Cubic B-Spline Collocation Method for Parabolic Partial Differential Equations

Dissertation submitted to the Faculty of Mathematics and Computer Science, South Asian University, in partial fulfillment of the requirement for the award of the degree of Master of Science

ANURAG TRIVEDI

Department of Mathematics
Faculty of Mathematics and Computer Science
South Asian University
New Delhi-110021, India
2017

This Dissertation is dedicated to

sister

for some special reason

Declaration and Certificate

Date: May 25, 2017

Declaration

We declare that the dissertation **"Cubic B-Spline Collocation Method for Parabolic Partial Differential Equations"** submitted by us in partial fulfillment of the requirement for the award of the degree of Master of Science of the South Asian University is our own work. The dissertation complies with the plagiarism rules of the University. It has not been submitted for any other degree or diploma of this University or any other University or for any publication.

Anurag

ANURAG TRIVEDI
Reg. no. SAU/AM(M)/2015/12

Certificate

We recommend that this dissertation be placed before the examiners for evaluation.

Mohanty

Prof. R K Mohanty
Dean, FMCS

Dr. Kapil Sharma
Associate Professor, FMCS
(Supervisor)

Acknowledgements

I would like to express my gratitude to my supervisor **Dr Kapil sharma Associate Professor** for introducing me to topic useful comments, remarks and engagement through the learning process of this master thesis.

We shall further like to thank the Dean of the Faculty of Mathematics and Computer Science, **Dr. R K Mohanty, Professor** for providing us with this great learning opportunity.

I would like to express my special gratitude and thanks to Mr. Pankaj Mishra and Mr. Gajendra Tiwari ,Research Scholar of South Asian University for giving me such attention and time. My thanks and appreciations also go to my colleagues in developing the project and people who have willingly helped me out with their abilities.

ANURAG TRIVEDI

Abstract

A variety of problems of physical interest are modeled in terms of parabolic partial differential equations. Well-known equations of this class are those of heat and diffusion equation. The present work is the outcome of a study of some aspects of these equations.

The aim of this Dissertation is to investigate the features of the numerical solution of parabolic partial differential equation(the targeted one dimensional heat equation $u_t = \alpha^2 u_{xx}$,0≤x≤L ,$t > 0$). This problem is one of the well-known second order linear partial differential equation. It shows that heat equation describes irreversible process and makes a distance between the previous and next steps. Such equations arise very often in various applications of science and engineering describing the variation of temperature (or heat distribution) in a given region . Cubic B-spline collocation method are used to solve the PDE and these method come under numerical method. The performence of the method is presented and the stability analysis is investigated by considering Fourier stability method. However, the principal aim in this thesis has been to develop and efficent methods .

In this thesis a brief introduction of Numerical solution of Differential Equations's and numerical Technique for PDE based on finite difference method ,finite element method ,finite volume method and more methods are discussed in chapter 1 , In chapter 2 some basic introduction of B-spline are given. we survey existing Literature review of B-spline. and chapter 3 A cubic B-spline collocation method is proposed. We provided numerical solution for test problem using our proposed method.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The history of research on partial differential equations (PDEs) goes back to the 18th century. Partial Differential equations (PDE) form the basis of many mathematical models of physical (Wave propagation,Fluid flow), chemical (air and water) and biological phenomena, and more recently their use has spread into economics,commerce, trade, financial forecasting, image processing and other fields. Numerical analysis can be considered as a branch of analytical applied mathematics. The numerical technique is such a bridge, and the corresponding approximate solution is termed the numerical solution. In the many case for solving PDE,numerical approximation is the best way. On the other ways , numerical simulation can play an important role in the design and study of complex systems. This activity generates a substantial literature, much of it highly specialized and technical. Meanwhile, mathematicians use analysis to probe new applications and to develop numerical simulation algorithms that are provably accurate and efficient. Such capability is of considerable importance, given the explosion of experimental and observational data and the spectacular acceleration of computing power. In this thesis, we will study the numerical model for problem solving of one dimensional heat equation. In particular we are interested the study between the numerical and the analytic solution.
There are many numerical techniques for solving PDEs, such as the finite difference method , finite element method , finite volume method . The finite element method and finite volume method are widely used in engineering to model problems.

Numerical Solution of Differential Equation's

A partial differential equation (PDE) is an equation stating a relationship between function of two or more independent variables and the partial derivatives of this function with respect to these independent variables. The order of the highest derivative defines the order of the equation. Partial differential equations (PDEs) arise in all field of engineering and science. Most real physical processes are governed by partial differential equations. In many cases, simplifying approximations are made to reduce the governing PDEs to ordinary differential equations (ODEs) or even to algebraic equations.

The Class of PDE is three type

1. Elliptic PDEs
2. Parabolic PDEs
3. Hyperbolic PDEs

linear PDEs as equations for which the dependent variable (and its derivatives) appear in terms with degree at most one. Anything else is called nonlinear PDE's.
following one dimensional heat equation

$$u_t = \alpha^2 u_{xx} \ , 0 \leq x \leq L \ , \ t > 0$$

which is a example of an Linear Second order linear PDE of Parabolic problem which has importance in diverse scientific fields, It describes irreversible process and makes a distance between the previous and next steps. Such equations arise very often in various applications of science and engineering describing the variation of temperature (or heat distribution) in a given region over some time . It can be expressed as the heat flow in the rod with diffusion $\alpha^2 u_{xx}$ along the rod, Where the coefficient α is the thermal diffusivity of the rod and L is the length of the rod . In this model, the flow of the heat in one-dimension that is insulated everywhere except at the two end points.
There are numeber of methods can be used to reduce the governing PDEs to a set of ordinary differential equations (ODEs).

1. Finite Difference Method
2. Finite Volume Method
3. Finite Element Method
4. Method of Weighted Residual
5. Least Squares Method
6. Galerkin Method
7. Sub-domain Method
8. Collocation method

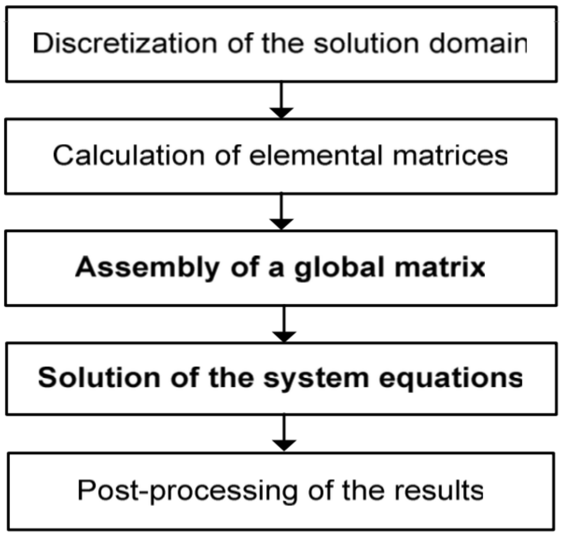
1.1 Finite Difference Method

The Finite Difference Method (FDM) is one of the numerical approximation methods that are frequently used to solve partial differential equations. To use a finite difference method to approximate the solution to a problem, one must first discretize the problem's domain. This is usually done by dividing the domain into a uniform grid. Note that this means that finite-difference methods produce sets of discrete numerical approximations to the derivative, often in a time-stepping manner. Appropriate types of differencing schemes and suitable methods of solution are chosen in different applications. For example, in fluid dynamics applications, depending upon the particular physics of the flows, which may include in viscid, viscous, incompressible, compressible, irrigational, rotational, laminar, turbulent, supersonic, or hypersonic flows, finite difference methods for new classes of problems is often to find an appropriate definition of stability that allow one to prove convergence and to estimate the error in approximation analyzed is very challenging. finite difference method Simple to construct and to analyse. Need some modification in regions where the solution is changing rapidly. but this method more difficult to apply when the solution region is not rectangular.

