Chapter 5

HIGH ACCURACY CUBIC SPLINE APPROXIMATION FOR TWO DIMENSIONAL QUASI-LINEAR ELLIPTIC BOUNDARY VALUE PROBLEMS

§

5.1 Introduction

When a physical system depends on more than one variable a general description of its behavior often leads to partial differential equation. These equations arise in such diverse subjects as meteorology, electromagnetic theory, heat transfer, nuclear physics and elasticity to name just a few. Often, a system of differential equations of various order with different boundary conditions occur in the study of obstacle, unilateral, moving and free boundary value problems, problems of deflection of beams and in a number of other scientific applications. Most of these equations arising in applications are not solvable analytically and one is obliged to devise techniques for the determination of approximate solutions. In this chapter, we aim to discuss the application of cubic spline functions to solve these boundary value problems. Use of cubic spline functions in the solution of nonlinear boundary value problems has been a challenging task for academic researchers. For the two dimensional linear elliptic equations, a number of constant mesh fourth order finite difference schemes have been designed by [13, 59, 60, 100, 101]. These linear systems have good numerical stability and provide high accuracy approximations. Later, using 9-point fourth order discretization for the solution of two dimensional nonlinear boundary value problems have been developed by Jain et al [46, 47] and Mohanty et al [63, 71, 73]. Theory of splines and their applications to two point linear boundary value problems have been studied in [1, 3, 4, 12, 34, 81, 93, 97]. Jain and Aziz [43] have derived fourth order cubic spline method for solving nonlinear two point boundary value problems with significant first derivative terms. Al-Said [8-9] has used cubic splines in the numerical solution of the second order boundary value problems. In the recent past, Mohanty et al [66-67] have discussed fourth order accurate cubic spline Alternate Group Explicit method for the solution of two point boundary value problems.

Khan et al [53-54] have analyzed the use of parametric cubic spline for the solution of linear two point boundary value problems. Later, Rashidinia et al [83-85] have derived the cubic spline method for the nonlinear singular two point boundary value problems. Houstis et al [42] have first used point iterative cubic spline collocation method for the solution of linear elliptic equations. Later, Hadjidimos et al [40] have extended their technique and used line iterative cubic spline collocation method for the solution of elliptic partial differential equation. Dai [19] has discussed a domain decomposition method for solving thin film

§ The results reported in this chapter have been accepted for publication in the Journal “Applied Mathematical Modelling” (2012) To appear
elliptic interface problems with variable coefficients and very lately has developed a compact finite difference scheme for solving an $N$-Carrier system with Neumann boundary conditions [21]. Recently, Mohanty et al [64, 65, 76, 77] have derived high accuracy finite difference methods for the numerical solution of nonlinear elliptic, hyperbolic and parabolic partial differential equations. To our knowledge, no high order cubic spline method for the solution of two dimensional quasi linear elliptic partial differential equations is known in the literature so far. We report here a new nine point compact discretization of order two in $y$- and order four in $x$-directions, based on cubic spline approximation, for the solution of two dimensional quasi-linear elliptic partial differential equations. We describe the complete derivation procedure of the method in details and also discuss how our discretization is able to handle Poisson’s equation in polar coordinates. The convergence analysis of the proposed cubic spline approximation for the nonlinear elliptic equation is discussed in details and we have shown under appropriate conditions the proposed method converges. Some physical examples and their numerical results are provided to justify the advantages of the proposed method.

$$A(x, y, u) \frac{\partial^2 u}{\partial x^2} + B(x, y, u) \frac{\partial^2 u}{\partial y^2} = f \left( x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \right), \quad (x, y) \in \Omega \quad (5.1.1)$$

defined in the bounded domain $\Omega = \{(x, y): 0 < x, y < 1\}$ with boundary $\partial \Omega$, where $A(x, y, u) > 0$ and $B(x, y, u) > 0$ in $\Omega$. The value of $u$ is being given on the boundary of the solution domain $\Omega$.

The corresponding Dirichlet boundary conditions are prescribed by

$$u(x, y) = \psi(x, y) , \quad (x, y) \in \partial \Omega \quad (5.1.2)$$

We assume that for $0 < x, y < 1$,

(i) $f \left( x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \right)$ is continuous, \hspace{1cm} (5.1.3a)

(ii) $\frac{\partial f}{\partial u}, \frac{\partial f}{\partial u_x}, \frac{\partial f}{\partial u_y}$ exist and are continuous, \hspace{1cm} (5.1.3b)

(iii) $\frac{\partial f}{\partial u} \geq 0, \left| \frac{\partial f}{\partial u_x} \right| \leq G$ and $\left| \frac{\partial f}{\partial u_y} \right| \leq H \quad (5.1.3c)$

where $G$ and $H$ are positive constants (see [46, 47]). Further, we may also assume that the coefficients $A(x, y, u)$ and $B(x, y, u)$ are sufficiently smooth and their required higher order partial derivatives exist in the solution domain $\Omega$.

The main aim of this work is to use cubic spline functions and their certain consistency relations, which are then used to develop a numerical method for computing smooth approximations to the solution of equation (5.1.1). Note that each discretization of the elliptic
differential equation (5.1.1) at an interior grid point is based on just 3 evaluations of the function $f$.

![9-point Computational Network](image)

**Organization.** This chapter is organized as follows. The next section presents the high accuracy numerical method based on cubic spline approximations. Third section discusses the derivation procedure of the scheme developed. Section 5.4 concerns with establishing the convergence analysis of the proposed method. In section 5.5, we discuss the application of proposed method to polar coordinate problems. Section 5.6 gives the results of the numerical experiments to verify the accuracy and computational efficiency of the proposed method. Finally, the chapter concludes with some brief remarks on the present work.

### 5.2 The cubic spline approximation and numerical scheme

In this section, we first aim to discuss a numerical method based on cubic spline approximation for the solution of nonlinear elliptic equation

$$A(x, y) \frac{\partial^2 u}{\partial x^2} + B(x, y) \frac{\partial^2 u}{\partial y^2} = f(x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}) \quad 0 < x, y < 1 \quad (5.2.1)$$

We consider our region of interest, a rectangular domain $\Omega = [0, 1] \times [0, 1]$. A grid with spacing $\Delta x > 0$ and $\Delta y > 0$ in the directions $x$- and $y$- respectively are first chosen, so that the mesh points $(x_l, y_m)$ denoted by $(l, m)$ are defined as $x_l = l\Delta x$ and $y_m = m\Delta y$, $l = 0, 1, ..., N + 1$, $m = 0, 1, ..., M + 1$, where $N$ and $M$ are positive integers such that $(N + 1)\Delta x = 1$ and $(M + 1)\Delta y = 1$.

Let us denote the mesh ratio parameter by $p = (\Delta y/\Delta x) > 0$. For convergence of the numerical scheme it is essential that our parameter remains in the range $0 < \sqrt{6}p < 1$ (shown later). Let $U_{l,m}$ and $u_{l,m}$ be the exact and approximation solution values of $u(x, y)$ at the grid point $(x_l, y_m)$, respectively. Similarly, let $A_{l,m} = A(x_l, y_m)$ and $B_{l,m} = B(x_l, y_m)$ be the exact values of $A(x, y)$ and $B(x, y)$ at the grid point $(x_l, y_m)$, respectively.

At the grid point $(x_l, y_m)$, we use the notation
Let $S_m(x)$ be the cubic spline interpolating polynomial of the value $u_{l,m}$ at the grid point $(x_l, y_m)$, and is given by

$$S_m(x) = \frac{(x_l-x)^3}{6\Delta x} M_{l-1,m} + \frac{(x-x_{l-1})^3}{6\Delta x} M_{l,m} + \left(u_{l-1,m} - \frac{\Delta x^2}{6} M_{l-1,m}\right) \frac{x-x}{\Delta x}, \quad x_{l-1} \leq x \leq x_l; \quad l = 1, 2, \ldots, N+1, \quad m = 0, 1, \ldots, M+1$$

which satisfies at $m$th-line parallel to $x$-axis the following properties.

**The Three Properties of Cubic Splines**

Our spline will need to conform to the following stipulations.

(i) $S_m(x)$ coincides with a polynomial of degree three on each $[x_{l-1}, x_l]$, $l = 1, 2, \ldots, N+1, \quad m = 0, 1, \ldots, M+1$;

(ii) $S_m(x) \in C^2[0,1]$, and

(iii) $S_m(x_l) = u_{l,m}$, $l = 0, 1, \ldots, N+1, \quad m = 0, 1, \ldots, M+1$.

The derivatives of the cubic spline function $S_m(x)$ are given by

$$S'_m(x) = -\frac{(x_l-x)^2}{2\Delta x} M_{l-1,m} + \frac{(x-x_{l-1})^2}{2\Delta x} M_{l,m} + \frac{u_{l,m} - u_{l-1,m}}{\Delta x} - \frac{\Delta x}{6} [M_{l,m} - M_{l-1,m}], \quad x \in [x_{l-1}, x_l]$$

$$S''_m(x) = -\frac{(x_{l+1}-x)^2}{2\Delta x} M_{l,m} + \frac{(x-x_l)^2}{2\Delta x} M_{l+1,m} + \frac{u_{l+1,m} - u_{l,m}}{\Delta x} - \frac{\Delta x}{6} [M_{l+1,m} - M_{l,m}], \quad x \in [x_l, x_{l+1}]$$

and where

$$m_{l,m} = S'_m(x_l) = U_{x_l,m}$$

and

$$M_{l,m} = S''_m(x_l) = U_{x_{l+1},m} = \frac{1}{A_{l,m}} \left[-B_{l,m} U_{y_{l+1},m} + f(x_l, y_m, U_{l,m}, m_{l,m}, U_{y,l,m})\right], \quad l = 0, 1, \ldots, N+1, \quad m = 0, 1, \ldots, M+1$$
From the continuity condition we may obtain the consistency relation for $S_m'(x_l)$. Furthermore, from above,

$$S_m(x_{l+1}) = U_{x,l+1,m} = \frac{u_{l+1,m} - u_{l,m}}{\Delta x} + \frac{\Delta x}{6} [M_{l,m} + 2M_{l+1,m}] \quad (5.2.11)$$

and

$$S_m(x_{l-1}) = U_{x,l-1,m} = \frac{u_{l,m} - u_{l-1,m}}{\Delta x} - \frac{\Delta x}{6} [M_{l,m} + 2M_{l-1,m}]. \quad (5.2.12)$$

Note that these are important properties of the cubic spline function $S_m(x)$ which are used in building up the difference scheme.

