

High-order finite difference schemes for incompressible flows

H. Fadel^{1,*}, M. Agouzoul¹, P.K. Jimack²

¹ ERD3M, Département Mécanique, Ecole Mohammadia d'Ingénieurs, BP765 Rabat-Agdal Morocco

² School of Computing, University of Leeds, Leeds LS2 9JT UK

SUMMARY

This paper presents a new high order approach to the numerical solution of the incompressible Stokes and Navier-Stokes equations. The class of schemes developed is based upon a velocity-pressure-pressure gradient formulation which allows: (i) high order finite difference stencils to be applied on non-staggered grids; (ii) high order pressure gradient approximations to be made using standard Padé schemes, and; (iii) a variety of boundary conditions to be incorporated in a natural manner. Results are presented in detail for a selection of two-dimensional steady-state test problems, using the fourth order scheme to demonstrate the accuracy and robustness of the proposed methods. Furthermore, extensions to higher orders and time-dependent problems are illustrated, whilst the extension to three-dimensional problems is also discussed. Copyright © 2009 John Wiley & Sons, Ltd.

KEY WORDS: *Incompressible flow; Steady flow; Velocity-Pressure-Pressure gradient formulation; Finite difference methods; High-order accuracy; Padé schemes*

1. INTRODUCTION

Over recent years, the development of high-order accurate schemes has become an attractive approach to improving the efficiency of numerical computations for more and more complex problems. Indeed, these schemes can be used either to reduce the computational costs for a given accuracy by reducing the mesh size or to increase the accuracy for a fixed mesh size.

The motivation for this work is to propose a new class of high-order finite difference schemes for steady flows of incompressible Newtonian fluids governed by Stokes or Navier-Stokes equations. In particular, our goal is to obtain high-order accurate solutions for both the velocity and the pressure, as well as the pressure derivatives, by choosing an appropriate formulation, without the need for any other approximations than the discretization of the equations governing the flow modelled. Although the numerical schemes and examples described here are all in two dimensions, the extension to three dimensional cases is straightforward.

*Correspondence to: fadel@emi.ac.ma

Contract/grant sponsor: Morocco CNRST; contract/grant number: a1/038

Contract/grant sponsor: Marie Curie; contract/grant number: MEST-CT-2005-020327

The problem can be stated as follows. Let Ω be the problem domain in \mathfrak{R}^2 , then the classical steady-state Navier-Stokes (NS) equation system in primitive variable form is written as:

$$-\nu \nabla^2 \mathbf{u} + \frac{1}{\rho} \nabla p + (\mathbf{u} \cdot \nabla) \mathbf{u} = \mathbf{f} \quad (1a)$$

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

where $\mathbf{u} = (u, v)^T$ denotes the velocity vector, p the static pressure field, ν the kinematic viscosity, ρ the fluid density, $\mathbf{f} = (f_x, f_y)^T$ an external force vector and $\mathbf{x} = (x, y)^T$ the Cartesian coordinate variable. We refer to equations (1a) as the momentum equations and to equation (1b) as the continuity equation.

To specify a unique solution, we consider the NS problem (1) together with the appropriate boundary conditions on $\partial\Omega = \partial\Omega_D \cup \partial\Omega_N$:

$$\mathbf{u} = \mathbf{u}_D \quad \text{on} \quad \partial\Omega_D, \quad (2a)$$

$$\nu \partial_n u_n - p = f_n \quad \text{on} \quad \partial\Omega_N, \quad (2b)$$

$$\nu \partial_n u_\tau = f_\tau \quad \text{on} \quad \partial\Omega_N. \quad (2c)$$

Here the subscripts D and N denote a boundary of Dirichlet and Neumann type respectively, n is the normal component and τ the tangential one. Equation (2a) is referred to as a Dirichlet boundary condition and equations (2b) and (2c) as open outflow boundary conditions.

Solving the system (1) under the boundary conditions (2), in the given form, using finite difference methods on a single grid leads to spurious pressure solutions. To avoid this a range of different approaches have been developed. The approach that we adopt in this work is to consider a velocity-pressure formulation obtained by replacing the incompressibility condition by a pressure Poisson equation. This equation is derived by taking the divergence of the momentum equations to obtain a consistent pressure equation and using the continuity equation to eliminate the divergence of the velocity to obtain a further simplification. A detailed discussion on this subject and related issues can be found in [1, 2]. We can now reformulate the problem by replacing the continuity equation in the Navier-Stokes system (1) by the following simplified pressure equation:

$$\nabla^2 p = \rho(-\nabla \cdot (\mathbf{u} \cdot \nabla \mathbf{u}) + \nabla \cdot \mathbf{f}). \quad (3)$$

Knowing that :

$$\nabla \cdot (\mathbf{u} \cdot \nabla \mathbf{u}) = \nabla \mathbf{u} : \nabla \mathbf{u} + u \cdot \nabla (\nabla \cdot \mathbf{u}) \quad (4)$$

and further use of the zero velocity divergence leads to the following form [3]:

$$\nabla^2 p = \rho(2(\partial_x u \partial_y v - \partial_y u \partial_x v) + \nabla \cdot \mathbf{f}). \quad (5)$$

In order to complete the mathematical formulation, it is necessary to specify appropriate boundary conditions for the pressure. This is complicated by the fact that, *a priori*, we have no explicit information about the pressure at the boundary. Nevertheless, such a condition may be obtained by taking the scalar product of the momentum equations with the unit normal vector \mathbf{n} at the boundary. On its own this is not sufficient however since the momentum equations are already satisfied on the boundary, so these conditions do not in themselves add any new information and the system remains undetermined. Moreover, the velocity-pressure formulation can not be equivalent to the primitive variable one without enforcing

the divergence-free condition on the boundaries [1, 4, 5, 6]. Hence, by using this condition and the following identity:

$$\nabla^2 \mathbf{u} = \nabla(\nabla \cdot \mathbf{u}) - \nabla \times \nabla \times \mathbf{u}, \quad (6)$$

the derived curl-curl consistent boundary condition for the pressure is:

$$\partial_n p = \rho \mathbf{n} \cdot (-\nu \nabla \times \nabla \times \mathbf{u} - (\mathbf{u} \cdot \nabla) \mathbf{u} + \mathbf{f}) \quad \text{on} \quad \partial\Omega_N. \quad (7)$$

Therefore, and when properly normalised, our reference problem is now expressed by:

$$-\frac{1}{Re} \nabla^2 \mathbf{u} + \nabla p + (\mathbf{u} \cdot \nabla) \mathbf{u} = \mathbf{f} \quad (8a)$$

$$\nabla^2 p = 2(\partial_x u \partial_y v - \partial_y u \partial_x v) + \nabla \cdot \mathbf{f}, \quad (8b)$$

with the appropriate boundary conditions:

$$\mathbf{u} = \mathbf{u}_D, \quad \text{on} \quad \partial\Omega_D, \quad (9a)$$

$$\frac{1}{Re} \partial_n u_n - p = f_n \quad \text{on} \quad \partial\Omega_N, \quad (9b)$$

$$\frac{1}{Re} \partial_n u_\tau = f_\tau \quad \text{on} \quad \partial\Omega_N, \quad (9c)$$

$$\partial_n p = \mathbf{n} \cdot \left(-\frac{1}{Re} \nabla \times \nabla \times \mathbf{u} - (\mathbf{u} \cdot \nabla) \mathbf{u} + \mathbf{f} \right) \quad \text{on} \quad \partial\Omega_N. \quad (9d)$$

Here, the Reynolds number is defined by $Re = U_0 L_0 / \nu$ for some characteristic velocity U_0 and length scale L_0 .

Now that the mathematical formulation is set, we propose to investigate a velocity-pressure-pressure gradient formulation to ensure a high order accuracy of the obtained results, especially for the pressure. This formulation is simply defined by the fact that the pressure gradient is considered as unknown in the previous velocity-pressure formulation. In order to approximate these additional variables we use the Padé schemes [7, 8, 9, 10], often referred to in the literature as compact schemes. Shih *et al.*, [11], proposed a similar approach (scheme (i) in [11]): using the primitive variable formulation but with the velocity and pressure first derivatives also considered as unknowns and approximated using a Padé 4th order scheme within the domain and a 4th order one-sided Taylor approximation on the boundaries. In our approach, we study the use of the 4th order Padé scheme only for the pressure derivative and the same order Padé non-central scheme on the boundaries.

An outline of the remainder of this paper is as follows. In sections 2 and 3 we introduce the numerical methods together with the solution strategy for Stokes and Navier-Stokes equations respectively. In section 4 we then discuss results obtained for different test cases. Finally, our conclusions and perspectives on future work are presented in section 5.

2. NUMERICAL METHODS FOR STOKES EQUATIONS

2.1. Discretization

In the particular case where it is possible to assume that convection effects may be safely neglected and $Re \rightarrow 0$, the problem (8-9) may be reformulated to the following Stokes

equations:

$$-\nabla^2 \mathbf{u} + \nabla p = \mathbf{f} \quad (10a)$$

$$\nabla^2 p = \nabla \cdot \mathbf{f}, \quad (10b)$$

together with the appropriate boundary conditions:

$$\mathbf{u} = \mathbf{u}_D, \quad \text{on } \partial\Omega_D, \quad (11a)$$

$$\partial_n u_n - p = f_n \quad \text{on } \partial\Omega_N, \quad (11b)$$

$$\partial_n u_\tau = f_\tau \quad \text{on } \partial\Omega_N, \quad (11c)$$

$$\partial_n p = \mathbf{n} \cdot (-\nabla \times \nabla \times \mathbf{u} + \mathbf{f}) \quad \text{on } \partial\Omega_N. \quad (11d)$$

To discretize the system's momentum equations (10a) we express them as Poisson equations, this allows the system (10) to be written as:

$$\nabla^2 w = g \quad \text{on } \Omega \quad (12)$$

where

$$w = \begin{pmatrix} u \\ v \\ p \end{pmatrix} \quad \text{and} \quad g = \begin{pmatrix} \partial_x p - f_x \\ \partial_y p - f_y \\ \partial_x f_x + \partial_y f_y \end{pmatrix}. \quad (13)$$

The solution w and the forcing function g are assumed to be sufficiently smooth and have the required continuous partial derivatives.

