ISTANBUL KULTUR UNIVERSITY

NUMERICAL METHODS FOR PARTIAL DIFFERENTIAL EQAUTIONS

SERHAT YILMAZ

MA THESIS

JANUARY 2010

ISTANBUL KULTUR UNIVERSITY

INSTITUTE OF SCIENCE

DEPARMENT OF MATHEMATICS AND COMPUTER

NUMERICAL MTHDS

NAME AND SURNAME: Serhat YILMAZ

REGISTRATION NUMBER: 0809041044

SUPERVISOR AND CHAIRPERSON : Yard. Doç. Dr. Hikmet ÇAĞLAR

ABSTRACT

NUMERICAL METHODS FOR PARTIAL DIFFERENTIAL EQUATIONS

Numerical methods are used to find an approximation solution to problems in practice of science and engineering are often either diffucult or imppossible to solve analytically. In this study, we deal to find numerical solutions of some kinds of partial differential equations(PDE). PDE are used to formulate, and thus aid the solution of, problems involving functions of severable variables; such as the propagation of sound or heat, electrostatics, electrodynamics, fluid flow, and elasticity. Seemingly, distinct physical phenomena may have identical mathematical formulations, and thus be governed by the same underlying dynamic.

Here, we develop numerical hybrid methods to solve PDE. The first method based on non-poylnomial cubic splines in the space direction and finite difference in the time direction. We have seen by using spline functions additional smoothness can be achieved. In the second method we use finite elements methods with Galerkin method instead of splines in the space direction but ıt gives a heavy calculation and not beter results. Several numerical techniques have been proposed fort he numerical solution of PDE. These tecniques are compared giving by numerical examples, and all numerical results are illustrated using MATLAB 7.0.

KEYWORDS: Partial differential equations, Numeric methods, Collection methods, Non-polynomial cubic splines, Finite element methods, Galerkin methods.

ÖZET

KISMİ TÜREVLİ DİFERANSİYEL DENKLEMLERİN SAYISAL ÇÖZÜMLERİ

Numerik metodlar fizik ve mühendislik uygulamalarında ortaya çıkan ve çözümü çok zor veya mümkün olmayan problemler için yaklaşım çözümleri üretmek için kullanılır. Bu çalışmada, biz parçalı diferansiyel denklemlere sayısal çözümler bulunmasına odaklandık. Kısmi türevli diferansiyel denklemler(PDE) birkaç bilinmeyen değişkene bağlı fonksiyonlar içeren problemleri formülize etmekte ve tabii ki çözümlerinde kullanılmaktadır. PDE, Isı dağılımı, elektrostatik, elektrodinamik, akışkan dinamiği gibi farklı fizik problemlerinde karşımıza çıkarlar.

Bu çalışmada, PDE'i çözmek için sayısal hibrid metodlar gelişdirdik. İlk metot, uzay boyutunda polinom olmayan kübik splineların, zaman boyutunda ise sonlu farkların kullanılması üzerine inşa edilmiştir. İkinci metotda ise uzay boyutunda splineların yerine sonlu elamanlar Galerkin metoduyla birlikte kullanılmıştır. Sonuç olarak splineların daha yakınsak bir sonuç verdiği, buna karşılık sonlu fark metodunun hem daha ağır hesaplamalar gerektirdiği hem de daha iyi olmayan sonuçlar verdiği görülmüşdür. Farklı metotlar bizim metodumuzla sayısal örnekler verilerek kıyaslanmıştır. Sayısal çözümlerin hepsi MATLAB 7.0 kullanılarak hesaplanmışdır.

ANAHTAR KELİMELER VE SÖZCÜKLER: Parçalı diferansiyel denklemler, Sayısal Metodlar, Kolakasyon metodları, polinom olmayan splinelar, Sonlu Elemanlar, Galerkin metod.

To my family

Acknowledgements

thanks

Contents

Chapter 2

Partial Differential Equations

2.1 Preliminaries

A partial differential equation (PDE) describes a relation between an unknown function and its partial derivatives. PDEs appear frequently in all areas of physics and engineering. Moreover, in recent years we have seen a dramatic increase in the use of PDEs in areas such as biology, chemistry, computer sciences (particularly in relation to image processing and graphics) and in economics (finance). In fact, in each area where there is an interaction between a number of independent variables, we attempt to define functions in these variables and to model a variety of processes by constructing equations for these functions. When the value of the unknown function(s) at a certain point depends only on what happens in the vicinity of this point, we shall, in general, obtain a PDE. The general form of a PDE for a function $u(x_1, x_2, ..., x_n)$ is $F(x_1, x_2, ..., x_n, u, u_{x1}, u_{x2}, ..., u_{x11}, ...)$ =0, where $x_1, x_2, ..., x_n$ are the independent variables, u is the unknown function, and u_{xi} denotes the partial derivative du/dx_i. The equation is, in general, supplemented by additional conditions such as initial conditions (as we have often seen in the theory of ordinary differential equations (ODEs)) or boundary conditions.

The analysis of PDEs has many facets. The classical approach that dominated the nineteenth century was to develop methods for finding explicit solutions. Because of the immense importance of PDEs in the different branches of physics, every mathematical development that enabled a solution of a new class of PDEs was accompanied by significant progress in physics. Thus, the method of characteristics invented by Hamilton led to major advances in optics and in analytical mechanics. The Fourier method enabled the solution of heat transfer and wave propagation, and Greens methodwas instrumental in the development of the theory of electromagnetism. The most dramatic progress in PDEs has been achieved in the last 50 years with the introduction of numerical methods that allow the use of computers to solve PDEs of virtually every kind, in general geometries and under arbitrary external conditions (at least in theory; in practice there are still a large number of hurdles to be overcome).

2.2 Classification

We can say that PDEs are often classified into different types. In fact, there exist several classifications. Some of them will be describedin this section.
The first classification is according to the order of the equation. The order is defined to be the order of the highest derivative in the equation. If the highest derivative is of order k, then the equation is said to be of order k. Thus, for example, the equation $u_{tt} - u_{xx} = f(x,t)$ is called a second-order equation, while $u_t + u_{xxxx} = 0$ is called a fourth-order equation.
Another classification is into two groups: linear versus nonlinear equations. An equation is called linear if in (1.1), F is a linear function of the unknown function u and its derivatives. Thus, for example, the equation $x^3 u_x + e^y u_y + sin(x^2 + y^2)u = x^4$ is a linear equation, while $u_x^2 + u_y^2 = 1$ is a nonlinear equation. The nonlinear equations are often further classified into subclasses according to the type of the nonlinearity. Generally speaking, the nonlinearity is more pronounced when it appears in a higher derivative. For example, the following two equations are both nonlinear:
$u_{xx} + u_{yy} = u^2$ (1.2))
$u_{xx} + u_{yy} = |\nabla u|^2 u.$ (1.3)
Here $|\nabla u|$ denotes the norm of the gradient of u. While (1.3) is nonlinear, it is still linear as a function of the highest-order derivative. Such a nonlinearity is called quasilinear. On the other hand in (1.2) the nonlinearity is only in the unknown function. Such equations are often called semilinear.
Here we have to mention that there is a classification of the family of second-order linear equations for functions in two independent variables into three distinct types: hyperbolic (e.g., the wave equation), parabolic (e.g., the heat equation), and elliptic equations (e.g., the Laplace equation). It turns out that solutions of equations of the

same type share many exclusive qualitative properties.Also by a certain change of variables any equation of a particular type can be transformed into a canonical form which is associated with its type. An equation that has the form
$L[u] = au_{xx} + bu_{xy} + cu_{yy} + du_x + eu_y + fu = g$, (1.4)
where a, b, . . . , f, g are given functions of x, y, and u(x, y) is the unknown function. We assume that the coefficients a, b, c do not vanish simultaneously.The operator $L_0[u] = au_{xx} + bu_{xy} + cu_{yy}$ that consists of the second-(highest-)order terms of the operator L is called the principal part of L. It turns out that many fundamental properties of the solutions of (1.4) are determined by its principal part, and, more precisely, by the sign of the discriminant $\delta(L) := b^2 - 4ac$ of the equation. We classify the equation according to the sign of $\delta(L)$.Equation (1.4) is said to be hyperbolic at a point (x, y) if $\delta(L)(x, y) > 0$,it is said to be parabolic at (x, y) if $\delta(L)(x, y) = 0$, and it is said to be elliptic at (x, y) if $\delta(L)(x, y) < 0$.
Finally, a single PDE with just one unknown function is called a scalar equation. In contrast, a set of m equations with l unknown functions is called a system of m equations.

2.3 Associated conditions

PDEs have in general infinitely many solutions. In order to obtain a unique solution one must supplement the equation with additional conditions. What kind of conditions should be supplied? It turns out that the answer depends on the type of PDE under consideration. In this section we briefly review the common conditions.
Let us consider the convection equation $C_t + \vec{\nabla} \bullet (C\vec{u}) = 0$ in one spatial dimension as a prototype for equations of first order. The unknown function C(x, t) is a surface defined over the (x, t) plane. It is natural to formulate a problem in which one supplies the concentration at a given time t_0, and then to deduce from the equation the concentration at later times. Namely, we solve the problem consisting of the convection equation with the condition $C(x, t_0) = C_0(x)$. This problem is called an initial value problem. Geometrically speaking, condition determines a curve through which the solution surface must pass. We can generalize this condition by imposing a curve Γ that must lie on the solution surface, so that the projection of Γ on the (x,

t) plane is not necessarily the x axis. The last example involve PDEs with just a first derivative with respect to t. In analogy with the theory of initial value problems for ODEs, we expect that equations that involve second derivatives with respect to t will require two initial conditions. Therefore it is natural to supply two initial conditions, one for the initial location , and one for its initial velocity:
$u(x, 0) = u_0(x), u_t(x, 0) = u_1(x)$.
Another type of constraint for PDEs that appears in many applications is called boundary conditions. As the name indicates, these are conditions on the behavior of the solution (or its derivative) at the boundary of the domain under consideration. As an example, consider the heat equation with a spatial domain Ω:
$u_t = k(u_{xx} + u_{yy} + u_{zz})$ $(x, y, z) \in \Omega$, $t > 0$. (1.45)
We shall assume in general that Ω is bounded. It turns out that in order to obtain a unique solution, one should provide (in addition to initial conditions) information on the behavior of u on the boundary $\partial\Omega$. Excluding rare exceptions, we encounter in applications three kinds of boundary conditions. The first kind, where the values of the temperature on the boundary are supplied, i.e.
$u(x, y, z, t) = f(x, y, z, t)$ $(x, y, z) \in \Omega$, $t > 0$, (1.46)
is called a Dirichlet condition in honor of the German mathematician Johann Lejeune Dirichlet (1805.1859). For example, this condition is used when the boundary temperature is given through measurements, or when the temperature distribution is examined under a variety of external heat conditions. Alternatively one can supply the normal derivative of the temperature on the boundary; namely, we impose
$\partial_n u(x, y, z, t) = f(x, y, z, t)$ $(x, y, z) \in \Omega$, $t > 0$, (1.46)
This condition is called a Neumann condition after the German mathematician Carl Neumann (1832.1925). We have seen that the normal derivative $\partial_n u$ describes the flux through the boundary. For example, an insulating boundary is modeled by condition (1.47) with $f = 0$.
A third kind of boundary condition involves a relation between the boundary values of u and its normal derivative: $\alpha(x, y, z) + \partial_n u(x, y, z, t) + u(x, y, z, t) = f(x, y, z, t)$ $(x, y, z) \in \Omega$, $t > 0$, (1.46)
Such a condition is called a condition of the third kind. Sometimes it is also called the Robin condition.
Although the three types of boundary conditions defined above are by far the most common conditions seen in applications, there are exceptions. For example, we can

supply the values of u at some parts of the boundary, and the values of its normal derivative at the rest of the boundary. This is called a mixed boundary condition. Another possibility is to generalize the condition of the third kind and replace the normal derivative by a (smoothly dependent) directional derivative of u in any direction that is not tangent to the boundary. This is called an oblique boundary condition. Also, one can provide a nonlocal boundary condition. For example, one can provide a boundary condition relating the heat flux at each point on the boundary to the integral of the temperature over the whole boundary.