We consider the following approximations:

$$\bar{U}_{y,l,m} = (u_{l,m+1} - u_{l,m-1})/(2\Delta y) \quad (5.2.13a)$$

$$\bar{U}_{y,l+1,m} = (u_{l+1,m+1} - u_{l+1,m-1})/(2\Delta y) \quad (5.2.13b)$$

$$\bar{U}_{y,l-1,m} = (u_{l-1,m+1} - u_{l-1,m-1})/(2\Delta y) \quad (5.2.13c)$$

$$\bar{U}_{yy,l,m} = (u_{l,m+1} - 2u_{l,m} + u_{l,m-1})/\Delta y^2 \quad (5.2.13d)$$

$$\bar{U}_{yy,l+1,m} = (u_{l+1,m+1} - 2u_{l+1,m} + u_{l+1,m-1})/\Delta y^2 \quad (5.2.13e)$$

$$\bar{U}_{yy,l-1,m} = (u_{l-1,m+1} - 2u_{l-1,m} + u_{l-1,m-1})/\Delta y^2 \quad (5.2.13f)$$

$$\bar{m}_{l,m} = \bar{U}_{x,l,m} = (u_{l+1,m} - u_{l-1,m})/(2\Delta x) \quad (5.2.14a)$$

$$\bar{m}_{l+1,m} = \bar{U}_{x,l+1,m} = (3u_{l+1,m} - 4u_{l,m} + u_{l-1,m})/(2\Delta x) \quad (5.2.14b)$$

$$\bar{m}_{l-1,m} = \bar{U}_{x,l-1,m} = (-3u_{l-1,m} + 4u_{l,m} - u_{l+1,m})/(2\Delta x) \quad (5.2.14c)$$

$$\bar{U}_{xx,l,m} = (u_{l+1,m} - 2u_{l,m} + u_{l-1,m})/\Delta x^2 \quad (5.2.14d)$$

$$\bar{F}_{l,m} = f(x_{l,y_m}, u_{l,m}, \bar{m}_{l,m}, \bar{U}_{y,l,m}) \quad (5.2.15a)$$

$$\bar{F}_{l+1,m} = f(x_{l+1,y_m}, u_{l+1,m}, \bar{m}_{l+1,m}, \bar{U}_{y,l+1,m}) \quad (5.2.15b)$$

$$\bar{F}_{l-1,m} = f(x_{l-1,y_m}, u_{l-1,m}, \bar{m}_{l-1,m}, \bar{U}_{y,l-1,m}) \quad (5.2.15c)$$

$$\bar{M}_{l,m} = \frac{1}{A_{\alpha 0}} [-B_{00} \bar{U}_{yy,l,m} + \bar{F}_{l,m}] \quad (5.2.16a)$$

$$\bar{M}_{l+1,m} = \frac{1}{A_{\alpha 0}} \left( 1 - \frac{\Delta x A_{\alpha 0}}{A_{\alpha 0}} \right) [-B_{l+1,m} \bar{U}_{yy,l+1,m} + \bar{F}_{l+1,m}] \quad (5.2.16b)$$

$$\bar{M}_{l-1,m} = \frac{1}{A_{\alpha 0}} \left( 1 + \frac{\Delta x A_{\alpha 0}}{A_{\alpha 0}} \right) [-B_{l-1,m} \bar{U}_{yy,l-1,m} + \bar{F}_{l-1,m}] \quad (5.2.16c)$$

$$\bar{m}_{l+1,m} = \bar{U}_{x,l+1,m} = \frac{u_{l+1,m} - u_{l,m}}{\Delta x} + \frac{\Delta x}{6} [\bar{M}_{l,m} + 2\bar{M}_{l+1,m}] \quad (5.2.17a)$$
The cubic spline approximations (5.2.16a)-(5.2.16c) and (5.2.17a), (5.2.17b) are discussed in details in [43]. Then at each internal grid point \((x_{l+1}, y_{m})\), the cubic spline method with accuracy of \(O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)\) for the solution of nonlinear elliptic partial differential equation (5.2.1) may be written as

\[
\begin{align*}
L_u & \equiv p^2 \left[ A_{00} - \frac{\Delta x^2}{6} A_{10} A_{10} + \frac{\Delta x^2}{12} A_{20} \right] \delta_x^2 U_{l,m} \\
& + \frac{\Delta y^2}{12} \left[ \left(1 - \frac{\Delta x A_{10}}{A_{00}}\right) B_{l+1,m} \delta_{yy} U_{l+1,m} + \left(1 + \frac{\Delta x A_{10}}{A_{00}}\right) B_{l-1,m} \delta_{yy} U_{l-1,m} + 10B_{l,m} \delta_{yy} U_{l,m} \right] \\
& = \frac{\Delta y^2}{12} \left[ \left(1 - \frac{\Delta x A_{10}}{A_{00}}\right) \delta_{x}^2 U_{l+1,m} + \left(1 + \frac{\Delta x A_{10}}{A_{00}}\right) \delta_{x}^2 U_{l-1,m} + 10 \delta_{x}^2 U_{l,m} \right] + \delta_{x}^2 U_{l,m},
\end{align*}
\]

where, \(\delta_x U_l = \left( U_{l+1} - U_{l-1} \right) \) and \(\mu_x U_l = \frac{1}{2} \left( U_{l+1} + U_{l-1} \right) \) are the central and average difference operators with respect to \(x\)-direction and the local truncation error \(T_{l,m} = O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta y^2 \Delta x^4)\).

### 5.3 Derivation of the cubic spline scheme

For the derivation of the numerical method (5.2.19) for the solution of partial differential equation (5.2.1), we follow the ideas given by Jain and Aziz [43].

At the grid point \((x_l, y_m)\), we may write the differential equation (5.2.1) as

\[
A_{l,m} U_{xx,l,m} + B_{l,m} U_{yy,l,m} = f(x_l, y_m, U_{l,m}, U_{x,l,m}, U_{y,l,m}) \equiv F_{l,m} \text{ (say)}
\]

Using Taylor’s expansion, we may write the approximations as

\[
\begin{align*}
\bar{U}_{y,l,m} &= \frac{(U_{l,m+1} - U_{l,m-1})}{(2\Delta y)} = U_{y,l,m} + \frac{\Delta y^2}{6} U_{03} + O(\Delta y^4) \\
\bar{U}_{y,l+1,m} &= \frac{(U_{l+1,m+1} - U_{l+1,m-1})}{(2\Delta y)} = U_{y,l+1,m} + \frac{\Delta y^2}{6} U_{03} + \frac{\Delta x \Delta y^2}{6} U_{13} + O(\Delta y^2 \Delta x^2)
\end{align*}
\]
Using Taylor series expansion about the grid point \((x_l, y_m)\), from equation (5.2.1) in the absence of first derivative terms we obtain

\[
\bar{U}_{y l-1,m} = \frac{(U_{l-1,m+1} - U_{l-1,m-1})}{(2\Delta y)} = U_{y l-1,m} + \frac{\Delta y^2}{6} U_{03} - \frac{\Delta x \Delta y^2}{6} U_{13} + O(\Delta y^2 \Delta x^2) \quad (5.3.2c)
\]

\[
\bar{U}_{y l,m} = \frac{(U_{l+1,m+1} - 2U_{l,m} + U_{l-1,m-1})}{\Delta y^2} = U_{y l,m} + \frac{\Delta y^2}{12} U_{04} + O(\Delta y^4) \quad (5.3.2d)
\]

\[
\bar{U}_{y l+1,m} = \frac{(U_{l+1,m+1} - 2U_{l+1,m} + U_{l+1,m-1})}{\Delta y^2} = U_{y l+1,m} + \frac{\Delta y^2}{12} U_{04} + \frac{\Delta x \Delta y^2}{12} U_{14} + O(\Delta y^2 \Delta x^2) \quad (5.3.2e)
\]

\[
\bar{U}_{y l-1,m} = \frac{(U_{l-1,m+1} - 2U_{l-1,m} + U_{l-1,m-1})}{\Delta y^2} = U_{y l-1,m} + \frac{\Delta y^2}{12} U_{04} - \frac{\Delta x \Delta y^2}{12} U_{14} + O(\Delta y^2 \Delta x^2) \quad (5.3.2f)
\]

\[
\bar{U}_{xx l,m} = \frac{U_{l+1,m} - 2U_{l,m} + U_{l-1,m}}{\Delta x^2} = U_{xx l,m} + \frac{\Delta x^2}{12} U_{40} + O(\Delta x^4) \quad (5.3.2g)
\]

\[
\bar{m}_{l,m} = \bar{U}_{xl,m} = \frac{(U_{l+1,m} - U_{l-1,m})}{(2\Delta x)} = U_{xl,m} + \frac{\Delta x^2}{6} U_{30} + O(\Delta x^4) \quad (5.3.3a)
\]

\[
\bar{m}_{l+1,m} = \bar{U}_{xl+1,m} = \frac{(3U_{l+1,m} - 4U_{l,m} + U_{l-1,m})}{(2\Delta x)} = U_{xl+1,m} - \frac{\Delta x^2}{3} U_{30} + O(\Delta x^3 + \Delta x^4) \quad (5.3.3b)
\]

\[
\bar{m}_{l-1,m} = \bar{U}_{xl-1,m} = \frac{(-3U_{l-1,m} + 4U_{l,m} - U_{l+1,m})}{(2\Delta x)} = U_{xl-1,m} - \frac{\Delta x^2}{3} U_{30} + O(-\Delta x^3 + \Delta x^4) \quad (5.3.3c)
\]

Using Taylor series expansion about the grid point \((x_l, y_m)\), from equation (5.2.1) in the absence of first derivative terms we obtain

\[
L_l = \frac{\Delta y^2}{12} \left[ \left( 1 - \frac{\Delta x A_{10}}{A_{00}} \right) F_{l+1,m} + \left( 1 + \frac{\Delta x A_{10}}{A_{00}} \right) F_{l-1,m} + 10F_{l,m} \right] + O(\Delta y^4 + \Delta y^2 \Delta x^2 + \Delta y^2 \Delta x^4) \quad ; \ l = 1, 2, \ldots, N, m = 1, 2, \ldots, M \quad (5.3.4)
\]

Let us denote \( \alpha_{l,m} = \left( \frac{\partial f}{\partial U_{x,l,m}} \right) \). \( \alpha_{l,m} = \left( \frac{\partial f}{\partial U_{x,l,m}} \right) \). \( \alpha_{l,m} = \left( \frac{\partial f}{\partial U_{x,l,m}} \right) \). \( \alpha_{l,m} = \left( \frac{\partial f}{\partial U_{x,l,m}} \right) \).

Then,

\[
\alpha_{l+1,m} = \alpha_{l,m} + \Delta x \alpha_{l,m} + O(\Delta x^2)
\]

\[
\alpha_{l-1,m} = \alpha_{l,m} - \Delta x \alpha_{l,m} + O(\Delta x^2)
\]

With the help of the approximations (5.3.2a)-(5.3.2c) and (5.3.3a)-(5.3.3c), we obtain

\[
F_{l,m} = f(x_l, y_m, U_{l,m}, \bar{m}_{l,m}, U_{y l,m})
\]

\[
= f(x_l, y_m, U_{l,m}, U_{xl,m} + \frac{\Delta x^2}{6} U_{30} + O(\Delta x^4), U_{yl,m} + O(\Delta y^2))
\]

162
\[ F_{l,m} = F_{l,m} + \frac{\Delta x^2}{6} U_{30} \alpha_{l,m} + O(\Delta y^2 + \Delta x^4) \]  \hspace{0.5cm} (5.3.6)

\[ \bar{F}_{l+1,m} = f(x_{l+1}, y_{l+1}, U_{l+1,m}, \bar{U}_{y_{l+1},m}, U_{x_{l+1},m}) \]

\[ = f(x_{l+1}, y_{l+1}, U_{l+1,m}, U_{x_{l+1},m} - \frac{\Delta x^2}{3} U_{30} + O(\Delta x^3 + \Delta x^4), U_{y_{l+1},m} + \frac{\Delta y^2}{6} U_{03} \]

\[ + O(\Delta x \Delta y^2 + \Delta y^2 \Delta x^2)) \]

\[ = F_{l+1,m} [-\frac{\Delta x^2}{3} U_{30} + O(\Delta x^3 + \Delta x^4)] \alpha_{l+1,m} + O(\Delta y^2 + \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \]

\[ \Rightarrow \bar{F}_{l+1,m} = F_{l+1,m} - \frac{\Delta x^2}{3} U_{30} \alpha_{l,m} + O(\Delta x^3 + \Delta x^4 + \Delta y^2 + \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \]  \hspace{0.5cm} (5.3.7)

\[ \bar{F}_{l-1,m} = f(x_{l-1}, y_{l-1}, U_{l-1,m}, \bar{U}_{y_{l-1},m}) \]

\[ = f(x_{l-1}, y_{l-1}, U_{l-1,m}, U_{x_{l-1},m} - \frac{\Delta x^2}{3} U_{30} + O(-\Delta x^3 + \Delta x^4), U_{y_{l-1},m} \]

\[ + O(\Delta y^2 - \Delta x \Delta y^2 + \Delta y^2 \Delta x^2)) \]

\[ = F_{l-1,m} - \frac{\Delta x^2}{3} U_{30} \alpha_{l-1,m} + O(-\Delta x^3 + \Delta x^4 + \Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \]

\[ \Rightarrow \bar{F}_{l-1,m} = F_{l-1,m} - \frac{\Delta x^2}{3} U_{30} \alpha_{l,m} + O(-\Delta x^3 + \Delta x^4 + \Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \]  \hspace{0.5cm} (5.3.8)