1.2 Finite Volume Method

The finite volume method is a discretizations method for the approximation of single or a system of differential equation. The finite volume method is based on Integral conservation law (I) rather then Partial differential Equation. The intergral Conservation law is enforced for small control volumes defined by the computational mesh to be specified concrete choice of control volumes type of approximation inside them and numerical method for evaluation of integrals and fluxes. FVM is in common use for discretizing computational fluid dynamics equations. finite volume methods discretize the balance equation directly, an obvious virtue of finite volume methods is the conservation property comparing with finite element methods based on the weak formulation. This property can be fundamental for the simulation of many physical models, e.g., in oil recovery simulations and in computational fluid dynamics in general. On the other hand, the function space and the control volume can be constructed based on general unstructured triangulations for complex geometry domains. The boundary condition can be easily built into the function space or the variational form. Thus FVM is more flexible than standard finite difference methods which mainly defined on the structured grids of simple domains.

Figure 1.1: Finite Element Method Algorithm Process for PDE



1.3 Finite Element Method

Finite element method (FEM) represents a powerful and general class of techniques for the approximate solution of partial differential equations. **The finite element method replaces the original function with a function that has some degree of smoothness over the global domain but is piecewise polynomial on simple cells, such as small triangles or rectangles.**

The basic idea of the method is approach the continuous function for exact solution of the PDE using piecewise approximation , Generally its polynomial . In the system is construct with node which is make a grid called a mesh. This mesh is programmed by the material and structural properties which define how the it will react to predefined loading conditions in the case of structural analysis.
The basic flow chart for Finite element method is shown in figure(1.1) the Variation of geometrical data is analyzed in spatial domain and it's sub-divided by geometric discretization and different strategies using material properties. Generally domain of solution is descretized into trianguler elements. Then the matices formed the system of equations are assembled and solved.
The finite element method (FEM) has been widely used for engineering analysis. This method was first developed in 1956 for the analysis of aircraft structural problems. The age of computers accelerated the use and understanding of FEM. Finite element Methods is more difficult to set up than the Finite difference method, but much better in irregular shaped domains. This method is not so easy to analyse at a simple level, but some rigorous results can be found with the aid of variational calculus.

1.4 Method of Weighted Residual

Before start the development of finite element method there exist an approximation technique for solving differential equations(DE) called the Method of weight residual . This method divided in to the two step. In the first step an approximate solution based on our assumed dependent variable behaviour. We assume solution for satisfy the boundary condition for ϕ. Our assumed solution substituted in the differential equation (DE). But its only approximate and it does not generally satisfy differential equations . Hence in the result we get error , which is called a residual . The residual is then made to vanish in some average sense over the entire solution domain to produce a system of algebraic equations, and then the second step we solve the system of equation result we get from previous step subject is prescribed from boundary condition for the approximate solution.
Let us consider differential Equation(DE)
$Ł(u) = 0$
with initial condition $I(u) = 0$ and Boundary condition $S(u) = 0$. The approximate solution satisfies the essential Boundary condition(BC) , but not natural BC and it is also may not be satisfy DE exactly.
The solution of Diffrental Equation $U(x) = 0$ is approximated by a series of finite funtions $\phi_k(x) = 0$ as fallows,

$$U(x) = U_0(x) + \sum_{k=1}^{N} a_k\phi_k(x).$$

Where $U_0(x)$ is satisfy intial and boundary condition and $\phi_k(x)$ is the trial and basis function and k and a are the coefficent of the detained by diffrential equation and N is the no of function. This is different approche where exact solution of the equation are known and add the boundry condition satisfying approximately.
Residual

$$\frac{d^2\tilde{u}}{dx^2} + p(x) = R(x)$$

We want to minimize the residual by multiplying with weight W and intergrate over the domain.

$$\int_a^b R(x)W(x)dx$$

$W(x)$ is called weight function and if it is satisfying for any $W(x)$,then $R(x)$ will approaches zero,and the approximate solution will approach the exact solution, and Depending on choice of W(x): least square error method, collocation method, Petrov-Galerkin method, and Galerkin method.

1.5 Least Squares Method

The method of least-squares we try to minimize the residual in a least-squares sense, the weight functions are chosen to be the derivatives of residual with respect to unknown fitting coefficients $c_i s$ $(i = 1, 2, 3....n)$of the approximate solution. So, we set

$$W_i = \frac{\partial R}{\partial c_i}$$

In the problem of one dimensional with the interval $[a, b]$, the weight residual integral by.

$$\int_a^b R(x)W_i(x)dx = \int_a^b R(x)\frac{\partial R}{\partial c_i}dx$$

1.6 Galerkin Method

This method developed by the Russian mathematician Boris Grigoryevich Galerkin. The method is modification of least square method. Galerkin method is one of the widely used meshfree methods for solving partial differential equations. In this method the derivative of the residual with respect to the unknown a_i, the derivative of the approximating function is used. That is, if the function is approximated as in (1.5) , then the weight functions are

$$W_i = \frac{\partial \tilde{u}}{\partial a_i}$$

Then this identical to the orignal trial or basis function.

$$W_i(x) = \frac{\partial \tilde{u}}{\partial a_i} = \varphi(x)$$

The best things of Galerkin method is construct a variational principal for a partial differential system and more complex problem. We will try to explore some properties of variational principles with a goal of developing a more thorough understanding of Galerkin's method and of answering the questions raised.

1.7 Sub-domain Method

In this method domain is divided in to the I subdomains D_i where

$$W_i = \begin{cases} 1 & \in D_i \\ 0 & else \end{cases}$$

hence this method minimized the residual error in each freely choice sub-domain. the best choice is if domain is divided in equal sub-domain. However, if higher resolution (and a corresponding smaller error) in a particular area is desired, a non-uniform choice may be more appropriate.

1.8 Collocation method

The collocation method is developed for the numerical solution of ordinary differential equations(ODE), partial differential equations(PDE) and integral equations(IE). In this method very elegant approach to construction method of given order of accuracy uses the idea of collocation. The method of collocation requires that the approximate solution satisfied the partial differential equations at certain preselected collocation points and approximate solution of the problem by collocation on a subinterval satisfied interpolating conditions. In this method Minimize residual by forcing it to pass through zero at a finite number of discrete points within Ω. So here

$$w_i(x) = \delta(x - x_i^c), \qquad x_i^c \in \Omega$$

where x_i^c is the $i-th$ collocation point. The choice of collocation points is an important consideration in the method. We aim to have

$$\int_\Omega r(x)w_i(x)\mathrm{d}x = \int_\Omega r(x)\delta(x - x_i^c)\mathrm{d}x = r(x_i^c) = 0$$

in the other ways to say , at each collocation point the trial functions are required to satisfy the differential equation exactly. The number of collocation points is related to the number of c_j.

Let us consider one- dimensional equation with two point Boundary value problem $L_u(t) = f(t)$, t $\in [a, b]$ $u(a) = \alpha$, $u(b) = \beta$

and Let us know about trail or basis function (approximation space) spanϕ_1, ϕ_2, ϕ_3..........Φ_n (the space of polynomials or splines of a certain degree, or some radial basis function space), Now we can write the approximate solution in the form

$$U(t) = \sum_{k=1}^{N} c_k \phi_k(x)$$

To determine the n unknown coefficient c_1, c_2.......c_n in this formulation we impose n collocation condition to obtain and $n \times n$ system of equation for c_k. In this method very useful feature that we have obtain a continuous approximation of solution $U(t)$ on each interval [a,b].

From the last few year B-spline collocation method for solving numerical technique day by day increases in the mathematical modelling ,physical process and research area of engineering. B-spline collocation method is one of the most emulative numerical technique for solving differential equation. B-spline collocation technique are very use full in finite element and finite difference scheme. The advantage of the B-spline collocation method over finite element Galerkin method is that calculation of coefficient matrices is very easy since no integrals need to evaluate or approximation. In the experiment of numerical process this method provide a high order of accuracy in many physical process.

