

High-Order Finite Difference Schemes for the First Derivative in Von Mises Coordinates

S. O. Alharbi*, M. H. Hamdan

Department of Mathematical Sciences, University of New Brunswick, Saint John, Canada
Email: *hamdan@unb.ca

Received 30 January 2016; accepted 26 March 2016; published 29 March 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Third- and fourth-order accurate finite difference schemes for the first derivative of the square of the speed are developed, for both uniform and non-uniform grids, and applied in the study of a two-dimensional viscous fluid flow through an irregular domain. The von Mises transformation is used to transform the governing equations, and map the irregular domain onto a rectangular computational domain. Vorticity on the solid boundary is expressed in terms of the first partial derivative of the square of the speed of the flow in the computational domain, and the schemes are used to calculate the vorticity at the computational boundary grid points using up to five computational domain grid points. In all schemes developed, we study the effect of coordinate clustering on the computed results.

Keywords

Von Mises Transformation, Non-Uniform Grid, Coordinate Clustering

1. Introduction

Fluid flow through irregular geometries (that is, geometries that do not conform to known coordinate lines) is encountered in many natural and industrial phenomena. Natural occurrences include flow in channels and in rivers over the curvilinear channel bottom, flow into a dam with a curvilinear free-surface, and groundwater and oil flow through the curvilinear earth layers and reservoirs. Industrial applications of flow through irregular domains include flow of air over wings and streamlined bodies, flow in curvilinear ducts, pipes and conduits, flow through turbines and propeller systems, and flow through artificial human and animal tissues, [1]. Solutions to,

*Corresponding author. On leave from Majmaah University, Kingdom of Saudi Arabia.

and analysis of initial and boundary value problems are important in the above-mentioned processes, and many involve solving the Navier-Stokes equations.

The inherent difficulty in solving the Navier-Stokes equations analytically encouraged the use of the more flexible numerical techniques to obtain approximate, yet accurate, solutions to initial and boundary value problems. The method of finite differences proved to be a flexible approach to a large number of flow problems, and has extensively been used due to its simplicity. This method has become an indispensable tool in the numerical treatment of initial and boundary value problems in Mathematical Physics, and its use in the simulation of two- and three-dimensional fluid flow problems is well-documented and frequently reported in the fluid dynamics literature, (cf. [1]-[4] and the references therein). Promotion of the status of finite differences as a means of producing accurate numerical solutions to fluid flow problems in general domains has been advanced by the introduction of grid-generation and the multi-grid method (cf. [1] and the references therein), while its utility in simulating flow in complex and curvilinear geometries has been advanced by the introduction of domain decomposition and coordinate transformation techniques (cf. [1] [2] [4]-[7] and the references therein).

In solving boundary value problems that involve viscous fluid flow in two space dimensions, the Navier-Stokes equations are usually cast in vorticity-streamfunction form. The streamfunction possesses Dirichlet conditions if the solid boundary is assumed to be a streamline of the flow. However, no explicit expression exists for the vorticity on a solid boundary. This forces one to impose a Neumann (derivative) boundary condition on the vorticity. Neumann boundary condition is derived using the definition of the vorticity as the Curl of the velocity vector, [2] [8] [9]. Vorticity at the solid boundary therefore remains a quantity to be solved for. Accurate computation of boundary vorticity has a great influence on the local and global accuracies of the numerical solution of quantities in the flow domain. Furthermore, local accuracy of the computation of boundary vorticity affects the overall global accuracy of the numerical scheme used. This emphasizes the urgent need of accurately computing the vorticity at the boundary [8].

In order to capture the boundary effects more accurately, the use of non-uniform grid with grid clustering at the boundary proved to be a viable tool. Grid refinement has been extensively studied and widely used, and a fairly large number of algebraic and mapping methods have been proposed to produce grid clustering for any given boundary shape, [4] [5] [7] [10] [11]. However, grid clustering also produces unfavourable effects on the convergence and the global accuracy of numerical solution obtained. These effects are due in part to the altering of the coefficient matrix that represents the finite difference form of the governing equations, and the potential loss of diagonal dominance of the coefficient matrix, [8] [11]. Additionally, non-smoothness characteristics of non-uniform grid have been shown to adversely affect the performance of the numerical scheme employed [7].

In order to overcome the above shortcomings, forward and backward finite difference schemes of various orders of local accuracy continue to be developed to better capture the boundary effects (including better approximation to the vorticity on a solid boundary), [12]. A large number of research articles has dealt with the testing of high order (although mainly second- and third-order) schemes when a non-uniform grid is employed (cf. [4], [6] [7] [11] [13] [14] and the references therein). However, there does not seem to exist a consensus as to which scheme or what order is best. The governing Navier-Stokes equations are usually approximated using second-order central differencing schemes. It has been argued (cf. [8] and the references therein) that a higher-order scheme for the vorticity on the solid wall may better approximate the vorticity there, however at the expense of a deteriorating and unstable solution to the governing equations.

The above conclusion has been reached for the case of uniform grid. However, many authors have reached a similar conclusion when a non-uniform grid is used, and some, [14], emphasized that the accuracy of the solution to the governing equations depends on the order of the scheme and on other factors such, as the stretching parameters employed, the error tolerance used in the solution methodology, and the iterative procedure employed. In addition, oscillations in the solutions have been pointed out by Pandit, Kalita and Dalal, [14], who stated that *“For fourth-order approximation of the vorticity at boundary points, some spurious oscillations were observed. When third-order approximation was used for the same, initially slight oscillations were seen.”*

Various authors emphasized that discarding schemes due to their low order of accuracy are *“premature”* when a non-uniform grid is used, (cf. [6] [12] and the references therein), and they argue that low order schemes may *“have a lesser impact on the loss of diagonal dominance”*. Agarwal, [8], expressed a similar point of view for the case of uniform grid, where a *“first-order scheme for the vorticity at a solid wall may have lesser impact on deteriorating the accuracy of the numerical solution to the governing Navier-Stokes equations”*.

In summary, computation of boundary vorticity accurately is tantamount to finding a most suitable finite dif-

ference expression for the first derivative of the tangential velocity component; one that minimizes the unfavourable effects on the coefficient matrix and one that produces the best approximation to the boundary vorticity, whether one employs a uniform or a non-uniform grid.

In the case of fluid flow in curvilinear domains, the domain is first mapped onto a rectangular domain (or a regular domain) where finite differences can be applied. Numerical solution is then sought for transformed governing equations and boundary conditions, [1] [5]-[7] [9] [12] [15] [16]. In particular, the vorticity Neumann boundary condition is transformed into the new coordinate system and a suitable differencing scheme is employed to accurately approximate the boundary vorticity. Numerical approximation to the boundary vorticity is influenced by a number of factors, including:

- The type of grid and spacing used.
- Entry condition (say to a channel).
- The use of flow-field grid points.

In order to shed some light on the above factors, Awartani, Ford and Hamdan, [16], and Siyyam, Ford and Hamdan, [14], considered the flow of a viscous fluid in a two-dimensional curvilinear channel. The curvilinear physical domain is mapped onto a rectangular computational domain using the von Mises coordinates, [1] [5] [15] [16], and the governing equations and boundary conditions are cast in von Mises variables. Vorticity condition on the physical boundary is expressed in terms of the first derivative of the square of the speed of the flow on the boundary of the transformed plane. Their use of the von Mises coordinates reduces the effects of non-smoothness in coordinate clustering. This is due to the fact that when a uniform grid is chosen in the physical domain, use of the von Mises transformation produces a naturally smooth clustered grid in the computational domain. In the above previous work, [12] [16], schemes of first-, second- and third-order local accuracy were developed for the first derivative and used in the calculation of vorticity on the flow domain boundary. In those schemes, stencils involving up to five computational grid points were used, and a comparison was offered between uniform and non-uniform grids. Evaluation of vorticity at a single corner point in the flow domain (where the exact solution is known) demonstrated the superiority of non-uniform grid when the von Mises transformation is used.

The above reported previous work, [5] [16], however, remained silent about the use of schemes of higher-order local accuracy, and the impact of using clustering in the physical domain on the computed vorticity on the boundary. This motivates the current work in which we develop and validate finite difference expressions, for the first derivative, that can be used in the accurate evaluation of vorticity on the flow domain boundary. Ultimately, finite difference expressions that yield the best approximation for the vorticity on the boundary are recommended for implementation in the full numerical solution of the channel flow problem described in this work. The intention is to develop a fourth-order scheme of local accuracy to approximate the vorticity on the boundary for both uniform and non-uniform grids using five grid-point stencils, and to address the following points when transforming a curvilinear physical domain into a rectangular computational domain:

- The type of grid spacing used: Typically, clustered grid near the boundary is expected to produce more accurate results. However, in some coordinate transformation procedures there are factors limiting how “*compact*” the grid points can be. An important question to ask is whether a uniform grid in the computational domain can produce a clustered grid in the physical domain.
- Entry conditions to a channel: In case of a uniform entry condition to a channel, there is a correspondence between a grid in the computational domain and a grid in the physical domain, in the sense that clustering in the physical plane produces the same clustering in the computational domain. However, when a more realistic parabolic entry profile is used, the correspondence is lost and the use of clustered grid becomes necessary in the computational domain. We will consider the effects of a parabolic entry profile when the von Mises transformation is used with both uniform and non-uniform grids.
- Use of interior grid points in the approximation: A question that needs to be answered is how many internal grid points should be used in approximating the vorticity on the boundary, and how far from the boundary can grid points be taken for uniform and clustered grids. This work will demonstrate that a natural order of grid points that are closest to the boundary result in more accurate approximations to boundary vorticity.

