COLLEGE OF NATURAL AND COMPUTATIONAL SCIENCES

DEPARTEMENT OF MATHEMATICS

Numerics in one-dimensional wave propagation

Submitted in partial fulfilment of the requirement for the Degree of master of science in Mathematics

By: **Fetene Molla**

Advisor: Dr. Tadesse Abdi

June, 2019

Addis Ababa Ethiopia

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to God for giving me patience. I am also grateful to my advisor Dr. Tadesse Abdi for the continuous support and giving motivation and immense knowledge in doing my project. Secondly I would like to extend my thanks to my families and all who encouraged me to complete my project and their heartful help while writing the paper. Lastly, I would like to thanks to Department of mathematics, Addis Ababa University, for giving the necessary materials throughout the preparation of my project.

Contents

Abstract

This project provides a practical overview of numerical solutions to the wave equation of finite string using the finite difference method. The second order central differences for time and space is applied to a problems involving the one-dimensional wave equation of finite string which lead to an explicit numerical scheme. It also allows the reader to show with the consistency, stability and convergence of explicit finite difference scheme for finite string and an example with working Matlab code for the scheme is presented.

0.1 Notations

$\mathbb{R}$ - The set of all real numbers.

$\mathcal{O}(h^n)$ - Discretization error or Truncation error.

Δx - The local distance between adjacent points in space.

Δt - The local distance between adjacent time steps.

$u^{(n)}$ - The n^{th} derivative of u

c - Velocity

0.2 Introduction

The finite difference approximations for derivatives are one of the common and of the oldest methods to solve differential equations numerically. The advent of finite difference techniques in numerical applications began in the early 1950's and their development was stimulated by the emergence of computers that offered a convenient framework for dealing with complex problems of science and technology.
The finite difference method consists of replacing each derivative by a difference quotient in the classic formulation. In a sense, a finite difference method formulation offers a more direct approach to the numerical solution of partial differential equations.
Wave equation, which is a partial differential equation, is a very important equation in Applied mathematics. It has analytical solution but it is time consuming. It has analytical solution but it is time consuming. Therefore one needs to use numerical methods for solving this equation. For this we investigate finite difference method for one dimensional wave equation. We implement the numerical scheme by computer programming for initial-boundary value problem and verify the qualitative behavior of the numerical solution of the wave equation.

Chapter 1

Preliminaries

In this section we will consider basic definitions, methodologies and practices that are important in this project.

1.1 Definition and example of Wave equation

The wave equation is a Hyperbolic partial differential equation. It typically concerns a time variable t with a spatial variable x, and a scalar function $u = u(x,t)$ whose values could model, for example, the mechanical displacement of a wave. The wave equation for u is

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2} \tag{1.1}$$

where, c is a constant. Solutions of this equation describe propagation of disturbances out from the region at a fixed speed in one or in all spatial directions, as do physical waves from plane or localized sources, the constant c is identified with the propagation speed of the wave. A unique solution for the above equation is usually obtained by setting a problem with further conditions such as initial and boundary conditions.

1.2 Finite difference method

The finite difference method is one of the premier mathematical tool to solve partial differential equations. It is a means of obtaining numerical solutions to partial differential equations.

The application of finite difference method to a particular differential equation problem includes the following steps:

1. Construction of a discrete finite-difference model of the problem:
   - Create a coverage of the computational domain by a space-time grid.
   - Approximations to derivatives, functions, initial and/or boundary condition all at the grid point.
   - Construction of a system of the finite-difference (i.e., algebraic) equations.
2. Analysis of the finite-difference model:
   - Consistency and order of the approximation
   - Stability
   - Convergence
3. Numerical computation using Matlab.

1.2.1 Grid(Mesh) point

One way to numerically solve Wave equation is to approximate all the derivatives by finite differences. These methods are derived from the truncated Taylor's series where a given PDE and boundary and initial conditions are replaced by set of algebraic equations that are then solved by varies well known numerical techniques. We partition the domain in space using a mesh $x_0, ..., x_L$ and in time using a mesh $t_0, ..., t_M$. The discretization of the given differential equation is obtained by dividing the given domain into a finite number of elements. The points at which those finite elements with functional values are called **nodes**. We assume a uniform partition both in space and in time, so the difference between two consecutive space points will be h and between two consecutive time points will be k. The domain is partitioned

in space and in time and approximations of the solution are computed at the space or time points. In addition, there are some practically useful schemes that can fail to yield a solution for bad combinations of Δx and Δt. We will compute the solution of the wave equation problem for $0 \leq x \leq L$, $0 \leq t \leq T$ using a uniform grid. The usual formulas are used, namely,
$t_j = t_0 + jk$, for $j = 0, 1, 2, ..., M$,
$x_i = x_0 + ih$, for $i = 0, 1, 2, ..., N$, where $k = T/M$ and $h = L/N$.
Here, Δx is usually called grid spacing and Δt is called time step since t usually represents time. At the grid points a function $u(x, t)$ is to be approximated by a grid function $u(x_i, t_j)$. A value of $u(x_i, t_j)$ can be denoted by u_{ij}.