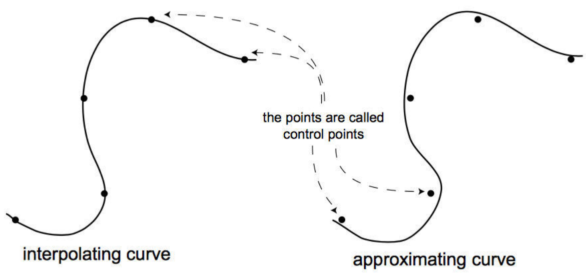
Chapter 2

Basics of B-spline

2.1 Spline

The Basic terminology of spline is mathematical representation of interface which is allow a user to design and control the shape of complex system. In the other way user enters a sequence of point for constructing a curve and there point are called control points. If curve passes each control point is called interpolating curve and if curve passes near the control point then it is called approximating curve. The basic theory of spline is a piecewise polynomial whose curve derivatives are continuous across the various sctions of curve.
Let we have n+1 point x_0,x_1,x_2.....x_n have been sapecified and satisfy $x_0 < x_1 < x_2 ..... < x_n$. These point are called konts and this sequence is called kont sequence. Let spline fuction of degree k having knots x_0,x_1,x_2.....x_n is a function S such that , on each interval $[x_{i-1}, x_i]$, S is a polynomial of degree $\leq$K, and S has continuous (k-1)th derivative on $[x_0, x_n]$.
Since S be the at most K degree piecewise polynomial . which is all $(k-1)^{th}$ derivative

Figure 2.1: Spline



are continuous. A one degree spline is piecewise linear and has the form

$$S_(x) = \begin{cases} \mathrm{p}_1(x) = a_1 + b_1 x & x \in [\mathrm{x}_0, x_1) \\ \mathrm{p}_2(x) = a_2 + b_2 x & x \in [\mathrm{x}_1, x_2) \\ . \\ . \\ . \\ \mathrm{p}_n(x) = a_n + b_n x & x \in [\mathrm{x}_{n-1}, x_n] \end{cases}$$

in this thesis We will only consider spline interpolation using liner spline , quardratic spline and cubic spline. Generalization to splines of general order is relatively straightforward.

2.1.1 Bezier Curves

Bezier Curves is Discovered by the French engineer Pierre Bezier. These Curve can be generated under the control of other points. Approximate tangents by using control points are used to generate curve. The Bezier curve can be represented mathematically as

$$\sum_{k=0}^{n} P_i B_i^n(t)$$

Where p_i is the set of points and $B_i^n(t)$ represents the Bernstein polynomials which are given by

$$B_i^n(t) = \binom{n}{i}(1-t)^{n-i}t^i$$

Where n is the degree of the polynomial , i is the index, and t is the variable.

2.2 B-Spline

B-spline (basis of spline) is spline function that has minimal support with respect to a given degree, smoothness and domain partition. In a given degree of spline function can be expreesed as a linear combination of B-spline of that degree. In B-spline curve can move control point for modifying the shape of the curve just like Bezier curves. Bezier Curve Developed by Paul de Casteljau (1959) and independently by Pierre Bezier (1962). B-spline Curves have higher degree of freedom for curve design. B-spline curve have two advantages first one The degree of B-spline polynomial can be set independently of the number of control points and second is B-spline allow local control over the shape of spline curve(or surface).

2.2.1 B-Spline Basis Function

"B-spline basis functions defined by kont sequence and by the polynomial degree of the curve". B-spline basis function will be used same way as Bezier Basis function . Two interesting Property that the domain is subdivided by knots and basis fuction are non-zero on the entire interval.
Let $k \in N_0$:= 0,1,2,....., $m \in N$:= 1, 2, 3...., $m \geq k+1$, and X:=($x_0 < x_1 < x_2 .....x_m$). a non-decreasing sequence of real numbers. Futhermore ,Let $t \in R$, and $N_{i,k}$, $i \in N_0$, $i \leq m-k-1$, denote the i-th normalized B-spline of degree k with respect to X, Then, for k=0 we have

$$B_{i,0}(t) = \begin{cases} 1 & t \in [x_i, x_{i+1}) \\ 0 & otherwise \end{cases} \tag{2.1}$$

Moreover , for $k \geq 1$ it holds

$$B_{i,k}(t) = \frac{(t-x_i)}{(x_{i+k-1}-x_i)} B_{i,k-1} + \frac{(x_{i+k}-t)}{(x_{i+k}-x_{i+1})} B_{i+1,k-1}(t) \tag{2.2}$$

where

$$B_{i,k}(t) = s_{i,k}(t) B_{i,k-1} + (1 - s_{i+1,k}(t)) B_{i+1,k-1}(t) \tag{2.3}$$

where

$$S_{i,k}(t) = \begin{cases} \frac{t-x_i}{x_{i+k-1}-x_i} & x_i \neq x_{i+k-1} \\ 0 & otherwise \end{cases} \tag{2.4}$$

Note that $B_{i,k-1}$=0 if x_i=x_{i+k},
This is ith degree Cox de- Boor recursion formula .
Now Generalization to B-splines of general order is relatively straightforward for builds the function of higher order .For degree $k \geq 1$ basis function $B_{i,k}(t)$ is a linear combination of two $(k-1)^{th}$ degree basis function.

2.2.1.1 B-spline of degree One(1)

one degree B-spline, also called Linear B-spline it can be obtained by Cox and Boor recursion formula. Put k=1 in (2.1)The formula of linear spline can given as :-

$$B_{i,1}(t) = \begin{cases} \frac{t-x_i}{x_{i+1}-x_i} & t \in [x_i, x_{i+1}) \\ \frac{x_{i+2}-t}{x_{i+2}-x_{i+1}} & t \in [x_{i+1}, x_{i+2}) \\ 0 & otherwise \end{cases}$$

The plot of this is hat function and $B_{i,1}$ consists of linear polynomial pieces. Linear B-spline discontinuous at a double knot ,but continuous at simple knots.

2.2.1.2 B-spline of degree Two(2)

This spline as called a quardratic B-spline. A quadratic B-spline is a C^1 piecewise quardratic polynomial. Put k=2 in (2.1)The formula of Quardratic spline can given as

$$B_{i,2}(t) = \begin{cases} \frac{(t-x_i)^2}{(x_{i-2}-x_i)(x_{i+1}-x_i)} & t \in [x_i, x_{i+1}) \\ \frac{(t-x_i)(x_{i+2}-t)}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})} + \frac{(x_{i+3}-t)(t-x_{i+1})}{(x_{i+3}-x_{i+1})(x_{i+2}-x_{i+1})} & t \in [x_{i+1}, x_{i+2}) \\ \frac{(x_{i+3}-t)^2}{(x_{i+3}-x_{i+1})(x_{i+3}-x_{i+2})} & t \in [x_{i+2}, x_{i+3}) \\ 0 & otherwise \end{cases}$$

At a single knot a quadratic B-spline is continuous and has a continuous derivative, at a double knot it is continuous, while at a triple knot it is discontinuous.

2.2.1.3 B-spline of degree Three(3)

This spline as called a cubic B-spline. A cubic B-spline ia C^2 piecewise cubic polynomial. put K=3 in (2.1) the formula for cubic b-spline can given as

$$B_{i,3}(t) = \quad \frac{(t-x_i)}{(x_{i+3}-x_i)} B_{i,2} + \frac{(x_{i+3}-t)}{(x_{i+3}-x_{i+1})} B_{i+1,2}(t)$$

For a cubic we can require C^2 continuity or position, slope and curvature continuity. cubic B-spline with a knot of Multiplicity 4 is discontinuous at the knot.