Chapter 3

Collocation

3.1 Introduction

Let X be a linear subspace of $L_2[D]$,the space of square integrable functions on D, where D is some subset of the real line E^1 or of the real plane E^2.And, Let L be a linear operator whose domain is X and whose range is also in X. Let $\{\Phi_1, \Phi_2,...,\Phi_N$ be a linearly independent subset of X, and let $X_N = span\{\Phi_1, \Phi_2, ..., \Phi_N\}$ be an N-dimensional subspace of X. Suppose we are given the linear equation:

$$Lx = y, \tag{3.1}$$

where y is given function from X. Approximating the solution x(t) of (3.1) by the method of collocation consists of finding a function $x_N(t) = a_1\Phi_1(t) + a_2\phi_2(t) + ... + a_n\phi_N(t)$ in X_N solving the $N \times N$ system of linear equations

$$Lx_N(t_i) = \sum_{j=1}^{N} a_j L\phi_j(t_i) = y(t_i), 1 \leq i \leq N, \tag{3.2}$$

where $t_1, t_2, ...t_N$ are N distinct points of D at which all the terms of (2) are defined. The function $x_N(t)$, if it exists, is said to collocate y(t) at the points $t_1, t_2, ...t_N$. Any function f(t) so obtained is referred to as an approximate solution obtained by the method of collocation.

In the following sections, we use method of collocation with non-polynomial cubic splines and with b-splines.

3.2 Non-polynomial Cubic Splines

Consider the polynomial interpolation, that is using approximations based on the idea of finding a polynomial which agrees with, or interpolates, the data and using this polynomial in place of the original function to estimate its values at other points. The use of local polynomials clearly improved the performance but at he expense of smoothness in the approximating function. However, additional smoothness can be achived with local polynomial interpolation by using spline functions in which different low-degree polynomials are used on each interval $[x_i, x_{i+1}]$ together with the imposition of smoothness conditions to ensure that the overall interpolating function has high a degree of continuity as possible at each of the nodes, x_i.

Definition 3.2.1. *Let $x_0 < x_1 < ... < x_N$ be an increasing sequence of nodes. The function s is a spline function of degree k if:*
(a) s is a polynomial of degree no more than k on each of the subintervals $[x_i, x_{i+1}]$.
(b) $s, s^{', ..., s^{k-1}}$ are all continous on the interval $[x_i, x_{i+1}]$.

By far the most commonly used splines for interpolation purposes are the cubic splines, and it is on these that we will concentrate our attention. The cubic spline functions have the form $span\,\{1, x, x^2, x^3$. In the present section, we apply non-polynomial cubic spline functions that have a polynomial and trigonometric parts to develop numerical methods for obtaining smooth approximations to the solution of partial differential equations. The spline functions we propose in this section have the form $span\ , x, coskx, sinkx$ where k is the frequency of the trginometric part of the spline functions which will be used to raise the accuracy of the method.
We turn now to the specific problem of obtaining a nonpolynomial cubic spline function which interpolates the fuction f at $x_0, x_1, ..., x_N$.
Let u(x) be a function defined on $[a, b]$. We divide the interval $[a, b]$ into n equal subintervals using the grid points

$$x_i = a + ih, i = 0, 1, 2, ..., n,$$

with

$$x_0 = a, x_n = b, h = (b - a)/n$$

where n is an arbitrary positive integer.
Let $u(x)$ be the exact solution and u_i be an approximation to $u(x_i)$ obtained by the non-polynomial cubic $S_i(x)$ passing through the points (x_i, u_i) and (x_{i+1}, u_{i+1}), we do not only require that $S_i(x)$ satisfies interpolatory conditions at x_i and x_{i+1}, but also the continuity of first derivative at the common nodes (x_i, u_i) are fulfilled. We write $S_i(x)$ in the form:

$$S_i(x) = a_i + b_i(x - x_i) + c_i sin\tau(x - x_i) + d_i cos\tau(x - x_i), i = 0, 1, ..., n - 1 \quad (3.3)$$

where a_i, b_i, c_i and d_i are constants and τ is a free parameter.

A non-polynomial function $S(x)$ of class $C^2[a, b]$ interpolates $u(x)$ at the grid points $x_i, i = 0, 1, 2, ..., n$, depends on a parameter τ, and reduces to ordinary cubic spline $S(x)$ in $[a, b]$ as $\tau \to 0$.

To derive expression for the coefficients of Eq. (3.3) in term of u_i, u_{i+1}, M_i and M_{i+1}, we first define:

$$S_i(x_i) = u_i, S_i(x_{i+1}) = u_{i+1}, S^{''}(x_i) = M_i, S^{''}(x_{i+1}) = M_{i+1}. \quad (3.4)$$

From algebraic manipulation, we get the following expression:

$$\begin{aligned} a_i &= u_i + \frac{Mi}{\tau^2}, \\ b_i &= \frac{u_{i+1} - u_i}{h} + \frac{M_{i+1} - M_i}{\tau\theta}, \\ c_i &= \frac{M_i cos\theta - M_{i+1}}{\tau^2 sin\theta}, \\ d_i &= -\frac{M_i}{\tau^2}, \end{aligned}$$

where θ=τh and $i = 0, 1, 2, ..., n - 1$.

Using the continuity of the first derivative at (x_i, u_i), that is $S^{'}_{i-1}(x_i) = S^{'}_i(x_i)$ we obtain the following relations for i=$1, ..., n - 1$.

$$aM_{i+1} + bM_i + aM_{i-1} = (1/h^2)(u_{i+1} - 2u_i + u_{i-1}) \quad (3.5)$$

where $a = (-1/\theta^2 + 1/\theta \sin\theta)$, $b = (1/\theta^2 - \cos\theta/\theta \sin\theta)$ and $\theta = \tau h$. The method is fourth-order convergent if $b = 5/12$ and $a = 1/12$ [9].
We next develop approximations for some examples:

Example 3.2.2. *Firstly we consider the second-order linear hyperbolic equation:*

$$u_{tt}(x,t) + 2\alpha u_t(x,t) + \beta^2 u(x,t) = u_{xx}(x,t) + f(x,t), \quad x \in (a,b), \quad t > 0 \tag{3.6}$$

with initial conditions

$$u(x,0) = \Phi(x), \quad u_t(x,0) = \Psi(x)$$

and boundary conditions

$$u(a,t) = g_1(t), \quad u(b,0) = g_2(t)$$

where α and β are constants.

The equation above represents a damped wave equation and a telegraph equation, the existence and approximations of the solutions investigated, see[1].

In recent years, many research has been done in developing and implementing modern high resolutions methods for the numerical solution of the second-order linear hyperbolic equation(1), *see*[1 − 3]. *Mohanty and Jain*[4 − 5] *and Mohanty*[6] *developed three-level implicit schemes for linear hyperbolic equations. Recently, Gao and Chi*[7] *proposed two semi-discretion methods to solve the one-space dimensional linear hyperbolic equation(1). Also, Huan-Wen Liu and Li-Bin Liu solved*[8] *linear hyperbolic equation. In this paper, we propose a non-polynomial cubic spline difference scheme to solve the linear hyperbolic equation*(1). *For every x_i, $i = 1, 2, ..., n-1$, by using the Taylor expansion in the time direction, we have the following difference schemes*

$$u(x_i,t_j) = \frac{u(x_i,t_{j+1}) + 2u(x_i,t) + u(x_i,t_{j-1})}{4} + O(k^2), \tag{3.7}$$

$$u_{xx}(x_i,t_j) = \frac{u_{xx}(x_i,t_{j+1}) + u_{xx}(x_i,t_{j-1})}{2} + O(k^2), \tag{3.8}$$

$$u_t(x_i,t_j) = \frac{u(x_i,t_{j+1}) - u(x_i,t_{j-1})}{2k} + O(k^2), \tag{3.9}$$

$$u_{tt}(x_i,t_j) = \frac{u(x_i,t_{j+1}) - 2u(x_i,t_j) + u(x_i,t_{j-1})}{k^2} + O(k^2). \tag{3.10}$$

The given eq. (1) *can be discretized as*

$$\begin{aligned}&\frac{u(x_i,t_{j+1})-2u(x_i,t_j)+u(x_i,t_{j-1})}{k^2}\\&+2\alpha\frac{u(x_i,t_{j+1})-u(x_i,t_{j-1})}{2k}\\&+\beta^2\frac{u(x_i,t_{j+1})+2u(x_i,t)+u(x_i,t_{j-1})}{4}\\&=\frac{u_{xx}(x_i,t_{j+1})+u_{xx}(x_i,t_{j-1})}{2}+f(x_i,t_j)+O(k^2).\end{aligned}\tag{3.11}$$

$i=1,2,...,n-1,\ j=1,2,...$

We can rewrite (3.5) *in a new form :*

$$(1+\frac{1}{12}\delta_x^2)M(x_i,t_j)=\frac{1}{h^2}\delta_x^2u(x_i,t_j), i=1,...,n-1\tag{3.12}$$

where

$$\begin{aligned}\delta_xM(x_i,t_j)&=M(x_{i+\frac{1}{2}},t_j)-M(x_{i-\frac{1}{2}},t_j),\\ \delta_x^2M(x_i,t_j)&=\delta_x(\delta_xM(x_i,t_j))=M(x_{i+1},t_j)-2M(x_i,t_j)+M(x_{i-1},t_j)\end{aligned}$$

for $i=1,...,n-1$*. Putting (3.8) and (3.12),it follows that;*

$$\begin{aligned}(1+\frac{1}{12}\delta_x^2)[u_{xx}(x_i,t_{j+1})+u_{xx}(x_i,t_{j-1})]&=(1+\frac{1}{12}\delta_x^2)[M(x_i,t_{j+1})+M(x_i,t_{j-1})+O(h^4)]\\&=\frac{1}{h^2}\delta_x^2[u(x_i,t_{j+1})+u(x_i,t_{j-1})]+O(h^4)\end{aligned}\tag{3.13}$$

