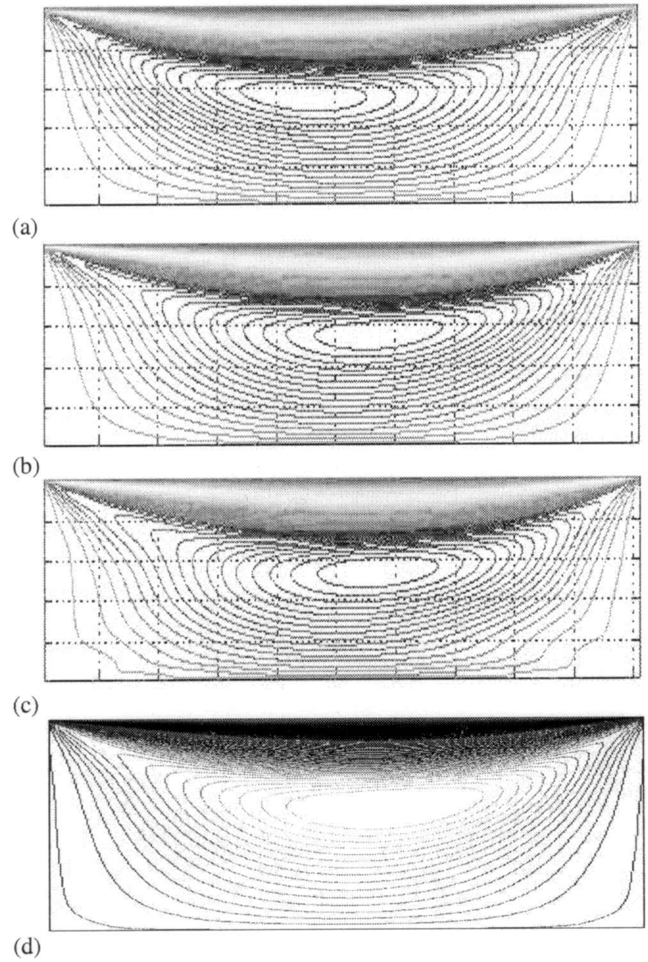

Figure 4. Iso-velocity contours of u profiles using (*a*) central differencing, (*b*) upwinding, (*c*) Sinc collocation, and (*d*) Fluent.

right-hand-side corners of the cavity. In the neighborhood of those singular endpoints, the Sinc algorithm appeared to outperform other methods by spreading an increasingly larger number of points as the corners were approached. In the cavity-driven problem, more accuracy was uniformly obtained with the Sinc results than with finite-difference methods that employed 23 times more computation cells. However, the improved accuracy with fewer nodes was hampered by the need to repeatedly solve dense matrices. Another difficulty was encountered in evaluating the derivatives near the boundaries, where singularities occurred. In order to avoid ill-conditioning, a better methodology to define endpoint derivatives than prescribed by conventional Sinc practices is deemed necessary. In addition to proposing a more suitable scheme for calculating derivatives, we suggest a careful combination of upwinding with Sinc collocation and the use of parallel computing

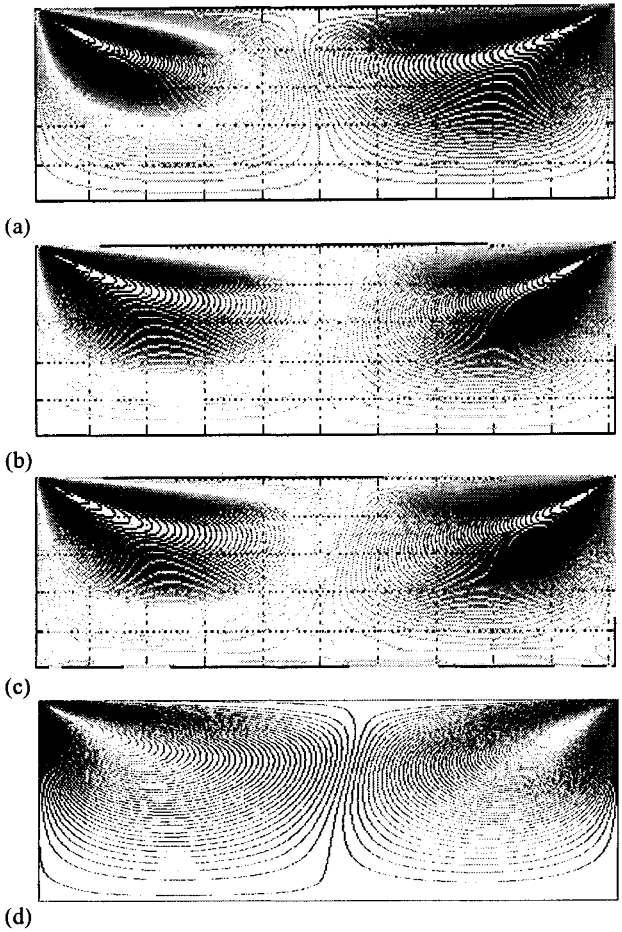

Figure 5. Iso-velocity contours of v profiles using (*a*) central differencing, (*b*) upwinding, (*c*) Sinc collocation, and (*d*) Fluent.

to reduce the time needed for convergence. The problem described in this study illustrates a successful application of Sinc collocation to the treatment of nonlinear partial differential equations.

REFERENCES

1. F. Stenger, A Sinc-Galerkin Method of Solution of Boundary Value Problems, *Math. Comput.*, vol. 33, pp. 85–109, 1979.
2. F. Stenger, Numerical Methods Based on Whittaker Cardinal, or Sinc Functions, *SIAM Rev.*, vol. 23, pp. 165–224, 1981.
3. F. Stenger, *Numerical Methods Based on Sinc and Analytic Functions*, Springer-Verlag, New York, 1993.