We have \( M_{l,m} = S_m''(x_l) = U_{xxl,m} \)

\[ \Rightarrow \bar{M}_{l,m} = \bar{U}_{xxl,m} = \frac{1}{A_{00}} [-B_{00} \bar{U}_{yyl,m} + \bar{F}_{l,m}] \]  \hspace{0.5cm} (5.3.9)

\[ \bar{M}_{l+1,m} = \bar{U}_{xxl+1,m} = \frac{1}{A_{l+1,m}} [-B_{l+1,m} \bar{U}_{yyl+1,m} + \bar{F}_{l+1,m}] \]

\[ = \frac{1}{(A_{l,m} + \Delta x A_{10} + \Delta x^2 A_{20} + \ldots)} [-B_{l+1,m} \bar{U}_{yyl+1,m} + \bar{F}_{l+1,m}] \]

\[ = \frac{1}{A_{l,m}(1 + \Delta x \frac{A_{10}}{A_{l,m}} + \ldots)^{-1}} [-B_{l+1,m} \bar{U}_{yyl+1,m} + \bar{F}_{l+1,m}] \]

\[ = \frac{1}{A_{l,m}} (1 - \Delta x \frac{A_{10}}{A_{l,m}} + \ldots)^{-1} [-B_{l+1,m} \bar{U}_{yyl+1,m} + \bar{F}_{l+1,m}] \]  \hspace{0.5cm} (5.3.10)

We may write (5.3.10) using the notation (5.2.2) as

\[ \bar{M}_{l+1,m} = \frac{1}{A_{00}} (1 - \frac{\Delta x A_{10}}{A_{00}}) [-B_{l+1,m} \bar{U}_{yyl+1,m} + \bar{F}_{l+1,m}] \]  \hspace{0.5cm} (5.3.11)

Similarly,

\[ \bar{M}_{l-1,m} = \bar{U}_{xxl-1,m} = \frac{1}{A_{l-1,m}} [-B_{l-1,m} \bar{U}_{yyl-1,m} + \bar{F}_{l-1,m}] \]
Now, we have
\[ U_{xx} = \frac{1}{A(x,y)} [-B(x,y)U_{yy} + F] \]
\[ \Rightarrow U_{xxx} = \frac{-A_x}{A^2} (-BU_{yy} + F) + \frac{1}{A} (-B_x U_{yy} - BU_{xyy} + F_x) \]

Using (5.3.9)-(5.3.12) we obtain
\[
\bar{m}_{t+1,m} = \bar{U}_{x,t+1,m} = \frac{U_{t+1,m} - U_{t,m}}{\Delta x} + \frac{\Delta x}{6} \left[ \bar{M}_{t,m} + 2\bar{M}_{t+1,m} \right]
\]
\[ \Rightarrow \bar{m}_{t+1,m} = U_{x,t,m} + \Delta x U_{20} + \frac{\Delta x^2}{2} U_{30} + O(\Delta x^3 + \Delta x^4 + \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \] \hspace{1em} (5.3.13)

and
\[
\bar{m}_{t-1,m} = \bar{U}_{x,t-1,m} = \frac{U_{t,m} - U_{t-1,m}}{\Delta x} - \frac{\Delta x}{6} \left[ \bar{M}_{t,m} + 2\bar{M}_{t-1,m} \right]
\]
\[ \Rightarrow \bar{m}_{t-1,m} = U_{x,t,m} - \Delta x U_{20} + \frac{\Delta x^2}{2} U_{30} + O(-\Delta x^3 + \Delta x^4 + \Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \] \hspace{1em} (5.3.14)

Also considering,
\[
\bar{U}_{x+1,t,m} - U_{x+1,t,m} = U_{x,t,m} + \Delta x U_{20} + \frac{\Delta x^2}{2} U_{30} + O(\Delta x^3 + \Delta x^4 + \Delta y^2 + \Delta y^2 \Delta x)
\]
\[ + \Delta y^2 \Delta x^2) - [U_{x,t,m} + \Delta x U_{20} + \frac{\Delta x^2}{2} U_{30} + O(\Delta x^3)] \]
\[ \Rightarrow \bar{U}_{x+1,t,m} = U_{x,t+1,m} + O(\Delta x^3 + \Delta x^4 + \Delta y^2 + \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \]
or,
\[
\bar{m}_{t+1,m} = m_{t+1,m} + O(\Delta x^3 + \Delta x^4 + \Delta y^2 + \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \] \hspace{1em} (5.3.15)

As done above, we may obtain
\[
\bar{U}_{x-1,t,m} - U_{x-1,t,m} = U_{x,t,m} - \Delta x U_{20} + \frac{\Delta x^2}{2} U_{30} + O(\Delta x^4 + \Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2)
\]
\[ + \Delta y^2 \Delta x^2) - [U_{x,t,m} - \Delta x U_{20} + \frac{\Delta x^2}{2} U_{30} - O(\Delta x^3)] \]
\[ \Rightarrow \bar{U}_{x-1,t,m} = U_{x,t-1,m} + O(-\Delta x^3 + \Delta x^4 + \Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \]
or, \[ \bar{m}_{l-1,m} = m_{l-1,m} + O(-\Delta x^3 + \Delta x^4 + \Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \] (5.3.16)

Now we need \( O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \)-approximation for \( \bar{U}_{x,l,m} \).

Let us consider

\[
\bar{U}_{x,l,m} = \bar{U}_{x,l,m} + a \Delta x [ \bar{F}_{l+1,m} - \bar{F}_{l-1,m} ] + b \Delta x [ \bar{U}_{y,y+l+1,m} - \bar{U}_{y,y-l-1,m} ] \\
+ c \Delta x^2 \bar{U}_{x,x+l,m} + d \Delta x^2 \bar{U}_{y,y+l,m} 
\] (5.3.17)

where ‘\( a \)’, ‘\( b \)’, ‘\( c \)’ and ‘\( d \)’ are free parameters to be determined.

\[
\Rightarrow \bar{U}_{x,l,m} = U_{x,l,m} + \frac{\Delta x^2}{6} U_{30} + a \Delta x [ F_{l+1,m} - F_{l-1,m} ] + b \Delta x [ \bar{U}_{y,y+l+1,m} - \bar{U}_{y,y-l-1,m} ] \\
+ c \Delta x^2 U_{x,x+l,m} + d \Delta x^2 U_{y,y+l,m} + O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \\
= U_{x,l,m} + \frac{\Delta x^2}{6} U_{30} + 2a \Delta x^2 F_{x,l,m} + 2b \Delta x^2 U_{x,x+l,m} + c \Delta x^2 U_{x,x+l,m} + d \Delta x^2 U_{y,y+l,m} \\
+ O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \\
= U_{x,l,m} + \frac{\Delta x^2}{6} U_{30} + 2a \Delta x^2 [ A_{00} U_{x,x+l,m} + A_{10} U_{x,x+l,m} + B_{00} U_{x,y+l,m} + B_{10} U_{y,y+l,m} ] \\
+ 2b \Delta x^2 U_{y,y+l,m} + c \Delta x^2 U_{x,x+l,m} + d \Delta x^2 U_{y,y+l,m} + O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \\
\Rightarrow \bar{U}_{x,l,m} = m_{l,m} + \frac{\Delta x^2}{6} [(1 + 12aA_{00}) U_{30} + 12(aB_{00} + b) U_{12} + (6c + 12aA_{10}) U_{20} + \\
(12aB_{10} + 6d) U_{02}] + O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \\
\]

Now, \( \bar{U}_{x,l,m} = m_{l,m} + O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \) if coefficient of \( \Delta x^2 \) is zero which means

\[ 1 + 12aA_{00} = 0 \]
\[ aB_{00} + b = 0 \]
\[ 6c + 12aA_{10} = 0 \]

and
\[ 12aB_{10} + 6d = 0 \]

From above, it is easily seen that, \( a = -\frac{1}{12 A_{00}}, b = \frac{B_{00}}{12 A_{00}}, c = \frac{A_{10}}{6 A_{00}} \) and \( d = \frac{B_{10}}{6 A_{00}} \),

so that (5.3.17) can now be rewritten as

\[
\bar{U}_{x,l,m} = U_{x,l,m} - \frac{\Delta x}{12 A_{00}} [ \bar{F}_{l+1,m} - \bar{F}_{l-1,m} ] + \frac{\Delta x}{12 A_{00}} B_{00} [ \bar{U}_{y,y+l+1,m} - \bar{U}_{y,y-l-1,m} ] \\
+ \frac{\Delta x^2 A_{10}}{6 A_{00}} U_{x,x+l,m} + \frac{\Delta x^2 B_{10}}{6 A_{00}} U_{y,y+l,m} \\
= m_{l,m} + O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) 
\] (5.3.18)

Now,
\[ F_{t+1,m} = f(x_{t+1}, y_m, U_{t+1,m}, \bar{m}_{t+1,m}, \bar{U}_{y_{t+1,m}}) \]
\[ = f(x_{t+1}, y_m, U_{t+1,m}, m_{t+1,m} + O(\Delta x^3 + \Delta x^4 + \Delta y^2 \Delta x + \Delta y^2 \Delta x^2), \]
\[ U_{y_{t+1,m}} + O(\Delta y^2 + \Delta y^2 \Delta x + \Delta y^2 \Delta x^2)) \]
\[ = F_{t+1,m} + O(\Delta x^3 + \Delta x^4 + \Delta y^2 + \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \]  \hspace{1cm} (5.3.19)

\[ F_{t-1,m} = f(x_{t-1}, y_m, U_{t-1,m}, \bar{m}_{t-1,m}, \bar{U}_{y_{t-1,m}}) \]
\[ = f(x_{t-1}, y_m, U_{t-1,m}, m_{t-1,m} + O(-\Delta x^3 + \Delta x^4 + \Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2), \]
\[ U_{y_{t-1,m}} + O(\Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2)) \]
\[ = F_{t-1,m} + O(-\Delta x^3 + \Delta x^4 + \Delta y^2 - \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \]  \hspace{1cm} (5.3.20)

\[ F_{t,m} = f(x_t, y_m, U_{t,m}, \bar{U}_{x_{t,m}}, \bar{U}_{y_{t,m}}) \]
\[ = f(x_t, y_m, U_{t,m}, m_{t,m} + O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)) \]
\[ = F_{t,m} + O(\Delta x^4 + \Delta y^2 + \Delta y^2 \Delta x^2) \]  \hspace{1cm} (5.3.21)

Finally, using the preceding approximations, from (5.2.19) and (5.3.4), we obtain the local truncation error as \( \tilde{T}_{t,m} = O(\Delta y^4 + \Delta y^2 \Delta x^2 + \Delta y^2 \Delta x^4) \).

Now we describe the numerical method of \( O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \) for the solution of the quasi-linear elliptic equation (5.1.1). Whenever, the coefficients \( A \) and \( B \) are functions of \( x, y \) and \( u \), the difference scheme (5.2.10) needs to be modified. For this purpose, we use the following central differences.