Applying the operator $(1+\frac{1}{12}\delta_x^2)$ *to two sides of Eq.* (3.11) *and using Eq.* (3.13)*,then it is obtained as follows*

$$\begin{aligned}&\frac{1}{k^2}(1+\frac{1}{12}\delta_x^2)u(x_i,t_j)+\frac{\alpha}{k}(1+\frac{1}{12}\delta_x^2)\delta_tu(x_i,t_j)\\&+\frac{\beta^2}{4}(1+\frac{1}{12}\delta_x^2)[u(x_i,t_{j+1}+2u(x_i,t_j)+u(x_i,t_{j-1})]\\&-\frac{1}{2h^2}\delta_x^2[u(x_i,t_{j+1})u(x_i,t_{j-1})]\\&=(1+\frac{1}{12}\delta_x^2)f(x_i,t_j)+O(k^2+h^4)i=1(1)n-1,j=1,2,...\end{aligned}\tag{3.14}$$

The proposed scheme (3.14) *is an implicit three level scheme. To start any computation, it is necessary to know the value of* $u(x,t)$ *at the nodal points of the first time level, that is, at* $t=k$*. Following* [12]*,a taylor series expansion at t=k may be written as*

$$u(x,k)=u(x,0)+ku_t(x,0)+\frac{k^2}{2}u_{tt}(x,0)+\frac{k^3}{6}u_{ttt}(x,0)+O(k^4).\tag{3.15}$$

Using the initial values, from (1) *we can calculate*

$$u_{tt}(x,0) = \phi_{xx}(x,0) + f(x,0) - 2\alpha u_t(x,0) - \beta^2 u(x,0), \tag{3.16}$$

$$u_{ttt}(x,0) = \psi_{xx}(x,0) + f_t(x,0) - 2_{tt}(x,0) - \beta^2 u_t(x,0). \tag{3.17}$$

We can obtain the numerical solution of u *by using initial values in*(3.16) *and* (3.17) *for* $t = k$.

Now let us consider a specific equation:

$$u_{tt}(x,t) + 4u_t(x,t) + 2u(x,t) = u_{xx}(x,t), \quad x \in (a,b), \quad t > 0 \tag{3.18}$$

with initial conditions

$$u(x,0) = sinx, \quad u_t(x,0) = -sinx$$

and boundary conditions

$$u(0,t) = 0, \quad u(\pi,0) = 0.$$

The exact solution of the above problem is $u(x,t) = e^{-t}sinx$. *The problem is solved by using the scheme*(3.14) *in this paper. The absolute errors given by the scheme* (16) *in* [7], *by the scheme* (23) *in* [8] *and by present scheme (13) are listed in Tables 3.1-3.4, respectively. It can be seen from tables, when*$h = \frac{\pi}{300}$ *and* $k = 0.1$, *the accuracy of solutions obtained by using the scheme* (16) *in* [7] *is much better than those by using the present scheme*(13). *The reason is that the error orders of the scheme* (16) *in* [7] *is approximately* $O(k^5)$ *as the step length* h *is quite small. When* k *decreases to* $k = 0.1$ *and* $k = 0.01$, *respectively ,since* k *is now quite small in comparison with* h, *the errors of numerical solutions mainly come from the approximation in space direction,therefore the absolute errors by using the present scheme*(13) *is much better than those by using the scheme* (16) *in* [7]. *Finally, we have to mention that the absolute errors of scheme* (23) *in* [8] *are similar to those by using the present scheme*(3.14) *where scheme* (23) *in* [8] *using quartic spline functions, we use non-polynomial spline functions.*

Table 3.1: Absolute errors of the scheme(16) *in*[7], *the scheme* (23) *in* [8] *and the present scheme*(3.14) ($h = \frac{\pi}{300}, k = 0.1$).

	t	$x=\frac{\pi}{30}$	$x=\frac{8\pi}{30}$	$x=\frac{15\pi}{30}$	$x=\frac{22\pi}{30}$	$x=\frac{29\pi}{30}$
[7]	*1.0*	*0.0105e-05*	*0.0747e-05*	*0.1005e-05*	*0.0747e-05*	*0.0105e-05*
[8]	*1.0*	*0.0379e-03*	*0.2969e-03*	*0.4033e-03*	*0.3026e-03*	*0.0463e-03*
[*Thepresents.*]	*1.0*	*0.0429e-03*	*0.3056e-03*	*0.4112e-03*	*0.3056e-03*	*0.0429e-03*
[7]	*2.0*	*0.1015e-06*	*0.7215e-06*	*0.9709e-06*	*0.7215e-06*	*0.1015e-06*
[8]	*2.0*	*0.0389e-03*	*0.3039e-03*	*0.4128e-03*	*0.3096e-03*	*0.0475e-03*
[*Thepresents.*]	*2.0*	*0.0435e-03*	*0.3092e-03*	*0.4161e-03*	*0.3092e-03*	*0.0435e-03*

Table 3.2: Absolute errors of the scheme(16) *in*[7], *the scheme* (23) *in* [8] *and the present scheme*(3.14) ($h = \frac{\pi}{30}, k = 0.1$).

	t	$x=\frac{\pi}{30}$	$x=\frac{8\pi}{30}$	$x=\frac{15\pi}{30}$	$x=\frac{22\pi}{30}$	$x=\frac{29\pi}{30}$
[7]	*1.0*	*0.0904e-04*	*0.6479e-04*	*0.8928e-04*	*0.7069e-04*	*0.0904e-04*
[8]	*1.0*	*0.0321e-03*	*0.6995e-03*	*0.2033e-03*	*0.6995e-03*	*0.0412e-03*
[*Thepresents.*]	*1.0*	*0.0298e-03*	*0.3055e-03*	*0.4111e-03*	*0.3055e-03*	*0.0429e-03*
[7]	*2.0*	*0.0884e-04*	*0.6337e-04*	*0.8731e-04*	*0.6913e-04*	*0.0884e-04*
[8]	*2.0*	*0.0532e-03*	*0.5065e-03*	*0.3128e-03*	*0.5065e-03*	*0.0331e-03*
[*Thepresents.*]	*2.0*	*0.0349e-03*	*0.3092e-03*	*0.4161e-03*	*0.3092e-03*	*0.0865e-03*

Table 3.3: Absolute errors of the scheme(16) *in*[7], *the scheme* (23) *in* [8] *and the present scheme*(3.14) ($h = \frac{\pi}{30}, k = 0.01$).

	t	$x=\frac{\pi}{10}$	$x=\frac{3\pi}{10}$	$x=\frac{5\pi}{10}$	$x=\frac{7\pi}{10}$	$x=\frac{9\pi}{10}$
[7]	*1.0*	*0.2947e-04*	*0.7717e-04*	*0.9539e-04*	*0.7717e-04*	*0.2947e-04*
[8]	*1.0*	*0.0886e-05*	*0.3170e-05*	*0.4242e-05*	*0.3170e-05*	*0.0886e-05*
The present s.	*1.0*	*0.1320e-05*	*0.3456e-04*	*0.4272e-05*	*0.3456e-05*	*0.1320e-05*
[7]	*2.0*	*0.2883e-04*	*0.7548e-04*	*0.9330e-04*	*0.7548e-04*	*0.2883e-04*
[8]	*2.0*	*0.0871e-05*	*0.3114e-05*	*0.4167e-05*	*0.3114e-05*	*0.0871e-05*
The present s.	*2.0*	*0.1295e-05*	*0.3116e-05*	*0.4193e-05*	*0.3392e-05*	*0.1295e-05*

Table 3.4: Absolute errors of the scheme(16) in[7], the scheme (23) in [8] and present scheme(3.14),($h = \frac{\pi}{30}, k = 0.001$).

	t	$x = \frac{\pi}{10}$	$x = \frac{3\pi}{10}$	$x = \frac{5\pi}{10}$	$x = \frac{7\pi}{10}$	$x = \frac{9\pi}{10}$
[7]	*1.0*	*0.2947e-04*	*0.7717e-04*	*0.9539e-04*	*0.7717e-04*	*0.2947e-0*
[8]	*1.0*	*0.1839e-08*	*0.6574e-08*	*0.8799e-08*	*0.6574e-08*	*0.1839e-0*
The present s.	*1.0*	*0.1096e-08*	*0.7134e-08*	*0.8819e-08*	*0.7134e-08*	*0.1096e-0*
[7]	*2.0*	*0.28834e-04*	*0.75489e-04*	*0.93309e-04*	*0.75489e-04*	*0.28834e-*
[8]	*2.0*	*0.1799e-08*	*0.6433e-08*	*0.8609e-08*	*0.6433e-08*	*0.1799e-0*
The present s.	*2.0*	*0.1992e-08*	*0.7029e-08*	*0.8398e-08*	*0.7029e-08*	*0.1992e-0*

Example 3.2.3. *Secondly we consider the generalized Fisher's equation:*

$$u_t(x,t) = u_{xx}(x,t) + \alpha u(x,t)(1 - u^{\beta}(x,t)) \qquad a < x < b, \qquad t > 0 \qquad (3.19)$$

with initial condition

$$u(x,0) = \Phi(x),$$

and boundary conditions

$$u(a,t) = g_1(t), \quad u(b,t) = g_2(t)$$

where α and β are constants.

The classic and simplest case of the nonlinear reaction-diffusion equation is when β=1. It was suggested by Fisher as a deterministic version of a stochastic model for the spatial spread of a favored gene in a population [1].

$$u_t(x,t) = u_{xx}(x,t) + \alpha u(x,t)(1 - u(x,t)) \qquad a < x < b, \qquad t > 0 \qquad (3.20)$$

This equation is referred to as the Fisher equation, the discovery, investigation and analysis of traveling waves in chemical reactions was first presented by Luther [2]. *In the last century, the Fisher's equation has became the basis for a variety of models for spatial spread, for example, in logistic population growth models* $[3,4]$, *flame propagation* $[5,6]$, *neurophysiology* [7], *autocatalytic chemical reactions* $[8-10]$, *branching*

Brownian motion processes [11], *gene-culture waves of advance* [12], *the spread of early farming in Europe* [13, 14], *and nuclear reactor theory* [15]. *It is incorporated as an important constituent of nonscalar models describing excitable media, e.g., the Belousov-Zhabotinsky reaction* [16]. *In chemical media, the function $u(x,t)$ is the concentration of the reactant and the positive constant α represents the rate of the chemical reaction. In media of other natures, u might be temperature or electric potential.*

The mathematical properties of equation (1) *have been studied extensively and there have been numerous discussions in the literature. The most remarkable summaries have been provided by Brazhnik and Tyson* [17]. *One of the first numerical solutions was presented in literature with a pseudo-spectral approach. Implicit and explicit finite differences algorithms have been reported by different authors such as Parekh and Puri and Twizell et al. A Galerkin finite element method was used by Tang and Weber whereas Carey and Shen* [18] *employed a least-squares finite element method. A collocation approach based on Whittakers sinc interpolation function* [19] *was also considered in* [20]. *Our solution based on non-polynomial spline method. In this paper, we propose a spline difference scheme to solve eq.* (2).
Let the region $R = (a,b) \times (0,\infty]$, be discretized by a set of points $R_{h,k}$ which are the vertices of a grid points (x_i, t_j), where $x_i = ih$, $i = 0, 1, 2, ..., n, nh = 1$, and $t_j = jk$, $j = 0, 1, 2, 3$. Here h and k are mesh size in the space and time directions respectively.