The solution of equation (12) can be approximated using the standard fourth-order scheme derived by Rosser [12, 13] and represented here by the molecular display:

$$\begin{bmatrix} 1 & b & 1 \\ a & -20 & a \\ 1 & b & 1 \end{bmatrix} w = c \begin{bmatrix} 0 & 1 & 0 \\ 1 & 8 & 1 \\ 0 & 1 & 0 \end{bmatrix} g, \quad (14)$$

with

$$a = 2 \frac{(5\Delta y^2 - \Delta x^2)}{\Delta x^2 + \Delta y^2}, \quad b = 2 \frac{(5\Delta x^2 - \Delta y^2)}{\Delta x^2 + \Delta y^2}, \quad c = \frac{(\Delta x^2 \Delta y^2)}{\Delta x^2 + \Delta y^2}$$

such that Δx and Δy are the grid size in the x -direction and the y -direction respectively.

The last point remaining to complete the discretization of our reference problem is the treatment of the Neumann boundary conditions (11b), (11c) and (11d). These include first order, second order and mixed second order partial derivatives that are approximated using standard Taylor series, consistent with the order of our scheme (see appendix A for full details).

2.2. Padé Schemes

As stated previously, the schemes developed in this work are based on a velocity-pressure-pressure gradient formulation. To define the additional equations required for the pressure gradient, that is considered as unknown, we propose to consider the standard Padé schemes.

For a function f , these schemes are defined by :

$$\beta f'_{i-2} + \alpha f'_{i-1} + f'_i + \alpha f'_{i+1} + \beta f'_{i+2} = c \frac{f_{i+3} - f_{i-3}}{6h} + b \frac{f_{i+2} - f_{i-2}}{4h} + a \frac{f_{i+1} - f_{i-1}}{2h}, \quad (15)$$

such that i indexes the nodes uniformly dividing a mesh of size h , and f'_i is an approximation of the first derivative of f at the node i . The coefficients a , b , c , α and β (note that a , b , and c are different from the previous subsection) are derived by developing all the variables in the Taylor series for the mesh node i and requiring all the coefficients of the resulting expansion to vanish up to the required order. The first non-zero coefficient sets the formal truncation error, and hence the order of accuracy, of the scheme [10].

To obtain fourth-order accuracy, the relation between these coefficients is given by:

$$a + 2^2b + 3^2c = 2 \frac{3!}{2!} (\alpha + 2^2\beta) . \quad (16)$$

Here we are interested in the family of tridiagonal 4th order schemes that are defined by:

$$\beta = 0, \quad a = \frac{2}{3}(\alpha + 2), \quad b = \frac{1}{3}(4\alpha - 1), \quad c = 0. \quad (17)$$

By choosing $\alpha = 1/4$, we recover the classical Padé scheme cited by Collatz [8]:

$$f'_{i-1} + 4f'_i + f'_{i+1} = \frac{3}{h}(-f_{i-1} + f_{i+1}), \quad (18)$$

with a leading truncation error of $(2/5!)h^4 f^{(5)}$. However, this scheme is central and therefore it can only be applied inside the domain Ω . We therefore need a boundary closure formula for nodes on the boundary. Such a 4th order forward formula, at a grid node $i = 0$, is given by:

$$f'_i + 3f'_{i+1} = \frac{1}{6h}(-17f_i + 9f_{i+1} + 9f_{i+2} - f_{i+3}), \quad (19)$$

with a leading truncation error of $(6/5!)h^4 f^{(5)}$.

2.3. Solution strategy

It is now clear that our problem contains five unknowns, rather than three, at each non-Dirichlet node. These are the velocity components u and v , the pressure field p and the pressure derivatives $\partial_x p$ and $\partial_y p$. To summarise, the scheme developed for the Stokes flow is based on the following discretization strategy.

- Velocity solution:
 - In Ω : 9-point Poisson's Formula (14)
 - On $\partial\Omega_D$: \mathbf{u} values treated as known values
 - On $\partial\Omega_N$: Taylor approximations for u and v derivatives.
- Pressure solution:
 - In Ω : 9-point Poisson's Formula (14)
 - On $\partial\Omega_D$: one p value specified to avoid the hydrostatic pressure mode
 - On $\partial\Omega_N$: Taylor approximations for u , v and p derivatives.
- Pressure gradient solution:
 - In Ω : Standard Padé scheme (18)
 - On $\partial\Omega$: Padé boundary closure (19).

The list of unknowns and difference equations at each node is detailed in appendix A. Due to the linearity of the system of discretized equations they may be written in matrix form. This leads to an unsymmetric matrix and, in the case of a problem with Dirichlet BC for the velocity components and Newmann BC for the pressure, the system takes the following form:

$$Ax = \begin{bmatrix} A_1 & 0 & 0 & A_2 & 0 \\ 0 & A_1 & 0 & 0 & A_2 \\ A_3 & A_4 & A_5 & 0 & 0 \\ 0 & 0 & A_6 & A_7 & 0 \\ 0 & 0 & A_8 & 0 & A_9 \end{bmatrix} \begin{pmatrix} u \\ v \\ p \\ \partial_x p \\ \partial_y p \end{pmatrix} = F. \quad (20)$$

The sparsity patterns of each of the sub-matrices A_1 to A_9 are plotted for a mesh size $[10 \times 10]$ in figure 1.

3. NUMERICAL METHODS FOR NAVIER-STOKES EQUATIONS

3.1. Discretization

For the Navier-Stokes problem (8-9), as for the Stokes case, we consider the momentum equations in the Poisson equation form (18), where

$$w = \begin{pmatrix} u \\ v \\ p \end{pmatrix} \quad \text{and} \quad g = g_1 + g_2, \quad (21)$$

with

$$g_1 = \begin{pmatrix} Re(\partial_x p - f_x) \\ Re(\partial_y p - f_y) \\ \partial_x f_x + \partial_y f_y \end{pmatrix} \quad \text{and} \quad g_2 = \begin{pmatrix} Re(u\partial_x u + v\partial_y u) \\ Re(u\partial_x v + v\partial_y v) \\ 2(\partial_y u\partial_x v - \partial_x u\partial_y v) \end{pmatrix}. \quad (22)$$

Here g_1 is equal to g in (13) up to a constant but, due to the nonlinearity of g_2 , the discretization is not so straightforward as for the Stokes problem. Details of the linearization techniques used are therefore considered in the next subsection.

3.2. Linearization Techniques

3.2.1. Picard method Picard iteration, also known as successive substitution, is a simple method to implement and typically converges from a large basin of attraction. It consists of taking one of the vectors u from the previous iteration level whilst the other is taken at the new level. This allows us to rewrite the vector g_2 in (22) as:

$$g_2 = \begin{pmatrix} Re(u^k \partial_x u^{k+1} + v^k \partial_y u^{k+1}) \\ Re(u^k \partial_x v^{k+1} + v^k \partial_y v^{k+1}) \\ 2(\partial_y u^k \partial_x v^{k+1} - \partial_x u^{k+1} \partial_y v^k) \end{pmatrix}, \quad (23)$$

where k denotes the previous iteration level and $k + 1$ the new one. A similar linearization is applied to the Neumann boundary condition for the pressure (9d):

$$\partial_n p = \mathbf{n} \cdot (-\nabla \times \nabla \times \mathbf{u}^{k+1} - (\mathbf{u}^k \cdot \nabla) \mathbf{u}^{k+1} + f). \quad (24)$$