In order to present our procedures and report the findings of this study, this work is organized as follows. In the following sections, we present a formulation of the problem of viscous fluid flow in a two-dimensional curvilinear configuration and transform the governing equations into von Mises variables. We present a discretization of the computational domain and derive third and fourth order accurate standard finite difference schemes

for the first derivative and we test these schemes on a uniform grid and four clustered grids.

2. Problem Formulation

2.1. The Physical Domain

We consider a physical problem that gives rise to the need for finite difference expressions for the first derivative. Non-uniform grid arises in a number of ways, including clustering of coordinates and where coordinate transformation is involved. In the current work we consider the von Mises transformation which gives rise to the need for non-uniform grid.

Consider the flow in a two-dimensional, symmetric, dimensionless channel bounded below and above by solid, curvilinear boundaries. The channel is shown in **Figure 1**, and is described by:

$$\{(x, y) | -g(x) \leq y \leq g(x); a \leq x \leq b\} \text{ where } g(x) \text{ is a known smooth function.}$$

The flow of a viscous, incompressible fluid through the channel is governed by the equations of continuity and linear momentum, which take the following dimensionless forms, (cf. [9] [16]):

$$\nabla \cdot \mathbf{v} = 0 \quad (1)$$

$$\mathbf{v} \cdot \nabla \mathbf{v} = -\nabla p + \frac{1}{\text{Re}} \nabla^2 \mathbf{v} \quad (2)$$

wherein, for two-dimensional flow, $\mathbf{v} = (u, v)$ is the velocity vector field, p is the pressure, Re is the Reynolds number, $\nabla \equiv (\partial_x, \partial_y)$, $\nabla^2 \equiv \partial_{xx} + \partial_{yy}$, and subscript notation denotes partial differentiation. Equations (1) and (2) represent a system of three scalar equations that are to be solved for the primitive variables $u(x, y)$, $v(x, y)$, and $p(x, y)$, subject to the no-slip velocity condition on the solid lower and upper boundaries. At the inlet and exit of the channel, the normal velocity component vanishes, that is $v(a, y) = v(b, y) = 0$, while the tangential velocity component, u , can be taken as a function of y (assuming the channel is long enough to allow for this condition to be imposed at the exit).

The dimensionless continuity Equation (1) implies the existence of a dimensionless streamfunction, $\psi(x, y)$, such that:

$$\psi_y = u \quad (3)$$

and

$$\psi_x = -v. \quad (4)$$

Elimination of the pressure term from Equation (2) and the introduction of the dimensionless vorticity $\Omega(x, y)$ that is defined in terms of the velocity components as

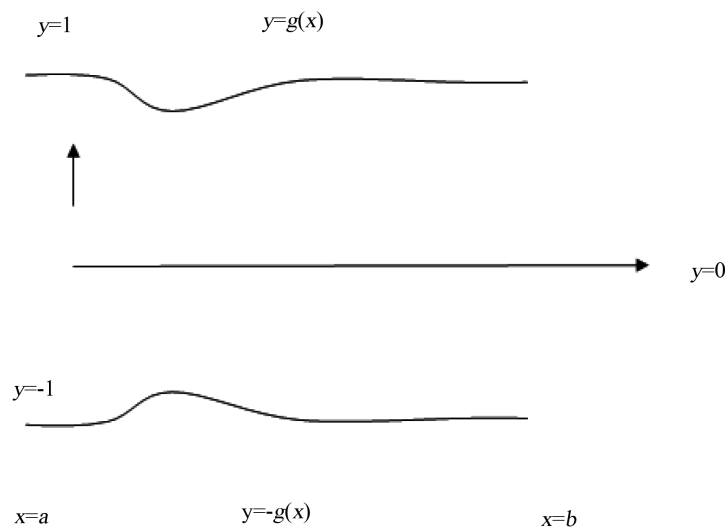


Figure 1. Representative sketch: physical domain.

$$\Omega = \nabla \times \mathbf{v} = v_x - u_y \tag{5}$$

facilitate casting the flow equations in the following vorticity-streamfunction form:

Streamfunction equation:

$$\psi_{xx} + \psi_{yy} = -\Omega \tag{6}$$

Vorticity equation:

$$\Omega_{xx} + \Omega_{yy} = \text{Re}[\psi_y \Omega_x - \psi_x \Omega_y]. \tag{7}$$

In vorticity-streamfunction, the problem has been reduced from that of solving three equations in three primitive variables to that of solving two equations in the two unknowns ψ and Ω .

Appropriate boundary and inlet conditions are the no-slip on the upper and lower walls of the channel, and a parallel, parabolic inlet profile at $x = a$. We assume that the channel is of sufficient length so that inlet and exit conditions are the same, namely:

$$u = 1 - y^2 \quad \text{at } x = a, \tag{8a}$$

$$v = 0 \quad \text{at } x = a, \tag{8b}$$

$$u = v = 0 \quad \text{on } g(x) \text{ and on } -g(x). \tag{8c}$$

The absence of explicit vorticity boundary conditions necessitates imposing appropriate inlet, exit and solid boundary conditions. Assuming that Equation (5) is valid on all boundaries then, with $v_x = 0$, $\Omega = -u_y$ can be used on all boundaries of the configuration at hand (even at the exit of the channel, since the profile of u is assumed to be known there). The above conditions translate into the following conditions on ψ (obtained from Equations (3)-(4), (8a) and (8b)), and Ω (obtained from Equations (5), (8a) and (8b)) along the channel inlet and exit, $x = a, b$:

$$\psi = y - \frac{y^3}{3} \quad \text{for } -1 \leq y \leq 1 \tag{9a}$$

$$\Omega = 2y \quad \text{for } -1 \leq y \leq 1. \tag{9b}$$

Along the lower and upper boundaries, respectively, we have:

$$\psi = -\frac{2}{3} \quad \text{and} \quad \psi = \frac{2}{3} \tag{9c}$$

$$\Omega = -u_y. \tag{9d}$$

In order to solve Equations (6) and (7), subject to conditions (9a)-(9d), numerically using finite differences, the flow domain is first transformed into a rectangular domain using the von Mises coordinates, and the governing equations and boundary conditions are cast in von Mises variables. The von Mises transformation and the transformed governing equations and boundary conditions are discussed in what follows.

2.2. The Computational Domain and Von Mises Transformation

Consider the curvilinear net (x, ψ) in which the curves $\psi = \text{constant}$ represent the streamlines of the flow. Now consider the transformation $(x, y) \rightarrow (x, \psi)$, defined by

$$y = y(x, \psi) \tag{10}$$

The Jacobian of transformation is given by

$$J = \left| \frac{\partial(x, y)}{\partial(x, \psi)} \right| = y_\psi. \tag{11}$$

If $0 < J < \infty$, then the inverse transformation exists and the following first partial derivative operators in the two coordinate systems are obtained, [5] [9] [16]:

$$\partial_x = \partial_x - \frac{y_x}{y_\psi} \partial_\psi \tag{12}$$

$$\partial_y = \frac{1}{y_\psi} \partial_\psi \tag{13}$$

Second partial derivative operators can be obtained by applying operators (12) and (13) onto themselves.

Applying the above transformation to the vorticity-streamfunction Equations (6) and (7), respectively, yields, [9] [10] [16]:

$$L(y) = \Omega(y_\psi)^3 \tag{14}$$

$$L(\Omega) = \Omega \Omega_\psi (y_\psi)^2 + \text{Re } y_\psi \Omega_x \tag{15}$$

where

$$L \equiv (y_\psi)^2 \partial_{xx} - 2y_x y_\psi \partial_{x\psi} + [1 + (y_x)^2] \partial_{\psi\psi} \tag{16}$$

The problem has thus been transformed from that of solving Equations (6) and (7) for $\psi(x, y)$ and $\Omega(x, y)$, subject to conditions (9a)-(9d), in the irregular (x, y) physical plane to that of solving Equations (14) and (15) for $y(x, \psi)$ and $\Omega(x, \psi)$ in the rectangular (x, ψ) computational plane, shown in **Figure 2**, where $\psi = \text{constant}$ represents the streamlines of the flow, subject to the following transformed conditions, [9] [16]:

$$\text{on the lower boundary } y = -g(x) \tag{17a}$$

$$\text{on the upper boundary } y = g(x) \tag{17b}$$

$$\text{at both inlet and exit } y - \frac{y^3}{3} = \psi \text{ for } -\frac{2}{3} \leq \psi \leq \frac{2}{3} \tag{17c}$$

$$\text{at inlet and exit, and on the lower and upper boundaries } \Omega = -\frac{1}{2}(q^2)_\psi \tag{17d}$$

The square of the speed of the flow is given by:

$$q^2(x, \psi) = u^2(x, \psi) + v^2(x, \psi) \tag{18}$$

and the velocity components are expressed as

$$u = \frac{1}{y_\psi}; \quad v = \frac{y_x}{y_\psi} = uy_x \tag{19}$$

Clearly, vorticity condition on the solid upper and lower channel boundary is given by (17d) in terms of the first derivative of the square of the speed of the flow. In the process of solving Equations (14) and (15) numerically, subject to conditions (17), we need to develop finite difference formulas for the first derivative in (17d).