2.3 Cubic B-spline with collocation method

In finite Element method based on cubic B-Spline collocation method has been developed to second order boundary value problem. This method is based on evaluating the accuracy of a differential equation at a finite set of collocation points. Let [a,b] be partitioned into a mesh by x_0,x_1,x_2.....x_N point such that

$$a = \quad \mathrm{x}_0 < x_1 < x_2 ..... < x_{N-1} < x_N = \mathrm{b} \tag{2.5}$$

where h=x_{i+1}-x_i , i= 0,1,2....N-1 The Cubic B-spline will used to approximate the solution U(x,t) of the modeled problem. The cubic B-splines B_j at the node x_i are defined to from a basis over the interval [a,b]. Thus, an approximation $U_N(x,t)$ to the exact solution $u(x,t)$ can be expressed in terms of the cubic B-splines as

$$U_N(x,t) = \quad \sum_{j=i-k+2}^{i+k-2} C_j(t) B_j(x) \tag{2.6}$$

Where C_j are time dependent parameters to be determined from boundary conditions and from of collocation differential equation. Support of cubic B-spline B_j lies in four interval , as

$$B_{i,3}(x) = \frac{1}{h^3} \begin{cases} (x - x_{i-2})^3 & x \in [x_{i-2}, x_{i-1}) \\ (x - x_{i-2})^3 - 4(x - x_{i-1})^3 & x \in [x_{i-1}, x_i) \\ (x_{i+2} - x)^3 - 4(x_{i+1} - x)^3 & x \in [x_i, x_{i+1}) \\ (x_{i+2} - x)^3 & x \in [x_{i+1}, x_{i+2}) \\ 0 & otherwise \end{cases} \tag{2.7}$$

Table 2.1: Knots Values of B, B', and B''.

	x_{i-1}	x_{i-2}	x_i	x_{i+1}	x_{i-2}
$B_{i,3}$	0	1	4	1	0
$B'_{i,3}$	0	3/h	0	-3/h	0
$B''_{i,3}$	0	6/h^2	-12/h^2	6/h^2	0

By putting K=3 in (2.6), we obtain the approximate solution

$$U_N(x,t) = \quad \sum_{j=i-1}^{i+1} C_j(t)B_j(x)$$

without loss of generality we can express

$$U(x_i,t) = \quad c_{i-1}B_{i-1}(x_i) + c_iB_i(x_i) + c_{i+1}B_{i+1}(x_i)$$

Moreover we can also write

$$U(x_i,t) = \quad c_{i-1}B_i(x_{i+1}) + c_iB_i(x_i) + c_{i+1}B_i(x_{i-1})$$

We evaluate cubic B-spline

$$U(x_i,t) = \quad c_{i-1}B_{i,3}(x_{i+1}) + c_iB_{i,3}(x_i) + c_{i+1}B_{i,3}(x_{i-1}) \tag{2.8}$$

$$U'(x_i,t) = \quad c_{i-1}B_{i,3}(x_{i+1}) + c_iB_{i,3}(x_i) + c_{i+1}B_{i,3}(x_{i-1}) \tag{2.9}$$

$$U''(x_i,t) = \quad c_{i-1}B_{i,3}(x_{i+1}) + c_iB_{i,3}(x_i) + c_{i+1}B_{i,3}(x_{i-1}) \tag{2.10}$$

Putting the Knots Value in the equation(2.8), (2.9) and (2.10) from table (1.1) we obtain

$$\begin{aligned} &U(x_i,t) = c_{i-1} + 4c_i + c_{i+1}, \\ &U'(x_i,t) = 3/h(c_{i-1} - c_{i+1}), \\ &U''(x_i,t) = 6/h^2(c_{i-1} - 2c_i + c_{i+1}). \end{aligned} \tag{2.11}$$

2.4 Importent Properties of B-spline Basis functions

The Bezier-curve produced by the Bernstein basis function has limited flexibility. First, the number of specified polygon vertices fixes the order of the resulting polynomial which defines the curve. The second limiting characteristic is that the value of the blending function is nonzero for all parameter values over the entire curve. The B-spline basis contains the Bernstein basis as the special case. The B-spline basis is

non-global.
A B-spline curve is defined as a linear combination of control points x_i (i=0,1,2,3....n) and B-spline basis function $N_{i,k}(t)$ given by
$C(t) = \sum_{i=0}^{n} x_i N_{i,k}(t)$, $n \geq k-1$,t$\in [tk-1, tn+1]$
Some properties of B-spline

1. B-splines are good for large continuous curves and surfaces.

2. k is the order of the polynomial segments of the B-spline curve. Order k means that the curve is made up of piecewise polynomial segments of degree k-1,

3. A B-spline curve is contained in the convex hull of its control polyline.

4. Clamped B-spline curve C(t) passes through the two end control points k_0 and k_n.

5. **Nonnegativity** – For all i, k and u, $N_{i,k}(t)$ is non-negative.

6. **Local Support** – $N_{i,k}(t)$ is a non-zero polynomial on $[u_i,u_{i+k+1})$.

7. On any span $[u_i,\ u_{i+1})$, at most k+1 degree k basis functions are non-zero, namely: $N_{i-k,k}(t)$, $N_{i-k+1,k}(t)$, $N_{i-k+2,k}(t)$, ..., and $N_{i,k}(t)$.

8. Basis function $N_{i,k}(t)$ is a composite curve of degree k polynomials with joining points at knots in $[u_i,\ u_{i+k+1}\ )$.

Chapter 3

Cubic B-Spline for 1-D Heat Equation

Numerical Solution of Heat Equation using cubic B-spline collocation method

3.1 Introduction

Consider The heat Equation in one space dimension:-

$$u_t = \alpha^2 u_{xx} \quad , 0 \leq x \leq L \ t > 0, \tag{3.1}$$

with initial condition

$$u(x, 0) = f(x) \tag{3.2}$$

and boundary conditions(t>0)

$$u(0, t) = u(L, t) = 0 \tag{3.3}$$

where u = u(x, t) is a function of two variables x and t. Here x is the space variable, so x ∈ [0, L], where L is the length of the rod. t is the time variable, so t≥ 0. heat flow in the rod with diffusion $\alpha^2 u_{xx}$ along the rod where the coefficient α is the thermal diffusivity.
The heat equation is a parabolic partial differential equation that describes the distribution of heat (or variation in temperature) in a given region over time. In this model, the flow of the heat in one-dimension that is insulated everywhere except at the two end points. Solutions of this equation are functions of the state along the rod and the time t . Researchers work on this problem over number of years. but it's still an intresting problem since many physical phenomena can be formuleted into PDE with boundary conditions. The heat equation is of fundamental importance in diverse scientific fields. it is the prototypical parabolic

partial differential equation in mathematics . In probability theory, the heat equation is connected with the study of Brownian motion via the FokkerPlanck equation. In financial mathematics it is used to solve the BlackScholes partial differential equation. The diffusion equation, a more general version of the heat equation, arises in connection with the study of chemical diffusion and other related processes.
The theory of B-spline was first proposed in 1946 by Schoenberg. Recurrence relations for the purpose of computing coefficients are given by Cox and de Boor. cubic B-spline collocation method was devoloped for Numerical Solution and Solving Burgers' equation. Various techniques of both the cubic spline and cubic B-spline collocation methods and their application have been developed to obtain the numerical solution of the differential equations. The advantage of numerical technique by using cubic B-spline collocation method procedures exhibit the following the desirable features: (1) obtained governing system is always diagonal which permits easy algorithms; (2) it provides low computer cost and easy problem formulation. which is robust in computation.
In this chapter A Cubic B-spline Collocation Method is used for solving heat problem with given initial (3.2) and Boundary conditions (3.3). We Proposed collocation method with cubic B-spline for solving numerical solution and the stability analysis is investigated by considering Fourier stability method.

3.2 Cubic B-spline Method for Heat Equation

Let [0,L] be partitioned into a mesh by points x_0,x_1,x_2.....x_N suct that

$$0 = \quad \mathrm{x}_0 < x_1 < x_2 ..... < x_{N-1} < x_N = \mathrm{L} \tag{3.4}$$

where h=x_{i+1}-x_i , i= 0,1,2....N-1 The Cubic B-spline will used to approximate solution U(x,t) of the equation(3.1). The cubic B-spline ϕ_j ,j=-1......N+1at the node x_i are defined to from a basis over the interval [0,L]. Numerical experiment for Prabolic PDE using the colloction method with cubic B-spline is to find an approximate solution $U_N(x,t)$ to the exact solution $u(x,t)$ in the form

$$U_N(x,t) = \quad \sum_{j=-1}^{N+1} \delta_j(t)\phi_j(x) \tag{3.5}$$

Where δ_j are time dependent parameters to be determined from boundary conditions and collocation from of the differential equation. Cubic B-spline covers four elements(mesh interval).

$$\phi_j(x) = \frac{1}{h^3}\begin{cases} (x-x_{j-2})^3 & x \in [x_{j-2}, x_{j-1}) \\ h^3 + 3h^2(x-x_{j-1}) + 3h(x-x_{j-1})^2 - 3(x-x_{j-1}) & x \in [x_{j-1}, x_j) \\ h^3 + 3h^2(x_{j+1}-x) + 3h(x_{j+1}-x)^2 - 3(x_{j+1}-x)^3 & x \in [x_j, x_{j+1}) \\ (x_{j+2}-x)^3 & x \in [x_{j+1}, x_{j+2}) \\ 0 & otherwise \end{cases} \tag{3.6}$$