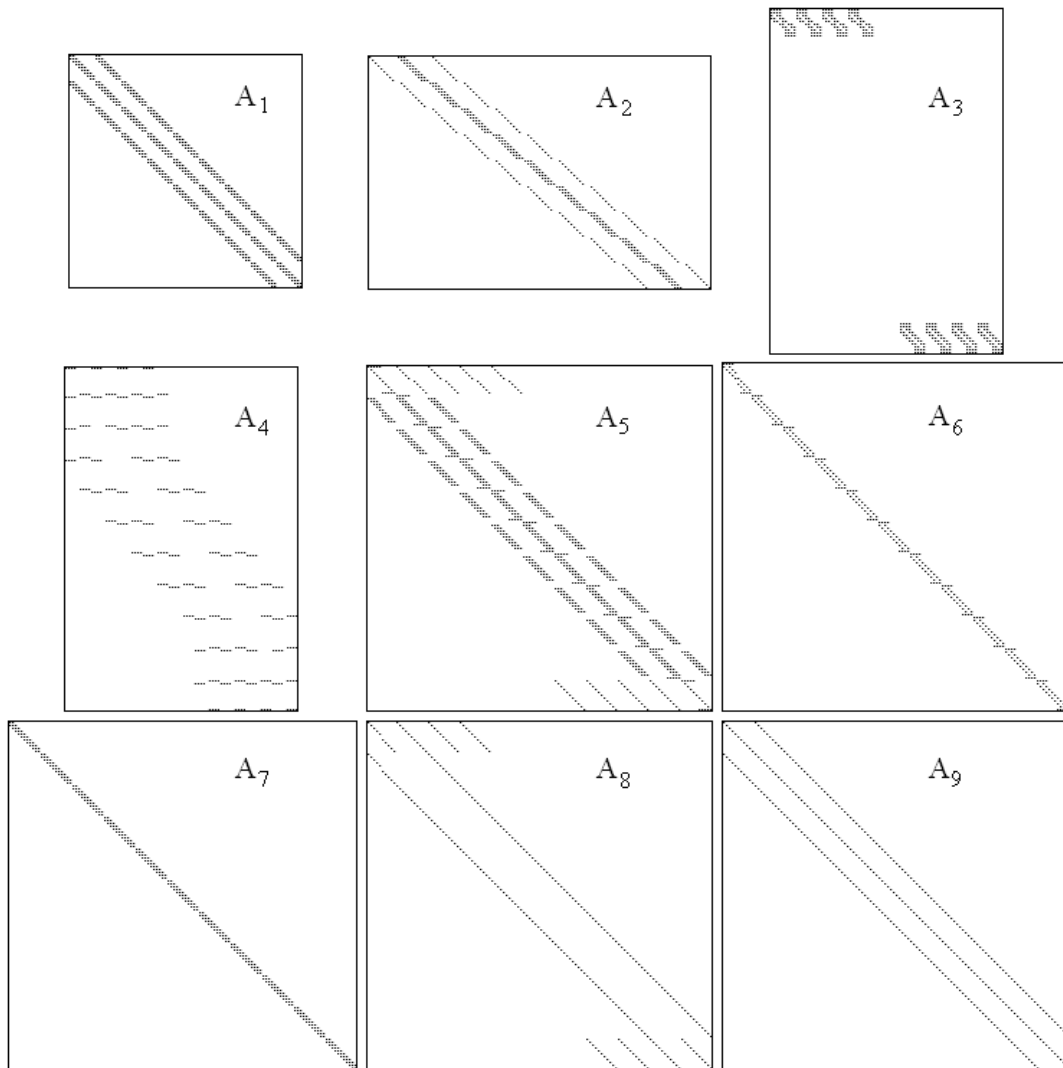


Figure 1. Sparsity patterns of the sub-matrices in the system (20)

Unfortunately, however, the convergence of the Picard method is generally linear and typically rather slow. Since it has good convergence properties it is often used to provide a suitable initial guess for the Newton method, which is introduced in the next section, and which converges much more rapidly provided a sufficiently good initial iterate is used.

3.2.2. Newton method Newton's method is another well-known procedure for the solution of nonlinear systems of algebraic equations. It generally achieves quadratic convergence when the current iteration value \mathbf{u}^k is sufficiently close to the true solution \mathbf{u} . The Newton iteration

process is now described.

Assuming that the system to be solved is of the form

$$\mathbf{N}(\mathbf{u}) = 0, \quad (25)$$

the first step consist of evaluating the Jacobian \mathbf{J}^k of \mathbf{N} at \mathbf{u}^k :

$$\mathbf{J}^k = \frac{\partial \mathbf{N}(\mathbf{u}^k)}{\partial \mathbf{u}}. \quad (26)$$

The linearised system to be solved is then given by:

$$\mathbf{J}^k \delta \mathbf{u}^k = -\mathbf{N}(\mathbf{u}^k), \quad (27)$$

and the updated solution is

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \delta \mathbf{u}^k. \quad (28)$$

For the finite difference approach described above, the linearization is obtained using a Taylor series expansions in two variables such that the series are truncated after the first derivative terms. This is illustrated as follows for an arbitrary function $f(x_1, x_2)$:

$$f(x_1^{k+1}, x_2^{k+1}) \approx f(x_1^k, x_2^k) + \partial_{x_1} f(x_1^k, x_2^k)(x_1^{k+1} - x_1^k) + \partial_{x_2} f(x_1^k, x_2^k)(x_2^{k+1} - x_2^k) \quad (29)$$

$$\approx f(x_1^{k+1}, x_2^k) + f(x_1^k, x_2^{k+1}) - f(x_1^k, x_2^k). \quad (30)$$

Applying this approach to each of the nonlinear terms in the vector g_2 in (22) leads to:

$$g_2 = g_{21} + g_{22}, \quad (31)$$

where

$$g_{21} = \begin{pmatrix} Re(u^k \partial_x u^{k+1} + u^{k+1} \partial_x u^k + v^k \partial_y u^{k+1} + v^{k+1} \partial_y u^k) \\ Re(u^k \partial_x v^{k+1} + u^{k+1} \partial_x v^k + v^k \partial_y v^{k+1} + v^{k+1} \partial_y v^k) \\ 2(\partial_y u^k \partial_x v^{k+1} + \partial_y u^{k+1} \partial_x v^k - \partial_x u^{k+1} \partial_y v^k - \partial_x u^k \partial_y v^{k+1}) \end{pmatrix} \quad (32)$$

and

$$g_{22} = - \begin{pmatrix} Re(u^k \partial_x u^k + v^k \partial_y u^k) \\ Re(u^k \partial_x v^k + v^k \partial_y v^k) \\ 2(\partial_x u^k \partial_y v^k - \partial_y u^k \partial_x v^k) \end{pmatrix}. \quad (33)$$

Finally, applying the same approach to the Neumann boundary condition for the pressure (9d) leads to:

$$\partial_n p = \mathbf{n} (-\nabla \times \nabla \times \mathbf{u} - (\mathbf{u}^k \cdot \nabla) \mathbf{u}^{k+1} - (\mathbf{u}^{k+1} \cdot \nabla) \mathbf{u}^k + (\mathbf{u}^k \cdot \nabla) \mathbf{u}^k + f). \quad (34)$$

3.3. Solution strategy

Three different solutions strategies, based upon the above linearization schemes, have been adopted in this work. For small Reynolds number the Newton method is applied, with an initial guess typically obtained by the solution to a corresponding Stokes problem. If this fails to converge then a hybrid method consisting of few Picard iterations as an initial guess for the Newton method is then applied.

For high Reynolds numbers even this strategy may be very inefficient and so an approach based upon numerical continuation is adopted. Here, in order to obtain an initial guess that is sufficiently close to the true solution for the Newton algorithm to converge, the problem (21) is solved with:

$$g = g_1 + \lambda g_2 \quad (35)$$

for gradually increasing values of λ such that $0 \leq \lambda \leq 1$.

4. NUMERICAL RESULTS

In this section, we present a selection of numerical results obtained using the schemes developed above for the solution of four typical test problems. The first two problems are selected to illustrate the numerical properties of the solver based upon known analytical solutions. It should be noted that for even simpler test problems with low order polynomial solutions, such as classical Poiseuille flow or the colliding flow suggested in [14], our solution schemes demonstrated superconvergent properties that recovered the exact solution on all meshes.

The second pair of test cases are selected to represent more realistic fluid mechanics applications characterized, from a physical point of view, by simple geometry and complex physics. The first problem represents a confined flow with Dirichlet BCs for the velocity, namely a leaky lid-driven cavity at $Re=100, 400$ and 1000 . This case exhibits some interesting physical features, specifically singularities at the cavity top corners due to discontinuous BCs, and a recirculation zone driven by the moving top wall. The second problem is an inflow/outflow test case, that is, a backward-facing step at $Re=800$. This flow is physically characterized by two recirculation zones, which requires appropriate outflow BCs in order to be successfully simulated.

In all four cases, the iterative solver used was the restarted GMRES [15]: with the restart dimension set to 30 and using only a simple diagonal preconditioner. This is clearly far from an optimal preconditioning strategy and its improvement will be the subject of further research, along with the use of alternative solvers for non-symmetric matrices.

4.1. Stokes flow test case

As a first test case for the Stokes flow, we chose a mathematical problem [16] with a known exact solution in order to assess the performance and the accuracy of the scheme developed.

Let $\Omega = [0, 1] \times [0, 1]$ be the flow domain and the exact solution given by:

$$u = 2\pi x^2(1-x)^2 \sin(\pi y) \cos(\pi y) \quad (36a)$$

$$v = -2x(x-1)(2x-1) \sin^2(\pi y) \quad (36b)$$

$$p = \sin(x) \cos(y) . \quad (36c)$$

Based upon this solution, the right-hand side of the system to be solved (10) is given by:

$$f_x = 2\pi(-1 + 6x + 2(\pi^2 - 3)x^2 - 4\pi^2 x^3 + 2\pi^2 x^4) \sin(2\pi y) + \cos(x) \cos(y) \quad (37a)$$

$$f_y = 4\pi^2 x(1 - 3x + 2x^2) \cos(2\pi y) - 12(1 - 2x) \sin(\pi y)^2 - \sin(x) \sin(y) \quad (37b)$$

$$\nabla \cdot f = 2 \sin(x) \cos(y) . \quad (37c)$$

$n_x \times n_y$	$\nabla \cdot \mathbf{u}$	Number of iterations
20×20	$8.14e^{-10}$	31
40×40	$5.23e^{-10}$	96
80×80	$1.13e^{-9}$	310
160×160	$6.32e^{-10}$	1023
320×320	$1.07e^{-9}$	2001

Table I. Grid refinement, continuity and the number of linear iterations for the Stokes test case (36)

$n_x \times n_y$	$\ E_u\ _\infty$	Order u	$\ E_v\ _\infty$	Order v
20×20	$1.24 \cdot 10^{-4}$	-	$1.24 \cdot 10^{-4}$	-
40×40	$7.58 \cdot 10^{-6}$	3.98	$7.84 \cdot 10^{-6}$	3.98
80×80	$4.95 \cdot 10^{-7}$	3.99	$4.88 \cdot 10^{-7}$	4.00
160×160	$3.11 \cdot 10^{-8}$	3.99	$3.07 \cdot 10^{-8}$	3.99
320×320	$1.95 \cdot 10^{-9}$	4.00	$1.92 \cdot 10^{-9}$	4.00