4. J. Lund and K. L. Bowers, *Sinc Methods for Quadrature and Differential Equations*, p. 304, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1992.
5. Y.-C. Jeng, Sinc Interpolation Errors in Finite Data Record Length, *IEEE Instrum. Meas. Technol. Conf.*, vol. 2, pp. 704–707, 1994.
6. T. Schanze, Sinc Interpolation of Discrete Periodic Signals, *IEEE Trans. Signal Process.*, vol. 43, no. 6, pp. 1502–1503, 1995.
7. V. Kober, M. A. Unser, and L. P. Yaroslavsky, Spline and Sinc Signal Interpolations in Image Geometrical Transforms, *Proc. SPIE Int. Soc. Optical Eng.*, vol. 2363, pp. 152–161, 1995.
8. F. Stenger, Sinc Convolution Solution of the Integral Equation Form of Lame's Equations, *ASME Mech. Behav. Adv. Mater.*, vol. 84, pp. 239–242, 1998.
9. B. Bialecki and P. Keast, Sinc Quadrature Subroutine for Cauchy Principal Value Integrals, *J. Comput. Appl. Math.*, vol. 112, pp. 3–20, 1999.
10. K. L. Bowers and J. Lund, Numerical Solution of Singular Poisson Problems via the Sinc-Galerkin Method, *SIAM J. Numer. Anal.*, vol. 24, pp. 36–51, 1987.
11. D. L. Lewis, J. Lund, and K. L. Bowers, Space-Time Sinc-Galerkin Method for Parabolic Problems, *Int. J. Numer. Meth. Eng.*, vol. 24, pp. 1629–1644, 1987.
12. R. C. Smith, G. A. Bogar, K. L. Bowers, and J. Lund, Sinc-Galerkin Method for Fourth-Order Differential Equations, *SIAM J. Numer. Anal.*, vol. 28, pp. 760–788, 1991.
13. R. C. Smith, K. L. Bowers, and J. Lund, Fully Sinc-Galerkin Method for Euler-Bernoulli Beam Models, *Numer. Meth. Partial Differential Equations*, vol. 8, pp. 171–202, 1992.
14. F. Stenger, Matrices of Sinc Methods, *J. Comput. Appl. Math.*, vol. 86, no. 1, pp. 297–310, 1997.
15. F. Stenger and M. J. O'Reilly, Computing Solutions to Medical Problems via Sinc Convolution, *IEEE Trans. Automat. Control*, vol. 43, no. 6, pp. 843–848, 1998.
16. A. C. Morlet, N. J. Lybeck, and K. L. Bowers, The Schwarz Alternating Sinc Domain Decomposition Method, *Appl. Numer. Math.*, vol. 25, no. 4, pp. 461–483, 1997.
17. A. C. Morlet, N. J. Lybeck, and K. L. Bowers, Convergence of the Overlapping Sinc Domain Decomposition Method, *Appl. Math. Comput.*, vol. 98, pp. 209–227, 1999.
18. N. J. Lybeck and K. L. Bowers, Sinc Methods for Domain Decomposition, *Appl. Math. Comput.*, vol. 75, pp. 13–41, 1996.
19. N. J. Lybeck and K. L. Bowers, Domain Decomposition in Conjunction with Sinc Methods for Poisson's Equation, *Numer. Meth. Partial Differential Equations*, vol. 12, pp. 461–487, 1996.
20. S. Narasimhan, K. Chen, and F. Stenger, A Harmonic-Sinc Solution of the Laplace Equation for Problems with Singularities and Semi-Infinite Domains, *Numer. Heat Transfer B*, vol. 33, no. 4, pp. 433–450, 1998.
21. S. Narasimhan, K. Chen, and F. Stenger, On the Solution of the Laplace Equation Using the Harmonic-Sinc Approximation Method, *Proc. Heat Transfer Fluid Mech. Inst.*, pp. 263–275, 1997.
22. A. Lippke, Analytical Solutions and Sinc Function Approximations in Thermal Conduction with Nonlinear Heat Generation, *ASME J. Heat Transfer*, vol. 113, pp. 5–11, 1991.
23. N. Bellomo and L. Ridolfi, Solution of Nonlinear Initial-Boundary Value Problems by Sinc Collocation-Interpolation Methods, *Comput. Math. Appl.*, vol. 29, no. 4, pp. 15–28, 1995.
24. E. Longo, G. Teppati, and N. Bellomo, Discretization of Nonlinear Models by Sinc Collocation-Interpolation Methods, *Comput. Math. Appl.*, vol. 32, no. 4, pp. 65–81, 1996.
25. L. Ridolfi and M. Macis, Identification of Source Terms in Nonlinear Convection Diffusion Phenomena by Sinc Collocation-Interpolation Methods, *Math. Comput. Model.*, vol. 26, no. 2, pp. 69–79, 1997.

26. D. F. Winter, K. L. Bowers, and J. Lund, Wind-Driven Currents in a Sea with Variable Eddy Viscosity Calculated via a Sinc-Galerkin Technique, *Int. J. Numer. Meth. Fluids*, vol. 33, pp. 1041–1073, 2000.
27. *Fluent UNS Theory Manual*, 4.8 ed., Fluent, Inc., Palo Alto, CA, 1998.
28. F. Stenger, *Sincpack—Summary of Basic Sinc Methods*, Department of Computer Science, University of Utah, Salt Lake City, UT, 1995.