1.2.2 Truncation error

Accuracy of a finite difference formula is a fundamental issue when discretizing differential equations. Truncation (or discretization) Error is caused when approximations are used to estimate some quantity. The error between the numerical solution and the exact solution is determined by the error that is commited by going from a differential operator to a difference operator.
As $(\Delta x\ ,\ \Delta t) \longrightarrow (0, 0)$ the numerical solution obtained with any useful scheme will approach the exact solution to the original differential equation. However, the rate at which the numerical solution approaches the exact solution varies with the scheme.

1.2.3 Derivation of finite difference method approximation

The principle of finite difference methods is close to the numerical schemes used to solve ordinary and partial differential equations. It consists in approximating the differential operator by replacing the derivatives in the equation using differential quotients. The domain is partitioned in space and in time and approximations of the solution are computed at the space or time points. Now, using the derivative of a function $u(x, t)$ at h:

$$u'(x) = \lim_{h \to 0} \frac{u(x + h, t) - u(x, t)}{h}$$

If h is sufficiently small then the derivative of $u(x)$ becomes:

$$u'(x) \approx \frac{u(x + h, t) - u(x, t)}{h} \tag{1.2}$$

Now consider the **Taylor series**, a way to approximate the value of a function by taking the sum of its derivatives at a given point, expansion of a function around a point $x = x_0$.
The Taylor series expansion of a function $u(x)$ at $x = x_0$ is given by the formula:

$$u(x_0) = \sum_{n=0}^{\infty} \frac{u^{(n)}(x_0)}{n!}(x - x_0)^n \tag{1.3}$$

where, $u^{(n)}(x_0) = \frac{d^n u}{dx^n}$ at $x = x_0$ and $u^0(x_0) = u(x_0)$.
If we let $x = x_0 + h$, then $x - x_0 = h$, and the series can be written as follows:

$$\begin{aligned} u(x_0 + h) &= \sum_{n=0}^{\infty} \frac{u^{(n)}(x_0)}{n!} h^n \\ &= u(x_0) + \frac{u'(x_0)}{1!} h + \frac{u''(x_0)}{2!} h^2 + \mathcal{O}(h^3) \end{aligned}$$

where, $\mathcal{O}(h^3)$ is the remaining terms of the series with leading term of order the error $\mathcal{O}(h^3)$ incurred in neglecting this part of the series expansion when calculating $u(x_0 + h)$.

Because h is a small quantity, we can write $0 < h < 1$, and $h > h^2 > h^3 > h^4 > ....$ Therefore, the remaining of the series represented by $O(h^3)$ provides the order of the error.

Now, we will replace the first order $\frac{du}{dx}$ at $x = x_0$, with the expression that $u'(x_0) \approx \frac{u(x_0+h)-u(x_0)}{h}$, $u''(x) = \frac{d^2u}{dx^2}$, *etc* selecting an appropriate value for h, and indicating that the error introduced in the calculation is of order $\mathcal{O}(h^n)$.

Now, from the above equation and using Taylor series expansion for first order derivatives for $h > 0$:

$$\begin{aligned} u(x_0 + h) &= u(x_0) + \frac{u'(x_0)}{1!} h + \mathcal{O}(h^2) = u(x_0) + u'(x_0)h + \mathcal{O}(h^2) \\ u(x_0 + h) &= u(x_0) + u'(x_0)h + \frac{u''(x_0)}{2!} h^2 + \mathcal{O}(h^3) \end{aligned}$$

and for $h < 0$, the Taylor series expansion becomes:

$$\begin{aligned} u(x_0 - h) &= u(x_0) - u'(x_0)h + \mathcal{O}(h^2) \\ u(x_0 - h) &= u(x_0) - u'(x_0)h + \frac{u''(x_0)}{2!} h^2 - \mathcal{O}(h^3) \end{aligned}$$