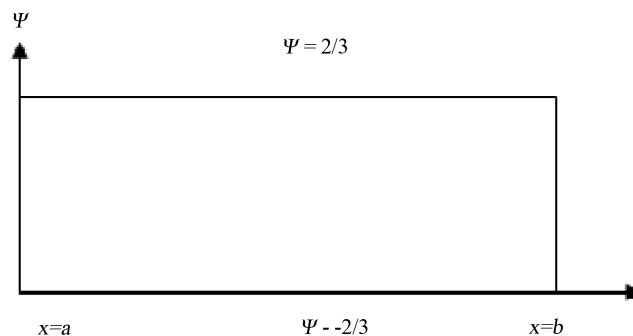


Figure 2. The transformed computational domain.

3. Finite Difference Expressions for the First Derivative

3.1. Discretizing the Flow Domain

The rectangular computational domain of **Figure 2** is discretized using either a uniform grid or a clustered grid, with vertical grid lines ranging from $I = 1$ at $x = a$ to $I = I_{max}$ at $x = b$, and horizontal grid lines ranging from $j = 1$ at the lower computational boundary $\left(\psi = -\frac{2}{3}\right)$ to $j = J_{max}$ at the upper computational boundary $\left(\psi = \frac{2}{3}\right)$.

It has been shown, [16], that if a uniform grid is fitted over the computational domain, the corresponding gridlines in the physical domain would be clustered away from the boundary and closer to the center of the channel. However, in order to capture the effects of boundary conditions more accurately, it is required to have the clustering near the boundary.

Clustering of grid lines near the computational boundary when the von Mises variables are used can be accomplished in two ways:

1) Selecting a uniform grid in the physical domain and calculating the grid spacing in the computational domain results in a “natural” clustering near the computational boundary. For the given parabolic inlet velocity profile, and the indicated range of streamlines, we can choose y to vary over the interval $[-1,1]$, and calculate the step size $\Delta\psi$ between two adjacent grid lines. Suppose that we wish to produce a uniform grid of step size

$\Delta y = 0.01$ in the physical domain, a clustered grid is computed using $y - \frac{y^3}{3} = \psi$ for $-\frac{2}{3} \leq \psi \leq \frac{2}{3}$. The first

three variable grid spacings in the computational domain, corresponding to a uniform grid in the physical domain with $\Delta y = 0.01$, are: $\Delta\psi_1 = 0.000099666$, $\Delta\psi_2 = 0.000297667$, $\Delta\psi_3 = 0.000493667$, where $\Delta\psi_j = \psi_{j+1} - \psi_j$ for $j = 1, 2, 3$. Clearly, the computational grid lines are clustered near the computational boundary, even though the physical grid lines were equally spaced (uniform grid). This points out the fact that the use of the von Mises variables has the added advantage of naturally producing clustered grid near the computational boundaries without resorting to an algebraic method to cluster the grid.

2) Selecting a grid in the physical domain that is non-uniform and clustered near the physical boundary produces a computational grid that is also clustered near the computational boundary.

Physical grid clustering can be accomplished in various ways, including the use of elementary functions. For the current work, we employ the square root, the cubic root and the square functions to define the clustering in the physical y -direction. These are illustrated in **Tables 1-5**, wherein we provide a comparison with the case of uniform grid in the physical domain (**Table 1**) and the case of uniform grid in the computational domain (**Table 5**) with $\Delta\psi = h = 0.01333333$. It should be noted that five grid points are chosen in this work (one boundary point, $j = 1$, and four internal points, $j = 2, 3, 4, 5$) since the development of a fourth-order accurate scheme requires a minimum of five grid points.

3.2. Calculating Square of the Speed at Inlet

At the inlet to the channel, the computational velocity conditions are: $v(x, \psi) = 0$, $u(x, \psi) = 1 - y^2$, for $-1 \leq y \leq 1$. The square of the speed at inlet to the computational domain is given by

$$q^2(x, \psi) = u^2(x, \psi) + v^2(x, \psi) = (1 - y^2)^2. \tag{20}$$

Table 1. Grid 1. Uniform grid in the physical domain $y_j = -y_j^*$ for $y_j^* = 1, 0.99, 0.98, 0.97, 0.96$.

j	y_j	Δy_j	ψ_j	$\Delta\psi_j$
1	$y_1 = -1$	0.01	$\psi_1 = -0.666666666$	$\Delta\psi_1 = 0.000099666$
2	$y_2 = -0.99$	0.01	$\psi_2 = -0.666567000$	$\Delta\psi_2 = 0.000297667$
3	$y_3 = -0.98$	0.01	$\psi_3 = -0.666269333$	$\Delta\psi_3 = 0.000493667$
4	$y_4 = -0.97$	0.01	$\psi_4 = -0.665775666$	$\Delta\psi_4 = 0.000687666$
5	$y_5 = -0.96$	0.01	$\psi_5 = -0.665088000$	

Table 2. Grid 2. Non-uniform grid in the physical domain. $y_j = -\sqrt{y_j^*}$ for $y_j^* = 1, 0.99, 0.98, 0.97, 0.96$.

j	y_j	Δy_j	ψ_j	$\Delta \psi_j$
1	$y_1 = -1$	0.005012567	$\psi_1 = -0.666666666$	$\Delta \psi_1 = 0.000025083$
2	$y_2 = -0.994987437$	0.005037944	$\psi_2 = -0.666641582$	$\Delta \psi_2 = 0.00007559$
3	$y_3 = -0.989949493$	0.005063713	$\psi_3 = -0.666565991$	$\Delta \psi_3 = 0.000126613$
4	$y_4 = -0.98488578$	0.005089883	$\psi_4 = -0.666439377$	$\Delta \psi_4 = 0.000178167$
5	$y_5 = -0.979795897$		$\psi_5 = -0.666261209$	

Table 3. Grid 3. Non-uniform grid in the physical domain. $y_j = -\sqrt[3]{y_j^*}$ for $y_j^* = 1, 0.99, 0.98, 0.97, 0.96$.

j	y_j	Δy_j	ψ_j	$\Delta \psi_j$
1	$y_1 = -1$	0.003344507	$\psi_1 = -0.666666666$	$\Delta \psi_1 = 0.000011173$
2	$y_2 = -0.996655493$	0.003367105	$\psi_2 = -0.666655493$	$\Delta \psi_2 = 0.000033772$
3	$y_3 = -0.993288388$	0.003390089	$\psi_3 = -0.666621721$	$\Delta \psi_3 = 0.000056756$
4	$y_4 = -0.989898299$	0.003413469	$\psi_4 = -0.666564965$	$\Delta \psi_4 = 0.000080136$
5	$y_5 = -0.986484829$		$\psi_5 = -0.666484829$	

Table 4. Grid 4. Non-uniform grid in the physical domain. $y_j = -(y_j^*)^2$ for $y_j^* = 1, 0.99, 0.98, 0.97, 0.96$.

j	y_j	Δy_j	ψ_j	$\Delta \psi_j$
1	$y_1 = -1$	0.0199	$\psi_1 = -0.666666666$	$\Delta \psi_1 = 0.000393383$
2	$y_2 = -0.9801$	0.0197	$\psi_2 = -0.666273283$	$\Delta \psi_2 = 0.001154077$
3	$y_3 = -0.9604$	0.0195	$\psi_3 = -0.665119206$	$\Delta \psi_3 = 0.001876541$
4	$y_4 = -0.9409$	0.0193	$\psi_4 = -0.663242665$	$\Delta \psi_4 = 0.002561929$
5	$y_5 = -0.9216$		$\psi_5 = -0.660680736$	

Table 5. Grid 5. Uniform grid in the computational domain with $\Delta \psi = h = 0.01333333$.

j	ψ_j	y_j
1	$\psi_1 = -0.666666666$	$y_1 = -1$
2	$\psi_2 = -0.653333333$	$y_2 = -0.989949493$
3	$\psi_3 = -0.640000000$	$y_3 = -0.979795897$
4	$\psi_4 = -0.626666667$	$y_4 = -0.969535971$
5	$\psi_5 = -0.613333337$	$y_5 = -0.959166304$

Using (20), we compute $(q^2)_j$ for each of the five types of grid considered, at the first five grid points. These are illustrated in **Table 6**.