Table II. Accuracy order of velocity components for the Stokes test case (36)

$n_x \times n_y$	$\ E_p\ _\infty$	Order p	$\ E_{\partial_x p}\ _\infty$	Order $\partial_x p$	$\ E_{\partial_y p}\ _\infty$	Order $\partial_y p$
20×20	$6.27 \cdot 10^{-3}$	-	$3.76 \cdot 10^{-2}$	-	$3.07 \cdot 10^{-2}$	-
40×40	$3.88 \cdot 10^{-4}$	4.01	$3.26 \cdot 10^{-3}$	3.53	$2.71 \cdot 10^{-3}$	3.50
80×80	$2.41 \cdot 10^{-5}$	4.01	$2.83 \cdot 10^{-4}$	3.52	$3.37 \cdot 10^{-4}$	3.51
160×160	$1.51 \cdot 10^{-6}$	4.00	$2.43 \cdot 10^{-5}$	3.54	$2.07 \cdot 10^{-5}$	3.52
320×320	$9.39 \cdot 10^{-8}$	4.01	$2.09 \cdot 10^{-6}$	3.54	$1.81 \cdot 10^{-6}$	3.52

Table III. Accuracy order of pressure field and derivatives for the Stokes test case (36)

The boundary conditions for this test case are constrained to those given by the solution (36). The values of \mathbf{u} are prescribed on all boundaries while the hydrostatic pressure mode is eliminated by fixing the pressure value at a single grid node. The tolerance of GMRES iteration is set to 10^{-14} (which is clearly excessively small however our goal here is not to assess the efficiency of the linear solver, rather the efficiency of the discretization scheme) and in table I we present the resulting maximum continuity check along with the number of linear iterations required on each grid used. As is typical in the absence of problem-specific preconditioning the iteration count approximately doubles each time the mesh spacing is halved.

In the tables II and III we present a grid refinement and accuracy analysis of the scheme developed. This clearly shows fourth order convergence in the maximum error for the velocity components and the pressure. For the pressure derivatives the order of accuracy is 3.5, which is a very good result knowing that the present Padé scheme is dispersive, and that no boundary conditions are specified. The convergence order was obtained using the following formula :

$$Order(\cdot) = \frac{\log(\|E_n\|_\infty / \|E_{2n}\|_\infty)}{\log 2}. \quad (38)$$

Additionally, the results obtained with the present scheme, denoted in the following by *fds1*, are compared with the ones obtained using three other schemes, defined as follows. The choice of these schemes allows comparisons to be made against more conventional finite difference

and finite element discretizations.

- Scheme *fds2* : modified version of *fds1* where the 9-points Poisson scheme used to discretize the equation (12) is replaced by the classical 5-points Poisson scheme [12] and second order Taylor approximations are used to approximate first and second order derivatives.
- Scheme *fds3* : modified version of *fds1* where the pressure derivatives are not considered as unknowns. These are then discretized in the momentum equation (right hand side g in (13)) by 4th order Taylor approximations while the Padé schemes are no longer necessary.
- Scheme *fes1* : classical mixed finite element solver for Stokes equations based on triangular P2-P1 elements [14].

The simulations are performed under the same conditions described above and each finite difference scheme uses the same linear solver than *fds1*, namely GMRES with a simple diagonal preconditioner, while the finite element scheme uses the MinRes linear solver [17] with a similar preconditioner. In table V we show average convergence orders obtained by running the same set of simulations performed previously for the four schemes. It should be noted that for the last two schemes, the pressure derivative errors and accuracy orders were post-computed by using 4th order Taylor approximations in *fds3* and by considering the gradient at each node as the average of each surrounding cell values in *fes1*. Our scheme, *fds1*, clearly shows superior convergence rates for p and its derivatives. It is also important to look at the actual errors and performance on a particular grid however. Table IV shows these results for a $[40 \times 40]$ mesh, and it is clear that the errors in the pressure and its derivatives are already superior (or at least as good in the finite element case) as the other schemes, which converge more slowly. For completeness of the comparison we also include the the number of non-zero matrix elements and the CPU times required by each scheme. These latter figures should be treated with great care however since we have not attempted to optimize any of the implementations developed.

Scheme	$\ E_u\ _\infty$	$\ E_v\ _\infty$	$\ E_p\ _\infty$	$\ E_{\partial_x p}\ _\infty$	$\ E_{\partial_y p}\ _\infty$	cpu T(s)	A size
<i>fds1</i>	$7.58 \cdot 10^{-6}$	$7.84 \cdot 10^{-6}$	$3.88 \cdot 10^{-4}$	$3.26 \cdot 10^{-3}$	$2.71 \cdot 10^{-3}$	61	75703
<i>fds2</i>	$1.76 \cdot 10^{-3}$	$1.29 \cdot 10^{-3}$	$6.13 \cdot 10^{-2}$	$5.41 \cdot 10^{-1}$	$4.98 \cdot 10^{-1}$	52	43635
<i>fds3</i>	$2.49 \cdot 10^{-6}$	$2.56 \cdot 10^{-6}$	$8.71 \cdot 10^{-4}$	$4.32 \cdot 10^{-3}$	$3.76 \cdot 10^{-3}$	50	88986
<i>fes1</i>	$7.50 \cdot 10^{-7}$	$3.62 \cdot 10^{-7}$	$2.28 \cdot 10^{-4}$	$1.61 \cdot 10^{-2}$	$1.47 \cdot 10^{-2}$	190	142746

Table IV. Uniform Errors and other simulation characteristics for a $[40 \times 40]$ mesh

Scheme	O u	O v	O p	O $\partial_x p$	O $\partial_y p$
<i>fds1</i>	3.99	3.99	4.01	3.53	3.51
<i>fds2</i>	1.97	1.98	1.99	1.35	1.34
<i>fds3</i>	3.76	3.77	3.55	3.10	3.03
<i>fes1</i>	3.87	3.73	2.63	1.34	1.25

Table V. Accuracy orders (average)

For the present test case, we discuss below the advantages and drawbacks of the compared schemes.

- *fds2* : as expected, this scheme is 2^{nd} order accurate for the velocity components and the pressure, whilst the pressure derivatives accuracy is only 1.35 despite the use of the 4^{th} order Padé scheme, which leads us to conclude that the global accuracy of our solution strategy is predominately driven by the Laplacian discretization.
- *fds3* : this scheme shows better performances in term of the time required for the simulation but it is slightly lower order than the scheme *fds1*. It should also be noticed that although the matrix size is smaller than the one obtained using *fds1*, the number of non-zero elements and therefore the matrix storage size is bigger. These inferior results for the pressure approximations also allow us to see the improving effect of the Padé scheme on the accuracy in these terms.
- *fes1* : this scheme shows poorer performance than the previous schemes, except for *fds2*, in term of accuracy, computing time and matrix storage size. This reflects the additional overhead associated with an unstructured finite element solver (that we would be unlikely to use on a simple geometry such as this).

4.2. Kovaszny Flow

Kovaszny flow [18] is an analytical solution of 2D steady-state Navier-Stokes equations, with no forcing term, that is similar to the laminar flow over a periodic array of cylinders. Unlike Poiseuille flow, this flow incorporates nonlinear effects and it is therefore a good test for the full Navier-Stokes solution algorithm. Let $\Omega = [-0.5, 1.0] \times [-0.5, 1.5]$ be the flow domain. The analytical solution has the form:

$$u(x, y) = 1 - e^{\alpha x} \cos(2\pi y) \quad (39a)$$

$$v(x, y) = \frac{\alpha}{2\pi} e^{\alpha x} \sin(2\pi y) \quad (39b)$$

$$p(x, y) = \frac{1}{2}(1 - e^{2\alpha x}), \quad (39c)$$

where the parameter α is given in terms of the Reynolds number Re by:

$$\alpha = \frac{Re}{2} - \sqrt{\frac{Re^2}{4} + 4\pi^2}. \quad (40)$$

As with the previous test case, the boundary conditions are given by the solution (39). The values of \mathbf{u} are prescribed on all the boundaries while the hydrostatic pressure mode is eliminated by fixing the pressure value at a single grid node. This problem was solved for a value of Re equal to 40 and the tolerance of the GMRES iteration was again set to 10^{-14} . For this test case the Newton linearization technique converged after 5 nonlinear iterations, the stopping criteria being set to 10^{-12} based on the nonlinear residual, (without the need for the Picard steps to find an initial iterate) for all mesh sizes considered. This is due to the relatively small value of Reynolds number.

Tables VI and VII show a grid refinement and accuracy analysis of the scheme. As for the Stokes test case, fourth order accuracy is confirmed for the velocity components and the pressure, whilst an order of accuracy of 3.5 is observed for the pressure derivatives.