Let
\[ A_{10} = (A_{t+1,m} - A_{t-1,m})/(2\Delta x) \] \hspace{1cm} (5.3.22a)
\[ A_{20} = (A_{t+1,m} - 2A_{t,m} + A_{t-1,m})/(\Delta x^2) \] \hspace{1cm} (5.3.22b)
\[ B_{10} = (B_{t+1,m} - B_{t-1,m})/(2\Delta x) \] \hspace{1cm} (5.3.22c)

where now \( A_{00} = A_{t,m} = A(x_t, y_m, U_{t,m}) \), \( A_{t \pm 1,m} = A(x_{t \pm 1,m}, y_m, U_{t \pm 1,m}) \) etc.

With the help of the approximations (5.3.22a)-(5.3.22c), we see that
\[ A_{00} = \frac{\Delta x^2 A_{10}}{6 A_{00}} A_{10} + \frac{\Delta x^2}{12} A_{20} \]
\[ = A_{t,m} - \frac{1}{24 A_{t,m}} (A_{t+1,m} - A_{t-1,m})^2 + \frac{1}{12} (A_{t+1,m} - 2A_{t,m} + A_{t-1,m}) + O(\Delta x^4) . \]

Thus substituting the central difference approximations (5.3.22a)-(5.3.22c) into (5.2.19), we obtain the required numerical method of \( O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \) based on cubic spline approximation for the solution of quasi-linear elliptic partial differential equation (5.1.1).
Note that, the Dirichlet boundary conditions are given by (5.1.3). Incorporating the boundary conditions, we can write the cubic spline method (5.2.19) in a tri-block diagonal matrix form. If the differential equation (5.1.1) is linear, we can solve the linear system using Gauss-Seidel (tri-diagonal solver) method; in the nonlinear case, we can use Newton-Raphson iterative method to solve the nonlinear system (see [41, 50, 89, 99]).

5.4 Convergence Theory

For simplicity, we consider the nonlinear elliptic differential equation

\[ u_{xx} + u_{yy} = f(x, y, u, u_x, u_y) \]  

(subject to the Dirichlet boundary conditions \( u(x, y) = \psi(x, y), (x, y) \in \partial \Omega \).)

Our cubic spline scheme (5.2.19) for which becomes

\[
p^2 \delta_x^2 U_{l,m} + \frac{\Delta y^2}{12} \left[ U_{yy, l+1,m} + U_{yy, l-1,m} + 10 U_{y, l,m} \right] \]

\[ = \frac{\Delta y^2}{12} \left[ \hat{F}_{l+1,m} + \hat{F}_{l-1,m} + 10 \hat{F}_{l,m} \right] + \hat{T}_{l,m} \]  

(5.4.2)

where \( p = (\Delta y/\Delta x) > 0 \) is the mesh aspect ratio and \( \hat{T}_{l,m} = O(\Delta y^4 + \Delta y^4 \Delta x^2 + \Delta y^2 \Delta x^4) \).

The difference scheme (5.4.2) then at each \((l, m), [l = 1(1)N, m = 1(1)M]\) may be written as the nine-point scheme

\[
\lambda_1 (U_{l+1,m} + U_{l-1,m}) + \lambda_2 (U_{l,m+1} + U_{l,m-1}) \\
+ \lambda_3 (U_{l+1,m+1} + U_{l+1,m-1} + U_{l-1,m+1} + U_{l-1,m-1} - (24 p^2 + 20) U_{l,m}) \\
= \frac{\Delta y^2}{12} \left[ \hat{F}_{l+1,m} + \hat{F}_{l-1,m} + 10 \hat{F}_{l,m} \right] + \hat{T}_{l,m} 
\]  

(5.4.3)

where \( \lambda_1 = p^2 - \frac{2}{12}, \ \lambda_2 = \frac{10}{12}, \ \lambda_3 = \frac{1}{12} \).

We next show that, under appropriate conditions, difference method (5.4.3) for elliptic equation (5.4.1) is \( O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \)-convergent. The condition which is usually imposed on equation (5.4.3) is that \( \lambda_1 > 0 \).

Let \( U = [U_{11}, U_{21}, \ldots, U_{N1}, U_{12}, U_{22}, \ldots, U_{N2}, \ldots, U_{1M}, U_{2M}, \ldots, U_{NM}]^T \) be the solution vector and \( T = [\hat{T}_{11}, \hat{T}_{21}, \ldots, \hat{T}_{N1}, \hat{T}_{12}, \hat{T}_{22}, \ldots, \hat{T}_{N2}, \ldots, \hat{T}_{1M}, \hat{T}_{2M}, \ldots, \hat{T}_{NM}]^T \) be the local truncation error vector.

Let \( \phi_{l,m} = \frac{\Delta y^2}{12} \left[ \hat{F}_{l+1,m} + \hat{F}_{l-1,m} + 10 \hat{F}_{l,m} \right] \) and

\[
\phi(U) = [\phi_{11}, \phi_{21}, \ldots, \phi_{N1}, \phi_{12}, \phi_{22}, \ldots, \phi_{N2}, \ldots, \phi_{1M}, \phi_{2M}, \ldots, \phi_{NM}]^T.
\]
Then the finite difference scheme in the matrix form can be written as

\[ DU + \phi(U) + T = 0 \]  (5.4.4)

where,

\[ D = [D_1, D_2, D_3]_{NM \times NM} \] is a tri-blockdiagonal matrix, where

\[ D_1 = [-\lambda_3, -\lambda_2, -\lambda_3]_{N \times N} \]

\[ D_2 = [-\lambda_1, (24\gamma^2 + 20)\lambda_3, -\lambda_1]_{N \times N} \] denote the \( N \times N \) tridiagonal matrices.

The method consists in finding approximations \( \mathbf{u} \) for \( \mathbf{U} \) by solving the \( NM \times NM \) system

\[ D\mathbf{u} + \phi(\mathbf{u}) = 0 \]  (5.4.5)

Let \( \epsilon_{l,m} = u_{l,m} - U_{l,m} \), \( l = 1(1)N, m = 1(1)M \)

and let the error vector be \( \mathbf{E} = \mathbf{u} - \mathbf{U} = [\epsilon_{11}, \epsilon_{21}, \ldots, \epsilon_{N1}, \ldots, \epsilon_{1M}, \epsilon_{2M}, \ldots, \epsilon_{NM}]^T \).  (5.4.6)

Also, let

\[ \bar{f}_{l \pm 1,m} = f(x_{l \pm 1}, y_m, u_{l \pm 1,m}, \bar{u}_{x,l \pm 1,m}, \bar{u}_{y,l \pm 1,m}) \approx \bar{f}_{l \pm 1,m} \]  (5.4.7a)

\[ \hat{f}_{l,m} = f(x_l, y_m, u_{l,m}, \bar{u}_{x,l \pm 1,m}, \bar{u}_{y,l \pm 1,m}) \approx \hat{f}_{l,m} \]  (5.4.7b)

Then we may write

\[ \bar{u}_{x,l \pm 1,m} - \bar{U}_{x,l \pm 1,m} = (\pm 3 \epsilon_{l \pm 1,m} \mp 4 \epsilon_{l,m} \pm \epsilon_{l \mp 1,m})/(2\Delta x) \]  (5.4.8a)

\[ \bar{u}_{x,l,m} - \bar{U}_{x,l,m} = (\epsilon_{l+1,m} - \epsilon_{l-1,m})/(2\Delta x) \]  (5.4.8b)

\[ \bar{u}_{y,l \pm 1,m} - \bar{U}_{y,l \pm 1,m} = (\epsilon_{l \pm 1,m+1} - \epsilon_{l \pm 1,m-1})/(2\Delta y) \]  (5.4.8c)

\[ \bar{u}_{y,l,m} - \bar{U}_{y,l,m} = (\epsilon_{l+1,m} - \epsilon_{l-1,m})/(2\Delta y) \]  (5.4.8d)

\[ \bar{u}_{yy,l \pm 1,m} - \bar{U}_{yy,l \pm 1,m} = (\epsilon_{l \pm 1,m+1} - 2 \epsilon_{l \pm 1,m} + \epsilon_{l \pm 1,m-1})/\Delta y^2 \]  (5.4.8e)

\[ \bar{f}_{l+1,m} - F_{l+1,m} = \epsilon_{l+1,m} P_{l+1,m}^{(1)} + (\bar{u}_{x,l+1,m} - \bar{U}_{x,l+1,m}) Q_{l+1,m}^{(1)} \]
\[ + (\bar{u}_{y,l+1,m} - \bar{U}_{y,l+1,m}) R_{l+1,m}^{(1)} \]  (5.4.9a)

\[ \bar{f}_{l-1,m} - F_{l-1,m} = \epsilon_{l-1,m} P_{l-1,m}^{(1)} + (\bar{u}_{x,l-1,m} - \bar{U}_{x,l-1,m}) Q_{l-1,m}^{(1)} \]
\[ + (\bar{u}_{y,l-1,m} - \bar{U}_{y,l-1,m}) R_{l-1,m}^{(1)} \]  (5.4.9b)

\[ \bar{f}_{l+1,m} - F_{l+1,m} = \epsilon_{l+1,m} P_{l+1,m}^{(1)} + (\bar{u}_{x,l+1,m} - \bar{U}_{x,l+1,m}) Q_{l+1,m}^{(1)} \]
\[ + (\bar{u}_{y,l+1,m} - \bar{U}_{y,l+1,m}) R_{l+1,m}^{(1)} \]  (5.4.10a)
\[ f_{l-1,m} - \tilde{f}_{l-1,m} = \varepsilon_{l-1,m} P_{l-1,m}^{(1)} + (\tilde{u}_{x,l-1,m} - \tilde{u}_{x,l-1,m}) Q_{l-1,m}^{(1)} \]
\[ + (\tilde{u}_{y,l-1,m} - \tilde{u}_{y,l-1,m}) R_{l-1,m}^{(1)} , \] (5.4.10b)

and
\[ \tilde{f}_{l,m} - \tilde{f}_{l,m} = \varepsilon_{l,m} P_{l,m}^{(2)} + (\tilde{u}_{x,l,m} - \tilde{u}_{x,l,m}) Q_{l,m}^{(2)} + (\tilde{u}_{y,l,m} - \tilde{u}_{y,l,m}) R_{l,m}^{(2)} \] (5.4.10c)

for suitable \( I_{l+1,m}^{(1)} \) and \( I_{l,m}^{(2)} \) where \( I = P, Q \) and \( R \).