Table 6. Square of the speed at inlet to computational domain for the five grids used.

j	q_j^2 Grid1	q_j^2 Grid2	q_j^2 Grid3	q_j^2 Grid4	$q_{i,j}^2$ Grid 5
1	0	0	0	0	0
2	0.00039601	0.0001	0.000044593	0.001552674	0.0491659667
3	0.00156816	0.0004	0.000178975	0.006026702	0.0948030258
4	0.00349281	0.0009	0.000404064	0.013157739	0.1380939921
5	0.00614656	0.0016	0.000720798	0.022696458	0.1794589879

3.3. Derivation of Finite Difference Schemes

The definition of vorticity is expressed at the lower boundary of the computational domain, in von Mises coordinates as the following derivative of the square of the speed of the flow:

$$\Omega_{i,1} = -\frac{1}{2} (q_\psi^2)_{i,1}. \tag{21}$$

In order to obtain finite difference expressions for $\Omega_{i,1}$, we first develop forward finite difference schemes for the first partial derivative on the Right-Hand-Side of Equation (21) using boundary and internal grid points. To accomplish this, we will use the Taylor’s series expansion approach, and discuss both standard and non-standard schemes that can be developed.

In order to derive forward differencing schemes of local accuracy n for the first derivative along the grid line $(i,1)$ using k_{\max} grid points, Siyyam, Ford and Hamdan, [12], assumed the schemes to take the form:

$$(q_\psi^2)_{i,1} + \sum_{j=1}^{k_{\max}} A_j (q^2)_{i,j} = E \tag{22}$$

where E is the local truncation error. The weights, A_j , are calculated using Taylor’s series expansion of $(q^2)_{i,m}$ about point $(i,1)$, for $m = 2, 3, \dots, k_{\max}$, namely

$$(q^2)_{i,m} = \sum_{j=1}^n \frac{(a_m)^{j-1}}{(j-1)!} \partial \psi^{j-1} q^2 \Big|_{(i,1)} + E_{n+1} \tag{23}$$

where

$$E_{n+1} = \sum_{j=n+1}^{\infty} \frac{(a_m)^{j-1}}{(j-1)!} \partial \psi^{j-1} q^2 \Big|_{(i,1)}. \tag{24}$$

The coefficients, a_m , appearing in (23) and (24) are defined by

$$a_m = \sum_{j=1}^{m-1} \Delta \psi_j \tag{25}$$

where

$$\Delta \psi_j = \psi_{j+1} - \psi_j ; \quad j = 1, 2, \dots, k_{\max-1}. \tag{26}$$

For uniform grid, $\Delta \psi_j = h = \text{constant}$ for all j , and (25) takes the form

$$a_m = \sum_{j=1}^{m-1} h = (m-1)h \tag{27}$$

Using (23) in (22), and equating to zero the coefficients of the first n partial derivatives (including the zero’th derivative) of q^2 with respect to ψ , leads to the following $n + 1$ conditions on A_j , $j = 1, 2, \dots, k_{\max}$:

$$\left. \begin{aligned} \sum_{j=1}^{k_{\max}} A_j &= 0 \\ 1 + \sum_{j=2}^{k_{\max}} a_j A_j &= 0 \\ \sum_{j=2}^{k_{\max}} \frac{(a_j)^m}{m!} A_j &= 0; m = 2, 3, \dots, n \end{aligned} \right\} \quad (28)$$

The order of the leading error term, E , will be the coefficient of $\frac{\partial^{n+1} q^2}{\partial \psi^{n+1}}$ is given by:

$$E = \sum_{j=2}^{k_{\max}} \frac{(a_j)^{n+1}}{(n+1)!} A_j. \quad (29)$$

Based on the above description, Siyyam, Ford and Hamdan, [12], observed the following:

1) The relationship between the order of the scheme, n , and the minimum number, k , of grid points used is that $k = n + 1$. This will render system (28) determinate consisting of $n + 1$ equations in as many unknowns. The resulting difference scheme is referred to as a standard scheme. If k_{\max} is the maximum number of grid points used, then $k_{\max} \geq k = n + 1$.

2) The number of standard schemes that use $k = n + 1$ grid points when k_{\max} grid points are employed, is given by the combination: $C(k_{\max}, k) = \frac{(k_{\max})!}{k!(k_{\max} - k)!}$.

3) If $k_{\max} > n + 1$, the number of unknown coefficients will be greater than the number of equations, and the resulting system will have an infinite number of solutions for the coefficients of the differencing scheme. In this case, assigning arbitrary values to the coefficients results in a non-standard or arbitrary scheme. In developing non-standard schemes, grid points are skipped. Skipping grid points results in deterioration of the solution obtained by the scheme.

4) If $k_{\max} < n + 1$, the number of unknowns will be less than the number of equations and the system will have no solution. This implies that in a first-order scheme, $n = 1$, and $k = n + 1 = 2$ is the minimum number of grid points used. Using less than 2 grid points results in no scheme. Similarly, there is no second-order scheme that uses less than 3 grid points, and no third-order schemes that uses less than 4 grid points, and so on.

The above observations are used to construct **Table 7**, below, that gives the number of possible schemes when $k_{\max} = 5$ grid points are employed.

3.4. Derivation of Standard Fourth-Order Accurate Scheme

We employ the procedure above to derive a standard fourth-order accurate scheme (that is, $n = 4$) using five grid points. We thus take $k_{\max} = k = n + 1 = 5$. The scheme will thus employ one boundary point and four internal grid points, namely grid points $j = 1, 2, 3, 4$ and 5. Assuming the scheme to be of the form

Table 7. Number of possible standard schemes corresponding to $k_{\max} = 5$.

Order of standard scheme	Minimum number of grid points required ($k = n + 1$)	Number of possible standard schemes when $k_{\max} = 5$	Number of possible standard schemes when $k_{\max} = 5$, and $j = 1$ is a grid point
First-order ($n = 1$)	2	$C(5,2) = 10$	$C(1,1).C(4,1) = 4$
Second-order ($n = 2$)	3	$C(5,3) = 10$	$C(1,1).C(4,2) = 6$
Third-order ($n = 3$)	4	$C(5,4) = 5$	$C(1,1).C(4,3) = 4$
Fourth order ($n = 4$)	5	$C(5,5) = 1$	$C(1,1).C(4,4) = 1$

$$(q^2)_{i,1} = -A_1(q^2)_{i,1} - A_2(q^2)_{i,2} - A_3(q^2)_{i,3} - A_4(q^2)_{i,4} - A_5(q^2)_{i,5} + E \tag{30}$$

or equivalently

$$(q^2)_{i,1} + \sum_{j=1}^5 A_j (q^2)_{i,j} = E \tag{31}$$

coefficients A_j are selected to satisfy the following:

$$\left. \begin{aligned} \sum_{j=1}^5 A_j &= 0 \\ \sum_{j=2}^5 a_{j-1} A_j &= -1 \\ \sum_{j=2}^5 \frac{(a_{j-1})^m}{m!} A_j &= 0; m = 2, 3, 4 \end{aligned} \right\} \tag{32}$$

The leading local truncation error has the following leading term that is the coefficient of $\frac{\partial^5 q^2}{\partial \psi^5}$:

$$E = \sum_{j=2}^5 \frac{(a_j)^5}{5!} A_j. \tag{33}$$

Equations (32) translate into the following five equations in the five coefficients:

$$\left. \begin{aligned} A_1 + A_2 + A_3 + A_4 + A_5 &= 0 \\ 1 + a_2 A_2 + a_3 A_3 + a_4 A_4 + a_5 A_5 &= 0 \\ \frac{a_2^2}{2} A_2 + \frac{a_3^2}{2} A_3 + \frac{a_4^2}{2} A_4 + \frac{a_5^2}{2} A_5 &= 0 \\ \frac{a_2^3}{6} A_2 + \frac{a_3^3}{6} A_3 + \frac{a_4^3}{6} A_4 + \frac{a_5^3}{6} A_5 &= 0 \\ \frac{a_2^4}{24} A_2 + \frac{a_3^4}{24} A_3 + \frac{a_4^4}{24} A_4 + \frac{a_5^4}{24} A_5 &= 0. \end{aligned} \right\} \tag{34}$$

Solution to this system of 5 equations in 5 unknowns is given by

$$A_1 = \frac{1}{a_2} + \frac{a_4 a_5}{a_3(a_3 - a_4)(a_3 - a_5)} + \frac{a_3 a_5}{a_4(a_4 - a_3)(a_4 - a_5)} + \frac{a_3 a_4}{a_5(a_5 - a_3)(a_5 - a_4)} \tag{35}$$

$$A_2 = -\frac{1}{a_2} - \frac{a_4 a_5}{(a_3 - a_2)(a_3 - a_4)(a_3 - a_5)} - \frac{a_3 a_5}{(a_4 - a_2)(a_4 - a_3)(a_4 - a_5)} - \frac{a_3 a_4}{(a_5 - a_2)(a_5 - a_3)(a_5 - a_4)} \tag{36}$$

$$A_3 = \frac{a_2 a_4 a_5}{a_3(a_3 - a_2)(a_3 - a_4)(a_3 - a_5)} \tag{37}$$

$$A_4 = \frac{a_2 a_3 a_5}{a_4(a_4 - a_2)(a_4 - a_3)(a_4 - a_5)} \tag{38}$$

$$A_5 = \frac{a_2 a_3 a_4}{a_5(a_5 - a_2)(a_5 - a_3)(a_5 - a_4)} \tag{39}$$

where

$$a_2 = \Delta\psi_1 \tag{40}$$

$$a_3 = \Delta\psi_1 + \Delta\psi_2 \tag{41}$$

$$a_4 = \Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 \tag{42}$$

$$a_5 = \Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4. \tag{43}$$

For the four types of non-uniform grids given in **Tables 1-4**, the values of a_2, a_3, a_4, a_5 are summarized in **Table 8**.