4.3. Lid-driven cavity flow

The lid-driven cavity flow is a classical and simple test case consisting of a confined fluid flow in a cavity driven by the moving upper wall, while the other walls are static. The problem to

$n_x \times n_y$	$\ E_u\ _\infty$	Order u	$\ E_v\ _\infty$	Order v
30×20	$2.6 \cdot 10^{-3}$	-	$2.51 \cdot 10^{-3}$	-
60×40	$1.65 \cdot 10^{-4}$	3.98	$1.58 \cdot 10^{-4}$	3.99
120×80	$1.04 \cdot 10^{-5}$	3.99	$9.87 \cdot 10^{-5}$	4.00
240×160	$6.48 \cdot 10^{-7}$	4.00	$6.13 \cdot 10^{-7}$	4.01
480×320	$4.05 \cdot 10^{-8}$	4.00	$3.83 \cdot 10^{-8}$	4.00

Table VI. Accuracy orders of velocity components for Kovaszny flow test case (39)

$n_x \times n_y$	$\ E_p\ _\infty$	Order p	$\ E_{\partial_x p}\ _\infty$	Order $\partial_x p$	$\ E_{\partial_y p}\ _\infty$	Order $\partial_y p$
30×20	$8.08 \cdot 10^{-3}$	-	$1.26 \cdot 10^{-2}$	-	$1.31 \cdot 10^{-2}$	-
60×40	$5.16 \cdot 10^{-4}$	3.97	$1.12 \cdot 10^{-3}$	3.49	$1.15 \cdot 10^{-3}$	3.51
120×80	$3.24 \cdot 10^{-5}$	3.99	$9.98 \cdot 10^{-5}$	3.49	$1.01 \cdot 10^{-4}$	3.52
240×160	$2.06 \cdot 10^{-6}$	3.99	$8.82 \cdot 10^{-6}$	3.50	$8.74 \cdot 10^{-6}$	3.52
480×320	$1.26 \cdot 10^{-7}$	4.02	$7.80 \cdot 10^{-7}$	3.50	$7.56 \cdot 10^{-7}$	3.53

Table VII. Accuracy order of pressure field and derivatives for Kovaszny flow test case (39)

be solved, in a unit square, is governed by the Navier-Stokes system of equations (8) together with the following Dirichlet BCs for the velocity components:

$$u(0, y) = 0 \quad , \quad v(0, y) = 0 \quad (41a)$$

$$u(1, y) = 0 \quad , \quad v(1, y) = 0 \quad (41b)$$

$$u(x, 0) = 0 \quad , \quad v(x, 0) = 0 \quad (41c)$$

$$u(x, 1) = 1 \quad , \quad v(x, 1) = 0 \quad (41d)$$

Our simulations were performed on a $[80 \times 80]$ uniform mesh ($\Delta x = \Delta y = 0.0125$) for Reynolds numbers of 100, 400 and 1000 such that the characteristic velocity and length are chosen to be equal to unity. Both GMRES and nonlinear residual tolerances were set to 10^{-9} . Using coarser meshes good asymptotic convergence was not achieved due to the poor resolution of the near wall flow, and leading to results different from those of Ghia [19]. We also noticed that without the use of the Taylor approximations (58b) and (58d) for the pressure boundary condition discretization, and only for this test case, the scheme failed to converge because of the singularities at the top corners of the domain.

Concerning the linearization technique, we have used the Newton method with 2 Picard iterations to obtain an initial guess. For $Re = 100$, no continuation procedure was necessary and only 5 Newton iterations assured convergence. For the higher Reynolds number test cases use of the continuation procedure was necessary: such that λ was incremented in steps of 0.25 for $Re = 400$ and 0.1 for $Re=1000$. The Newton iterations converged quickly in all cases, with only 4 to 5 iterations needed at each λ during the continuation.

In the figure 2 we show the u-velocity profiles along a vertical line passing through the geometric centre of the cavity for the three Reynolds numbers considered, whilst figure 3 shows the v-velocity profiles along a horizontal line, also passing through the geometric centre of the cavity. For both figures, the solid lines represent the numerical results obtained using our proposed scheme on a uniform grid, whilst symbols indicate the solutions reported in [19]. In order to visualize the streamlines, the stream function and the vorticity are obtained by solving

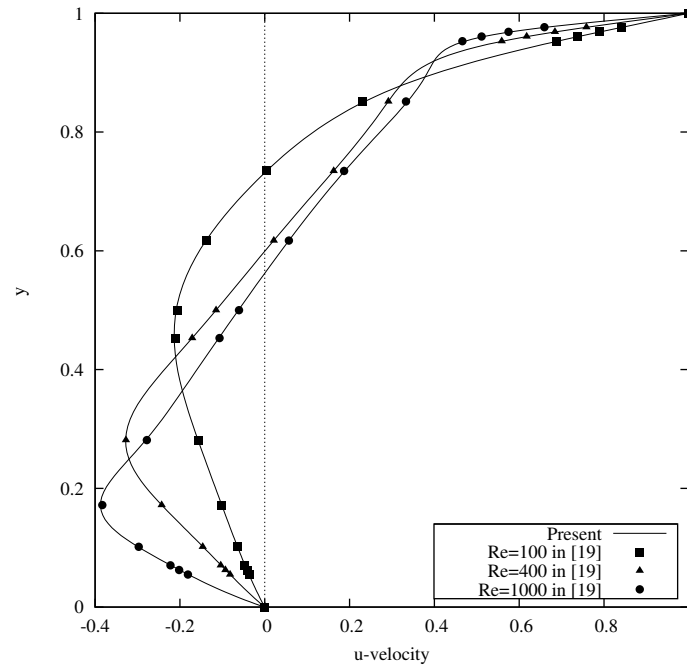


Figure 2. u-velocity at vertical line $x=0.5$ for the lid-driven cavity with $Re=100$, 400 and 1000

the linear system using the 9-point Poisson scheme (14). The calculations are performed as a post-processing step at the end of the flow simulations and figure 4 shows the computed streamlines for a Reynolds number of 1000.

The mesh dependency of the solution for a Reynolds number of 100 was examined for five mesh configurations, beginning from $[80 \times 80]$ and increasing by 20 in both directions. In table (VIII) we show the velocity components values at centre node $(0.5, 0.5)$ compared with values obtained in [19] that indicates clearly that the finer mesh lead to mesh-independent solutions.

Quantity	$[80 \times 80]$	$[100 \times 100]$	$[120 \times 120]$	$[140 \times 140]$	$[160 \times 160]$	Ghia [19]
$u(0.5, 0.5)$	-0.208033	-0.210836	-0.210892	-0.210894	-0.210895	-0.2109
$v(0.5, 0.5)$	0.053973	0.054487	0.054529	0.054537	0.054539	0.05454

Table VIII. Mesh dependency check at $(0.5, 0.5)$ for the lid-driven cavity at $Re = 100$

4.4. Backward facing step

The backward-facing step is a another standard benchmark problem. For this particular implementation the domain is a rectangular channel such that the re-entrant backward-facing step is simulated by imposing a parabolic velocity profile along the upper half of the inlet boundary and zero inflow velocity along the lower half of the inflow boundary. An open boundary condition is applied at the outflow and the no-slip velocity condition is imposed

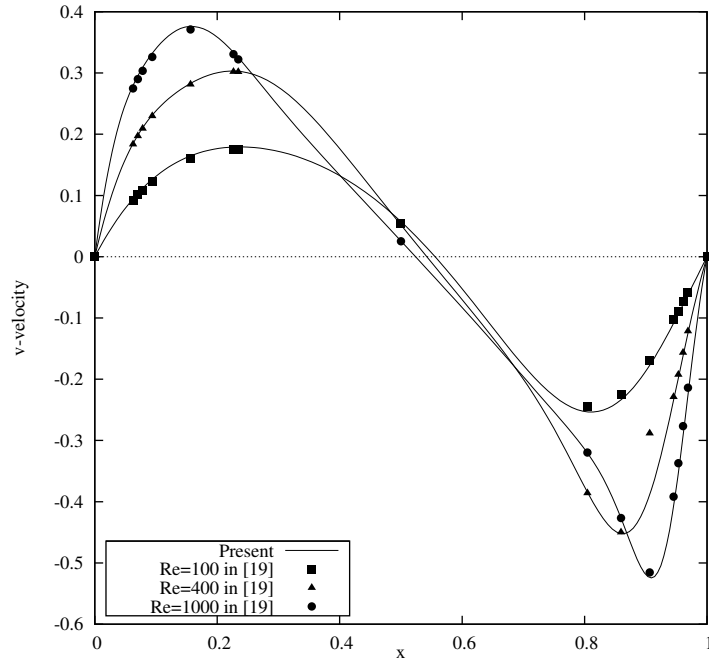


Figure 3. v -velocity at horizontal line $y=0.5$ for the lid-driven cavity with $Re=100, 400$ and 1000

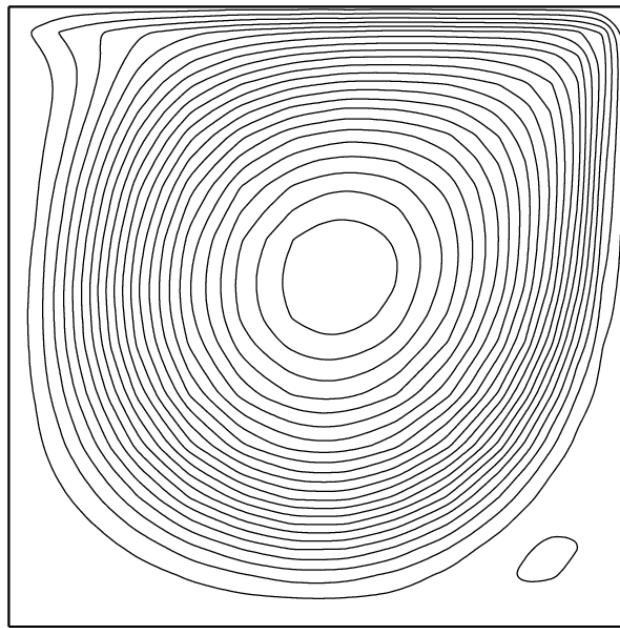


Figure 4. Streamlines for the lid-driven cavity flow at $Re=1000$

on the remaining (top and bottom) boundaries. Specifically and as shown in figure 5, the domain $\Omega = [0, 30] \times [-0.5, 0.5]$, the governing equations the same as for the previous test case, and the imposed boundary conditions are as follows:

$$u(0, 0 \leq y \leq 0.5) = 24y(0.5 - y) \quad , \quad v(0, 0 \leq y \leq 0.5) = 0 \quad (42a)$$

$$u(0, -0.5 \leq y < 0) = 0 \quad , \quad v(0, -0.5 \leq y < 0) = 0 \quad (42b)$$

$$u(x, -0.5) = 0 \quad , \quad v(x, -0.5) = 0 \quad (42c)$$

$$u(x, 0.5) = 0 \quad , \quad v(x, 0.5) = 0 \quad (42d)$$

$$Re^{-1} \partial_x u(30, y) - p = 0 \quad , \quad Re^{-1} \partial_x v(30, y) = 0 \quad (42e)$$