Also, we may write
\[
\begin{align*}
P_{l \pm 1,m}^{(1)} &= P_{l,m}^{(1)} \pm O(D\varepsilon) , \\
Q_{l \pm 1,m}^{(1)} &= Q_{l,m}^{(1)} \pm D\varepsilon Q_{x,l+1,m}^{(1)} + O(D\varepsilon^{2}), \\
R_{l \pm 1,m}^{(1)} &= R_{l,m}^{(1)} \pm D\varepsilon R_{x,l+1,m}^{(1)} + O(D\varepsilon^{2})
\end{align*}
\] (5.4.11)

Now,
\[
\begin{align*}
\tilde{u}_{x,l+1,m} - \tilde{u}_{x,l+1,m} &= \left( \tilde{u}_{x,l+1,m} - \tilde{u}_{x,l+1,m} \right) - \frac{\Delta x}{12} \left[ (\tilde{f}_{l+1,m} - \tilde{f}_{l+1,m}) - (\tilde{f}_{l-1,m} - \tilde{f}_{l-1,m}) \right] \\
&+ \frac{\Delta x}{12} \left[ (\tilde{u}_{y,l+1,m} - \tilde{u}_{y,l+1,m}) - (\tilde{u}_{y,l+1,m} - \tilde{u}_{y,l+1,m}) \right] \\
&= \frac{(\varepsilon_{l+1,m} - \varepsilon_{l-1,m})}{2\Delta x} \left[ \varepsilon_{l+1,m} P_{l+1,m}^{(1)} + \left( \frac{3\varepsilon_{l+1,m} - 4\varepsilon_{l,m} + \varepsilon_{l-1,m}}{2\Delta x} \right) Q_{l+1,m}^{(1)} \right] \\
&+ \left( \frac{(\varepsilon_{l+1,m} - \varepsilon_{l+1,m})}{2\Delta y} \right) R_{l+1,m}^{(1)} + \frac{\Delta x}{12} \left[ (\varepsilon_{l-1,m} - \varepsilon_{l-1,m}) \right] \\
&+ \left( \frac{(\varepsilon_{l-1,m} - \varepsilon_{l-1,m})}{2\Delta y} \right) Q_{l-1,m}^{(1)} + \left( \frac{(\varepsilon_{l-1,m} - \varepsilon_{l-1,m})}{2\Delta y} \right) R_{l-1,m}^{(1)} \\
&+ \frac{\Delta x}{12} \left[ (\varepsilon_{l+1,m} - \varepsilon_{l+1,m}) \right] - \left( \frac{(\varepsilon_{l-1,m} - \varepsilon_{l-1,m})}{2\Delta y} \right)
\end{align*}
\] (5.4.12)

\[
\begin{align*}
\tilde{u}_{x,l+1,m} - \tilde{u}_{x,l+1,m} &= \frac{(\varepsilon_{l+1,m} - \varepsilon_{l-1,m})}{\Delta x} + \frac{1}{6\Delta x} \left[ (\varepsilon_{l+1,m} - 2\varepsilon_{l,m} + \varepsilon_{l-1,m}) \right] \\
&- \frac{\Delta x}{3\Delta y^{2}} \left[ (\varepsilon_{l+1,m} + 2\varepsilon_{l,m} + \varepsilon_{l-1,m}) \right] \\
&+ \frac{\Delta x}{3} \left( \varepsilon_{l+1,m} + \varepsilon_{l-1,m} \right) Q_{l+1,m}^{(1)} \\
&+ \frac{\Delta x}{3} \left( \varepsilon_{l+1,m} + \varepsilon_{l-1,m} \right) R_{l+1,m}^{(1)}
\end{align*}
\] (5.4.13)

\[
\begin{align*}
\tilde{u}_{x,l+1,m} - \tilde{u}_{x,l+1,m} &= \frac{(\varepsilon_{l+1,m} - \varepsilon_{l-1,m})}{\Delta x} - \frac{1}{6\Delta x} \left[ (\varepsilon_{l+1,m} - 2\varepsilon_{l,m} + \varepsilon_{l-1,m}) \right] \\
&+ \frac{\Delta x}{3\Delta y^{2}} \left[ (\varepsilon_{l-1,m} + 2\varepsilon_{l,m} + \varepsilon_{l-1,m}) \right] - \frac{\Delta x}{3} \varepsilon_{l+1,m} P_{l+1,m}^{(1)} \\
&+ \frac{\Delta x}{3} (3\varepsilon_{l+1,m} + 4\varepsilon_{l,m} - \varepsilon_{l-1,m}) Q_{l+1,m}^{(1)}
\end{align*}
\]
Subtracting (5.4.4) from (5.4.5), we get
\[
\textbf{D}E + \phi(u) - \phi(U) = T
\]
where \( E \) is the error vector defined in (5.4.6).

Now,
\[
\phi(u) - \phi(U) = \frac{\Delta y^2}{12} \left[ (\bar{f}_{l+1,m} - \bar{f}_{l+1,m}) + (\bar{f}_{l-1,m} - \bar{f}_{l-1,m}) + 10 \left( \bar{f}_{l,m} - \bar{f}_{l,m} \right) \right]
\]
where
\[
\bar{f}_{l+1,m} - \bar{f}_{l+1,m} = \epsilon_{l+1,m} P_{l+1,m}^{(1)} + \left( \frac{\epsilon_{l+1,m} - \epsilon_{l,m}}{\Delta x} \right) + \frac{1}{6\Delta x} \left( \epsilon_{l+1,m} - 2 \epsilon_{l,m} + \epsilon_{l-1,m} \right)
\]
\[- \frac{\Delta y}{3} \left( \epsilon_{l+1,m+1} - 2 \epsilon_{l+1,m} + \epsilon_{l+1,m-1} \right) + \frac{\Delta x}{3} \epsilon_{l+1,m} P_{l+1,m}^{(1)}
\]
\[- \frac{1}{6} \left( 3 \epsilon_{l+1,m} - 4 \epsilon_{l,m} + \epsilon_{l-1,m} \right) Q_{l+1,m}^{(1)} + \frac{\epsilon_{l+1,m} - \epsilon_{l+1,m-1}}{2\Delta y} P_{l+1,m}^{(1)}
\]
\[
\bar{f}_{l-1,m} - \bar{f}_{l-1,m} = \epsilon_{l-1,m} P_{l-1,m}^{(1)} + \left( \frac{\epsilon_{l,m} - \epsilon_{l-1,m}}{\Delta x} \right) - \frac{1}{6\Delta x} \left( \epsilon_{l+1,m} - 2 \epsilon_{l,m} + \epsilon_{l-1,m} \right)
\]
\[- \frac{\Delta y}{3} \left( \epsilon_{l-1,m+1} - 2 \epsilon_{l-1,m} + \epsilon_{l-1,m-1} \right) - \frac{\Delta x}{3} \epsilon_{l-1,m} P_{l-1,m}^{(1)}
\]
\[- \frac{1}{6} \left( -3 \epsilon_{l-1,m} + 4 \epsilon_{l,m} - \epsilon_{l+1,m} \right) Q_{l-1,m}^{(1)}
\]
\[- \frac{\epsilon_{l-1,m} - \epsilon_{l-1,m-1}}{2\Delta y} P_{l-1,m}^{(1)} Q_{l-1,m}^{(1)} + \frac{\epsilon_{l-1,m} - \epsilon_{l-1,m-1}}{2\Delta y} P_{l-1,m}^{(1)} Q_{l-1,m}^{(1)}
\]
So that,
\[
\phi(u) - \phi(U) = \frac{\Delta y^2}{12} \epsilon_{l,m} \left[ - \frac{4}{h} Q_{l+1,m}^{(1)} + \frac{2}{h} Q_{l+1,m}^{(1)} + \frac{4}{h} Q_{l-1,m}^{(1)} + \frac{4}{h} Q_{l-1,m}^{(1)} \right]
\]
\[+ 10 P_{l,m}^{(2)} + \frac{4}{24} Q_{l+1,m}^{(2)} + \frac{4}{24} Q_{l-1,m}^{(2)} + \frac{2p}{3\Delta y} Q_{l+1,m}^{(1)}
\]
\[+ \frac{\Delta x}{3} P_{l+1,m}^{(1)} + \frac{1}{6\Delta x} Q_{l+1,m}^{(1)} + \frac{2}{h} Q_{l+1,m}^{(1)} - \frac{1}{6h} Q_{l-1,m}^{(1)}
\]
\[+ \frac{1}{6} Q_{l-1,m}^{(1)} - \frac{10}{2\Delta x} Q_{l-1,m}^{(2)} - \frac{10\Delta x}{12} P_{l+1,m}^{(1)} Q_{l-1,m}^{(2)}
\]
\[+ \frac{30}{24} Q_{l+1,m}^{(1)} Q_{l,m}^{(2)} + \frac{10}{24} Q_{l-1,m}^{(1)} Q_{l,m}^{(2)} - \frac{20p}{3\Delta y} Q_{l-1,m}^{(2)}
\]
\[+ \frac{\Delta y^2}{12} \epsilon_{l-1,m} \left[ P_{l-1,m}^{(1)} - \frac{1}{6\Delta x} Q_{l-1,m}^{(1)} + \frac{1}{6\Delta x} Q_{l+1,m}^{(1)} - \frac{2p}{3\Delta y} Q_{l-1,m}^{(1)}
\]
Then we may write

$$
\begin{align*}
& w = \begin{pmatrix} Q_{l-1,m}^{(1)} & Q_{l-1,m}^{(1)} & \frac{1}{6} Q_{l-1,m}^{(1)} \end{pmatrix} + \begin{pmatrix} \frac{1}{6} Q_{l-1,m}^{(1)} & \frac{1}{2} Q_{l-1,m}^{(1)} & \frac{1}{6} Q_{l-1,m}^{(1)} \end{pmatrix}
& - \frac{1}{24} Q_{l-1,m}^{(1)} Q_{l-1,m}^{(2)} - \frac{10}{24} Q_{l-1,m}^{(1)} Q_{l-1,m}^{(2)} + \frac{10}{24} Q_{l-1,m}^{(1)} Q_{l-1,m}^{(2)} + \frac{20}{12} Q_{l-1,m}^{(1)} Q_{l-1,m}^{(2)} \\
& + \frac{\Delta y^2}{12} \epsilon_{l,m} \left[ \pm \frac{10}{2} R_{l,m}^{(2)} \right] \\
& + \frac{\Delta y^2}{12} \epsilon_{l+1,m+1} \left[ - \frac{p}{3} R_{l+1,m}^{(1)} + \frac{p}{6} R_{l+1,m}^{(1)} Q_{l+1,m}^{(1)} + \frac{1}{2} R_{l+1,m}^{(1)} \\
& - \frac{10 p}{24} R_{l+1,m}^{(1)} Q_{l+1,m}^{(2)} + \frac{10 p}{12} Q_{l+1,m}^{(2)} \\
& + \frac{\Delta y^2}{12} \epsilon_{l-1,m+1} \left[ \frac{p}{3} Q_{l-1,m}^{(1)} - \frac{p}{6} R_{l-1,m}^{(1)} Q_{l-1,m}^{(1)} - \frac{1}{2} R_{l-1,m}^{(1)} \\
& + \frac{10 p}{24} R_{l-1,m}^{(1)} Q_{l-1,m}^{(2)} - \frac{10 p}{12} Q_{l-1,m}^{(2)} \\
& + \frac{\Delta y^2}{12} \epsilon_{l+1,m-1} \left[ - \frac{p}{3} Q_{l+1,m}^{(1)} - \frac{p}{6} R_{l+1,m}^{(1)} Q_{l+1,m}^{(1)} - \frac{1}{2} R_{l+1,m}^{(1)} \\
& + \frac{10 p}{24} R_{l+1,m}^{(1)} Q_{l+1,m}^{(2)} + \frac{10 p}{12} Q_{l+1,m}^{(2)} \\
& + \frac{\Delta y^2}{12} \epsilon_{l-1,m-1} \left[ \frac{p}{3} Q_{l-1,m}^{(1)} + \frac{p}{6} R_{l-1,m}^{(1)} Q_{l-1,m}^{(1)} - \frac{1}{2} R_{l-1,m}^{(1)} \\
& - \frac{10 p}{24} R_{l-1,m}^{(1)} Q_{l-1,m}^{(2)} - \frac{10 p}{12} Q_{l-1,m}^{(2)} \\
& + O(\Delta y^2 \Delta x^2) \\
& \phi(u) - \phi(U) = P.E \\
\end{align*}
$$