Table 3.1: Value of $\phi_j(x)$ and its derivative at nodal points.

	x_{i-1}	x_{i-2}	x_i	x_{i+1}	x_{i-2}
$\phi_{j,3}$	0	1	4	1	0
$\phi'_{j,3}$	0	3/h	0	-3/h	0
$\phi''_{j,3}$	0	6/h^2	-12/h^2	6/h^2	0

$\phi_j(x)$ is non negative and is locally supported on $[x_{j-2}\ ,\ x_{j+2}\ ]$. Besides it's easy to observe that $\phi_j(x)$=$\phi_{j+1}(x)$,(J=-1......N+1) and

$\sum_{j=-1}^{N+1}\phi_j(x)$= 1 (x$\in$[0,L]) for some trivial Computaion we obtain the knot value of $\phi_j(x)$. So the Considering the approximate solution (3.5) and cubic B-spline $\phi_j(x)$ defined in (3.6) and the nodal value of U and its derivatives U' ,U'' at the nodes are determined in term of the element parameters δ_j by

$$\begin{aligned} U(x_j,t) &= \delta_{j-1} + 4\delta_j + \delta_{j+1} \\ U'(x_j,t) &= 3/h(\delta_{j-1} - \delta_{j+1}) \\ U''(x_j,t) &= 6/h^2(\delta_{j-1} - 2\delta_j + \delta_{j+1}) \end{aligned} \tag{3.7}$$

We use the following differentiation notation in the one-dimensional case,

$$\dot{u} := u_t = \frac{\partial u}{\partial t}, u' := u_x = \frac{\partial u}{\partial x}, u'' := u_{xx} = \frac{\partial^2 u}{\partial x^2},$$

Now applying implict method scheme(weight averaged method) of solving parabolic second order one-dimensional (3.1) heat equation we obtained ,

$$(U_t)_j^n + (1-\theta)f_j^n + \theta f_j^{n+1} = 0, \tag{3.8}$$

where

$$f_j^n = \alpha^2(U_{xx})_j^n \tag{3.9}$$

Note that for $\theta = 0$ and $\theta = 1$, (3.9) yields the Explicit Forword Euler and Implicit Backword Euler respectively. A more popular scheme for implementation is when $\theta = 0.5$ which yields the Crank-Nicolson schem(Average of the explicit (forward Euler) and implicit (backward Euler) schemes) which is also unconditionally stable. Further, we discretize the time derivative by so we have:-

$$\frac{U_j^n - U_j^{n+1}}{\Delta t} + (1-\theta)\alpha^2(U_{xx})_j^n + \theta\alpha^2(U_{xx})_j^{n+1} = 0$$

$$U_j^n - U_j^{n+1} + \Delta t(1-\theta)\alpha^2(U_{xx})_j^n + \Delta t\theta\alpha^2(U_{xx})_j^{n+1} = 0 \tag{3.10}$$

Substituting(3.7) into (3.10) we obtained

$$\begin{aligned}&(\delta_{j-1}^{n}+4\delta_{j}^{n}+\delta_{j+1}^{n})-(\delta_{j-1}^{n+1}+4\delta_{j}^{n+1}+\delta_{j+1}^{n+1})\\&+\Delta t(1-\theta)\alpha^{2}6/h^{2}(\delta_{j-1}^{n}{}^{-}2\delta_{j}^{n}+\delta_{j+1}^{n})+\Delta t\theta\alpha^{2}6/h^{2}(\delta_{j-1}^{n+1}-2\delta_{j}^{n+1}+\delta_{j+1}^{n+1})\end{aligned}\tag{3.11}$$

which involves (n+3) unknown with (n+1) equations .

$$\begin{aligned}&\delta_{j-1}^{n+1}(1-6/h^{2}\Delta\theta\alpha^{2})+\delta_{j}^{n+1}(4+12/h^{2}\Delta\theta\alpha^{2})+\delta_{j+1}^{n+1}(1-6/h^{2}\Delta\theta\alpha^{2})\\&=L_{1}+\Delta t(1-\theta)\alpha^{2}L_{3}\end{aligned}\tag{3.12}$$

where

$$\begin{aligned}L_{1}&=(\delta_{j-1}^{n}+4\delta_{j}^{n}+\delta_{j+1}^{n})\\L_{2}&=3/h(\delta_{j+1}^{n}{}^{-}\delta_{j-1}^{n})\\L_{3}&=6/h^{2}(\delta_{j-1}^{n}-2\delta_{j}^{n}+\delta_{j+1}^{n})\end{aligned}\tag{3.13}$$

This set of equation is a recurrence relationship of element parameters vector d^{n}=δ_{-1}^{n},δ_{0}^{n},δ_{1}^{n}.....δ_{N}^{n},δ_{N+1}^{n}.
Now using given boundary conditions (3.3) and eleminating the parameters δ_{-1} and δ_{N+1}, in (3.13) then the system may be re written as,

$$U(x_{0})=\delta_{-1}^{n+1}+4\delta_{0}^{n+1}+\delta_{1}^{n+1}=0\Rightarrow\delta_{-1}^{n+1}=-(4\delta_{0}^{n+1}+\delta_{1}^{n+1})\tag{3.14}$$

$$U(x_{N})=\delta_{N-1}^{n+1}+4\delta_{N}^{n+1}+\delta_{N+1}^{n+1}=0\Rightarrow\delta_{N+1}^{n+1}=-(4\delta_{N}^{n+1}+\delta_{N-1}^{n+1})\tag{3.15}$$

with (3.14) (3.15) ,(3.11) turn out to be square system of (n+1) unknown at each time level t_{n}, n=0,1,2.... for J=0

$$\begin{aligned}&\delta_{-1}^{n+1}(1-6/h^{2}\Delta\theta\alpha^{2})+\delta_{0}^{n+1}(4+12/h^{2}\Delta\theta\alpha^{2})+\delta_{1}^{n+1}(1-6/h^{2}\Delta\theta\alpha^{2})\\&=\delta_{-1}^{n}(1+6/h^{2}\Delta(1-\theta)\alpha^{2})+\delta_{0}^{n}(4-12/h^{2}\Delta(1-\theta)\alpha^{2})+\delta_{1}^{n}(1+6/h^{2}\Delta(1-\theta)\alpha^{2})\end{aligned}\tag{3.16}$$

After simplification

$$\begin{aligned}&(-4\delta_{0}^{n+1}-\delta_{1}^{n+1})(1-6/h^{2}\Delta\theta\alpha^{2})+\delta_{0}^{n+1}(4+12/h^{2}\Delta\theta\alpha^{2})+\delta_{1}^{n+1}(1-6/h^{2}\Delta\theta\alpha^{2})\\&=(-4\delta_{0}^{n}-\delta_{1}^{n})(1+6/h^{2}\Delta(1-\theta)\alpha^{2})+\delta_{0}^{n}(4-12/h^{2}\Delta(1-\theta)\alpha^{2})\\&+\delta_{1}^{n}(1+6/h^{2}\Delta(1-\theta)\alpha^{2})\end{aligned}\tag{3.17}$$

Solve (3.17) we obtain

$$\tfrac{36}{h^{2}}\Delta\theta\alpha^{2}\delta_{0}^{n+1}=-\tfrac{36}{h^{2}}\Delta\theta\alpha^{2}\delta_{0}^{n}\tag{3.18}$$