Using (40)-(43) in (35)-(39), coefficients A_1, A_2, A_3, A_4, A_5 take the following forms in terms of non-uniform step-sizes, $\Delta\psi_j$:

$$A_1 = \frac{1}{\Delta\psi_1} + \frac{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_3)(\Delta\psi_3 + \Delta\psi_4)} - \frac{(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_3)(\Delta\psi_4)} + \frac{(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_3 + \Delta\psi_4)(\Delta\psi_4)} \tag{44}$$

$$A_2 = -\frac{1}{\Delta\psi_1} - \frac{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_2)(\Delta\psi_3)(\Delta\psi_3 + \Delta\psi_4)} + \frac{(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_2 + \Delta\psi_3)(\Delta\psi_3)(\Delta\psi_4)} - \frac{(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_3 + \Delta\psi_4)(\Delta\psi_4)} \tag{45}$$

$$A_3 = \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_2)(\Delta\psi_3)(\Delta\psi_3 + \Delta\psi_4)} \tag{46}$$

$$A_4 = -\frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_2 + \Delta\psi_3)(\Delta\psi_3)(\Delta\psi_4)} \tag{47}$$

$$A_5 = \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_3 + \Delta\psi_4)(\Delta\psi_4)} \tag{48}$$

For uniform grid with step size h , coefficients A_1, A_2, A_3, A_4, A_5 take the following forms:

$$A_1 = \frac{25}{12h}; A_2 = -\frac{9}{h}; A_3 = \frac{3}{h}; A_4 = -\frac{4}{3h}; A_5 = \frac{1}{4h} \tag{49}$$

and the fourth-order expression for the first derivative takes the form:

Table 8. Values of a_2, a_3, a_4, a_5 for the non-uniform grids used.

Grid #	1	2	3	4
a_2	0.000099666	0.000025083	0.000011173	0.000393383
a_3	0.000397333	0.000100673	0.000044945	0.00154746
a_4	0.000891	0.000227286	0.000101701	0.003424001
a_5	0.001578666	0.000405453	0.000181837	0.00598593

$$(q^2)_{i,1} = \frac{-25(q^2)_{i,1} + 108(q^2)_{i,2} - 36(q^2)_{i,3} + 16(q^2)_{i,4} - 3(q^2)_{i,5}}{12h} + E. \tag{50}$$

Values of the coefficients A_1, A_2, A_3, A_4, A_5 for both uniform and non-uniform grids are given in **Table 9**, below.

3.5. Standard Third-Order Accurate Schemes

In order to arrive at a third-order ($n = 3$) locally accurate forward differencing scheme for the first derivative, Awartani, Ford and Hamdan, [16], used the above procedure and employed four grid points ($k_{\max} = 4$) in a scheme of the form given by (22). Their derivation produced one standard scheme that uses grid points $j = 1, 2, 3, 4$. Their scheme is reproduced here for completeness and we derive other third-order accurate standard schemes.

Letting

$$(q^2)_{i,1} + A_1(q^2)_{i,1} + A_2(q^2)_{i,2} + A_3(q^2)_{i,3} + A_4(q^2)_{i,4} = E \tag{52}$$

then the coefficients satisfy the following conditions:

$$\left. \begin{aligned} A_1 + A_2 + A_3 + A_4 &= 0 \\ 1 + a_2A_2 + a_3A_3 + a_4A_4 &= 0 \\ \frac{a_2^2}{2}A_2 + \frac{a_3^2}{2}A_3 + \frac{a_4^2}{2}A_4 &= 0 \\ \frac{a_2^3}{6}A_2 + \frac{a_3^3}{6}A_3 + \frac{a_4^3}{6}A_4 &= 0. \end{aligned} \right\} \tag{52}$$

Equations (52) represent a determinate system consisting of 4 equations in 4 unknowns that has the following unique solution:

$$\left. \begin{aligned} A_1 &= \frac{a_4(a_2 + a_3) + a_2a_3}{a_2a_3a_4} \\ A_2 &= \frac{-a_3a_4}{a_2(a_4 - a_2)(a_3 - a_2)} \\ A_3 &= \frac{a_2a_4}{a_3(a_3 - a_2)(a_4 - a_3)} \\ A_4 &= \frac{-a_2a_3}{a_4(a_4 - a_2)(a_4 - a_3)}. \end{aligned} \right\} \tag{53}$$

where

Table 9. Values of coefficients A_1, A_2, A_3, A_4, A_5 for the five grids used; fourth order accurate scheme.

Grid	A_1	A_2	A_3	A_4	A_5
Grid 1	14,306.07323	-16,095.95974	2032.465715	-261.1816983	18.60248878
Grid 2	56,687.98222	-63,619.31368	7871.374257	-987.5683431	86.5377706
Grid 3	12,783.0685	-19,849,012,084	17,520.61174	-2180.649425	150.0190806
Grid 4	3647.395614	-142,573.05	543.2005178	-72.92902426	5.475543678
Grid 5 uniform	156.2500391	-300.000075	225.0000563	-100.000025	18.75000469

$$\left. \begin{aligned} a_2 &= \Delta\psi_1 \\ a_3 &= \Delta\psi_1 + \Delta\psi_2 \\ a_4 &= \Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 \end{aligned} \right\} \quad (54)$$

and

$$\Delta\psi_j = \psi_{j+1} - \psi_j; j = 1, 2, 3, 4. \quad (55)$$

The leading error term, $E = \sum_{j=2}^4 \frac{(a_j)^4}{4!} A_j$, is the coefficient of $\frac{\partial^4 q^2}{\partial \psi^4}$.

The four standard, third-order accurate schemes that use the four grid points $j = 1, 2, 3$ and 4 , are as follows:

(1,2,3,4) Scheme:

Solution (53), above, provides the following expression that was initially derived by Awartani, Ford and Hamdan, [3]:

$$\begin{aligned} (q^2)_{i,1} &= - \left\{ \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2) + (\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)(2\Delta\psi_1 + \Delta\psi_2)}{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)} \right\} (q^2)_{i,1} \\ &+ \left\{ \frac{(\Delta\psi_1 + \Delta\psi_2) + (\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{\Delta\psi_1\Delta\psi_2(\Delta\psi_2 + \Delta\psi_3)} \right\} (q^2)_{i,2} \\ &- \left\{ \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{\Delta\psi_2\Delta\psi_3(\Delta\psi_1 + \Delta\psi_2)} \right\} (q^2)_{i,3} \\ &+ \left\{ \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)}{\Delta\psi_3(\Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)} \right\} (q^2)_{i,4} + E \end{aligned} \quad (56)$$

where

$$\begin{aligned} E &= \left\{ \frac{-\Delta\psi_1^3(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{24\Delta\psi_2(\Delta\psi_2 + \Delta\psi_3)} \right. \\ &+ \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)^3(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{24\Delta\psi_2\Delta\psi_3} \\ &\left. - \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)^3}{24\Delta\psi_3(\Delta\psi_2 + \Delta\psi_3)} \right\} (q^2_{\psi\psi\psi\psi})_{i,1} + \dots \end{aligned} \quad (57)$$

For uniform grid, (57) yields the following scheme:

$$(q^2)_{i,1} = \frac{2(q^2)_{i,4} - 9(q^2)_{i,3} + 18(q^2)_{i,2} - 11(q^2)_{i,1}}{6h} - \frac{h^3}{4} (q^2_{\psi\psi\psi\psi})_{i,1} + \dots \quad (58)$$

Values of the coefficients A_1, A_2, A_3, A_4 for both uniform and non-uniform grids are given in **Table 10**, below.

Other standard third-order accurate schemes obtained in this work employ grid points $(1, \alpha, \beta, \gamma)$ and are of

Table 10. Values of coefficients of the (1,2,3,4) scheme.