(5.4.16)

where \( E \) is the error vector and \( P = (p_{r,s}), \) \( r = 1(1)NM, s = 1(1)NM \) is a tri-block-diagonal matrix with the following matrix elements

$$
\begin{align*}
P_{(m-1)N+l,(m-1)N+l} &= \frac{\Delta y^2}{12} \left[ 10 R_{l,m}^{(2)} - \frac{8}{3} Q_{l,m}^{(1)} - \frac{4}{3} Q_{l,m}^{(1)} Q_{l,m}^{(2)} + \frac{2}{3} Q_{l,m}^{(1)} Q_{l,m}^{(2)} \right] + O(\Delta y^2 \Delta x^2) \\
&= 1(1)N, m = 1(1)M \\
\end{align*}
$$

$$
\begin{align*}
P_{(m-1)N+l,(m-1)N+l+1} &= \frac{\Delta y^2}{12} \left[ \pm p Q_{l,m}^{(1)} \pm \frac{10}{3} p Q_{l,m}^{(2)} \right] \\
&+ \frac{\Delta y^2}{12} \left[ p_{l,m}^{(1)} + \frac{4}{3} Q_{l,m}^{(1)} + \frac{2}{3} p Q_{l,m}^{(1)} + \frac{2}{3} Q_{l,m}^{(1)} Q_{l,m}^{(1)} Q_{l,m}^{(2)} - \frac{40}{24} Q_{l,m}^{(1)} Q_{l,m}^{(2)} \right] + O(\pm \Delta y^2 \Delta x + \Delta y^2 \Delta x^2) \\
\end{align*}
$$

171
We obtain the error equation, in the absence of round-off errors, using relation (5.4.16) in equation (5.4.15) as

\[(D + P)E = T\]  

(5.4.17)

Let

\[P_* = \min_{(x,y) \in \Omega} \partial f \quad \text{and} \quad P^* = \max_{(x,y) \in \Omega} \partial f \quad \text{where} \quad \bar{\Omega} = \Omega + \partial \Omega\]

Then

\[0 < P_* \leq p_{l \pm 1,m}^{(1)}, p_{l,m}^{(2)} \leq P^*\]

and for \(L = Q\) and \(R\), let

\[0 < \left| L_{l \pm 1,m}^{(1)} \right|, \left| L_{l,m}^{(2)} \right| \leq L \quad \text{and} \quad \left| L_{x \in l,m}^{(1)} \right| \leq L^{(1)}\]

for some positive constant \(L^{(1)}\).

For sufficiently small \(\Delta y\), we have

\[
\left| p_{(m-1)N + l, (m-1)N + l \pm 1} \right| < \lambda_1 , \quad \left[ l = 1(1)N - 1, 2(1)N ; m = 1(1)M \right],
\]
\[
\left| p_{(m-1)N + l, (m-1)N + l \pm 1} \right| < \lambda_2 , \quad \left[ l = 1(1)N ; m = 1(1)M - 1, 2(1)M \right],
\]
\[
\left| p_{(m-1)N + l, mN + l \pm 1} \right| < \lambda_3 , \quad \left[ l = 1(1)N - 1, 2(1)N ; m = 1(1)M - 1 \right].
\]
We first show that the matrix $D + P$ is irreducible. This can be followed from the directed graph of $D + P$ (see Fig. 5.2):

The arrows indicate paths $i \to j$ for every non zero entry $(D + P)_{i,j}$ of $D + P$. For $\lambda_1 > 0$, there exists a direct path $(l_1, l_2), (l_3, l_4), \ldots, (l_r, l_j)$ connecting $i$ to $j$ for any ordered pair of nodes $i$ and $j$. Hence the graph is strongly connected and thus $D + P$ is irreducible.

Now let $S_k$ denote the sum of the elements in the $k^{th}$ row of $D + P$, then for $k = 1 \& N$,

$$S_k = (p^2 + 11\lambda_3) + \frac{\Delta y}{12} (b_k + \frac{\Delta y}{3} c_k) + \frac{\Delta y^2}{12} \left[ 10P_{1,k}^{(2)} + P_{1,k}^{(1)} \right] + O(\Delta y^2 \Delta x) \tag{5.4.18a}$$

where

$$b_k = \pm \frac{4p}{3} Q_{k,1}^{(1)} \pm \frac{25p}{6} Q_{k,1}^{(2)} + 5R_{k,1}^{(2)} + \frac{1}{2} R_{k,1}^{(1)},$$

$$c_k = (p^2 - 4) Q_{xk,1}^{(1)} - 2Q_{k,1}^{(1)} Q_{k,1}^{(1)} - 3Q_{k,1}^{(1)} Q_{k,1}^{(2)}$$

$$\pm \frac{p}{2} R_{k,1}^{(1)} Q_{k,1}^{(1)} \pm \frac{3p}{2} R_{k,1}^{(1)} - \frac{5p}{4} R_{k,1}^{(1)} Q_{k,1}^{(2)}$$

$$S_{(M-1)N+k} = (p^2 + 11\lambda_3) + \frac{\Delta y}{12} (b_{(M-1)N+k} + \frac{\Delta y}{3} c_{(M-1)N+k})$$

$$+ \frac{\Delta y^2}{12} \left[ 10P_{k,N}^{(2)} + P_{k,N}^{(1)} \right] + O(\Delta y^2 \Delta x) \tag{5.4.18b}$$

where
\[ b_{(M-1)N+k} = \pm \frac{4p}{3} Q_{k,N}^{(1)} + \frac{25p}{6} Q_{k,N}^{(2)} - 5 R_{k,N}^{(2)} - \frac{1}{2} R_{k,N}^{(1)}, \]

\[ c_{(M-1)N+k} = (p^2 - 4) Q_{x_k,1}^{(1)} - 2 Q_{k,N}^{(1)} Q_{k,N}^{(1)} - 3 Q_{k,N}^{(1)} Q_{k,N}^{(2)} \]

\[ \pm \frac{p}{2} R_{k,N}^{(1)} Q_{k,N}^{(1)} \pm \frac{3p}{2} R_{x_k,1}^{(1)} \pm \frac{5p}{4} R_{k,N}^{(1)} Q_{k,N}^{(2)} \]

For \( r = 2(1)M - 1 \):

\[ S_{(r-1)N+k} = p^2 + \frac{\Delta y}{12} \left[ b_{(r-1)N+k} + \frac{\Delta y}{6} c_{(r-1)N+k} \right] + \frac{\Delta y^2}{12} \left[ 10 p_{k,r}^{(2)} + p_{k,r}^{(1)} \right] \]

\[ + O(\Delta y^2 \Delta x) \] (5.4.18c)

where

\[ b_{(r-1)N+k} = \pm \left( p Q_{k,r}^{(1)} + 5 Q_{k,r}^{(2)} \right); \]

\[ c_{(r-1)N+k} = -8 Q_{x_k,r}^{(1)} - 4 Q_{k,r}^{(1)} Q_{k,r}^{(1)} - 6 Q_{k,r}^{(1)} Q_{k,r}^{(2)} \]

\[ \pm 3p R_{k,r}^{(1)} + \frac{p}{2} R_{x_k,r}^{(1)} \]

For \( q = 2(1)N - 1, \ k = 1 & M \):

\[ S_{(k-1)N+q} = 1 + \frac{\Delta y}{12} \left[ b_{(k-1)N+q} + \frac{\Delta y}{3} c_{(k-1)N+q} \right] + \frac{\Delta y^2}{12} \left[ 10 p_{k,q}^{(2)} + 2 p_{k,q}^{(1)} \right] \]

\[ + O(\Delta y^2 \Delta x) \] (5.4.18d)

where

\[ b_{(k-1)N+q} = \pm \left( R_{q,k}^{(1)} + 5 R_{q,k}^{(2)} \right); \]

\[ c_{(k-1)N+q} = 2p^2 Q_{x_{q,k}}^{(1)} - 8 Q_{q,k}^{(1)} Q_{q,k}^{(2)} \]

and finally,

\[ S_{(r-1)N+q} = 1 + \frac{\Delta y}{12} \left[ \frac{\Delta y}{3} (-8 Q_{q,r}^{(1)} Q_{q,r}^{(2)}) \right] + \frac{\Delta y^2}{12} \left[ 10 p_{q,r}^{(2)} + 2 p_{q,r}^{(1)} \right] + O(\Delta y^2 \Delta x) \] (5.4.18e)

for \( q = 2(1)N - 1, \ r = 2(1)M - 1 \)

Now, with the help of equations (5.4.9), for \( k = 1, N, (M - 1)N + 1 \ & \ NM \)

\[ |b_k| \leq \frac{11}{2} (pQ + R); \]

\[ |c_k| \leq (p^2 - 4) Q^{(1)} + 5Q^2 + \left( \frac{2p+5}{4} \right) RQ + \frac{3}{2} pR^{(1)} \]

For \( k = q \ & \ (M - 1)N + q, \ q = 2(1)N - 1 \)

\[ |b_k| \leq 6R; \ |c_k| \leq 8Q^2 + 2p^2 Q^{(1)} \]

and for \( k = (r-1)N + 1, \ & \ rN, \ r = 2(1)M - 1 \)
\[ |b_k| \leq 6pQ \ ; \ |c_k| \leq 8Q^{(1)} + 10Q^2 + 3pR + \frac{1}{2}pR^{(1)} \]

From the above set of equations, it follows that for sufficiently small \( \Delta y \),

\[ S_k > \frac{11}{12}\Delta y^2 P_k \ ; \ k = 1, N, (M - 1)N + 1 \ \& \ NM \quad (5.4.19a) \]
\[ S_k > \Delta y^2 P_k \ ; \ k = q \ \& \ (M - 1)N + q, \ q = 2(1)N - 1 \quad (5.4.19b) \]
\[ S_k > \frac{11}{12}\Delta y^2 P_k \ ; \ k = (r - 1)N + 1, \ \& \ rN, \ r = 2(1)M - 1 \quad (5.4.19c) \]
\[ S_{(r-1)N+q} \geq \Delta y^2 P_k \ ; \ q = 2(1)N - 1, \ r = 2(1)M - 1 \quad (5.4.19d) \]

Thus for sufficiently small \( \Delta y \), \( D + P \) is monotone. Hence, \( D + P \) is invertible and let \((D + P)^{-1} = J > 0\),

Here \( J = (J_{r,s}) \quad [r = 1(1)NM, s = 1(1)NM] \)

Since, \( \Sigma_{r=1}^{NM} J_{p,r} S_r = 1 \), \( p = 1(1)NM \), using the above equations with \( p = 1(1)NM \), we obtain

\[ J_{p,k} \leq \frac{1}{S_k} \leq \frac{12}{11\Delta y^2 P_k} \quad k = 1, N, (M - 1)N + 1 \ \& \ NM \quad (5.4.20a) \]
\[ \Sigma_{q=2}^{N-1} J_{p,k} \leq \frac{1}{\min_{2s \leq q \leq N - 1} S_k} \leq \frac{1}{\Delta y^2 P_k} \quad k = q \ \& \ (M - 1)N + q \quad (5.4.20b) \]
\[ \Sigma_{r=2}^{M-1} J_{p,k} \leq \frac{1}{\min_{2s \leq r \leq M - 1} S_k} \leq \frac{12}{11\Delta y^2 P_k} \quad k = (r - 1)N + 1, \ \& \ rN \quad (5.4.20c) \]
\[ \Sigma_{q=2}^{N-1} \Sigma_{r=2}^{M-1} J_{p,k} \leq \frac{1}{\min_{2s \leq q \leq N - 1, 2s \leq r \leq M - 1} S_k} \leq \frac{1}{\Delta y^2 P_k} \quad k = (r - 1)N + q \quad (5.4.20d) \]

We may write error equation (5.4.17) as

\[ ||E|| \leq ||J|| \cdot ||T|| \quad (5.4.21) \]

where,

\[ ||J|| = \max_{1 \leq p \leq NM} (J_{p,1} + \Sigma_{q=2}^{N-1} J_{p,q} + J_{p,N}) \]
\[ \quad + (\Sigma_{r=2}^{M-1} J_{p,(r-1)N+1} + \Sigma_{q=2}^{N-1} \Sigma_{r=2}^{M-1} J_{p,(r-1)N+q} + \Sigma_{r=2}^{M-1} J_{p,rN} + J_{p,(M-1)N+1} + \Sigma_{q=2}^{N-1} J_{p,(M-1)N+q} + J_{p,N}) \quad (5.4.22) \]

Substituting equations (5.4.20) in (5.4.22), from equation (5.4.21) we obtain, for sufficiently small \( \Delta y \),

\[ ||E|| \leq O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4). \]

Hence, \( O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4) \) convergence is established.
5.5 Numerical Method for Solving Singular problems

We propose here a modification to the approximating scheme developed earlier that reduces the numerical difficulties associated with singular problems.