For j=1

$$\delta_0^{n+1}(1-6/h^2\Delta\theta\alpha^2)+\delta_1^{n+1}(4+12/h^2\Delta\theta\alpha^2)+\delta_2^{n+1}(1-6/h^2\Delta\theta\alpha^2)$$
$$=\delta_0^{n}(1+6/h^2\Delta(1-\theta)\alpha^2)+\delta_1^{n}(4-12/h^2\Delta(1-\theta)\alpha^2)+\delta_2^{n}(1+6/h^2\Delta(1-\theta)\alpha^2) \tag{3.19}$$

for j=2

$$\delta_1^{n+1}(1-6/h^2\Delta\theta\alpha^2)+\delta_2^{n+1}(4+12/h^2\Delta\theta\alpha^2)+\delta_3^{n+1}(1-6/h^2\Delta\theta\alpha^2)$$
$$=\delta_1^{n}(1+6/h^2\Delta(1-\theta)\alpha^2)+\delta_2^{n}(4-12/h^2\Delta(1-\theta)\alpha^2)+\delta_3^{n}(1+6/h^2\Delta(1-\theta)\alpha^2) \tag{3.20}$$

and for J=N

$$\delta_{N-1}^{n+1}(1-6/h^2\Delta\theta\alpha^2)+\delta_N^{n+1}(4+12/h^2\Delta\theta\alpha^2)+\delta_{N+1}^{n+1}(1-6/h^2\Delta\theta\alpha^2)$$
$$=\delta_{N-1}^{n}(1+6/h^2\Delta(1-\theta)\alpha^2)+\delta_N^{n}(4-12/h^2\Delta(1-\theta)\alpha^2)+\delta_{N+1}^{n}(1+6/h^2\Delta(1-\theta)\alpha^2) \tag{3.21}$$

by (3.21), we get

$$\delta_{N-1}^{n+1}(1-6/h^2\Delta\theta\alpha^2)+\delta_N^{n+1}(4+12/h^2\Delta\theta\alpha^2)+(-\delta_{N-1}^{n+1}-4\delta_N^{n+1})(1-6/h^2\Delta\theta\alpha^2)$$
$$=\delta_{N-1}^{n}(1+6/h^2\Delta(1-\theta)\alpha^2)+\delta_N^{n}(4-12/h^2\Delta(1-\theta)\alpha^2)+$$
$$(-\delta_{N-1}^{n}-4\delta_N^{n})(1+6/h^2\Delta(1-\theta)\alpha^2) \tag{3.22}$$

further ,by solving above equation, we obtain

$$\tfrac{36}{h^2}\Delta\theta\alpha^2\delta_N^{n+1}=-\tfrac{36}{h^2}\Delta\theta\alpha^2\delta_N^{n} \tag{3.23}$$

Now put θ=$\frac{1}{2}$, Δ=k in (3.23) solving Tridiagonal systems because this system are particular easy to solve using Gaussian elimination. we have AC^{n+1}=BC^n Using given initial condition

1. $(\mathrm{U}_N)_x(x_0,0)=U_x(x_0)=f'(x_0)$
$\frac{3}{h}[\delta_{-1}^0-\delta_1^0]=f'(x_0)$
$U(x_0)=\delta_{-1}^0+4\delta_0^0+\delta_1^0$
$f(x_0)-4\delta_0^0-\delta_1^0=\delta_{-1}^0$
$\frac{3}{h}[f(x_0)-4\delta_0^0-\delta_1^0-\delta_1^0]=f'(x_0)$
$4\delta_0^0+2\delta_1^0=f(x_0)-\frac{h}{3}f'(x_0)$

2.$U_N(x_i,0)=U(x_i,0)$
$\delta_{i-1}^0+4\delta_i^0+\delta_{i+1}^0=f(x_j)$
for i=1, $\delta_0^0+4\delta_1^0+\delta_2^0=f(x_1)$
for i=2, $\delta_1^0+4\delta_2^0+\delta_3^0=f(x_2)$
for i=N-1, $\delta_{N-2}^0+4\delta_{N-1}^0+\delta_N^0=f(x_{N-1})$

$3.(U_N(x_N,0))_x = U_x(x_N,0)$
$4\delta_N^0 + 2\delta_{N+1}^0 = f(x_N) - \frac{h}{3}f'(x_N)$
Now we get initial matrix

$$\begin{pmatrix} 4 & 2 & & 0 \\ 1 & 4 & 1 & \ddots \\ & & & \\ & \ddots & \ddots & 1 \\ 0 & & 2 & 4 \end{pmatrix}_{(N+1)\times(N+1)} \begin{pmatrix} \delta_0^0 \\ \delta_1^0 \\ \vdots \\ \delta_{N-1}^0 \\ \delta_N^0 \end{pmatrix}_{(N+1)\times(1)} = \begin{pmatrix} f(x_0) - \frac{h}{3}f'(x_0) \\ f(x_1) \\ \vdots \\ f(x_{n-1}) \\ f(x_N) - \frac{h}{3}f'(x_N) \end{pmatrix}_{(N+1)\times(1)}$$

3.3 Stability Analysis

A very useful technique for analysing the stability of a finite difference method used to solve an initial value problem is the Fourier method, as same the stability analysis also using in finite element method. The stability of numerical schemes is closely associated with numerical error.
So now considering the equation

$$U_j^{n+1} - U_j^n - \Delta t(1-\theta)\alpha^2(U_{xx})_j^n - \Delta t\theta\alpha^2(U_{xx})_j^{n+1} = 0 \tag{3.24}$$

now sustituting (3.7)into (3.24) yields ,we obtain

$$\begin{aligned} &(\delta_{j-1}^{n+1} + 4\delta_j^{n-1} + \delta_{j+1}^{n+1}) - (\delta_{j-1}^n + 4\delta_j^n + \delta_{j+1}^n) \\ &-\Delta t(1-\theta)\alpha^2 6/h^2(\delta_{j-1}^n - 2\delta_j^n + \delta_{j+1}^n) - \Delta t\theta\alpha^2 6/h^2(\delta_{j-1}^{n+1} - 2\delta_j^{n+1} + \delta_{j+1}^{n+1}) \end{aligned} \tag{3.25}$$

Now re-write (3.25)

$$\begin{aligned} &\delta_{j-1}^{n+1}(1 - 6/h^2\Delta\theta\alpha^2) + \delta_j^{n+1}(4 + 12/h^2\Delta\theta\alpha^2) + \delta_{j+1}^{n+1}(1 - 6/h^2\Delta\theta\alpha^2) \\ &= \delta_{j-1}^n(1 + 6/h^2\Delta(1-\theta)\alpha^2) + \delta_j^n(4 - 12/h^2\Delta(1-\theta)\alpha^2) + \delta_{j+1}^n(1 + 6/h^2\Delta(1-\theta)\alpha^2) \end{aligned} \tag{3.26}$$

and using the fourier stability method, Let

$$\begin{aligned} &\delta_j^n = \xi^n e^{i\alpha j}, \delta_{j+1}^{n+1} = \xi^{n+1}e^{i\alpha(j+1} \\ &\delta_{j+1}^n = \xi^n e^{i\alpha(j+1)}, \delta_{j-1}^{n+1} = \xi^{n+1}e^{i\alpha(j-1)} \end{aligned} \tag{3.27}$$

puting the value(3.27) in the equation(3.26) we get

$$\begin{aligned} &\xi^{n+1}e^{i\alpha(j-1)}(1 - 6/h^2\Delta\theta\alpha^2) + \xi^{n+1}e^{i\alpha j}(4 + 12/h^2\Delta\theta\alpha^2) + \xi^{n+1}e^{i\alpha(j+1)}(1 - 6/h^2\Delta\theta\alpha^2) \\ &= \xi^n e^{i\alpha(j-1)}(1 + 6/h^2\Delta(1-\theta)\alpha^2) + \xi^n e^{i\alpha j}(4 - 12/h^2\Delta(1-\theta)\alpha^2) \\ &+\xi^n e^{i\alpha(j+1)}(1 + 6/h^2\Delta(1-\theta)\alpha^2) \end{aligned} \tag{3.28}$$

equation (3.28) divided by $\xi^n e^{i\alpha j}$ we get

$$\begin{aligned}
&\xi e^{-i\alpha}(1-6/h^2\Delta\theta\alpha^2)+\xi(4+12/h^2\Delta\theta\alpha^2)+\xi e^{i\alpha}(1-6/h^2\Delta\theta\alpha^2)\\
&=e^{-i\alpha}(1+6/h^2\Delta(1-\theta)\alpha^2)+(4-12/h^2\Delta(1-\theta)\alpha^2)\\
&+e^{i\alpha}(1+6/h^2\Delta(1-\theta)\alpha^2)
\end{aligned} \tag{3.29}$$