Grid	A_1	A_2	A_3	A_4
Grid 1	13,672.62702	-15,079.7727	1520.916705	-113.7705973
Grid 2	54,200.53239	-59,683.49924	5916.931052	-433.9642054
Grid 3	121,583.6377	-133,812.6289	13,190.00854	-961.00854
Grid 4	3480.328141	-3850.98425	401.9421534	-31.26164889
Grid 5 uniform	137.5000344	-225.0000563	112.5000281	-25.00000625

the form given by (52), with coefficients $A_1, A_\alpha, A_\beta, A_\gamma$. These are as follows:

(1,2,3,5) Scheme:

$$\begin{aligned}
 (q_\psi^2)_{i,1} = & - \left\{ \frac{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(2\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3) + \Delta\psi_1(\psi_1 + \Delta\psi_2)}{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_1 + \Delta\psi_2)} \right\} q_{i,1}^2 \\
 & + \left\{ \frac{(\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_1(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_2)} \right\} q_{i,2}^2 \\
 & - \left\{ \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_1 + \Delta\psi_2)\Delta\psi_2(\Delta\psi_3 + \Delta\psi_4)} \right\} q_{i,3}^2 \\
 & + \left\{ \frac{\Delta\psi_1(\psi_1 + \Delta\psi_2)}{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_4 + \Delta\psi_3)} \right\} q_{i,5}^2 + E
 \end{aligned} \tag{59}$$

where

$$\begin{aligned}
 E = & \frac{1}{24} \left\{ \frac{-\Delta\psi_1^3(\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_2 + \Delta\psi_3)} \right. \\
 & + \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)^3(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_2(\Delta\psi_3 + \Delta\psi_4)} \\
 & \left. - \frac{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)^3 \Delta\psi_1(\psi_1 + \Delta\psi_2)}{(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_3 + \Delta\psi_4)} \right\} q_{\psi\psi\psi\psi}^2 + \dots
 \end{aligned} \tag{60}$$

For uniform grid, (59) yields the following scheme:

$$(q_\psi^2)_{i,1} = \frac{-21q_{i,1}^2 + 32q_{i,2}^2 - 12q_{i,3}^2 + q_{i,5}^2}{12h} - \frac{h^3}{4} (q_{\psi\psi\psi\psi}^2)_{i,1} + \dots \tag{61}$$

Values of the coefficients A_1, A_2, A_3, A_5 for both uniform and non-uniform grids are given in **Table 11**, below.

(1,2,4,5) Scheme:

$$\begin{aligned}
 (q_\psi^2)_{i,1} = & - \left\{ \frac{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(2\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3) + \Delta\psi_1(\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)} \right\} q_{i,1}^2 \\
 & + \left\{ \frac{(\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_1(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_3 + \Delta\psi_2)} \right\} q_{i,2}^2 \\
 & - \left\{ \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_2\Delta\psi_4(\Delta\psi_2 + \Delta\psi_3)} \right\} q_{i,4}^2 \\
 & + \left\{ \frac{\Delta\psi_1(\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)\Delta\psi_4} \right\} q_{i,5}^2 + E
 \end{aligned} \tag{62}$$

where

Table 11. Values of coefficients of the (1,2,3,5) scheme.

Grid	A_1	A_2	A_3	A_5
Grid 1	13,183.73878	-14,295.48845	1126.106905	-14.35723799
Grid 2	52,267.16645	-56,598.30806	4384.864483	-53.72287539
Grid 3	117,250.3235	-126,909.795	9777.680061	-118.2085627
Grid 4	3355.33057	-3648.30425	297.0706287	-4.09648027
Grid 5 uniform	131.2500328	-200.00005	75.00001875	-6.250001563

$$E = \frac{1}{24} \left\{ \frac{-\Delta\psi_1^3 (\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_3 + \Delta\psi_2)} + \frac{\Delta\psi_1 (\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)^3 (\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_4 (\Delta\psi_2 + \Delta\psi_3)} + \frac{-\Delta\psi_1 (\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)^3 (\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{(\Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4) \Delta\psi_4} \right\} (q_{\psi\psi\psi\psi}^2)_{i,1} + \dots \tag{63}$$

For uniform grid, (62) yields the following scheme:

$$(q_{\psi}^2)_{i,1} = \frac{-19q_{i,1}^2 + 24q_{i,2}^2 - 8q_{i,4}^2 + 3q_{i,5}^2}{12h} - \frac{h^3}{4} (q_{\psi\psi\psi\psi}^2)_{i,1} + \dots \tag{64}$$

Values of the coefficients A_1, A_2, A_4, A_5 for both uniform and non-uniform grids are given in **Table 12**, below.

(1,3,4,5) Scheme:

$$(q_{\psi}^2)_{i,1} = - \left\{ \frac{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(2\Delta\psi_1 + 2\Delta\psi_2 + \Delta\psi_3) + (\Delta\psi_1 + \Delta\psi_2)(\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{(\Delta\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)} \right\} q_{i,1}^2 + \left\{ \frac{(\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_3 (\psi_1 + \Delta\psi_2)(\Delta\psi_3 + \Delta\psi_4)} \right\} q_{i,3}^2 - \left\{ \frac{(\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_3 \Delta\psi_4 (\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)} \right\} q_{i,4}^2 + \left\{ \frac{(\psi_1 + \Delta\psi_2)(\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{((\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)(\Delta\psi_2 + \Delta\psi_3) \Delta\psi_4)} \right\} q_{i,5}^2 + E \tag{65}$$

where

$$E = \frac{1}{24} \left\{ \frac{(\psi_1 + \Delta\psi_2)^3 (\psi_1 + \Delta\psi_2 + \Delta\psi_3)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_3 (\Delta\psi_3 + \Delta\psi_4)} - \frac{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3)^3 (\psi_1 + \Delta\psi_2)(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)}{\Delta\psi_3 \Delta\psi_4} + \frac{(\Delta\psi_1 + \Delta\psi_2 + \Delta\psi_3 + \Delta\psi_4)^3 (\psi_1 + \Delta\psi_2)(\psi_1 + \Delta\psi_2 + \Delta\psi_3)}{(\Delta\psi_2 + \Delta\psi_3) \Delta\psi_4} \right\} (q_{\psi\psi\psi\psi}^2)_{i,1} + \dots \tag{66}$$

For uniform grid, (65) yields the following scheme:

$$(q_{\psi}^2)_{i,1} = \frac{-13q_{i,1}^2 + 36q_{i,3}^2 - 32q_{i,4}^2 + 9q_{i,5}^2}{12h} - \frac{h^3}{4} (q_{\psi\psi\psi\psi}^2)_{i,1} + \dots \tag{67}$$

Table 12. Values of coefficients of the (1,2,4,5) scheme.

Grid	A_1	A_2	A_4	A_5
Grid 1	11789.2926	-12058.4901	324.5056047	-55.30810146
Grid 2	46733.75961	-47768.30974	1242.031037	-207.4808993
Grid 3	104833.6526	-107130.427	2753.697604	-456.9231722
Grid 4	3001.166302	-3074.036981	88.57579336	-15.70511441
Grid 5 uniform	118.7500297	-150.0000375	50.0000125	-18.75000469

Values of the coefficients A_1, A_3, A_4, A_5 for both uniform and non-uniform grids are given in **Table 13**, below.

3.6. Standard First- and Second-Order Accurate Schemes

Six standard, second-order accurate schemes have been developed by Awartani, Ford and Hamdan, [16], using combinations of grid points on the boundary and the four grid points closest to the boundary. Henceforth, the notation $(1, \alpha, \beta, \gamma)$ -scheme denotes the scheme that employs points on the grid lines $j = 1, \alpha, \beta, \gamma$. For the sake of comparison of results, we list below the natural order (1,2,3)-scheme that produced the most accurate results in [3]:

(1,2,3) Scheme:

$$(q_\psi^2)_{i,1} = -\frac{(2\Delta\psi_1 + \Delta\psi_2)}{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)}q_{i,1}^2 + \frac{(\Delta\psi_1 + \Delta\psi_2)}{\Delta\psi_1\Delta\psi_2}q_{i,2}^2 - \frac{\Delta\psi_1}{(\Delta\psi_1 + \Delta\psi_2)\Delta\psi_2}q_{i,3}^2 + E \tag{68}$$

where

$$E = \left\{ \frac{\Delta\psi_1(\Delta\psi_1 + \Delta\psi_2)}{3!} \right\} (q_{\psi\psi\psi}^2)_{i,1} + \dots \tag{69}$$

For uniform grid, (3.49) reduces to the following second-order accurate formula:

$$(q_\psi^2)_{i,1} = \frac{-3q_{i,1}^2 + 4q_{i,2}^2 - q_{i,3}^2}{2h} + \frac{h^2}{3}(q_{\psi\psi\psi}^2)_{i,1} + \dots \tag{70}$$

Combinations of standard first-order schemes that use the two grid points $(1, \beta)$ have been derived by Siyyam, Ford and Hamdan, [12], and take the following general form for $\beta = 2, 3, 4, 5$ (that is, when $k_{\max} = 5$ grid points are used):

$$(q_\psi^2)_{i,1} = \frac{(q^2)_{i,\beta} - (q^2)_{i,1}}{a_\beta} - \left(\frac{a_\beta}{2} \right) (q_{\psi\psi}^2)_{i,1} + \dots \tag{71}$$

For the sake of comparison of results, we list below the natural order (1,2)-scheme that produced the most accurate results in [12]:

(1,2)-Scheme:

$$(q_\psi^2)_{i,1} = \frac{(q^2)_{i,2} - (q^2)_{i,1}}{\Delta\psi_1} - \left(\frac{\Delta\psi_1}{2} \right) (q_{\psi\psi}^2)_{i,1} + \dots \tag{72}$$

For uniform grid, (72) reduces to:

$$(q_\psi^2)_{i,1} = \frac{(q^2)_{i,2} - (q^2)_{i,1}}{h} - \frac{h}{2}(q_{\psi\psi}^2)_{i,1} + \dots \tag{73}$$

Table 13. Values of coefficients of the (1,3,4,5) scheme.