We develop an approximation for eq.(2) *in which the time derivative is replaced by a finite difference approximation and the space derivative by the non-polynomial cubic spline function approximation. We need the following finite difference approximation for the time derivative of u. Let:*

$$\bar{u}_t(x_i, t_j) = \tfrac{u(x_i,t_{j+1}) - u(x_i,t_{j-1})}{2k} = u_t(x_i, t_j) + O(k^2). \quad (7)$$

At the grid point (i,j), the proposed differential equation (2) *may be discretized by:*

$$\bar{u}_t(x_i, t_j) = \bar{u}_{xx}(x_i, t_j) + \alpha u(x_i, t_j)(1 - u(x_i, t_j)). \quad (8)$$

By using (7) *in equation* (8) *we obtain*

$$\frac{u(x_i,t_{j+1})-u(x_i,t_{j-1})}{2k} = M(x_i,t_j) - \alpha u(x_i,t_j)(1-u(x_i,t_j)), \quad (9)$$

where $M(x_i,t_j) = S''(x_i)$ *is the second spline derivative at* (x_i,t_j). *From* (9) *we have*

$$M(x_i,t_j) = \frac{u(x_i,t_{j+1})-u(x_i,t_{j-1})}{2k} - \alpha u(x_i,t_j)(1-u(x_i,t_j)), \quad (10)$$

then we have

$$M(x_{i+1},t_j) = \frac{u(x_{i+1},t_{j+1})-u(x_{i+1},t_{j-1})}{2k} - \alpha u(x_{i+1},t_j)(1-u(x_{i+1},t_j)), \quad (11)$$

and

$$M(x_{i-1},t_j) = \frac{u(x_{i-1},t_{j+1})-u(x_{i-1},t_{j-1})}{2k} - \alpha u(x_{i-1},t_j)(1-u(x_{i-1},t_j)), \quad (12)$$

substituting (10), (11) *and* (12) *into* (6), *after simplifying we obtain:*

$$\begin{aligned}
&\lambda u(x_{i+1},t_{j+1}) + \mu u(x_i,t_{j+1}) + \lambda u(x_{i-1},t_{j+1}) \\
&+a\alpha(u(x_{i+1},t_{j+1}))^2 + 2b(u(x_i,t_{j+1}))^2 + a\alpha(u(x_{i-1},t_{j+1}))^2 \\
&\tfrac{a}{k}u(x_{i+1},t_{j-1}) - \tfrac{2b}{k}u(x_i,t_{j-1}) - \tfrac{a}{k}u(x_{i-1},t_{j-1}) = 0 \qquad (13)
\end{aligned}$$

where $\lambda = \frac{a}{k} - a\alpha - \frac{1}{h^2}$, $\mu = \frac{2b}{k} - 2b\alpha + \frac{2}{h^2}$ *and* a *is parameter.*

4. The similar scheme with using Taylor expansion

Expanding (13) in Taylor series in terms of $u(x_i,t_j)$ *and its derivatives, we can obtain another scheme(17). To do this, for simplicity, let* (x_i,t_j) *denote the grid points given by* $x_i = a + ih$,$i = 0,1,...,n$, *and* $t_j = jk$, $j = 0,1,2,...$*For every* x_i, $i = 1,2,...,n-1$, *by using the Taylor expansion in the time direction, we have the following difference schemes*

$$u(x_i,t_j) = \frac{u(x_i,t_{j+1})+2u(x_i,t_j)+u(x_i,t_{j-1})}{4} + O(k^2) \quad (14)$$

$$u_{xx}(x_i,t_j) = \frac{u_{xx}(x_i,t_{j+1})+u_{xx}(x_i,t_{j-1})}{2} + O(k^2) \tag{15}$$

$$u_t(x_i,t_j) = \frac{u(x_i,t_{j+1})-u(x_i,t_{j-1})}{2k} + O(k^2) \tag{16}$$

By using (14), (15) and (16),the given equation (2) *can be discretized as*

$$\frac{u(x_i,t_{j+1})-u(x_i,t_j-1)}{2k} - \alpha\frac{u(x_i,t_{j+1})+2u(x_i,t_j)+u(x_i,t_{j-1})}{4}\left(1 - \frac{u(x_i,t_{j+1})+2u(x_i,t_j)+u(x_i,t_{j-1})}{4}\right)$$

$$= \frac{u_{xx}(x_i,t_{j+1})+u_{xx}(x_i,t_{j-1})}{2} \qquad i = 1,...,n-1, j = 1,2,... \tag{17}$$

So, (6) *equality can be rewritten in a new form as follow :*

$$(1 + \frac{1}{12}\delta_x^2)M(x_i,t_j) = \frac{1}{h^2}\delta_x^2 u(x_i,t_j),\ i = 1,...,n-1 \tag{18}$$

where
$\delta_x M(x_i,t_j) = M(x_{i+\frac{1}{2}},t_j) - M(x_{i-\frac{1}{2}},t_j)$,
$\delta_x^2 M(x_i,t_j) = \delta_x(\delta_x M(x_i,t_j)) = M(x_{i+1},t_j) - 2M(x_i,t_j) + M(x_{i-1},t_j)$,

for $i = 1,...,n-1$. *Using (14) and (18), we have;*

$$(1 + \frac{1}{12}\delta_x^2)[u_{xx}(x_i,t_{j+1}) + u_{xx}(x_i,t_{j-1})] = (1 + \frac{1}{12}\delta_x^2)[M(x_i,t_{j+1}) + M(x_i,t_{j-1}) + O(h^4)]$$

$$= \frac{1}{h^2}\delta_x^2[u(x_i,t_{j+1}) + u(x_i,t_{j-1})] + O(h^4). \tag{19}$$

If we apply the operator $(1 + \frac{1}{12}\delta_x^2)$ *to two sides of* (17) *and use* (19), *we have*

$$\frac{1}{2k}(1 + \frac{1}{12}\delta_x^2)[u(x_i,t_{j+1}) - u(x_i,t_{j-1})] - \frac{\alpha}{4}(1 + \frac{1}{12}\delta_x^2)[u(x_i,t_{j+1}) + 2u(x_i,t_j) + u(x_i,t_{j-1})]$$

$$+ \frac{\alpha}{16}(1 + \frac{1}{12}\delta_x^2)[u(x_i,t_{j+1}) + 2u(x_i,t_j) + u(x_i,t_{j-1})]^2 - \frac{1}{2h^2}\delta_x^2[u(x_i,t_{j+1}) + u(x_i,t_{j-1})] = 0$$

$$i = 1,2,...,n-1, j = 1,2,... \tag{20}$$

The proposed scheme (20) *is an implicit three level scheme. To start any computation, it is necessary to know the value of* $u(x,t)$ *at the nodal points of the first time* *level, that is, at* $t = k$. *Following* [22], *a taylor series expansion at t=k may be written as*

$$u(x,k) = u(x,0) + ku_t(x,0) + \frac{k^2}{2}u_{tt}(x,0) + \frac{k^3}{6}u_{ttt}(x,0) + O(k^4) \qquad (21)$$

Using the initial values, from (1) *we can calculate*

$$u_{tt}(x,0) = \phi_{xx}(x,0) + f(x,0) - 2\alpha u_t(x,0) - \beta^2 u(x,0), \qquad (22)$$

$$u_{ttt}(x,0) = \psi_{xx}(x,0) + f_t(x,0) - 2\alpha u_{tt}(x,0) - \beta^2 u_t(x,0), \qquad (23)$$

Thus using initial value and (22), *from* (23), *we may obtain the numerical solution of* u *at* $t = k$.

5. Numerical example

In this section, we test our scheme on an example. All computations are done by using MATLAB 7.0.

We consider the numerical results obtained by apllying the schemes dicussed above to the following Fisher's equation

$$u_t(x,t) = u_{xx}(x,t) + 6u(x,t)(1 - u(x,t)), \quad 0 < x < 1, \ \ t > 0$$

with initial condition

$$u(x,0) = \frac{1}{(1+e^x)^2},$$

and boundary conditions

$$u(0,t) = \frac{1}{(1+e^{-5t})^2}, \quad u(1,t) = \frac{1}{(1+e^{1-5t})^2}$$

The exact solution of the above problem is $u(x,t) = \frac{1}{(1+e^{x-5t})^2}$. *The problem is solved by using the schemes* (13) *and* (20) *in this paper.The maximum absolute errors given by scheme* (13) *are listed in Table 1 and given by scheme* (20) *are listed in Table 2.The results prove that the second method with using Taylor expansion is more accurate than the first method in this paper.Also, numerical results given by scheme* (13) *and given by scheme* (20) *are shown in Fig. 1 and Fig. 2, respectively.*

Example 3.2.4. *Hyperbolic partial differential equations play a very important role modern applied mathematics due to their deep physical background. Hyperbolic differential equation subject to an integral conservation condition in one space dimensiona,feature in the mathematical modelling of many phenomena. Recently, much attention has been paid in the literature to the development, analsis and implementation of accurate methods.In this paper we will consider a non-classic hyperbolic equation [1]:*
We consider the following problem of this family of equations:

$$\frac{\partial u}{\partial t} = \gamma \frac{\partial^2 u}{\partial x^2}, (x,t) \in (0,1) \times (0,T] \qquad (1.1)$$

with initial condition

$$u(x,0) = f(x), 0 \leq x \leq 1, \qquad (1.2)$$

and boundary conditions
$u(0,t)=g(t),\ 0\leq t \leq T,$ (1.3)
$\int_0^x u(x,t)dx = m(t), 0 < x \leq T,$ (1.4)

where g *and* m *are known functions.*
$u'' + a_1(x)u' + a_2(x)u + a_3(x)v' + a_4(x)v = f_1(x),$

$$v'' + b_1(x)v' + b_2(x)v + b_3(x)u' + b_4(x)u = f_2(x)\ ,$$

(1)

with the following boundary conditions

$$u(0) = u(1) = 0, v(0) = v(1) = 0 \quad (2)$$

where $a_i(x), b_i(x), f_1(x)$ *and* $f_2(x)$ *are given functions for* $i = 1, 2, 3, 4$.