Consider the two spatial dimensions elliptic partial differential equation

$$\frac{\partial^2 u}{\partial x^2} + B(x) \frac{\partial^2 u}{\partial y^2} = D(x) \frac{\partial u}{\partial x} + g(x,y), \quad 0 < x, y < 1 \quad (5.5.1)$$

subject to appropriate Dirichlet boundary conditions prescribed. The coefficients $B(x)$, $D(x)$ and function $g(x,y) \in C^2(\Omega)$, where $C^m(\Omega)$ denotes the set of all functions of $x$ and $y$ with continuous partial derivatives upto order $m$, in $\Omega$.

On applying formula (5.2.19) to the elliptic equation (5.5.1), we obtain the following difference scheme

$$p^2 \delta_x^2 U_{l,m} + \frac{\Delta y^2}{12} \left[ B_{l+1} \bar{U}_{yy,l+1,m} + B_{l-1} \bar{U}_{yy,l-1,m} + 10B_l \bar{U}_{yy,l,m} \right]$$

subject to appropriate Dirichlet boundary conditions prescribed.

The coefficients $B(x)$, $D(x)$ and function $g(x,y)$ involve terms like $1/x$, $1/x^2$, $1/xy^3$ and so forth. For an example, if $D(x) = 1/x$, then $D_{l-1} = 1/x_{l-1}$ which blows to infinity at $l = 1$ (since $x_0 = 0$).

So, in order to handle the singularity at $x = 0$, we modify scheme (5.5.2) such that the order and accuracy of the solution is retained throughout the solution region.

For this purpose we would need the following approximations:

$$D_{l \pm 1} = D_{00} \pm \Delta x D_{10} + \frac{\Delta x^2}{2} D_{20} \pm O(\Delta x^3) \quad (5.5.3a)$$

$$B_{l \pm 1} = B_{00} \pm \Delta x B_{10} + \frac{\Delta x^2}{2} B_{20} \pm O(\Delta x^3) \quad (5.5.3b)$$

$$g_{l \pm 1,m} = g_{00} \pm \Delta x g_{10} + \frac{\Delta x^2}{2} g_{20} \pm O(\Delta x^3) \quad (5.5.3c)$$

where $g_{l,m} = g_{00} = g(x_l,y_m)$ etc.

Now, substituting the approximations (5.5.3a)-(5.5.3c) in the difference scheme (5.5.2) and merging the higher order terms in local truncation error, we obtain the modified scheme as

\[ -12p^2 + \frac{4\Delta y^2}{3} D_{10} - \Delta y^2 D_{00}^2 \delta_x^2 U_{l,m} \]

\[ + \left[ p \Delta y \left( 6D_{00} + \frac{\Delta x^2}{2} D_{20} \right) - \frac{\Delta y^2 \Delta x}{6} D_{00} D_{10} \right] (2\mu_x \delta_x) U_{l,m} \]
Note that, the modified scheme (5.5.4) is of $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$ accurate and applicable to both singular and non-singular elliptic differential equations of the form (5.5.1).

Now, consider the Poisson’s equation

$$\frac{\partial^2 u}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 u}{\partial \theta^2} + \frac{1}{r} \frac{\partial u}{\partial r} = [4 - \pi^2] \cos(\pi \theta), \quad 0 < r, \theta < 1$$  \hspace{1cm} (5.5.5)

The above equation represents two-dimensional Poisson’s equation in cylindrical polar coordinates in $r$-$\theta$ plane. This problem arises in the simulation of certain semi-bounded plasmas where the electric potential are to be computed. Replacing the variables $(x,y)$ by $(r, \theta)$ and substituting $B_{00} = 1/r_1^2, B_{10} = -2/r_1^3, B_{20} = 6/r_1^4, D_{00} = -1/r_1, D_{10} = B_{00}, D_{20} = B_{10}$ in (5.5.4), we obtain $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$ scheme for the solution of the elliptic equation (5.5.5).

Similarly, for the 2-dim Poisson’s equation in cylindrical polar coordinates in $r$-$z$ plane

$$\frac{\partial^2 u}{\partial r^2} + \frac{\partial^2 u}{\partial z^2} + \frac{1}{r} \frac{\partial u}{\partial r} = \cosh z \left[ 2 \cosh r + \frac{1}{r} \sinh r \right], \quad 0 < r, z < 1$$  \hspace{1cm} (5.5.6)

We replace the variables $(x,y)$ by $(r,z)$ and setting $B_{00} = 1, B_{10} = 0 = B_{20}, D_{00} = -1/r_1, D_{10} = 1/r_1^3, D_{20} = -2/r_1^3$ in (5.5.4), we can get $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$ scheme for the solution of elliptic equation (5.5.6).

Next consider the Convection-Diffusion equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \beta \frac{\partial u}{\partial x}$$  \hspace{1cm} (5.5.7)

where $\beta > 0$ is a constant and magnitude of $\beta$ determines the ratio of convection to diffusion. Substituting $B(x) = 1.0, D(x) = \beta$ and $g(x,y) = 0$ in the difference scheme (5.5.4) and simplifying, we obtain a difference scheme of $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$ accuracy for the solution of the convection-diffusion equation (5.5.7).

### 5.6 Numerical Illustrations

Substituting the approximations (5.2.13)-(5.2.14) in the differential equation (5.1.1), we obtain a central difference scheme of $O(\Delta y^2 + \Delta x^2)$ of the form

$$A_{l,m} \bar{U}_{xlm} + B_{l,m} \bar{U}_{yml,m} = f(x_l,y_m, U_{l,m}, \bar{U}_{xlm}, \bar{U}_{ylm}) + O(\Delta y^2 + \Delta x^2)$$  \hspace{1cm} (5.6.1)
Numerical experiments are carried out to illustrate our method and to demonstrate computationally its convergence. We solve the following two dimensional elliptic boundary value problems on unequal mesh both on rectangular and cylindrical polar coordinates whose exact solutions are known to us. The Dirichlet boundary conditions can be obtained using the exact solutions as a test procedure. We also compare our method with the central difference scheme (5.6.1) and the methods discussed in [71] in terms of solution accuracy. In all cases, we have taken the initial guess $u(x, y) = \theta$. The iterations were stopped when the absolute error tolerance became $\leq 10^{-10}$. All computations were carried out in double precision arithmetic using MATLAB. Graphs depicting exact and numerical solutions for selected parameters for each of the problems discussed in examples have been included (see Figs. 5.3-5.7).

**Example 5.1 (Convection-diffusion equation)**

The problem is to solve (5.5.7) in the solution region $0 < x, y < 1$ whose exact solution is given by

$$u(x, y) = e^{\frac{\beta y}{k}} \frac{\sin y}{\sin k} \left[ 2e^{\frac{\beta}{2}} \sinh kx + \sinh(1 - x) \right],$$

where $\sigma^2 = \pi^2 + \frac{\beta^2}{4}$. The maximum absolute errors for $u$ are tabulated in Table 5.1(a) and Table 5.1(b). Fig. 5.3(i) and Fig. 5.3(ii) demonstrate a comparison of the plots of the numerical and exact solution of $u(x, y)$ for the values $\beta = 30$ and $\gamma = (\Delta y / \Delta x^2) = 20$.

**Example 5.2 (Poisson’s equation in polar coordinates)**

The problems are to solve (5.5.5) and (5.5.6) in the solution regions $0 < r, \theta < 1$ and $0 < r, \gamma < 1$ respectively. The exact solutions are given by $u(r, \theta) = r^2 \cos \pi \theta$ and $u(r, \gamma) = \cosh r \cosh \gamma$. The maximum absolute errors for $u$ are tabulated in Table 5.2(a) and Table 5.2(b). A comparison of the plots of the numerical and exact solution of $u$ for the value $\gamma = (\Delta y / \Delta x^2) = 20$ is shown in the figures 5.4(i), 5.4(ii), 5.5(i) and 5.5(ii).

**Example 5.3 (Steady state Burgers’ Model Equation)**

$$\varepsilon \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = u \left( \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} \right) + e^x \sin \left( \frac{\pi y}{2} \right) \left[ \varepsilon \left( 1 - \frac{\pi^2}{4} \right) - e^x \left( \sin \left( \frac{\pi y}{2} \right) + \frac{\pi}{2} \cos \left( \frac{\pi y}{2} \right) \right) \right], \quad 0 < x, y < 1 \quad (5.6.2)$$

where $R_c = \varepsilon^{-1} > 0$ is called Reynolds number. The exact solution is given by $u(x, y) = e^x \sin \left( \frac{\pi y}{2} \right)$. The maximum absolute errors for $u$ are tabulated in Table 5.3(a) and Table 5.3(b) for various values of $R_c$. Fig. 5.6(i) and Fig. 5.6(ii) demonstrate a comparison...
of the plots of the numerical and exact solution of $u(x,y)$ for the values $R_e=100$ and $\gamma = (\Delta y/\Delta x^2) = 20$.

**Example 5.4** (Quasi-Linear Elliptic Equation)

\[
\frac{\partial^2 u}{\partial x^2} + (1 + u^2) \frac{\partial^2 u}{\partial y^2} = \alpha u \left( \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} \right) + \exp(xy) \left[ x^2 + y^2 + (\exp(xy))^2 x^2 - \alpha \exp(xy) (x + y) \right], \quad 0 < x, y < 1 \quad (5.6.3)
\]

The exact solution is $u(x,y) = \exp(xy)$. The maximum absolute errors for $u$ are tabulated in Table 5.4(a) and Table 5.4(b) for various values of $\alpha$. A comparison of the plots of the numerical and exact solution of $u$ for the value $\gamma = (\Delta y/\Delta x^2) = 20$ is shown in the figures 5.7(i) and 5.7(ii).