Now the equation by replacing $e^{-i\alpha}$ and $e^{i\alpha}$ in (3.39) ,we get

$$\begin{aligned}
&\xi(\cos\alpha-i\sin\alpha)(1-6/h^2\Delta\theta\alpha^2)+\xi(4+12/h^2\Delta\theta\alpha^2)+\xi(\cos\alpha+i\sin\alpha)(1-6/h^2\Delta\theta\alpha^2)\\
&=(\cos\alpha-i\sin\alpha)(1+6/h^2\Delta(1-\theta)\alpha^2)+(4-12/h^2\Delta(1-\theta)\alpha^2)\\
&+(\cos\alpha+i\sin\alpha)(1+6/h^2\Delta(1-\theta)\alpha^2)
\end{aligned} \tag{3.30}$$

Solve (3.30)

$$\begin{aligned}
&\xi((2\cos\alpha)(1-6/h^2\Delta\theta\alpha^2)+(4+12/h^2\Delta\theta\alpha^2))\\
&=(2\cos\alpha)(1+6/h^2\Delta(1-\theta)\alpha^2)+(4-12/h^2\Delta(1-\theta)\alpha^2)
\end{aligned} \tag{3.31}$$

simplify (3.31) equation and we get the value of ξ

$$\xi=\frac{(2\cos\alpha)(1+6/h^2\Delta(1-\theta)\alpha^2)+(4-12/h^2\Delta(1-\theta)\alpha^2)}{(2\cos\alpha)(1-6/h^2\Delta\theta\alpha^2)+(4+12/h^2\Delta\theta\alpha^2)} \tag{3.32}$$

Then the solution is stable for $\theta\in[\frac{1}{2},1]$ using the equation since the inequality $\mid\xi\mid\leq 1$ holds by Fourier stability method.

3.4 Numerical Results

In this section we consider test problem for our proposed method and We observed analytical and numerical solution:- The following formulas are used to find the absolute, L_2 and RMS errors

$$AbsoluteError = max|u^{num}-u^{exact}|,$$

$$L_2SpaceError=\sqrt{\Delta t\sum_{i=1}^{n}(u^{num}-u^{exact})^2},$$

$$RootMeanSquareError=\sqrt{\sum_{i=1}^{n}\frac{(u^{num}-u^{exact})^2}{n}},$$

Now consider example for numerical result.

$$\frac{\partial u}{\partial t}=\frac{\partial^2 u}{\partial x^2} \quad ,0\leq x\leq 1 \ t>0 \tag{3.33}$$

Table 3.2: Maximum absolute error at different time and time step length

t time	K=0.01	k=0.001	k=0.0001	k=0.00001
0.3	1.3707e-04	1.3851e-05	1.2611e-05	1.2606e-05
0.5	3.1707e-04	3.2065e-06	2.9190e-06	2.9185e-06
0.7	6.1608e-06	6.2353e-07	5.6760e-07	5.6748e-07
1.0	4.5506e-07	4.6116e-08	4.1974e-08	4.1966e-08
1.5	4.8983e-09	4.9277e-10	4.5270e-10	4.5268e-10
2.0	4.6867e-11	4.7234e-12	4.3401e-12	4.3399e-12
4.0	3.8453e-17	1.7883e-17	8.0518e-19	6.7635e-18

Table 3.3: L_2 space error at different time and time step length

t time	K=0.01	k=0.001	k=0.0001	k=0.00001
0.3	9.6926e-05	3.0972e-06	8.9173e-07	2.8190e-07
0.5	2.2420e-05	7.1700e-07	2.0640e-07	6.5259e-08
0.7	4.3563e-06	1.3942e-07	4.0135e-08	1.2689e-08
1.0	3.2178e-07	1.0310e-08	2.9680e-09	9.3840e-10
1.5	3.4636e-09	1.1018e-10	3.2011e-11	1.0122e-11
2.0	3.3140e-11	1.0562e-12	3.0689e-13	9.7044e-14
4.0	1.3812e-17	5.7900e-19	5.4881e-20	1.1827e-19

with initial condition

$$u(x,0) = \sin(\pi x) \quad , 0 \leq \text{x} \leq 1 \tag{3.34}$$

and boundary conditions

$$u(0,t) = u(1,t) = 0 \tag{3.35}$$

The exact solution is given by

$$u(x,t) = e^{-\pi^2 t} \sin(\pi x) \tag{3.36}$$

Numerical result of the example are shown in table 3.2-3.4 and fig 1-3 . In the table 3.2 we show absolute error at time step length k and different time. L_2 space error at time step length k and different time are shown in table 3.3. Root mean Square error at different time and time step length k. graphical representation of absolute error for k=0.00001 at t=2,t=4 are show in figure 1-2 and respectively graph of numerical solution at t=0.3,0.5,0.7 for k=0.00001 are show in figure 3.

Table 3.4: Root mean square error at different time and time step length

t time	K=0.01	k=0.001	k=0.0001	k=0.00001
0.3	9.6445e-05	9.7456e-06	8.8730e-06	8.8702e-06
0.5	2.2309e-05	2.2561e-06	2.0538e-06	2.0534e-06
0.7	4.3347e-06	4.3871e-07	3.9936e-07	3.9928e-07
1.0	3.2018e-07	3.2440e-08	2.9533e-08	2.9527e-08
1.5	3.4464e-09	3.4671e-10	3.1852e-10	3.1850e-10
2.0	3.2976e-11	3.3234e-12	3.0536e-12	3.0535e-12
4.0	1.3743e-17	1.8218e-18	5.4608e-19	3.7217e-18

Figure 3.1: Absolute error at t=2 and k=0.00001

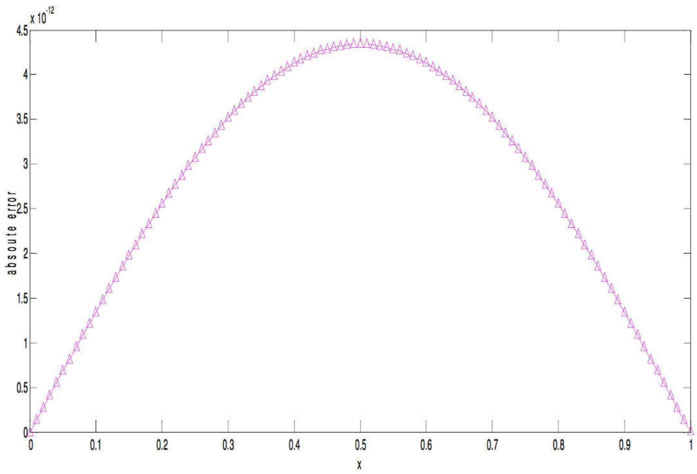


Figure 3.2: Absolute error at t=4 and k=0.00001

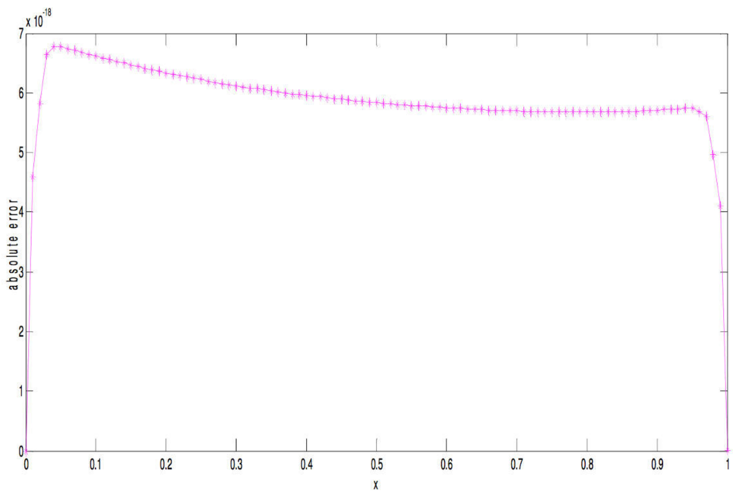
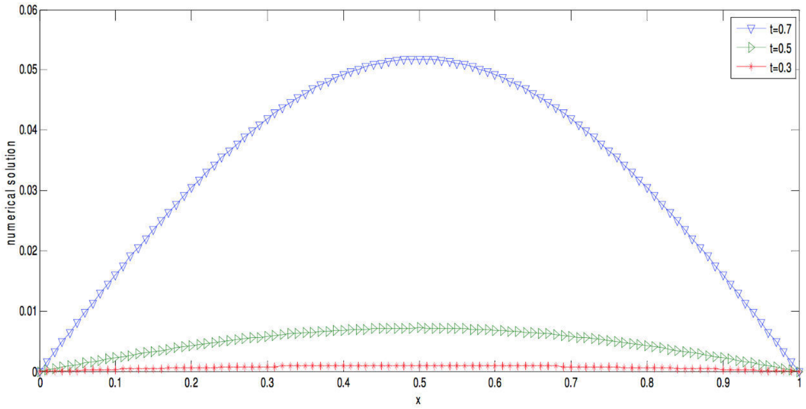


Figure 3.3: Numerical solution at different value of time t



3.5 Conclusion

In this thesis we have developed a numerical scheme based on B-spline collocation method.