The analytical solutions of eqs. (1) and (2) have been studied by a number of authors [Lu, Abbas, Sami]. In [Lu] the variational iteration method is applied for the solutions with the assumption that the solutions are unique. In [Sami] the homotopy analysis method (HAM) is applied for the solutions and a new modification of HAM is proposed. The comparasion between the modified HAM and standard HAM is also presented in [Sami]. Recently He's homotopy perturbation method (HPM) is applied to the problems (1) and (2) in [Abbas]. Their method consists of reducing the solutions to a system of integral equations and using the HPM for this system. In a recent work [Caglar1], we have discussed the numerical solutions of the linear system of the second-order BVPs using the third-degree B-splines. It has been shown that the B-spline method is workable and cable of solve the the linear system of the second-order BVPs.

On the other hand, the non-polynomial spline method is also very useful and effective tool and used for a large variety of problems by several authors, e.g. [islam,ras1,ras2,ras3]. Islam et al. [islam] considered the non-polynomial spline method for the solution of a system of second-order BVPs. Recently, Rashidinia et al. [ras1] introduced the non-polynomial spline method for the second-order hyperbolic equations with mixed boundary conditions. More recently, Rashidinia et al. [ras2, ras3] have also showed that this method can be successfully implemented to the numerical solution of non-linear singular BVPs. They used the quesilinearization technique to reduce the given non-linear problem to a sequence of linear problems.

In the present work, we present a numerical solution of Eq. (1) with boundary conditions (2) by using the non-polynomial spline method. Two linear test problems *are considered for the numerical illustration of the method and numerical results are illustrated using MATLAB 6.5. We have showed that the proposed method is a full computational method and a powerful tool for solving the linear system of the second-order BVPs.*

Example 3.2.5. *Parabolic partial differential equations play a very important role modern applied mathematics due to their deep physical background. Parabolic differential equation with non-local boundary conditions in one, two or three space dimentions,feature in the mathematical modelling of many phenomena. Recently, much attention has been paid in the literature to the development, analsis and implementation of accurate methods.In this paper we will consider a non-classic parabolic equation [1]:*

$$\frac{\partial u}{\partial t} = \gamma \frac{\partial^2 u}{\partial x^2}, (x,t) \in (0,1) \times (0,T] \qquad (1.1)$$

with initial condition

$$u(x,0) = f(x), 0 \leq x \leq 1, \qquad (1.2)$$

and boundary conditions

$u(0,t)=g(t),\ 0\leq t \leq T,$ (1.3)

$\int_0^x u(x,t)dx = m(t), 0 < x \leq T,$ (1.4)

where g and m are known functions.

As another type of non-local boundary value problem consider

$\frac{\partial u}{\partial t} = \gamma \frac{\partial^2 u}{\partial x^2} + q(x,t), (x,t) \in (0,1) \times (0,T],$ (1.5)

with initial condition (1.2) *and boundary condition*

$$u(1,t) = g(t), 0 < t \leq T, \qquad (1.6)$$

and non-local condition

$$\int_0^{b(x)} u(x,t)dx = m(t), 0 \leq x \leq T, 0 < b(t) < 1. \qquad (1.7)$$

There are many papers that deal with integral conditions giving the specification of mass, e.g. Dehghan [1-3], Cannon and Matheson [4], Cannon and van der Hoek [5]. Dehgan and Tatari [1] solved this problem using the radical basis functions for discretization of space and finite difference methods for discretization of time. Also Caglar considered third degree B-spline functions to solve one-dimentional heat equation [6]. In Ref. [7] a new method is proposed in the reproducing kernel space and the solution is given by the form of series. In the paper by Martin-Vaquero [10], Crandall's formulas is used to solve second-order parabolic equation subject to non-local conditions. The author of [11] applied new finite difference schemes to solve one-dimensional parabolic equation with boundary integral conditions.

A more extensive list of references as well as a survey on progress made on this class of problems may be found in Dehghan [2]
The organization of this paper as follows:In section $2,3$ *and* 4*, we investigate two different non-polynomial cubic spline methods. In section* 5*, we give two different examples of parabolic partial differential equations with non-local conditions. A conclusion is given in Section 6. Finally some references are presented at the end.*

Let the region $R = (0,1) \times (0,T]$*, be discretized by a set of points* $R_{h,k}$ *which are the vertices of a grid points* (x_i, t_i)*, where* $x_i = ih$*,* $i = 0,1,2,...,n$*,*$nh = 1$*, and* $t_j = jk$*,* $j = 0,1,2,3$*. Here* h *and* k *are mesh size in the space and time directions respectively.*
We develop an approximation for eq.(1.1) *in which the time derivative is replaced by a finite difference approximation and the space derivative by the non-polynomial cubic spline function approximation. We need the following finite difference approximation for the time derivative of* u*. Let:*

$$\overline{u}_t(x_i, t_j) = \frac{u(x_i, t_{j+1}) - u(x_i, t_{j-1})}{2k} = u_t(x_i, t_j) + O(k^2) \tag{7}$$

At the grid point (i,j), the proposed differential equation (1.1) *may be discretized by:*

$$\overline{u}_t(x_i,t_j)=\gamma\overline{u}_{xx}(x_i,t_j) \tag{8}$$

By using (7) *in equation* (8) *we obtain*

$$\frac{u(x_i,t_{j+1})-u(x_i,t_{j-1})}{2k\gamma}=M(x_i,t_j) \tag{9}$$

where $M(x_i,t_j)=S''(x_i)$ *is the second spline derivative at* (x_i,t_j)*. From* (9) *we have*

$$M(x_{i+1},t_j)=\frac{u(x_{i+1},t_{j+1})-u(x_{i+1},t_{j-1})}{2k\gamma}, \tag{10}$$

and

$$M(x_{i-1},t_j)=\frac{u(x_{i-1},t_{j+1})-u(x_{i-1},t_{j-1})}{2k\gamma}, \tag{11}$$

substituting (9),(10) *and* (11) *into* (6)*, after simplifying we obtain:*

$$\alpha\left(\frac{u(x_{i+1},t_{j+1})-u(x_{i+1},t_{j-1})}{2k\gamma}\right)+2\beta\left(\frac{u(x_i,t_{j+1})-u(x_i,t_{j-1})}{2k\gamma}\right)+\alpha\left(\frac{u(x_{i-1},t_{j+1})-u(x_{i-1},t_{j-1})}{2k\gamma}\right)$$
$$-\frac{1}{h^2}(u(x_{i+1},t_j)-2u(x_i,t_j)+u(x_{i-1},t_j))=0$$

4. The similiar scheme with using Taylor expansion

For every x_i*,* $i=1(1)(n-1)$*, by using the Taylor expansion in the time direction, we have the following difference schemes*

$$\left.\begin{aligned} u_{xx}(x_i,t_j) &= \frac{u_{xx}(x_i,t_{j+1})+u_{xx}(x_i,t_{j-1})}{2}+O(k^2)\\ u_t(x_i,t_j) &= \frac{u(x_i,t_{j+1})-u(x_i,t_{j-1})}{2k}+O(k^2)\end{aligned}\right\} \tag{12}$$

The given eq. (1.1) *may be discretized as*

$$\tfrac{u(x_i,t_{j+1})-u(x_i,t_{j-1})}{2k} = \gamma \tfrac{u_{xx}(x_i,t_{j+1})+u_{xx}(x_i,t_{j-1})}{2} + O(k^2), \tag{13}$$
$i = 1, 2, ..., n-1,\ j = 1, 2, ...$

So, (6) *equality can be rewritten in a new form as follow:*

$$(1 + \frac{1}{12}\delta_x^2)M(x_i, t_j) = \frac{1}{h^2}\delta_x^2 u(x_i, t_j), i = 1, ..., n-1 \tag{14}$$

where
$\delta_x M(x_i, t_j) = M(x_{i+\frac{1}{2}}, t_j) - M(x_{i-\frac{1}{2}}, t_j),$

$\delta_x^2 M(x_i, t_j) = \delta_x(\delta_x M(x_i, t_j)) = M(x_{i+1}, t_j) - 2M(x_i, t_j) + M(x_{i-1}, t_j),$

For $i = 1, ..., n-1$. *Using* (12) *and* (14), *we have;*

$(1 + \frac{1}{12}\delta_x^2)[u_{xx}(x_i, t_{j+1}) + u_{xx}(x_i, t_{j-1})] = (1 + \frac{1}{12}\delta_x^2)[S_{xx}(x_i, t_{j+1}) + S_{xx}(x_i, t_{j-1}) + O(h^4)] = (1 + \frac{1}{12}\delta_x^2)[S_{xx}(x_i, t_{j+1}) + S_{xx}(x_i, t_{j-1})] + O(h^4)$
$= \frac{1}{h^2}\delta_x^2[S(x_i, t_{j+1}) + S(x_i, t_{j-1})] + O(h^4)$
$= \frac{1}{h^2}\delta_x^2[u(x_i, t_{j+1}) + u(x_i, t_{j-1})] + O(h^4).$ *(15)*

Applying the operator $(1 + \frac{1}{12}\delta_x^2)$ *to two sides of Eq.* (13) *and using Eq.* (15), *we have*

$u(x_{i+1}, t_{j+1})(\frac{1}{24k} - \frac{\gamma}{2h^2}) + u(x_i, t_{j+1})(\frac{1}{2k} - \frac{1}{12k} + \frac{\gamma}{h^2}) + u(x_{i-1}, t_{j+1})(\frac{1}{24k} - \frac{\gamma}{2h^2}) = u(x_{i+1}, t_{j-1})(\frac{1}{24k} + \frac{\gamma}{2h^2}) + u(x_i, t_{j-1})(\frac{1}{2k} - \frac{1}{12k} - \frac{\gamma}{h^2}) + u(x_{i-1}, t_{j-1})(\frac{1}{24k} + \frac{\gamma}{2h^2})$
$i = 1(1)(n-1),\ j = 1, 2, ...$ *(16)*

The proposed scheme (16) *is an implicit three level scheme. To start any computation, it is necessary to know the value of* $u(x,t)$ *at the nodal points of the first time* *level, that is, at* $t = k$. *From Taylor series expansion at* $t = k$ *may be written as*

$$u(x,k) = u(x,0) + ku_t(x,0) + \frac{k^2}{2} u_{tt}(x,0) + O(k^3) \qquad (17)$$

Using the initial values, from (1.1) *we can calculate*

$$u_t(x,0) = \gamma u_{xx}(x,0) \qquad (18)$$

.

Thus using initial value and (18), *from* (17), *we may obtain the numerical solution of* u *at* $t = k$.