Now, rearranging the above equations for solving first order and second order derivative with Taylor series for $h > 0$ gives:

$$u'(x_0) = \frac{u(x_0 + h) - u(x_0)}{h} - \mathcal{O}(h)$$

and for $h < 0$ we get:

$$u'(x_0) = \frac{u(x_0) - u(x_0 - h)}{h} - \mathcal{O}(h)$$

For small space size h, $(h > 0)$, then $\mathcal{O}(h) \to 0$, then the above equations become:

$$u'(x_0) \approx \frac{u(x_0 + h) - u(x_0)}{h} \tag{1.4}$$

Equation (1.4) is called a first order forward finite difference approximation to $u'(x_0)$. This approximation is called a forward finite difference approximation since we start at x_0 and step forwards to the point $x_0 + h$, h is called the step size small and

$$u'(x_0) \approx \frac{u(x_0) - u(x_0 - h)}{h} \tag{1.5}$$

Equation (1.5) is called the backward difference formula because it involves the values of u at x_0 and $x_0 - h$. The order of magnitude of the truncation error for the backward difference approximation is the same as that of the forward difference approximation.

1.2.4 First order central difference method

Now, in order to obtain first order central difference method adding equation (1.4) and equation (1.5) and we get:

$$u(x_0 + h) - u(x_0 - h) = 2hu'(x_0) + \mathcal{O}(h^2) \tag{1.6}$$

To get good approximations to the continuous problem small h is chosen. When $0 < h \leq 1$; the truncation error for the first order central difference method goes to zero and solving for $u'(x_0)$ gives:

$$u'(x_0) \approx \frac{u(x_0 + h) - u(x_0 - h)}{2h} \tag{1.7}$$

This is the central difference approximation to $u'(x_0)$.

1.2.5 Second order central difference method

Approximation of the second derivative $u''(x_0)$ can be obtained in a similar manner to the first order method. Finite difference approximations to higher order derivatives can be obtained in the same way and only with the additional manipulations of the Taylor Series expansion about $u(x_0)$.

$$u(x_0 + h) = u(x_0) + u'(x_0)h + \frac{u''(x_0)}{2!}h^2 + \mathcal{O}(h^3) \tag{1.8}$$

$$u(x_0 - h) = u(x_0) - u'(x_0)h + \frac{u''(x_0)}{2!}h^2 - \mathcal{O}(h^3) \tag{1.9}$$

Now, adding the two equations, (1.8) and (1.9), and solving for small h then $u''(x_0)$ becomes:

$$u''(x_0) \approx \frac{u(x_0 + h) - 2u(x_0) + u(x_0 - h)}{h^2} \tag{1.10}$$

This is called the central difference method to the second derivative of u at x_0 and. In general, when $0 < h \ll 1$, the above methods we discussed can be summarized simply as follows.

methods	formula
First Order Forward Difference	$u'(x_0) \approx \frac{u(x_0+h)-u(x_0)}{h}$
First Order Backward Difference	$u'(x_0) \approx \frac{u(x_0)-u(x_0-h)}{h}$
First Order Central Difference	$u'(x_0) \approx \frac{u(x_0+h)-u(x_0-h)}{2h}$
Second Order Central Difference	$u''(x_0) \approx \frac{u(x_0+h)-2u(x_0)+u(x_0-h)}{h^2}$

Using the second order central difference method with time and space derivatives we derived a difference equation equivalent to the wave equation up to an error of order $\mathcal{O}(h^3)$, which lead to an explicit numerical scheme.

1.2.6 Properties of the Finite Difference Equation

The most important properties of the finite difference equation are consistency, stability and convergence. These notions cover different aspects of the relation between the partial differential equation and finite difference equation, and the exact and numerical solutions of the partial differential equation.

- **Consistency**: A finite difference method of a partial differential equation is consistent if the difference between the exact solution of partial differential equation and the approximated solution using finite difference method vanishes as the space and time step size approach zero. Consistency deals with how well the solution of finite difference solution approximates to the partial differential equation and it is the necessary condition for convergence.
- **Stability**: For a stable numerical scheme, the errors will not grow unboundedly with time.
- **Convergence**: A finite difference method of a partial differential equation is convergence when the solution of a finite difference equation approaches to the exact solution of the partial differential equation as both grid interval and time step sizes are reduced. The necessary and sufficient conditions for convergence are consistency and stability.