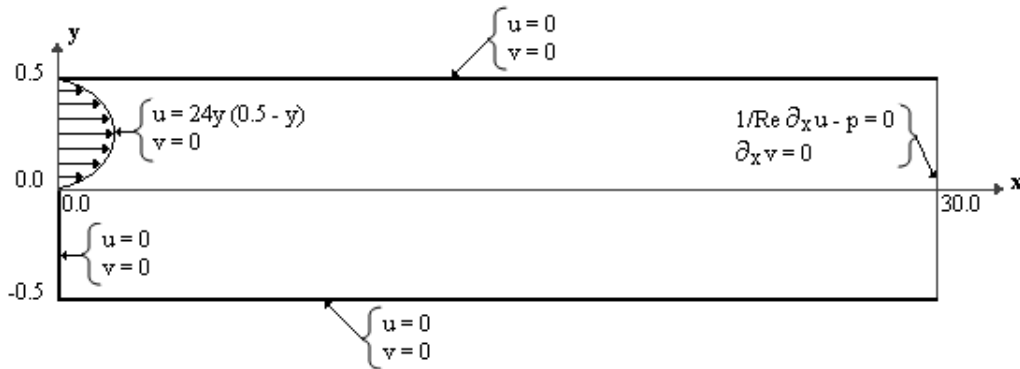


Figure 5. Backward facing step geometry and boundary conditions [21]

This problem was simulated on a $[20 \times 600]$, equally spaced, mesh ($\Delta x = \Delta y = 0.05$) and for a Reynolds number of 800 such that the characteristic velocity is the average inflow velocity and equal to unity and characteristic length in the channel height and also equal to unity. The GMRES and nonlinear residual tolerances were set to 10^{-9} . The resulting flow has been demonstrated to be steady and stable [20], and our computational results are compared with those provided by Gartling [21]. As with the previous test case, simulations on a coarser mesh than that used for the results presented here fail to capture all of the features of the resulting fluid flow. Also as for the previous test case, we have used the continuation approach to obtain a converged solution at this Reynolds number: this time based upon increments of λ equal to 0.125. Each of these nonlinear problems was solved with Newton's method using 2 Picard iterations to obtain the initial guess.

We show in figure 6 the u -velocity profiles along the two vertical lines $x = 7$ and $x = 15$ and in figure 7 the v -velocity profiles along the same vertical lines. In both figures, the solid lines represent the numerical results obtained using our proposed scheme, while symbols indicate the solutions reported in [21].

At a Reynolds number of 800 the flow studied here is characterized by two separation regions. The first of these starts at the step corner and continues downstream for a distance of approximately 12 step heights, whilst the second is on the upper wall and occupies a region from approximately 10 to 20 step heights downstream. To illustrate these, figure 8 shows plots

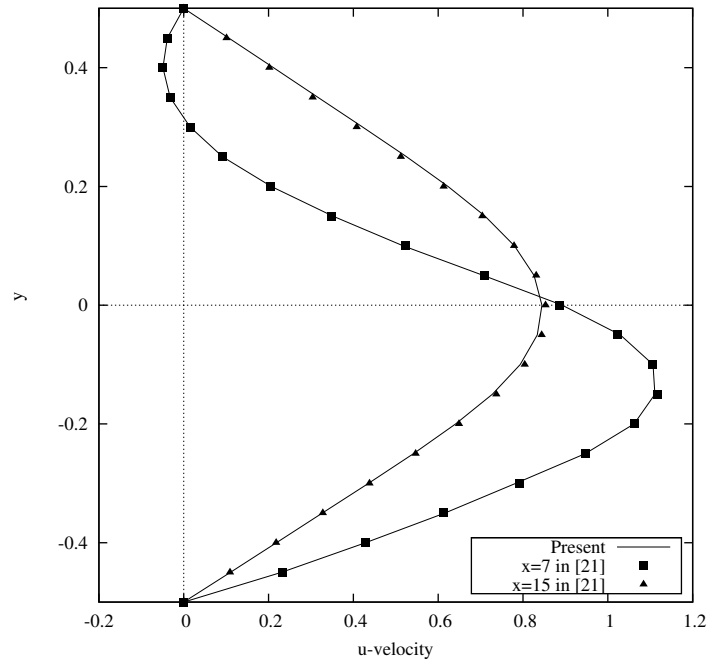


Figure 6. u-velocity for the backward-facing step at vertical lines $x=7$ and $x=15$

of the pressure and stream function contours for the first ten step heights downstream of the inflow boundary: these plots show a good match with the equivalent results presented in [21]. Additionally, table IX shows a comparison of separation and reattachment point locations in the recirculating regions obtained using the present scheme versus those obtained in [21, 20].

Quantity	Present study	Gartling[21]	Gresho and al.[20]
Lower eddy reattachment x location	6.10	6.10	6.10
Upper eddy separation x location	4.85	4.86	4.85
Upper eddy reattachment x location	10.50	10.48	10.49

Table IX. Comparison of present results with Gartling[21] and Gresho and al.[20]

5. EXTENSIONS

5.1. Higher order Schemes :

Numerous extensions to the work presented here are possible. For example, the framework that we have described naturally permits the use of higher order stencils and higher order Padé schemes. This may be confirmed by extending the accuracy to 6th order, and in tables X and XI we present the results obtained by such an extension for the Stokes test case of section

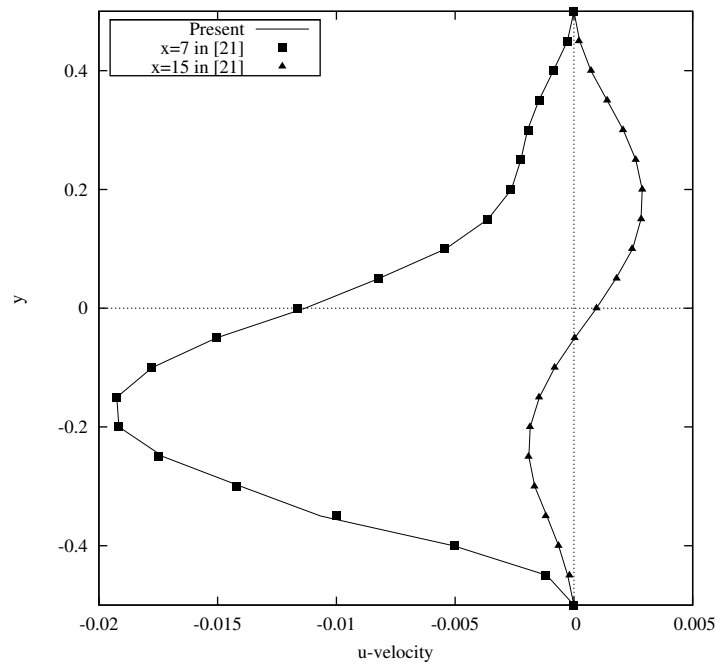
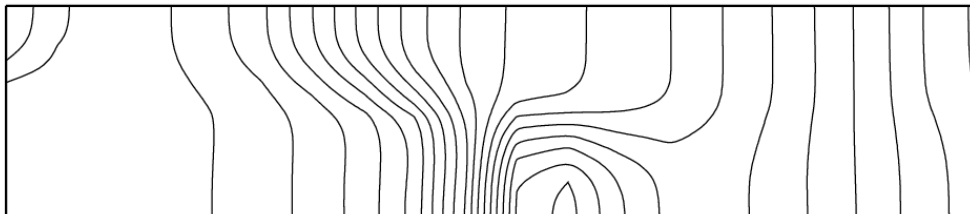


Figure 7. v -velocity for the backward-facing step at vertical lines $x=7$ and $x=15$

a - Pressure Contours



b - Streamlines

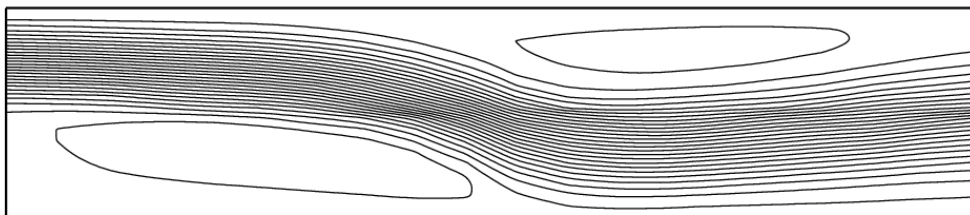


Figure 8. Pressure and stream function contours for the flow over a backward-facing step

4.1. Our numerical strategy remains the same but we now use the 6th order Poisson equation scheme introduced by Rosser [12] and the same order Padé schemes and Taylor approximations. The grid refinement sequences shown in tables X and XI demonstrate the expected sixth order accuracy for velocity and pressure, and an order of accuracy of 5.5 for the pressure gradient. Extension to even higher orders is of course possible in a similar manner.