Appendix A

Thomas Algorithm

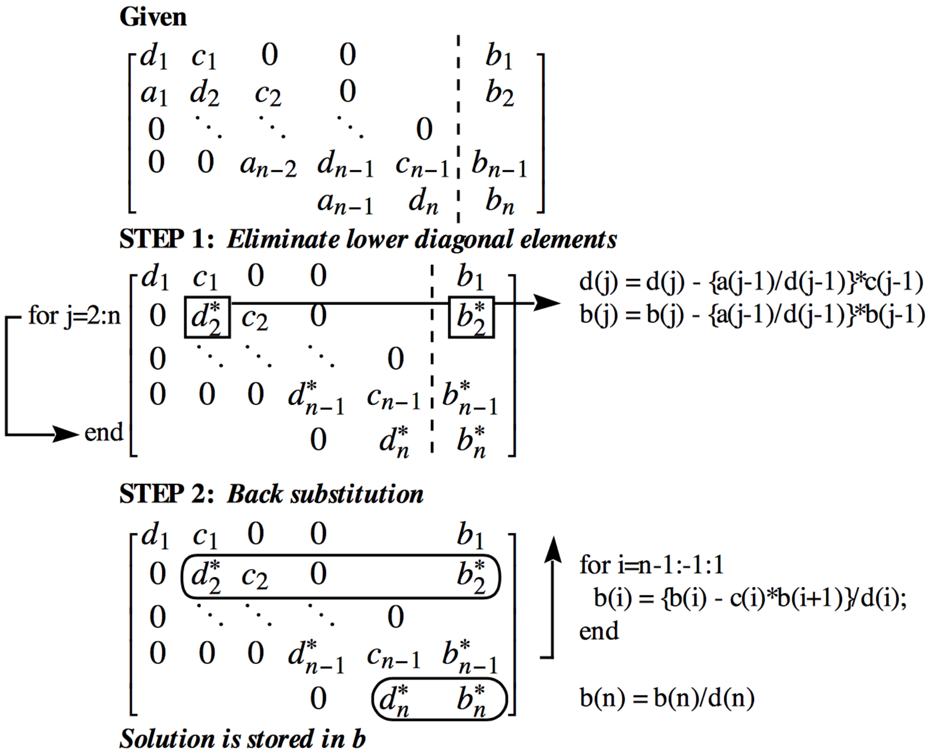
In this thesis we solving the heat equation via Tridiagonal Matrix . The Tridiagonal Matrix Algorithm also known as the Thomas Algorithm is an application of gaussian eliminated to a banded matrix. We saw how to take advantage of the banded structure of the finite difference generated matrix equation to create an efficient algorithm to numerically solve the heat equation.
When a system of linear equations has a special shape (symmetric, or tridiagonal), then it is recommended to use a method specifically developed for this kind of equation. Such methods are not only more efficient in term of computational time and computer memory, but also accumulate smaller round-off errors.
later thesis solution have involve solving a system of linear equations T x = b with a tridiagonal matrix structure like this

$$T = \begin{pmatrix} d_1 & c_1 & & 0 \\ a_1 & d_2 & \ddots & \\ & \ddots & \ddots & c_{n-1} \\ 0 & & a_{n-1} & d_n \end{pmatrix}.$$

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix}.$$

$$b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-1} \\ b_n \end{pmatrix}.$$

(A.0)



Thomas Algorithm

Bibliography

[1] H.S. Carslaw, J.C. Jaeger, *Conduction of Heat in solids*, Oxford University, Pres, 1959.

[2] D.V. Widder, *The Heat Equation*, Academic Press, Pres, 1976.

[3] J.R. Cannon, *The One-Dimensional Heat Equation*, Cambridge University Pres, 1984.

[4] S. Dhawan, S. Kumar, *A Numerical Solution of One Dimensional Heat Equation Using Cubic B-Spline Basis Functions*, International Journal of Research and Reviews in Applied Sciences, 1 (1) (2009) 71-77.

[5] A.M. Wazwaz, *Partial Differential Equations Methods and Applications*, Saint Xavier University, 2002.

[6] I. Dag , B. Saka , D. Irk, *Application Cubic B-splines for Numerical Solution of the RLW Equation*, Appl. Maths. and Comp., 159 (2004) 373389.

[7] C. de Boor, *A Practicle guide to splines*, Applied Mathematical Sciences Springer-Verlag, 2001.

[8] C. de Boor, *On Calculating with B-splines*, Journal of Approximation Theory, 6 (1972) 50- 62.

[9] S. Kutluay , A.R. Bahadr , A. Ozdes , *Numerical Solution of Onedimensional Burger Equation: Explicit and Exact-explicit Finite Difference Methods*, J. Comp. App. Math., 103 (1999) 251-261.

[10] I. Dag , D. Irk , B. Saka , *A Numerical Solution of the Burgers Equation Using Cubic B-splines*, Appl. Maths. and Comp. , 163 (2005) 199211.

[11] S.G. Rubin , P.K. Khosla , *Higher-Order Numerical Solutions Using Cubic Splines*, AIAA J. , 14 (1976) 851-858.

[12] S.G. Rubin, R.A. Graves , *Cubic Spline Approximation for Problems in Fluid Mechanics*, Nasa TR R-436 , Washington, DC, 1975.

[13] P.M. Prenter , *Splines and Variational Methods*, John Wiley , New York, 1975.

[14] A.K. Khalifa , K.R. Raslana , H.M. Alzubaidi , *A Collocation Method with Cubic B-splines for Solving the MRLW Equation*, Journal of Computational and Applied Mathematics , 212 (2008) 406 418.

[15] G. Sewell , *The Numerical Solution of Ordinary and Partial Differential Equations*, John Wiley and Sons, 2005.

[16] H. Caglar , M.Ozer, N.Caglar , *The Numerical Solution of the One-dimensional Heat Equation by Using Third Degree B-spline Functions*, Chaos, Solitons Fractals, 38 (2008) 1197-1201.

[17] A.R. Bahadr, *Application of Cubic B-spline Finite Element Technique to the Thermistor Problem*, Applied Mathematics and Computation, 149 (2004) 379387.

[18] B. Saka , I. Dag , *Quartic B-spline Collocation Method to the Numerical Solutions of the Burgers' Equation*, Chaos, Solitons Fractals, 32 (2007) 1125-1137.

[19] M.A. Ramadan , T.S. El-Danaf , F.E.I. Abd Alaal , *A Numerical Solution of the Burgers Equation Using Septic B-splines*, Chaos, Solitons Fractals, 26 (2005) 795-804.

[20] R.C. Mittal and Ram Jiwari, *A differential quadrature method for solving Burgers- type equation, International Journal of numerical methods for Heat and Fluid flow*, 22 (7), (2012), 880-895.

[21] Ram Jiwari, R.C. Mittal and K. K Sharma, *A numerical scheme based on weighted average differential quadrature method for the numerical solution of Burgers' equation*, Applied Mathematics and Computation, 219 (2013) 6680-6691.

[22] R.C. Mittal and Ram Jiwari, *A numerical scheme for singularly perturbed burger-huxley equation*, J. Appl. Math. Informatics, 29 (2011), No. 3-4, 813-829.