Chapter 2

Explicit Finite difference method

Our aim is to determine the vertical displacement $u(x,t)$ of a point x at time t. We assume that the horizontal displacement is so small relative to the vertical displacement as to be negligible and the maximum displacement of each point on the string is small in comparison with the length L of the string. Both the time and space derivatives are replaced by finite differences.
We assume that the region $\mathbf{R} = \{(x,t) \mid 0 \leq x \leq L, 0 \leq t \leq T\}$ to be subdivided into rectangles. To derive a difference equation for the solution we start by replacing the space derivative by the second order center difference formula.
Consider the following initial-boundary value problem for the Wave equation:

$$\mathbf{u_{tt} = c^2 u_{xx}} \tag{2.1}$$

$$B.C : u(0,t) = 0, u(L,t) = 0$$

$$I.C : u(x,0) = f(x), u_t(x,0) = g(x).$$

Step 1:- Define a discretization in space(h) and time(t) with uniform grid. The usual formula we used are namely,

$$t_j = t_0 + jk$$

$$x_i = x_0 + ih$$

where, $k = T/M$ and $h = L/(N+1)$.
Ste 2 - Discretize the PDE. Use a second order central difference scheme for both space and time derivatives. In preparation for introducing finite difference approximations we evaluate the differential equation at the grid point $(x,t) = (x_i, t_j)$ to obtain $u_{tt}(x_i,t_j) = c^2 u_{xx}(x_i,t_j)$.
Now, using second order central difference to approximate the derivatives and gives us:

$$\frac{u(x_i,t_{j+1}) - 2u(x_i,t_j) + u(x_i,t_{j-1})}{k^2} - \mathcal{O}(k^2) = c^2 \frac{u(x_{i+1},t_j) - 2u(x_i,t_j) + u(x_{i-1},t_j)}{h^2} - \mathcal{O}(h^2)$$

Now, rearranging the equation and can be rewritten as:

$$u(x_i,t_{j+1}) = \lambda^2 u(x_{i+1},t_j) + 2(1-\lambda^2)u(x_i,t_j) + \lambda^2 u(x_{i-1},t_j) - u(x_i,t_{j-1}) + k^2\tau_{ij},$$

where, $\tau_{ij} = \mathcal{O}(k^2) - \mathcal{O}(h^2)$ is the truncation error and $\lambda = \frac{ck}{h}$.
Dropping the truncation error for $\mathcal{O}(h^2) \to 0$ and $\mathcal{O}(k^2) \to 0$ gives us the following finite difference approximation to the wave equation:

$$u_{i,j+1} = \lambda^2 u_{i+1,j} + 2(1-\lambda^2)u_{ij} + \lambda^2 u_{i-1,j} - u_{i,j-1}$$

and rearranging terms led to the iterative system:

$$u_{i,j+1} = \lambda^2(u_{i+1,j} + u_{i-1,j}) + 2(1-\lambda^2)u_{ij} - u_{i,j-1} \tag{2.2}$$

which is called the explicit finite difference scheme and it can be expressed in the form of tridiagonal $N \times N$ matrix form $u^{i+1} = Ku^i - u^{i-1} + b$.
where

$$\text{K}=\begin{bmatrix} 2(1-\lambda^2) & \lambda^2 & \cdots & \cdots \\ \lambda^2 & 2(1-\lambda^2) & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \cdots & 2(1-\lambda^2) \end{bmatrix}$$

$$u^i = \begin{pmatrix} u_{i,1} \\ u_{i,2} \\ \vdots \\ u_{i,n} \end{pmatrix} \text{ and } b = \begin{pmatrix} \lambda^2 u_{i,0} \\ 0 \\ \vdots \\ \lambda^2 u_{i,n} \end{pmatrix}$$

This is the finite difference approximation to the wave equation. The value at the $(i+1)^{th}$ time step is computed from the values at the i^{th} and the $(i-1)^{th}$ time steps.

Now, from discretizing the Boundary and Initial Conditions we get:
$u_{0,j} = u_{N+1,j} = 0$, and the first initial condition translates into $u_{i,0} = f_i$ and this can be determined using the second initial condition. It is important to preserve the quadratic truncation error, and possibilities include introducing a ghost point(or boundary point) or to use a higher-order one-sided difference. However, for this particular problem there is a more direct approach using Taylor's theorem. Keeping in mind that $u_{tt} = c^2 u_{xx}$ then, for small k,

$$\begin{aligned} u(x_i, t_1) &= u(x_i, k) \\ &= u(x_i, 0) + k u_t(x_i, 0) + \frac{1}{2} k^2 u_{tt}(x_i, 0) + \mathcal{O}(k^3) \\ &= u(x_i, 0) + k u_t(x_i, 0) + \frac{1}{2} k^2 c^2 u_{xx}(x_i, 0) + \mathcal{O}(k^3) \\ &= f(x_i) + k g(x_i) + \frac{1}{2} k^2 c^2 f''(x_i) + \mathcal{O}(k^3) \end{aligned}$$

With this, we have that,

$$u_{i,1} = f_i + k g_i + \frac{\lambda^2}{2}(f_{i+1} - 2f_i + f_{i-1}) \tag{2.3}$$

for $i = 1, ..., L$.