$n_x \times n_y$	$\ E_u\ _\infty$	Order(u)	$\ E_v\ _\infty$	Order(v)
20×20	4.54 10 ⁻⁵	-	4.57 10 ⁻⁵	-
40×40	7.25 10 ⁻⁷	5.97	7.20 10 ⁻⁷	5.99
80×80	1.15 10 ⁻⁸	5.99	1.12 10 ⁻⁸	6.00
160×160	1.79 10 ⁻¹⁰	6.00	1.73 10 ⁻¹⁰	6.02
320×320	2.80 10 ⁻¹²	6.00	2.65 10 ⁻¹²	6.03

Table X. Accuracy order of velocity components for Stokes test case (36) using 6th order scheme

$n_x \times n_y$	$\ E_p\ _\infty$	Order(p)	$\ E_{\partial_x p}\ _\infty$	Order($\partial_x p$)	$\ E_{\partial_y p}\ _\infty$	Order($\partial_y p$)
20×20	8.24 10 ⁻⁴	-	9.72 10 ⁻³	-	9.51 10 ⁻³	-
40×40	1.31 10 ⁻⁵	5.98	2.27 10 ⁻⁴	5.42	2.25 10 ⁻⁴	5.40
80×80	2.05 10 ⁻⁷	5.99	5.27 10 ⁻⁶	5.43	5.30 10 ⁻⁶	5.41
160×160	3.21 10 ⁻⁹	6.00	1.22 10 ⁻⁷	5.43	1.23 10 ⁻⁷	5.43
320×320	4.98 10 ⁻¹¹	6.01	2.83 10 ⁻⁹	5.43	2.87 10 ⁻⁹	5.42

Table XI. Accuracy order of pressure field and derivatives for Stokes test case (36) using 6th order scheme

5.2. The Unsteady Case :

We consider in the following the unsteady Stokes equation under appropriate boundary and initial conditions:

$$\partial_t \mathbf{u} - \nabla^2 \mathbf{u} + \nabla p = \mathbf{f} \quad (43a)$$

$$\nabla^2 p = \nabla \cdot \mathbf{f} , \quad (43b)$$

The spatial discretization strategy remains the same as described in section 2, leading to a semi-discretized problem that can be expressed in the following matrix form:

$$M \frac{dx}{dt} = -Ax + F \quad (44)$$

The matrix A , the vector F and the unknown vector x are defined as in (20), whilst the matrix M given by:

$$M = \begin{bmatrix} M_1 & 0 & 0 & 0 & 0 \\ 0 & M_1 & 0 & 0 & 0 \\ M_2 & M_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (45)$$

Δt	$n_x \times n_y$	$\ E_u\ _\infty$	Order(u)	$\ E_v\ _\infty$	Order(v)
1/10	10×10	9.33 10 ⁻⁷	-	5.08 10 ⁻⁶	-
1/20	20×20	6.01 10 ⁻⁸	3.96	3.15 10 ⁻⁷	4.01
1/40	40×40	3.83 10 ⁻⁹	3.97	1.94 10 ⁻⁸	4.02
1/80	80×80	2.38 10 ⁻¹⁰	4.01	1.21 10 ⁻⁹	4.01
1/160	160×160	1.48 10 ⁻¹¹	4.00	7.49 10 ⁻¹¹	4.01

Table XII. Accuracy order of velocity components for unsteady Stokes test case (46)

Δt	$n_x \times n_y$	$\ E_p\ _\infty$	Order p	$\ E_{\partial_x p}\ _\infty$	Order $\partial_x p$	$\ E_{\partial_y p}\ _\infty$	Order $\partial_y p$
1/10	10×10	1.26 10 ⁻⁴	-	3.58 10 ⁻⁴	-	3.01 10 ⁻⁴	-
1/20	20×20	8.08 10 ⁻⁶	3.96	3.21 10 ⁻⁵	3.48	3.19 10 ⁻⁵	3.24
1/40	40×40	5.12 10 ⁻⁷	3.98	2.93 10 ⁻⁶	3.45	3.44 10 ⁻⁶	3.21
1/80	80×80	3.26 10 ⁻⁸	3.97	2.72 10 ⁻⁷	3.44	3.69 10 ⁻⁷	3.22
1/160	160×160	2.08 10 ⁻⁹	3.98	2.51 10 ⁻⁸	3.44	3.86 10 ⁻⁸	3.26

Table XIII. Accuracy order of pressure field and derivatives for unsteady Stokes test case (46)

To perform the temporal differencing, we use here the Backward Difference Formula (BDF) [22] family of methods.

In order to validate our unsteady version of the scheme for the Stokes flow, let's consider the following exact test case defined on $\Omega = [0, 1] \times [0, 1]$ such that the solution is given by:

$$u = \sin(x) \sin(y + t) \quad (46a)$$

$$v = \cos(x) \cos(y + t) \quad (46b)$$

$$p = \cos(x) \sin(y + t) . \quad (46c)$$

Based upon this solution, the right-hand side of the system to be solved (43) is given by:

$$f_x = \sin(x)(\cos(y + t) + \sin(y + t)) \quad (47a)$$

$$f_y = \cos(x)(3 \cos(y + z) - \sin(y + z)) \quad (47b)$$

$$\nabla \cdot f = 2 \cos(x) \sin(y + z) . \quad (47c)$$

The boundary conditions for this test case are defined by the solution (46) as well as the initial conditions with $t = 0$. The values of \mathbf{u} are given on all boundaries while the pressure value is fixed at a single grid node. The tolerance of GMRES iteration is set to 10^{-14} and the solutions were computed up to time 10.

In this work the 4th order BDF scheme is used, to be compatible with 4th order spatial discretizations, using exact solutions as given by (46) for the first three steps. The numerical results showing the order of accuracy, computed using formula (38), are displayed in tables XII and XIII. As predicted, it demonstrate clearly that the scheme is fourth-order accurate in time for all the variables, whilst the pressure derivatives have an order approaching 3.5.

6. CONCLUSIONS

In this paper we have presented a new class of high-order finite difference schemes for the accurate and stable solution of the incompressible Stokes and Navier-Stokes equations.

Our approach is based upon a velocity-pressure-pressure gradient formulation, where all components are approximated on a single mesh using high order finite difference stencils for the velocity and pressure, and corresponding Padé schemes for the pressure gradient. The solution strategy for the new class of schemes presented in this paper can be summarised by three major points:

1. The momentum and pressure equations are written in a Poisson equation form, such that schemes developed for this equation could be used.
2. The pressure derivatives are considered as unknown, which leads us to use Padé schemes to discretize them.
3. Other derivatives involved in the equations are discretized using Taylor approximations.

A detailed derivation of the 4th order spatial discretization has been presented and numerical tests, that confirm the 4th order accuracy for the velocity and pressure components, have been described. Furthermore, these numerical tests show that the order of accuracy obtained for the pressure derivatives is 3.5.

There are many other potential extensions which we have yet to implement. In principle, at least, the extension from two dimensions to three dimensions should be straightforward. Similarly, the extension to time-dependent problems using high order implicit time-stepping has been shown to be feasible without substantial further development: so long as the time-step size is not too large the Newton solver should converge using an initial iterate based upon a simple extrapolation of the solution at the previous time levels. This is the subject of our current research, where we seek to apply time-dependent Stokes and Navier-Stokes equations to fluid-structure interaction problems. The results of this work will be the subject of subsequent papers treating both small and large structure deformations.

Of course, further research is also needed to improve the efficiency of the schemes' practical implementation. Specifically, the current linear solver is far from optimal as it is based upon a very simple diagonal preconditioner, which fails to control the growth in the number of iterations required as the mesh size is refined. Optimizing (or at least substantially improving) this preconditioner is an important next step. Furthermore, we currently solve the linear systems to the same accuracy at each Newton iteration: further efficiency improvements should be made by reducing significantly the level of accuracy required at the first and second Newton iterations. Perhaps more challenging, but also of great potential value, would be the extension of these higher order schemes to non-uniform grids. This would be desirable for flows, often occurring at higher Reynolds number, which exhibit boundary layers, or similar phenomena.

REFERENCES

1. Gresho PM and Sani RL. *Incompressible Flow and Finite Element Method, Volume 2, Isothermal Laminar Flow*. Wiley Chichester, England, 1998.
2. Gresho PM and Sani RL. On pressure boundary-condition for the incompressible Navier-Stokes equations, *International Journal for Numerical Methods in Fluids* 1987; **7**: 1111–1145. Chichester, 1998.
3. Roache PJ *Computational Fluid Dynamics*. Hermosa Publishers, Albuquerque, New Mexico, USA, 1992.
4. Karniadakis GE, Israeli M and Orszag SA. High-order splitting methods for the incompressible Navier-Stokes equations, *Journal of Computational Physics* 1991; **97**: 414–443.
5. Petersson NA. Stability of Pressure Boundary Conditions for Stokes and Navier-Stokes Equations, *Journal of Computational Physics* 2001; **172**: 40–70.