We need two more equations. The two end conditions can be derivated as follows

$$u_0 = 0 \qquad (19)$$

$$\frac{h}{3}(u_0 + 4u_1 + 2u_2 + 4u_3 + ... + 4u_{n-1} + u_n) = m(t) \qquad (20)$$

The method is described in matrix form for Eqs.(16),(19),(20):

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & ... & 0 & 0 & 0 \\ (\frac{1}{24k} - \frac{\gamma}{2h^2}) & (\frac{5}{12k} + \frac{\gamma}{h^2}) & (\frac{1}{24k} - \frac{\gamma}{2h^2}) & 0 & ... & 0 & 0 & \\ 0 & (\frac{1}{24k} - \frac{\gamma}{2h^2}) & (\frac{5}{12k} + \frac{\gamma}{h^2}) & (\frac{1}{24k} - \frac{\gamma}{2h^2}) & 0 & ... & 0 & \\ . & . & . & . & ... & . & . & . \\ . & . & . & . & ... & . & . & . \\ . & . & . & . & ... & . & . & . \\ 0 & 0 & 0 & 0 & ... & (\frac{1}{24k} - \frac{\gamma}{2h^2}) & (\frac{5}{12k} + \frac{\gamma}{h^2}) & (\frac{1}{24k} - \frac{\gamma}{2h^2}) \\ 1 & 4 & 2 & 4 & ... & 2 & 4 & 1 \end{bmatrix}$$

$$B=\begin{bmatrix} 0 \\ -f(x_1)(\frac{1}{24k}+\frac{\gamma}{2h^2})-f(x_2)(\frac{5}{12k}-\frac{\gamma}{h^2})-f(x_3)(\frac{1}{24k}+\frac{\gamma}{2h^2}) \\ -f(x_2)(\frac{1}{24k}+\frac{\gamma}{2h^2})-f(x_3)(\frac{5}{12k}-\frac{\gamma}{h^2})-f(x_4)(\frac{1}{24k}+\frac{\gamma}{2h^2}) \\ . \\ . \\ . \\ -f(x_{n-2})(\frac{1}{24k}+\frac{\gamma}{2h^2})-f(x_{n-1})(\frac{5}{12k}-\frac{\gamma}{h^2})-f(x_n)(\frac{1}{24k}+\frac{\gamma}{2h^2}) \\ \frac{3}{h}m(t) \end{bmatrix}, \quad (43$$

Finally we find out the approximation solution as
$U = [u_0, u_1, ..., u_n]'$
$AU = B$

4. Test Examples

In this section, to illustrate our methods we have solved the boundary value problem for the one-dimentional heat equation with nonlocal initial condition. All computations are done by using MATLAB 6.5.

Example 1.

We first consider the following equations

$$\left.\begin{array}{l} f(x) = cos(\frac{\Pi}{2}x), 0 < x < 1, \\ g(t) = exp(-\frac{\Pi^2}{4}t), 0 < t < 1, \\ m(t) = \frac{2}{\Pi}exp(-\frac{\Pi^2}{4}t), 0 < t < 1 \end{array}\right\} \quad (50)$$

with the exact solution of the above problem is
$u(x,t) = exp(-\frac{\Pi^2}{4}t)cos(\frac{\Pi}{2}x)$

Example 3.2.6. *We consider a non-linear system of second-order BVPs of the form [1,2,3,5,6]:*

$$\left.\begin{aligned} u'' + a_1(x)u' + a_2(x)u + a_3(x)v' + a_4(x)v + H_1(x,u,v) = f_1(x), \\ v'' + b_1(x)v' + b_2(x)v + b_3(x)u' + b_4(x)u + H_2(x,u,v) = f_2(x), \end{aligned}\right\} \quad (1)$$

with the following boundary conditions

$$u(0) = u(1) = 0, v(0) = v(1) = 0 \qquad (2)$$

where $0 < x < 1, H_1, H_2$ *are nonlinear functions of* u *and* v*,* $a_i(x), b_i(x), f_1(x)$*, and* $f_2(x)$*, are given functions, and* $a_i(x), b_i(x)$ *are continuous,* $i = 1, 2, 3, 4$.

The existence and approximations of the solutions to non-linear systems of second-order BVPs have investigated by many authors[1-6]. In [1] the sinc-collocation method is presented for solving second-order systems. Their method consists of reducing the solution of Eq.(1) to a set of algebric equations by expanding $u(x)$ *and* $v(x)$ *as sinc functions with unknown coefficients. New method is presented to solve Eq.(1) used in the form of series in the reproducing kernel space in [2]. The variation iteration method is applied for the solution with the assumption that the solutions are unique in [3]. He's homotopy perturbation method (HPM) is proposed for the solution of systems in [5]. A new modification of the homotopy analysis method (HAM) is presented for solving systems of second-order BVPs in [6].*

The section of this paper are organized as follows: In the next section we describe the basic formulation of the spline function required for our subsequent development. In section 3 the method are used to analysis to solution of problem (1) and (2). In section 4 some numerical result, that are illustrated using MATLAB 6.5, are given to clarify the method. Section 5 ends this paper with a brief conclusion. Note that we have computed the numerical results by MATLAB 6.5.

To illustrate the application of the Spline method developed in the previous section we consider the non-linear system of second-order BVP that is given in Eq. (1). At the grid point (x_i, u_i)*, the proposed non-linear system of second-order BVP in Eq. (1) may be discretized by*

$$\left.\begin{aligned} u'' + a_1(x_i)u' + a_2(x_i)u + a_3(x_i)v' + a_4(x_i)v + H_1(x,u,v) = f_1(x_i), \\ v'' + b_1(x_i)v' + b_2(x_i)v + b_3(x_i)u' + b_4(x_i)u + H_2(x,u,v) = f_2(x_i). \end{aligned}\right\} \tag{7}$$

Substituting $M_i = u''$ and $N_i = v''$ in equation system (5):

$$\left.\begin{aligned} M_i + a_1(x_i)u_i' + a_2(x_i)u_i + a_3(x_i)v_i' + a_4(x_i)v_i + H_1(x_i,u_i,v_i) = f_1(x_i), \\ N_i + b_1(x_i)v_i' + b_2(x_i)v_i + b_3(x_i)u_i' + b_4(x_i)u_i + H_2(x_i,u_i,v_i) = f_2(x_i). \end{aligned}\right\} \tag{8}$$

Solving Eq. (8) for M_i and N_i, we get

$$\left.\begin{aligned} M_i = -a_1(x_i)u_i' - a_2(x_i)u_i - a_3(x_i)v_i' - a_4(x_i)v_i - H_1(x_i,u_i,v_i) + f_1(x_i) \\ N_i = -b_1(x_i)v_i' - b_2(x_i)v_i - b_3(x_i)u_i' - b_4(x_i)u_i - H_2(x_i,u_i,v_i) + f_2(x_i) \end{aligned}\right\} \tag{9}$$

The following approximations for the first-order derivative of u and v in Eq. (9) can be used

$$\left.\begin{aligned} & u_i' \cong \tfrac{u_{i+1}-u_{i-1}}{2h}, \\ & u_{i+1}' \cong \tfrac{3u_{i+1}-4u_i+u_{i-1}}{2h}, \\ & u_{i-1}' \cong \tfrac{-u_{i+1}+4u_i-3u_{i-1}}{2h}, \\ & v_i' \cong \tfrac{v_{i+1}-u_{i-1}}{2h}, \\ & v_{i+1}' \cong \tfrac{3v_{i+1}-4v_i+v_{i-1}}{2h}, \\ & v_{i-1}' \cong \tfrac{-v_{i+1}+4v_i-3v_{i-1}}{2h}. \end{aligned}\right\} \tag{10}$$

So Eq. (9) becomes

$$\left.\begin{aligned} M_i = & -a_1(x_i)\tfrac{u_{i+1}-u_{i-1}}{2h} - a_2(x_i)u_i - a_3(x_i)\tfrac{v_{i+1}-v_{i-1}}{2h} \\ & -a_4(x_i)v_i - H_1(x_i,u_i,v_i) + f_1(x_i) \end{aligned}\right\} \tag{11a}$$

$$\left.\begin{aligned} M_{i+1} = & -a_1(x_{i+1})\tfrac{3u_{i+1}-4u_i+u_{i-1}}{2h} - a_2(x_{i+1})u_i - a_3(x_{i+1})\tfrac{3v_{i+1}-4v_i+v_{i-1}}{2h} \\ & -a_4(x_{i+1})v_i - H_1(x_{i+1},u_{i+1},v_{i+1}) + f_1(x_{i+1}) \end{aligned}\right\} \tag{11b}$$

$$\left.\begin{aligned} M_{i-1} = -a_1(x_{i-1})\tfrac{-u_{i+1}+4u_i-3u_{i-1}}{2h} - a_2(x_{i-1})u_i - a_3(x_{i-1})\tfrac{-v_{i+1}+4v_i-3v_{i-1}}{2h} \\ -a_4(x_{i-1})v_i - H_1(x_{i-1},u_{i-1},v_{i-1}) + f_1(x_{i-1}) \end{aligned}\right\} \quad (11c)$$

and

$$\left.\begin{aligned} N_i = -b_1(x_i)\tfrac{v_{i+1}-v_{i-1}}{2h} - b_2(x_i)v_i - b_3(x_i)\tfrac{u_{i+1}-u_{i-1}}{2h} \\ -b_4(x_i)u_i - H_2(x_i,u_i,v_i) + f_2(x_i) \end{aligned}\right\} \quad (12a)$$

$$\left.\begin{aligned} N_{i+1} = -b_1(x_{i+1})\tfrac{3v_i+1-4v_i+v_{i-1}}{2h} - b_2(x_{i+1})v_i - b_3(x_{i+1})\tfrac{3u_i+1-4u_i+u_i-1}{2h} \\ -b_4(x_{i+1})u_i - H_2(x_{i+1},u_{i+1},v_{i+1}) + f_2(x_{i+1}) \end{aligned}\right\} \quad (12b)$$

$$\left.\begin{aligned} N_{i-1} = -b_1(x_{i-1})\tfrac{-v_i+1+4v_i-3v_i-1}{2h} - b_2(x_{i-1})v_i - b_3(x_{i-1})\tfrac{-u_{i+1}+4u_i-3u_{i-1}}{2h} \\ -b_4(x_{i-1})u_i - H_2(x_{i-1},u_{i-1},v_{i-1}) + f_2(x_{i-1}) \end{aligned}\right\} \quad (12c)$$

Substituting Eqs. (11a-11c)-(12a-12c) in Eqs. (6a) and (6b) respectively, we find the following $2(n-1)$ *linear algebraic equations in the* $2(n+1)$ *unknowns for* $i = 0, 1, ..., n$.