Grid	A_1	A_3	A_4	A_5
Grid 1	4272.561304	-6070.25437	2073.745892	-276.0528255
Grid 2	16,799.27007	-23,721.13304	7961.140282	-1039.277311
Grid 3	37,581.59177	-52,958.56974	17,668.47142	-2291.493454
Grid 4	1105.334666	-1590.210946	562.7195912	-77.84331141
Grid 5 uniform	81.25002031	-225.0000563	200.00005	-56.25001406

4. Results and Discussion

An expression for the vorticity on the lower computational channel boundary is given by $\Omega_{i,1} = -\frac{1}{2}(q_w^2)_{i,1}$. The lower, left-corner vorticity expression is given by $\Omega_{1,1} = -\frac{1}{2}(q_w^2)_{1,1}$, with exact value of this corner vorticity $\Omega_{1,1} = -2$, as obtained from the channel inlet condition $\Omega = 2y$ where $y = -1$.

In order to approximate $\Omega_{1,1}$, we approximate the first derivative of the square of the speed, $(q_w^2)_{1,1}$, using the schemes developed in this work. These values are computed on the uniform and non-uniform grids given in **Tables 1-5**, using the square of the speed values of **Table 6**.

In the following Tables, we list the computed values of $\Omega_{1,1}$ together with the *Percentage Relative Error (PRE)* defined as:

$$PRE = \frac{\Omega_{1,1} |\text{Exact} - \Omega_{1,1}|_{\text{Computed}}}{\Omega_{1,1} |\text{Exact}} \times 100\% . \quad (74)$$

Comparison of the schemes that use the natural order of grid points in conjunction with the five grids used, is given in **Table 14**. This table demonstrates that the fourth order accurate scheme produces a corner vorticity that is closest to the exact value, with a Percentage Relative Error of 0.125% when Grid 3 is used. The fourth order accurate scheme consistently produces lower Percentage Relative Error for all of the grids employed.

Standard, first-order accurate scheme of Equation (72) produces the results in **Table 14**, below, which shows that using Grid 3 produces the most accurate computed values of vorticity and least Percentage Relative Errors. This may be ascribed to the fact that Grid 3 uses grid points that are closest to the boundary, hence better able to capture boundary effects. Grid 5 (uniform grid) produces the least accurate computed values of vorticity and largest Percentage Relative Error. Contrasted with clustered grid, the uniform Grid 5 has its largest step-size (grid spacing) closest to the boundary, which explains the large Percentage Relative Error associated with it.

Standard, second-order accurate scheme that uses the natural order of grid points produces results illustrated in **Table 14** which, consistently, emphasizes the superiority of the (1,2,3) scheme when using Grid 3. The Percentage Relative Error associated with this combination is 0.146%. The four, third-order accurate schemes produce results reported in **Table 15**, and echo similar conclusions to the superiority of the natural-order scheme and the use of Grid 3.

Table 14 and **Table 15** support the following conclusions:

- 1) The Percentage Relative Error decreases with increasing accuracy of the scheme employed, for both variable and uniform grids.
- 2) Grids with lowest Percentage Relative Errors for all standard schemes are, respectively: Grid 3, Grid 2, Grid 1, Grid 4, Grid 5.
- 3) Schemes that employ the natural order of grid points produce smaller Percentage Relative Errors, for all grids employed.

Table 14. Comparison of values of $\Omega_{1,1}$ and pre using standard first, second, third and fourth-order accurate schemes of natural order of grid points.

Grid	Scheme			
	1 st order (1,2) scheme	2 nd order (1,2,3) scheme	3 rd order (1,2,3,4) scheme	4 th order (1,2,3,4,5) scheme
Grid 1	-1.9867357 (0.666%)	-1.991132154 (0.443%)	-1.9920387 (0.398%)	-1.992431358 (0.378%)
Grid 2	-1.993381972 (0.331%)	-1.995622482 (0.218%)	-1.996072644 (0.196%)	-1.996257356 (0.187%)
Grid 3	-1.995569677 (0.221%)	-1.997066733 (0.146%)	-1.997366868 (0.131%)	-1.997498503 (0.125%)
Grid 4	-1.973488941 (1.325%)	-1.982419735 (0.879%)	-1.984135078 (0.793%)	-1.98505092 (0.747%)
Grid 5 (uniform)	-1.843726 (7.813%)	-1.909891241 (4.509%)	-1.9246767643 3.766%	-1.931826679 (3.408%)

4) Using clustered Grid 3 produces the lowest Percentage Relative Error, for all standard schemes used.

5. Generalization of the Schemes to Computational Flow Field Grid Points

Forward-difference schemes that approximate the derivative at a typical grid point (i,j) in the computational flow domain, using up-to 5 grid points, employ the gridlines $j, j + 1, j + 2, j + 3, j + 4$, shown in **Figure 3**.

The standard forward-difference uniform and non-uniform schemes that are based on the natural order of grid lines are listed below for the sake of completeness and ease of reference.

$(j, j + 1)$ -Scheme: first order

For non-uniform grid:

$$(q^2)_{i,j} \approx \frac{(q^2)_{i,j+1} - (q^2)_{i,j}}{\Delta\psi_j} \tag{75}$$

For uniform grid:

$$(q^2)_{i,j} \approx \frac{(q^2)_{i,j+1} - (q^2)_{i,j}}{h} \tag{76}$$

$(j, j + 1, j + 2)$ Scheme: second order

For non-uniform grid:

$$(q^2)_{i,j} \approx -\frac{(2\Delta\psi_j + \Delta\psi_{j+1})}{\Delta\psi_j(\Delta\psi_j + \Delta\psi_{j+1})}q_{i,j}^2 + \frac{(\Delta\psi_j + \Delta\psi_{j+1})}{\Delta\psi_j\Delta\psi_{j+1}}q_{i,j+1}^2 - \frac{\Delta\psi_j}{(\Delta\psi_j + \Delta\psi_{j+1})\Delta\psi_{j+1}}q_{i,j+2}^2 \tag{77}$$

For uniform grid:

$$(q^2)_{i,j} \approx \frac{-3q_{i,j}^2 + 4q_{i,j+1}^2 - q_{i,j+2}^2}{2h} \tag{78}$$

$(j, j + 1, j + 2, j + 3)$ Scheme: third order

Table 15. Values of $\Omega_{1,1}$ and PRE using standard third-order accurate schemes.

Scheme Grid	(1,2,3,4) scheme	(1,2,3,5) scheme	(1,2,4,5) scheme	(1,3,4,5) scheme
Grid 1	-1.9920387 (0.398%)	-1.9917441 (0.412%)	-1.990900404 (0.454%)	-1.986352479 (0.682%)
Grid 2	-1.996072644 (0.196%)	-1.995920807 (0.203%)	-1.99548624 (0.225%)	-1.99313533 (0.343%)
Grid 3	-1.997366868 (0.131%)	-1.997266348 (0.136%)	-1.996973186 (0.151%)	-1.995385341 (0.231%)
Grid 4	-1.984135078 (0.793%)	-1.983628605 (0.818%)	-1.981979149 (0.901%)	-1.973188713 (1.341%)
Grid 5 uniform	-1.92467676435 (3.766%)	-1.92229302 (3.885%)	-1.917526191 (4.124%)	-1.903225703 (4.838%)

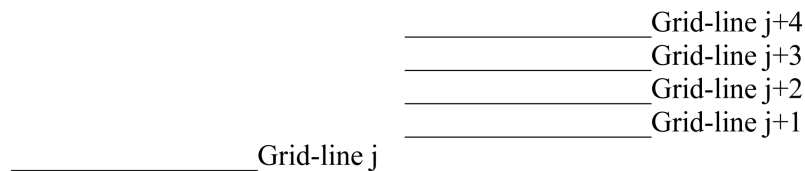


Figure 3. Grid lines $j, j + 1, j + 2, j + 3, j + 4$, used in forward-difference schemes.