6. Henshaw WD and Petersson NA. A split-step scheme for the incompressible Navier-Stokes equations, in *Numerical simulations of incompressible flows* (Half Moon Bay, CA, 2001), 108125, World Sci. Publishing, River Edge, NJ, 2003.
7. Hirsh RS. Higher Order Accurate Difference Solutions of Fluid Mechanics Problems by a Compact Differencing Technique, *Journal of Computational Physics* 1975; **19** : 90–109.
8. Collatz L. *The Numerical Treatment of Differential Equations*, Springer-Verlag, New York, 2005.
9. Collatz L. Hermitian Methods for Initial Value Problems in Partial Differential Equations, *Topics in Numerical Analysis, Proceedings of the Royal Irish Academy Conference on Numerical Analysis* 1972 : 41–61.
10. Lele SK Compact Finite Difference Schemes with Spectral-like Resolution, *Journal of Computational Physics* 1992; **103**: 16–42.
11. Shih TM, Tan CH and Hwang BC. Effect of Grid Staggering on Numerical Schemes, *International Journal for Numerical Methods in Fluids* 1989; **9**: 193–212.
12. Rosser JB. Nine-point difference solution for Poisson's equation, *Computers and Mathematics with Applications* 1975; **1**: 351–360.
13. Strikwerda JC. *Finite Difference Schemes and Partial Differential Equations*, Wadsworth & Brooks/Cole, Pacific Grove, CA, 1989.
14. Elman HC, Silvester DJ and Wathen AJ. *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics*, Oxford University Press, Oxford, 2005.
15. Saad Y and Schultz MH. GMRES: a generalized minimal residual method for solving nonsymmetric linear systems, *SIAM Journal on Scientific and Statistical Computing* 1986; **7**: 856–869.
16. Bochev PB, Gunzburger MD and Lehoucq RB. On stabilized finite element methods for the Stokes problem in the small time step limit. *International Journal for Numerical Methods in Fluids* 2007; **53**: 573–597.
17. Paige CC and Saunders MA. Solution of sparse indefinite systems of linear equations. *SIAM Journal on Numerical Analysis* 1975; **12**:617–629.
18. Kovasznay LIG. Laminar flow behind a two-dimensional grid. *Proc. Cambridge Philos. Soc.* 1948; **44**:58–62.
19. Ghia U, Ghia KN and Shin CT. High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method *Journal of Computational Physics* 1982; **48**: 387–411.
20. Gresho PM, Gartling DK, Torczynski JR, Cliffe KA, Winters KH, Garratt TJ, Spence A and Goodrich JW. Is the steady viscous incompressible two-dimensional flow over a backward-facing step at Re = 800 stable? *International Journal for Numerical Methods in Fluids* 1993; **17**:501–541.
21. Gartling DK. A test problem for outflow boundary conditions - Flow over a backward facing step. *International Journal for Numerical Methods in Fluids* 1990; **11**:953-967.
22. Brenan KE, Campbel SL, and Petzold LR. *The Numerical Solution of Initial Value Problems in Differential Algebraic Equations*, Elsevier Science Publishing Co., New York, 1989.

7. APPENDIX A

This appendix provides a fully-expanded description of the discretised equations that we solve in the case of the Stokes problem. In the domain Ω , we use the 9-point Poisson scheme (14) for both the pressure and the velocity components, and the Padé scheme (18) for the pressure derivatives. These lead to the following discretisations:

- For the x-velocity u :

$$\begin{aligned}
 & a(u_{i-1,j} + u_{i+1,j}) + b(u_{i,j-1} + u_{i,j+1}) + u_{i-1,j-1} + u_{i+1,j-1} + u_{i-1,j+1} - u_{i+1,j+1} \\
 & \quad - 20u_{i,j} - c(8\partial_x p_{i,j} + \partial_x p_{i-1,j} + \partial_x p_{i+1,j} + \partial_x p_{i,j-1} + \partial_x p_{i,j+1}) \quad (48) \\
 & \quad = -c(8f_{xi,j} + f_{xi-1,j} + f_{xi+1,j} + f_{xi,j-1} + f_{xi,j+1}) .
 \end{aligned}$$

- For the y-velocity v :

$$\begin{aligned}
 & a(v_{i-1,j} + v_{i+1,j}) + b(v_{i,j-1} + v_{i,j+1}) + v_{i-1,j-1} + v_{i+1,j-1} + v_{i-1,j+1} - v_{i+1,j+1} \\
 & \quad - 20v_{i,j} - c(8\partial_y p_{i,j} + \partial_y p_{i-1,j} + \partial_y p_{i+1,j} + \partial_y p_{i,j-1} + \partial_y p_{i,j+1}) \quad (49) \\
 & \quad = -c(8f_{yi,j} + f_{yi-1,j} + f_{yi+1,j} + f_{yi,j-1} + f_{yi,j+1}) .
 \end{aligned}$$

- For the pressure p :

$$\begin{aligned} a(p_{i-1,j} + p_{i+1,j}) + b(p_{i,j-1} + p_{i,j+1}) + p_{i-1,j-1} + p_{i+1,j-1} + p_{i-1,j+1} - p_{i+1,j+1} \\ - 20p_{i,j} = c(8\partial_x f_{xi,j} + \partial_x f_{xi-1,j} + \partial_x f_{xi+1,j} + \partial_x f_{xi,j-1} + \partial_x f_{xi,j+1} \\ + 8\partial_y f_{yi,j} + \partial_y f_{yi-1,j} + \partial_y f_{yi+1,j} + \partial_y f_{yi,j-1} + \partial_y f_{yi,j+1}). \end{aligned} \quad (50)$$

- For the pressure partial derivatives $\partial_x p$:

$$\partial_x p_{i-1,j} + 4\partial_x p_{i,j} + \partial_x p_{i+1,j} + \frac{3}{\Delta x}(p_{i-1,j} - p_{i+1,j}) = 0. \quad (51)$$

- For the pressure partial derivatives $\partial_y p$:

$$\partial_y p_{i,j-1} + 4\partial_y p_{i,j} + \partial_y p_{i,j+1} + \frac{3}{\Delta y}(p_{i,j-1} - p_{i,j+1}) = 0. \quad (52)$$

On the boundary $\partial\Omega_N$, the pressure boundary condition (11d) and its partial derivatives needs to be discretised.

- For the pressure p :

– at the corners

$$n_x(\partial_x p_{i,j} + \partial_{xy}^2 v_{i,j} - \partial_{yy}^2 u_{i,j}) + n_y(\partial_y p_{i,j} + \partial_{xy}^2 u_{i,j} - \partial_{xx}^2 v_{i,j}) = n_x f_{xi,j} + n_y f_{yi,j} \quad (53)$$

with

$$n_x = (\mathbf{n} \cdot \mathbf{e}_x) \frac{\Delta x}{\sqrt{\Delta x^2 + \Delta y^2}} \quad \text{and} \quad n_y = (\mathbf{n} \cdot \mathbf{e}_y) \frac{\Delta y}{\sqrt{\Delta x^2 + \Delta y^2}};$$

where n_x and n_y are the components of the outward normal vector \mathbf{n} while \mathbf{e}_x and \mathbf{e}_y are versors of the two-dimensional Cartesian coordinate system.

– along horizontal boundaries

$$\partial_y p_{i,j} + \partial_{xy}^2 u_{i,j} - \partial_{xx}^2 v_{i,j} = f_{yi,j}; \quad (54)$$

– along vertical boundaries

$$\partial_x p_{i,j} + \partial_{xy}^2 v_{i,j} - \partial_{yy}^2 u_{i,j} = f_{xi,j}. \quad (55)$$

- For the pressure partial derivatives $\partial_x p$:

$$\partial_x p_{i,j} + 3\partial_x p_{i+1,j} + \frac{1}{6\Delta x}(17p_{i,j} - 9p_{i+1,j} - 9p_{i+2,j} + p_{i+3,j}) = 0. \quad (56a)$$

- For the pressure partial derivatives $\partial_y p$:

$$\partial_y p_{i,j} + 3\partial_y p_{i,j+1} + \frac{1}{6\Delta y}(17p_{i,j} - 9p_{i,j+1} - 9p_{i,j+2} + p_{i,j+3}) = 0. \quad (57a)$$

In the above equations we use the appropriate 4th order Taylor approximation in (58) for first order derivatives $\partial_x p_{i,j}$ and/or $\partial_y p_{i,j}$, a combination of the appropriate approximations in (58) for mixed second order derivatives $\partial_{xy}^2 u_{i,j}$ and/or $\partial_{xy}^2 v_{i,j}$, and the appropriate approximation in (59) for second order derivatives $\partial_{xx}^2 v_{i,j}$ and/or $\partial_{yy}^2 u_{i,j}$:

$$i = 0, \quad f'_i = \frac{1}{12h}(-25f_i + 48f_{i+1} - 36f_{i+2} + 16f_{i+3} - 3f_{i+4}) \quad (58a)$$

$$i = 1, \quad f'_i = \frac{1}{12h}(-3f_{i-1} - 10f_i + 18f_{i+1} - 6f_{i+2} + f_{i+3}) \quad (58b)$$

$$1 < i < n - 2, \quad f'_i = \frac{1}{12h}(f_{i-2} - 8f_{i-1} + 8f_{i+1} - f_{i+2}) \quad (58c)$$

$$i = n - 2, \quad f'_i = \frac{1}{12h}(3f_{i+1} + 10f_i - 18f_{i-1} + 6f_{i-2} - f_{i-3}) \quad (58d)$$

$$i = n - 1, \quad f'_i = \frac{1}{12h}(25f_i - 48f_{i-1} + 36f_{i-2} - 16f_{i-3} + 3f_{i-4}); \quad (58e)$$

$$i = 0, \quad f''_i = \frac{1}{12h^2}(45u_i - 154u_{i+1} + 214u_{i+2} - 156u_{i+3} + 61u_{i+4} - 10u_{i+5}) \quad (59a)$$

$$i = n - 1, \quad f''_i = \frac{1}{12h^2}(45u_i - 154u_{i-1} + 214u_{i-2} - 156u_{i-3} + 61u_{i-4} - 10u_{i-5}) \quad (59b)$$