$$\left.\begin{aligned} &[\tfrac{\alpha a_1(x_{i-1})}{2h} - \tfrac{2\beta a_1(x_i)}{2h} - \tfrac{3\alpha a_1(x_{i+1})}{2h} - \alpha a_2(x_{i+1}) - \tfrac{1}{h^2}]u_{i+1} \\ &+[\tfrac{-4\alpha a_1(x_{i-1})}{2h} - 2\beta a_2(x_i) + \tfrac{4\alpha a_1(x_{i+1})}{2h} + \tfrac{2}{h^2}]u_i \\ &+[\tfrac{3\alpha a_1(x_{i-1})}{2h} - \alpha a_2(x_{i-1}) + \tfrac{2\beta a_1(x_i)}{2h} - \tfrac{\alpha a_1(x_{i+1})}{2h} - \tfrac{1}{h^2}]u_{i-1} \\ &+[\tfrac{\alpha a_3(x_{i-1})}{2h} - \tfrac{2\beta a_3(x_i)}{2h} - \tfrac{3\alpha a_3(x_{i+1})}{2h} - \alpha a_4(x_{i+1})]v_{i+1} \\ &+[\tfrac{-4\alpha a_3(x_{i-1})}{2h} - 2\beta a_4(x_i) + \tfrac{4\alpha a_3(x_{i+1})}{2h}]v_i \\ &+[\tfrac{3\alpha a_3(x_{i-1})}{2h} - \alpha a_4(x_{i-1}) + \tfrac{2\beta a_3(x_i)}{2h} - \tfrac{\alpha a_3(x_{i+1})}{2h}]v_{i-1} \\ &-\alpha H_1(x_{i-1},u_{i-1},v_{i-1}) - 2\beta H_1(x_i,u_i,v_i) - \alpha H_1(x_{i+1},u_{i+1},v_{i+1}) = \\ &-\alpha f_1(x_{i-1}) - 2\beta f_1(x_i) - \alpha f_1(x_{i+1}) \end{aligned}\right\} \quad (13)$$

and

$$\left.\begin{array}{l}
[\frac{\alpha b_1(x_{i-1})}{2h} - \frac{2\beta b_1(x_i)}{2h} - \frac{3\alpha b_1(x_{i+1})}{2h} - \alpha b_2(x_{i+1}) - \frac{1}{h^2}]v_{i+1} \\
+[\frac{-4\alpha b_1(x_{i-1})}{2h} - 2\beta b_2(x_i) + \frac{4\alpha b_1(x_{i+1})}{2h} + \frac{2}{h^2}]v_i \\
+[\frac{3\alpha b_1(x_{i-1})}{2h} - \alpha b_2(x_{i-1}) + \frac{2\beta b_1(x_i)}{2h} - \frac{\alpha b_1(x_{i+1})}{2h} - \frac{1}{h^2}]v_{i-1} \\
+[\frac{\alpha b_3(x_{i-1})}{2h} - \frac{2\beta b_3(x_i)}{2h} - \frac{3\alpha b_3(x_{i+1})}{2h} - \alpha b_4(x_{i+1})]u_{i+1} \\
+[\frac{-4\alpha b_3(x_{i-1})}{2h} - 2\beta b_4(x_i) + \frac{4\alpha b_3(x_{i+1})}{2h}]u_i \\
+[\frac{3\alpha b_3(x_{i-1})}{2h} - \alpha b_4(x_{i-1}) + \frac{2\beta b_3(x_i)}{2h} - \frac{\alpha b_3(x_{i+1})}{2h}]u_{i-1} \\
-\alpha H_2(x_{i-1}, u_{i-1}, v_{i-1}) - 2\beta H_2(x_i, u_i, v_i) - \alpha H_2(x_{i+1}, u_{i+1}, v_{i+1}) = \\
-\alpha f_2(x_{i-1}) - 2\beta f_2(x_i) - \alpha f_2(x_{i+1})
\end{array}\right\} \quad (14)$$

We need four more equations. The four end conditions can be derivated as follows:

$$\left. u_0 = 0, u_n = 0, v_0 = 0, v_n = 0 \right\} \quad (15)$$

This leads to the system

$$X_{1i} = \frac{\alpha a_1(x_{i-1})}{2h} - \frac{2\beta a_1(x_i)}{2h} - \frac{3\alpha a_1(x_{i+1})}{2h} - \alpha a_2(x_{i+1}) - \frac{1}{h^2} \quad (16)$$

$$Y_{1i} = \frac{-4\alpha a_1(x_{i-1})}{2h} - 2\beta a_2(x_i) + \frac{4\alpha a_1(x_{i+1})}{2h} + \frac{2}{h^2} \quad (17)$$

$$Z_{1i} = \frac{3\alpha a_1(x_{i-1})}{2h} - \alpha a_2(x_{i-1}) + \frac{2\beta a_1(x_i)}{2h} - \frac{\alpha a_1(x_{i+1})}{2h} - \frac{1}{h^2} \quad (18)$$

$$X_{2i} = \frac{\alpha a_3(x_{i-1})}{2h} - \frac{2\beta a_3(x_i)}{2h} - \frac{3\alpha a_3(x_{i+1})}{2h} - \alpha a_4(x_{i+1}) \quad (19)$$

$$Y_{2i} = \frac{-4\alpha a_3(x_{i-1})}{2h} - 2\beta a_4(x_i) + \frac{4\alpha a_3(x_{i+1})}{2h} \quad (20)$$

$$Z_{2i} = \frac{3\alpha a_3(x_{i-1})}{2h} - \alpha a_4(x_{i-1}) + \frac{2\beta a_3(x_i)}{2h} - \frac{\alpha a_3(x_{i+1})}{2h} \quad (21)$$

$$g_i = \frac{a_1(x_i)}{2h}\ , \quad h_i = a_2(x_i)\ , \quad k_i = \frac{a_3(x_i)}{2h}\ , \quad l_i = a_4(x_i) \quad (22)$$

$$X_{1i} = \alpha g_{i-1} - 2\beta g_i - 3\alpha g_{i+1} - \alpha h_{i+1} - \frac{1}{h^2} \quad (23)$$

$$Y_{1i} = -4\alpha g_{i-1} + 4\alpha g_{i+1} - 2\beta h_i + \frac{2}{h^2} \quad (24)$$

$$Z_{1i} = 3\alpha g_{i-1} + 2\beta g_i - \alpha g_{i+1} - \alpha h_{i-1} - \tfrac{1}{h^2} \tag{25}$$

$$X_{2i} = \alpha k_{i-1} - 2\beta k_i - 3\alpha k_{i+1} - \alpha l_{i+1} \tag{26}$$

$$Y_{2i} = -4\alpha k_{i-1} + 4\alpha k_{i+1} - 2\beta l_i \tag{27}$$

$$Z_{2i} = 3\alpha k_{i-1} + 2\beta k_i - \alpha k_{i+1} - \alpha l_{i-1} \tag{28}$$

$$X_{3i} = \tfrac{\alpha b_3(x_{i-1})}{2h} - \tfrac{2\beta b_3(x_i)}{2h} - \tfrac{3\alpha b_3(x_{i+1})}{2h} - \alpha b_4(x_{i+1}) \tag{29}$$

$$Y_{3i} = \tfrac{-4\alpha b_3(x_{i-1})}{2h} - 2\beta b_4(x_i) + \tfrac{4\alpha b_3(x_{i+1})}{2h} \tag{30}$$

$$Z_{3i} = \tfrac{3\alpha b_3(x_{i-1})}{2h} - \alpha b_4(x_{i-1}) + \tfrac{2\beta b_3(x_i)}{2h} - \tfrac{\alpha b_3(x_{i+1})}{2h} \tag{31}$$

$$X_{4i} = \tfrac{\alpha b_1(x_{i-1})}{2h} - \tfrac{2\beta b_1(x_i)}{2h} - \tfrac{3\alpha b_1(x_{i+1})}{2h} - \alpha b_2(x_{i+1}) - \tfrac{1}{h^2} \tag{32}$$

$$Y_{4i} = \tfrac{-4\alpha b_1(x_{i-1})}{2h} - 2\beta b_2(x_i) + \tfrac{4\alpha b_1(x_{i+1})}{2h} + \tfrac{2}{h^2} \tag{33}$$

$$Z_{4i} = \tfrac{3\alpha b_1(x_{i-1})}{2h} - \alpha b_2(x_{i-1}) + \tfrac{2\beta b_1(x_i)}{2h} - \tfrac{\alpha b_1(x_{i+1})}{2h} - \tfrac{1}{h^2} \tag{34}$$

$$m_i = \tfrac{b_1(x_i)}{2h}, \quad p_i = b_2(x_i), \quad r_i = \tfrac{b_3(x_i)}{2h}, \quad s_i = b_4(x_i) \tag{35}$$

$$X_{3i} = \alpha r_{i-1} - 2\beta r_i - 3\alpha r_{i+1} - \alpha s_{i+1} \tag{36}$$

$$Y_{3i} = -4\alpha r_{i-1} + 4\alpha r_{i+1} - 2\beta s_i \tag{37}$$

$$Z_{3i} = 3\alpha r_{i-1} + 2\beta r_i - \alpha r_{i+1} - \alpha s_{i-1} \tag{38}$$

$$X_{4i} = \alpha m_{i-1} - 2\beta m_i - 3\alpha m_{i+1} - \alpha p_{i+1} - \tfrac{1}{h^2} \tag{39}$$

$$Y_{4i} = -4\alpha m_{i-1} + 4\alpha m_{i+1} - 2\beta p_i + \tfrac{2}{h^2} \tag{40}$$

$$Z_{4i} = 3\alpha m_{i-1} + 2\beta m_i - \alpha m_{i+1} - \alpha p_{i-1} - \tfrac{1}{h^2} \tag{41}$$