For non-uniform grid:

$$\begin{aligned}
 (q^2)_{i,j} \approx & - \left\{ \frac{\Delta\psi_j (\Delta\psi_j + \Delta\psi_{j+1}) + (\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})(2\Delta\psi_j + \Delta\psi_{j+1})}{\Delta\psi_j (\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})} \right\} (q^2)_{i,j} \\
 & + \left\{ \frac{(\Delta\psi_j + \Delta\psi_{j+1}) + (\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})}{\Delta\psi_j \Delta\psi_{j+1} (\Delta\psi_{j+1} + \Delta\psi_{j+2})} \right\} (q^2)_{i,j+1} \\
 & - \left\{ \frac{\Delta\psi_j (\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})}{\Delta\psi_{j+1} \Delta\psi_{j+2} (\Delta\psi_j + \Delta\psi_{j+1})} \right\} (q^2)_{i,j+2} \\
 & + \left\{ \frac{\Delta\psi_j (\Delta\psi_j + \Delta\psi_{j+1})}{\Delta\psi_{j+2} (\Delta\psi_{j+1} + \Delta\psi_{j+2})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})} \right\} (q^2)_{i,j+3}
 \end{aligned} \tag{79}$$

For uniform grid:

$$(q^2)_{i,j} \approx \frac{2(q^2)_{i,j+3} - 9(q^2)_{i,j+2} + 18(q^2)_{i,j+1} - 11(q^2)_{i,j}}{6h}. \tag{80}$$

(j, j + 1, j + 2, j + 3, j + 4) Scheme: fourth order

For non-uniform grid:

$$(q^2)_{i,j} \approx -A_1 (q^2)_{i,j} - A_2 (q^2)_{i,j+1} - A_3 (q^2)_{i,j+2} - A_4 (q^2)_{i,j+3} - A_5 (q^2)_{i,j+4}. \tag{81}$$

where

$$\begin{aligned}
 A_1 = & \frac{1}{\Delta\psi_j} + \frac{(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})}{(\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_{j+2})(\Delta\psi_{j+2} + \Delta\psi_{j+3})} \\
 & - \frac{(\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})}{(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})(\Delta\psi_{j+2})(\Delta\psi_{j+3})} \\
 & + \frac{(\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})}{(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})(\Delta\psi_{j+2} + \Delta\psi_{j+3})(\Delta\psi_{j+3})}
 \end{aligned} \tag{82}$$

$$\begin{aligned}
 A_2 = & -\frac{1}{\Delta\psi_j} - \frac{(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})}{(\Delta\psi_{j+1})(\Delta\psi_{j+2})(\Delta\psi_{j+2} + \Delta\psi_{j+3})} \\
 & + \frac{(\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})}{(\Delta\psi_{j+1} + \Delta\psi_{j+2})(\Delta\psi_{j+2})(\Delta\psi_{j+3})} \\
 & - \frac{(\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})}{(\Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})(\Delta\psi_{j+2} + \Delta\psi_{j+3})(\Delta\psi_{j+3})}
 \end{aligned} \tag{83}$$

$$A_3 = \frac{\Delta\psi_j (\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})}{(\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_{j+1})(\Delta\psi_{j+2})(\Delta\psi_{j+2} + \Delta\psi_{j+3})} \tag{84}$$

$$A_4 = -\frac{\Delta\psi_j ((\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+1} + \Delta\psi_{j+3}))}{(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})(\Delta\psi_{j+1} + \Delta\psi_{j+2})(\Delta\psi_{j+2})(\Delta\psi_{j+3})} \tag{85}$$

$$A_5 = \frac{\Delta\psi_j (\Delta\psi_j + \Delta\psi_{j+1})(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2})}{(\Delta\psi_j + \Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})(\Delta\psi_{j+1} + \Delta\psi_{j+2} + \Delta\psi_{j+3})(\Delta\psi_{j+2} + \Delta\psi_{j+3})(\Delta\psi_{j+3})} \tag{86}$$

For uniform grid:

For uniform grid with step size h , coefficients A_1, A_2, A_3, A_4, A_5 take the following forms:

$$A_1 = \frac{25}{12h}; A_2 = -\frac{9}{h}; A_3 = \frac{3}{h}; A_4 = -\frac{4}{3h}; A_5 = \frac{1}{4h} \quad (87)$$

and the fourth-order expression for the first derivative takes the form:

$$(q^2)_{i,j} \approx \frac{-25(q^2)_{i,j} + 108(q^2)_{i,j+1} - 36(q^2)_{i,j+2} + 16(q^2)_{i,j+3} - 3(q^2)_{i,j+4}}{12h}. \quad (88)$$

6. Conclusions

The main theme of this work has been the development and testing of numerical schemes for the first derivative using uniform and non-uniform grids. The schemes have been applied to the computation of corner vorticity in the case of viscous fluid flow through a two-dimensional curvilinear channel that has been mapped onto a rectangular computational domain using the von Mises transformation.

The following has been accomplished in this work.

- 1) A fourth-order accurate scheme that uses 5 grid points has been developed and tested.
- 2) Third-order accurate schemes not previously reported have been developed and tested in this work.

Results have clearly demonstrated that:

- 1) Non-uniform grids produce more accurate values of corner vorticity than uniform grids.
- 2) The clustered grid that consistently produced the best approximation to corner vorticity is Grid 3, which is based on a stretching of physical coordinates using the cubic root function. This produced the smallest grid-spacing closest to the boundary in both the physical and computational domains.
- 3) The schemes that produced the best approximation to corner vorticity are the standard schemes which employ the natural order of grid points.

Acknowledgements

S. O. Alharbi, gratefully acknowledges support for this work through his PhD scholarship from *Majmaah University, Kingdom of Saudi Arabia*.

References

- [1] Hamdan, M.H. (1997) Natural Coordinate System Approach to Coupled n-Phase Fluid Flow in Curved Domains. *Applied Mathematics and Computation*, **85**, 297-304. [http://dx.doi.org/10.1016/S0096-3003\(96\)00144-0](http://dx.doi.org/10.1016/S0096-3003(96)00144-0)
- [2] Fletcher, C.A.J. (1988) Computational Techniques for Fluid Dynamics. Vol. II, Chapter 17, Springer-Verlag. <http://dx.doi.org/10.1007/978-3-642-97071-9>
- [3] Gupta, M.M. (1983) A Survey of Some Second-Order Difference Schemes for the Steady State Convection-Diffusion Equation. *International Journal for Numerical Methods in Fluids*, **3**, 319-331. <http://dx.doi.org/10.1002/flid.1650030403>
- [4] Kalnay de Rivas, E. (1972) On the Use of Non-Uniform Grids in Finite-Difference Equations. *Journal of Computational Physics*, **10**, 202-210. [http://dx.doi.org/10.1016/0021-9991\(72\)90060-5](http://dx.doi.org/10.1016/0021-9991(72)90060-5)
- [5] Barron, R.M. (1989) Computation of Incompressible Potential Flow Using von Mises Coordinates. *Mathematics and Computers in Simulation*, **31**, 177-188. [http://dx.doi.org/10.1016/0378-4754\(89\)90157-2](http://dx.doi.org/10.1016/0378-4754(89)90157-2)
- [6] Spatz, W.F. and Carey, G.F. (1998) Formulation and Experiments with High-Order Compact Schemes for Non-Uniform Grids. *International Journal of Numerical Methods for Heat & Fluid Flow*, **8**, 288-303. <http://dx.doi.org/10.1108/09615539810206357>
- [7] Veldman, A.E.P. and Rinzema, K. (1992) Playing with Non-Uniform Grids. *Journal of Engineering Mathematics*, **26**, 119-130. <http://dx.doi.org/10.1007/BF00043231>
- [8] Agarwal, R.K. (1981) A Third-Order-Accurate Upwind Scheme for Navier-Stokes Solutions in Three Dimensions. In: *Computers in Flow Predictions and Fluid Dynamics Experiments, Proceedings of the Winter Annual Meeting*, Washington DC, 15-20 November 1981, 73-82.
- [9] Alderson, T.L., Allan, F.M. and Hamdan, M.H. (2007) On the Universality of the von Mises Transformation. *International Journal of Applied Mathematics*, **20**, 109-121.

-
- [10] Hamdan, M.H. (2009) Recent Developments in the von Mises Transformation and Its Applications in the Computational Sciences: Plenary Lecture. In: Perlovsky, L., *et al.*, Eds., *Mathematical Methods, System Theory and Control*, MAMECTIS09, 180-189.
- [11] Shukla, R.K., Shukla and Zhong, X. (2005) Derivation of High-Order Compact Finite Difference Schemes for Non-Uniform Grid Using Polynomial Interpolation. *Journal of Computational Physics*, **204**, 404-429. <http://dx.doi.org/10.1016/j.jcp.2004.10.014>
- [12] Siyyam, H.I., Ford, R.A. and Hamdan, M.H. (2009) First-Order Accurate Finite Difference Schemes for Boundary Vorticity Approximations in Curvilinear Geometries. *Applied Mathematics and Computation*, **215**, 2378-2387. <http://dx.doi.org/10.1016/j.amc.2009.08.028>
- [13] Castillo, J.E., Hyman, J.M., Shashkov, J.M. and Steinberg, S. (1995) The Sensitivity and Accuracy of Fourth Order Finite-Difference Schemes on Non-Uniform Grids in One Dimension. *Computers and Mathematics with Applications*, **30**, 41-55. [http://dx.doi.org/10.1016/0898-1221\(95\)00136-M](http://dx.doi.org/10.1016/0898-1221(95)00136-M)
- [14] Pandit, S.K., Kalita, J.C. and Dalal, D.C. (2008) A Fourth Order Accurate Compact Scheme for the Solution of Steady Navier-Stokes Equations on Non-Uniform Grids. *Computers and Fluids*, **37**, 121-134. <http://dx.doi.org/10.1016/j.compfluid.2007.04.002>
- [15] Visbal, M.R. and Gaitonde, D.V. (2002) On the Use of Higher-Order Finite Difference Schemes on Curvilinear and Deforming Meshes. *Journal of Computational Physics*, **181**, 155-185. <http://dx.doi.org/10.1006/jcph.2002.7117>
- [16] Awartani, M.M., Ford, R.A. and Hamdan, M.H. (2005) Computational Complexities and Streamfunction Coordinates. *Applied Mathematics and Computation*, **169**, 758-777. <http://dx.doi.org/10.1016/j.amc.2004.09.088>