Finally, Table 5.5 shows that our method works as a fourth order method with fixed mesh parameter $\gamma = \Delta y/\Delta x^2$. The order of convergence may be obtained by using the formula

\[
\log \left( \frac{e_{\Delta x_1}}{e_{\Delta x_2}} \right) / \log \left( \frac{\Delta x_1}{\Delta x_2} \right) \quad (5.6.4)
\]

where $e_{\Delta x_1}$ and $e_{\Delta x_2}$ are maximum absolute errors for two uniform mesh widths $\Delta x_1$ and $\Delta x_2$, respectively. For computation of order of convergence of the proposed method, we have considered errors for last two values of $\Delta x$, i.e., $\Delta x_1 = \frac{1}{20}, \Delta x_2 = \frac{1}{40}$ for the above discussed elliptic partial differential equations.
Table 5.1(a)

**Example 5.1**: The maximum absolute errors \((p = \frac{\Delta y}{\Delta x} = 0.8)\)

<table>
<thead>
<tr>
<th>(\Delta x)</th>
<th>Proposed (O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4))-method</th>
<th>(O(\Delta y^2 + \Delta x^2))-method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(\beta = 20)</td>
<td>(\beta = 50)</td>
</tr>
<tr>
<td>1/16</td>
<td>(0.2439E-02) secs.</td>
<td>(0.3662E-01) secs.</td>
</tr>
<tr>
<td>1/32</td>
<td>(0.2253E-03) secs.</td>
<td>(0.3789E-02) secs.</td>
</tr>
<tr>
<td>1/64</td>
<td>(0.3683E-04) secs.</td>
<td>(0.2410E-03) secs.</td>
</tr>
<tr>
<td>1/128</td>
<td>(0.8396E-05) secs.</td>
<td>(0.1769E-04) secs.</td>
</tr>
</tbody>
</table>

Table 5.1(b)

**Example 5.1**: The maximum absolute errors \((\gamma = \frac{\Delta y}{\Delta x^2} = 20)\)

<table>
<thead>
<tr>
<th>(\Delta x)</th>
<th>Proposed (O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4))-method</th>
<th>(O(\Delta y^4 + \Delta y^2 \Delta x^2 + \Delta x^4))-method discussed in [71]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(\beta=10)</td>
<td>(\beta=20)</td>
</tr>
<tr>
<td>(\frac{1}{10})</td>
<td>(0.1062E-01)</td>
<td>(0.1823E-01)</td>
</tr>
<tr>
<td>(\frac{1}{20})</td>
<td>(0.6971E-03)</td>
<td>(0.1213E-02)</td>
</tr>
<tr>
<td>(\frac{1}{40})</td>
<td>(0.4352E-04)</td>
<td>(0.7360E-04)</td>
</tr>
</tbody>
</table>
Comparison of plots of solution of Example 5.1
Table 5.2(a)

Example 5.2: The maximum absolute errors ($p = \frac{\Delta y}{\Delta x} = 0.8$)

<table>
<thead>
<tr>
<th>$\Delta x$</th>
<th>Proposed $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$-method</th>
<th>$O(\Delta y^2 + \Delta x^2)$-method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Eq. (5.5.5)</td>
<td>Eq. (5.5.6)</td>
</tr>
<tr>
<td>$\frac{1}{16}$ cpuntime</td>
<td>0.1917E-03 0.388811 secs.</td>
<td>0.2183E-04 0.279115 secs.</td>
</tr>
<tr>
<td>$\frac{1}{32}$ cpuntime</td>
<td>0.4870E-04 2.161357 secs.</td>
<td>0.5668E-05 1.607198 secs.</td>
</tr>
<tr>
<td>$\frac{1}{64}$ cpuntime</td>
<td>0.1207E-04 8.529112 secs.</td>
<td>0.1405E-05 8.255509 secs.</td>
</tr>
<tr>
<td>$\frac{1}{128}$ cpuntime</td>
<td>0.2975E-05 69.298359 secs.</td>
<td>0.1340E-06 24.550988 secs.</td>
</tr>
</tbody>
</table>

Table 5.2(b)

Example 5.2: The maximum absolute errors ($\gamma = \frac{\Delta y}{\Delta x^2} = 20$)

<table>
<thead>
<tr>
<th>$\Delta x$</th>
<th>Proposed $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$-method</th>
<th>$O(\Delta y^2 + \Delta x^2)$-method discussed in [71]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Eq. (5.5.5)</td>
<td>Eq. (5.5.6)</td>
</tr>
<tr>
<td>$\frac{1}{10}$</td>
<td>0.2976E-02</td>
<td>0.3574E-03</td>
</tr>
<tr>
<td>$\frac{1}{20}$</td>
<td>0.1917E-03</td>
<td>0.2343E-04</td>
</tr>
<tr>
<td>$\frac{1}{40}$</td>
<td>0.1202E-04</td>
<td>0.1448E-05</td>
</tr>
</tbody>
</table>
Fig 5.4(i) Poisson’s equation ($r$-$z$ plane)  
$\gamma = 20$ [Numerical Solution]

Fig 5.4(ii) Poisson’s equation ($r$-$z$ plane)  
$\gamma = 20$ [Exact Solution]

Comparison of plots of solution of Example 5.2(a)
Fig 5.5(i) Poisson’s equation (r-θ plane) 
\( \gamma = 20 \) [Numerical Solution]

Fig 5.5(ii) Poisson’s equation (r-θ plane) 
\( \gamma = 20 \) [Exact Solution]

Comparison of plots of solution of Example 5.2(b)
\textbf{Table 5.3(a)}

*Example 5.3*: The maximum absolute errors \((p = \frac{\Delta y}{\Delta x} = 0.8)\)

<table>
<thead>
<tr>
<th>(\Delta x)</th>
<th>Proposed (O(\Delta y^2 + \Delta y^4 \Delta x^2 + \Delta x^4))-method</th>
<th>(O(\Delta y^2 + \Delta x^2))-method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(R_e=50)</td>
<td>(R_e=100)</td>
</tr>
<tr>
<td>(\frac{1}{32}) cputime</td>
<td>0.1744E-03 0.127566 secs.</td>
<td>0.1851E-03 0.127566 secs.</td>
</tr>
<tr>
<td>(\frac{1}{64}) cputime</td>
<td>0.4441E-04 1.173741 secs.</td>
<td>0.4527E-04 1.173741 secs.</td>
</tr>
<tr>
<td>(\frac{1}{128}) cputime</td>
<td>0.1111E-04 13.292612 secs.</td>
<td>0.1142E-04 12.640269 secs.</td>
</tr>
</tbody>
</table>

\textbf{Table 5.3(b)}

*Example 5.3*: The maximum absolute errors \((y = \frac{\Delta y}{\Delta x^2} = 20)\)

<table>
<thead>
<tr>
<th>(\Delta x)</th>
<th>Proposed (O(\Delta y^2 + \Delta y^4 \Delta x^2 + \Delta x^4))-method</th>
<th>(O(\Delta y^4 + \Delta y^2 \Delta x^2 + \Delta x^4))-method discussed in [71]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(R_e=10)</td>
<td>(R_e=100)</td>
</tr>
<tr>
<td>(\frac{1}{10})</td>
<td>0.1022E-01</td>
<td>0.8190E-02</td>
</tr>
<tr>
<td>(\frac{1}{20})</td>
<td>0.5887E-03</td>
<td>0.7330E-03</td>
</tr>
<tr>
<td>(\frac{1}{40})</td>
<td>0.3683E-04</td>
<td>0.4347E-04</td>
</tr>
</tbody>
</table>
Fig 5.6(i) Steady state Burger’s equation
\( \gamma = 20, \text{Re} = 100 \) [Numerical Solution]

Fig 5.6(ii) Steady state Burger’s equation
\( \gamma = 20, \text{Re} = 100 \) [Exact Solution]

Comparison of plots of solution of Example 5.3
Table 5.4(a)

Example 5.4: The maximum absolute errors ($p = \frac{\Delta y}{\Delta x} = 0.8$)

<table>
<thead>
<tr>
<th>$\Delta x$</th>
<th>Proposed $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$-method</th>
<th>$O(\Delta y^2 + \Delta x^2)$-method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\alpha=5$</td>
<td>$\alpha=10$</td>
</tr>
<tr>
<td>$\frac{1}{16}$ cpu time</td>
<td>0.2853E-04</td>
<td>0.4999E-04</td>
</tr>
<tr>
<td></td>
<td>0.599231 secs.</td>
<td>0.465740 secs.</td>
</tr>
<tr>
<td>$\frac{1}{32}$ cpu time</td>
<td>0.8114E-05</td>
<td>0.1446E-04</td>
</tr>
<tr>
<td></td>
<td>4.542767 secs.</td>
<td>2.745459 secs.</td>
</tr>
<tr>
<td>$\frac{1}{64}$ cpu time</td>
<td>0.2088E-05</td>
<td>0.3740E-05</td>
</tr>
<tr>
<td></td>
<td>31.642228 secs.</td>
<td>23.121967 secs.</td>
</tr>
<tr>
<td>$\frac{1}{128}$ cpu time</td>
<td>0.4325E-06</td>
<td>0.8828E-06</td>
</tr>
<tr>
<td></td>
<td>390.85138 secs.</td>
<td>292.2078 secs.</td>
</tr>
</tbody>
</table>

Table 5.4(b)

Example 5.4: The maximum absolute errors ($\gamma = \frac{\Delta y}{\Delta x^2} = 20$)

<table>
<thead>
<tr>
<th>$\Delta x$</th>
<th>Proposed $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$-method</th>
<th>$O(\Delta y^4 + \Delta y^2 \Delta x^2 + \Delta x^4)$-method discussed in [71]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\alpha=5$</td>
<td>$\alpha=10$</td>
</tr>
<tr>
<td>$\frac{1}{10}$</td>
<td>0.5014E-03</td>
<td>0.8912E-03</td>
</tr>
<tr>
<td>$\frac{1}{20}$</td>
<td>0.3248E-04</td>
<td>0.5791E-04</td>
</tr>
<tr>
<td>$\frac{1}{40}$</td>
<td>0.2017E-05</td>
<td>0.3607E-05</td>
</tr>
</tbody>
</table>
Fig 5.7(i) Quasi-linear elliptic equation
\( \gamma = 20, \alpha = 20 \) [Numerical Solution]

Fig 5.7(ii) Quasi-linear elliptic equation
\( \gamma = 20, \alpha = 20 \) [Exact Solution]

Comparison of plots of solution of Example 5.4
Table 5.5

Fourth order convergence: $\Delta x_1 = \frac{1}{20}$, $\Delta x_2 = \frac{1}{40}$, $\gamma = \frac{\Delta y}{\Delta x^2} = 20$

<table>
<thead>
<tr>
<th>Example</th>
<th>Parameters</th>
<th>Order of the Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>$\beta = 10$</td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>$\beta = 20$</td>
<td>4.04</td>
</tr>
<tr>
<td></td>
<td>$\beta = 30$</td>
<td>4.08</td>
</tr>
<tr>
<td>5.2</td>
<td>Eq. (5.5.5)</td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>Eq. (5.5.6)</td>
<td>4.01</td>
</tr>
<tr>
<td>5.3</td>
<td>$R_c = 10$</td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>$R_c = 100$</td>
<td>4.07</td>
</tr>
<tr>
<td>5.4</td>
<td>$\alpha = 5$</td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>$\alpha = 10$</td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>$\alpha = 20$</td>
<td>3.99</td>
</tr>
</tbody>
</table>
5.7 Conclusion and Observations

Available numerical methods based on cubic spline approximations for the numerical solution of quasi-linear elliptic equations are of $O(\Delta y^2 + \Delta x^2)$ accurate. Although 9-point finite difference approximations of $O(\Delta y^4 + \Delta y^2 \Delta x^2 + \Delta x^4)$ accurate for the solution of non-linear and quasi-linear elliptic differential equations are discussed in [64, 71], but these methods require five evaluations of the function $f$. In this article, using the same number of grid points and three evaluations of the function $f$, we have derived a new stable cubic spline method of $O(\Delta y^2 + \Delta y^2 \Delta x^2 + \Delta x^4)$ accuracy for the solution of quasi-linear elliptic equation (5.1.1). However, for a fixed parameter $\gamma = \frac{\Delta y}{\Delta x^2}$, the proposed method behaves like a fourth order method. The accuracy of the proposed method is exhibited from the computed results. Further, we have reported the numerical results for the solution of 2D nonlinear Burger’s equation in Table 5.3 for $p = 0.8$. The stability of the method plays an important role for computation. In general, the stability condition for 2D nonlinear Burger’s equation cannot be determined theoretically. During computation for nonlinear Burger’s equation, we found that for $(\Delta x, R) = (\frac{1}{32}, 50)$, $(\frac{1}{32}, 100)$ and $(\frac{1}{64}, 100)$, the $O(\Delta y^2 + \Delta x^2)$-method becomes unstable and errors overflow in these cases, whereas the proposed method is stable in these cases. In other cases though, both the methods are stable. The proposed method is applicable to Poisson’s equation in polar coordinates, and two dimensional Burgers’ equation, which is main highlight of the work.