The method is described in matrix form in the following way for Eqs. (16)-(41):

$$A = \begin{bmatrix} A_1 & | & \mathrm{A}_2 \\ - & - & - \\ \mathrm{A}_3 & | & \mathrm{A}_4 \end{bmatrix}, \tag{42}$$

$$\mathrm{B}= \begin{bmatrix} 0 \\ \text{-}\alpha f_1(x_0) - 2\beta f_1(x_1) - \alpha f_1(x_2) \\ \text{-}\alpha f_1(x_1) - 2\beta f_1(x_2) - \alpha f_1(x_3) \\ . \\ . \\ . \\ \text{-}\alpha f_1(x_{n-2}) - 2\beta f_1(x_{n-1}) - \alpha f_1(x_n) \\ 0 \\ 0 \\ \text{-}\alpha f_2(x_0) - 2\beta f_2(x_1) - \alpha f_2(x_2) \\ \text{-}\alpha f_2(x_1) - 2\beta f_2(x_2) - \alpha f_2(x_3) \\ . \\ . \\ . \\ \text{-}\alpha f_2(x_{n-2}) - 2\beta f_2(x_{n-1}) - \alpha f_2(x_n) \\ 0 \end{bmatrix}, \tag{43}$$

$$H = \begin{bmatrix} 0 \\ \text{-}\alpha H_1(x_0,u_0,v_0) - 2\beta H_1(x_1,u_1,v_1) - \alpha H_1(x_2,u_2,v_2) \\ \text{-}\alpha H_1(x_1,u_1,v_1) - 2\beta H_1(x_2,u_2,v_2) - \alpha H_1(x_3,u_3,v_3) \\ . \\ . \\ . \\ \text{-}\alpha H_1(x_{n-2},u_{n-2},v_{n-2}) - 2\beta H_1(x_{n-1},u_{n-1},v_{n-1}) - \alpha H_1(x_n,u_n,v_n) \\ 0 \\ 0 \\ \text{-}\alpha H_2(x_0,u_0,v_0) - 2\beta H_2(x_1,u_1,v_1) - \alpha H_2(x_2,u_2,v_2) \\ \text{-}\alpha H_2(x_1,u_1,v_1) - 2\beta H_2(x_2,u_2,v_2) - \alpha H_2(x_3,u_3,v_3) \\ . \\ . \\ . \\ \text{-}\alpha H_2(x_{n-2},u_{n-2},v_{n-2}) - 2\beta H_2(x_{n-1},u_{n-1},v_{n-1}) - \alpha H_2(x_n,u_n,v_n) \\ 0 \end{bmatrix},$$

$$U = [u_0, u_1, ..., u_n, v_0, v_1, ..., v_n]'. \tag{44}$$

Here the four submatrices A_1, A_2, A_3 and A_4 are defined as

$$A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & ... & 0 & 0 \\ X_{11} & Y_{11} & Z_{11} & 0 & ... & 0 & 0 \\ 0 & X_{12} & Y_{12} & Z_{12} & 0 & ... & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & ... & 0 & 0 & X_{1(n-2)} & Y_{1(n-2)} & Z_{1(n-2)} \\ . & . & . & . & 0 & 0 & 1 \end{bmatrix}, \tag{45}$$

$$A_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & ... & 0 & 0 \\ X_{21} & Y_{21} & Z_{21} & 0 & ... & 0 & 0 \\ 0 & X_{22} & Y_{22} & Z_{22} & 0 & ... & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & ... & 0 & 0 & X_{2(n-2)} & Y_{2(n-2)} & Z_{2(n-2)} \\ . & . & . & . & 0 & 0 & 0 \end{bmatrix}, \tag{46}$$

$$A_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & ... & 0 & 0 \\ X_{31} & Y_{31} & Z_{31} & 0 & ... & 0 & 0 \\ 0 & X_{32} & Y_{32} & Z_{32} & 0 & ... & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & ... & 0 & 0 & X_{3(n-2)} & Y_{3(n-2)} & Z_{3(n-2)} \\ . & . & . & . & 0 & 0 & 0 \end{bmatrix}, \tag{47}$$

$$A_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & ... & 0 & 0 \\ X_{41} & Y_{41} & Z_{41} & 0 & ... & 0 & 0 \\ 0 & X_{42} & Y_{42} & Z_{42} & 0 & ... & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & ... & 0 & 0 & X_{4(n-2)} & Y_{4(n-2)} & Z_{4(n-2)} \\ . & . & . & . & 0 & 0 & 1 \end{bmatrix}. \tag{48}$$

Finally the approximate solution is obtained by solving the nonlinear system using Levenberg-Marquardt optimization method [7] and Matlab 6.5.

$$AU + H = B. \tag{49}$$

4. Numerical examples

In this section, to illustrate our methods we have solved two non-linear system of second-order BVP . All computations are done by using MATLAB 6.5.

Example 1.

Consider the following equations

$$\left.\begin{array}{l} u''(x) - xv'(x) + u(x) = f_1(x) \\ v''(x) + xu'(x) + u(x)v(x) = f_2(x) \end{array}\right\} \tag{50}$$

subject to the boundary conditions

$$u(0) = u(1) = 0, v(0) = v(1) = 0, \tag{51}$$

where $0 < x < 1, f_1(x) = x^3 - 2x^2 + 6x$ and $f_2(x) = x^2 - x$.

The exact solutions of $u(x)$ and $v(x)$ are given as $x^3 - x$ and $x^2 - x$ respectively. The observed maximum absolute errors of $u(x)$ and $v(x)$ for $n = 21$ (nodal points) are given in Table 1. The numerical results of $u(x)$ and $v(x)$ are also illustrated in Figures 1 and 2.

3.3 B-splines

B-spline method is so named because of use of splines as basis function. In this section, we will focus on B-spline basis and their definitions. Secondly, we will look the results of this method on diffusion equation and compare them with other methods. Let $\Omega_n beapartitionof$[a,b]$\subset \Re$. A B-spline of order k is a spline from S$_k$(Ω_n)$withminimalsupportandthepartitionofunityholding.Toexplainthis, letusdefinedB_{i,j}(x)u$
$\in Z$ is a B-spline of degree k, the left end of which support is equal to x$_i$, $andthenwehavethefollow$

,1. Supp($B_{k,i}$) $= [x_i, x_{i+k+1}]$
2.$B_{k,i}(x) \geq 0, \forall x \Re$
3. $\sum_{i=-\infty}^{\infty} B_{k,i}(x) = 1, \forall x \Re$
A detailed description of B-spline functions generated by subdivision can be found in [13]. Consider equally-spaced knots of a partition $\pi : a = x_0 < x_1 < ... < x_n = b$ on [a,b]. Let $S_3[\pi]$ be the space of continuously-differentiable, piecewise, third-degree polynomials on π. That is, $S_3[\pi]$ is the space of third-degree splines on π. Consider the B-splines basis in $S_3[\pi]$. The third-degree B-splines are defined as

$$B_0(x) = \frac{1}{6h^3} \begin{cases} x^3 & 0 \leq x < h \\ -3x^3 + 12hx^2 - 12h^2x + 4h^3 & h \leq x < 2h \\ 3x^3 - 24hx^2 + 60h^2x - 44h^3 & 2h \leq x < 3h \\ -x^3 + 12hx^2 - 48h^2x + 64h^3 & 3h \leq x < 4h \end{cases} \quad (5)$$

$$B_{i-1}(x) = B_0(x - (i-1)h),\ i = 2, 3, ...,$$

To solve singularly perturbed convection-dominated diffusion equation, B_i ,
B_i' *and* B_i'' evaluated at the nodal points are needed. Their coefficients are summarized in Table 1.

Table 3.1: Values of B_i, B_i' *and* B_i''

	$\mathbf{x}_i$	$\mathbf{x}_{i+1}$	$\mathbf{x}_{i+2}$	$\mathbf{x}_{i+3}$	$\mathbf{x}_{i+4}$
B_i	0	1/6	4/6	1/6	0
B_i'	0	-3/6h	0/6h	3/6h	0
B_i''	0	6/6h^2	-12/6h^2	6/6h^2	0

Chapter 4

Finite Element Method

4.1 Introduction

In this chapter we construct so-called finite element approximations to solutions to partial differential equations. The term "finite element method" has come to be associated with using piecewise polynomials in one, two, and three dimensions together with so-called Rayleigh-Ritz metod and its more general counterpart, the Galerkin method, to approximate solutions to operator equations.In this chapter we concentrate on Galerkin method with splines.

4.2 Galerkin Method

Galerkin was born in 1871 in Russia. He bagan doing research in engineering while he was in prison in $1906 - 19007$ for his participation in the anti-tsarist revolutionary movement. His method was introduced in a paper on elasticity published in 1925. Galerkin's method for solving a general differential equation is based on seeking an approximation solution, which is
1. Easy to differentiate and integrate
2. Spanned by a set of nearly orthogonal basis functions in a finite-dimensional space.

Example 4.2.1. *Let u(t) be the solution to the ordinary differential equation given buy $u'(t) - \lambda u(t) = 0$, and let U(t) be the approximation solution spanned by the basis functions* $1, , t, t^2$. *Thus*

$U(t) = A \cdot 1 + B \cdot t + C \cdot t^2 and U'(t) = B + 2C \cdot t.$
Inserting U(t) and U'(t) in the differential equation, we get
$B + 2C \cdot t - \lambda(A \cdot 1 + B \cdot t + C \cdot t^2) = 0$, *and thus*
$-\lambda C t^2 + (2C - \lambda B)t + B - \lambda A = 0.$

This is a simple algebraic equation, however we need three different equations to calculate A, B, C.
The Galerkin method using the Galerkin orthogonality property of the approximate solution U(t) avoids this complexity.

Definition 4.2.2. *A usual scalar product for two real valued functions u(x) and v(x) is defined by*
$< u, v >= \int_0^T u(x)v(x)dx,$

Definition 4.2.3. *u(x) and v(x) are orthogonal if ¡u,v¿ = 0.*

Definition 4.2.4. *A norm associated with this scalar product is defined by*
$\|u\| = \sqrt{< u, u >} = (\int_0^T |u(x)|^2 dx)^{\frac{1}{2}}$

Let U(t) be an approximation of the real solution $u'(t) = u(t)$ (1) of a differential equation then
$u'(t) - \cdot u(t) = 0$ and
$U'(t) - \cdot U(t) \neq 0$

Definition 4.2.5. *If U(t) is an approximation of u(t), then*

$R(U(t)) = U'(t) - \lambda \cdot U(t)$
is called the residual error of U(t).

Let v^n be an orthogonal basis function, multiply the equation (1) by v(t) and integrate
$\int_0^T U'(t)v(t) = \lambda \cdot \int_0^T U(t) \cdot v(t)dt$, then
$\int_0^T (U'(t)v(t) - \lambda U(t) \cdot v(t))dt = 0$
So, now the problem is to find an approximate solution U(t) in the subspace.

Chapter 5

Conclusion

5.1 compare

References

H. Radjvi & P. Rosenthal, *Simultaneous Triangularization*, Springer Verlag , Berlin, 2000.

G. C. Rota and W. G. Strang, A note on the joint spectral radius, *Indag. Math.*, 1960.

F. F. Bonsall & J. Duncan, *Numerical Ranges of Operators on Normed Algebras I*, Londan Math. Soc. Lect. Note, Ser. 2, Cambridge Univ. Press, Londan, 1971.

V. S. Shulman, On invariant subspace of Volterra Operators, *Funk. Anal. i Prilozen*, Russian, 1984.

N. Aronszajn & K. T. Smith, Invariant subspaces of completely continuous operators, *Ann. of Math* , 1954.

V. Lomonosov, Invariant subspaces for the family of operators commuting with compact operators, *Functional. Anal. i. Prilozen*, 1973.

Y. V. Turovskii, Volterra Semigroups have invariant subspaces, *J.Funct.Anal.*, 1999.

H. Radjvi & P. Rosenthal, *Invariant Subspaces*, Springer Verlag, New York,1973.

Curriculum Vitae(CV)

2001 METU.