This is the derivation of the finite difference approximation for Wave equation.

Example 1. *Solve $u_{tt} = 4u_{xx}$, with boundary conditions $u(0,t) = 0$, $u(4,t) = 0$, $t > 0$ and the initial conditions.*
$u(x, 0) = x(4 - x)$ *and* $u_t(x, 0) = 0$,
solution 1. *We have $c^2 = 4$. Let us assume that we use an explicit method with $h = 1$ and $k = 0.5$. Let the number of time steps up to which the computations are to be performed be 4. Then,we have $\lambda = \frac{ck}{h} = \frac{2(0.5)}{1} = 1$.*
Now, using an explicit formula given in and (2.2) we get:
$u_{i,j+1} = u_{i+1,j} + u_{i-1,j} - u_{i,j-1}$, $i = 1, 2, 3$ *and* $j = 0, 1, 2, 3$.
The boundary conditions give the values $u(0, j) = 0 = u(4, j)$, for all j and the initial conditions give the following values.

$$\begin{aligned} u(x, 0) &= x(4 - x), u_{0,0} = 0, u_{1,0} = 3 \\ u_{2,0} &= u(2, 0) = 4, u_{3,0} = 3, u_{4,0} = 0 \end{aligned}$$

Now, using first order central difference method to $u_t(x,0)=0$ *which gives* $u_{i,-1}=u_{i,1}$. *We have the following results:*
For $j=1$*: calculate* $u_{i,2}=u_{i+1,1}+u_{i-1,1}-u_{i,0}$.

- $i=1: u_{1,2}=u_{2,1}+u_{0,1}-u_{1,0}=3+0-3=0,$
- $i=2: u_{2,2}=u_{3,1}+u_{1,1}-u_{2,0}=2+2-4=0,$
- $i=3: u_{3,2}=u_{4,1}+u_{2,1}-u_{3,0}=0+3-3=0.$

These are the values at the interior points on the time level $t=1.0$.
For $j=2$ *calculate* $u_{i,3}=u_{i+1,2}+u_{i-1,2}-u_{i,1}$.

- $i=1: u_{1,3}=u_{2,2}+u_{0,2}-u_{1,1}=0+0-2=-2,$
- $i=2: u_{2,3}=u_{3,2}+u_{1,2}-u_{2,1}=0+0-3=-3,$
- $i=3: u_{3,3}=u_{4,2}+u_{2,2}-u_{3,1}=0+0-2=-2.$

These are the values at the interior points on the time level $t=1.5$
For $j=3$ *calculate* $u_{i,4}=u_{i+1,3}+u_{i-1,3}-u_{i,2}$.

- $i=1: u_{1,4}=u_{2,3}+u_{0,3}-u_{1,2}=-3+0-0=-3,$
- $i=2: u_{2,4}=u_{3,3}+u_{1,3}-u_{2,2}=-2-2-0=-4,$
- $i=3: u_{3,4}=u_{4,3}+u_{2,3}-u_{3,2}=0-3-0=-3.$

These are the values at the interior points on the required fourth time level $t=2.0$.

2.1 Consistency, Stability, Convergence, Dispersion and Dissipation

When choosing a method for solving a differential equation problem it is necessary to have some knowledge about how to analyze the result if the method with respect to these concepts. The solution method should have certain properties. In most cases, it not possible to analyze the complete solution method. The most important properties are summarized below.

2.1.1 Consistency

A numerical scheme is consistent if the discrete numerical equation tends to the exact solution of a differential equation as the mesh size (represented by Δx and Δt) tends to zero. Consistency is used to indicate the accuracy of the method. Consider the expression of equation (2.3) to follow the consistency condition that is satisfied by $\tau_{ij} \to 0$ as $h \to 0$ and $k \to 0$.
Since finite difference method of a partial differential equation is consistent, if the difference between the solution of partial differential equation and finite difference method vanishes as the space and time step size approach zero. Consistency deals with how well the finite difference equation approximates the partial differential equation and it is the necessary condition for convergence.

2.1.2 Stability

A numerical solution is said to be stable if it does not magnify the errors that appear in the course of numerical solution process. Stability guarantees that the method produces a bounded solution whenever the exact solution of a differential equation is bounded. For iterative methods, a stable method is one that does not diverge.
Solutions of partial differential equations using explicit numerical methods need a means of limiting the timestep so that the solution is stable. A criterion that is usually used to constrain the step size is called the **Courant–Friedrichs–Lewy**(or CFL) condition. The CFL condition for stability requires that the characteristics emanating from a mesh point (x_i, t_j) must, for $0 \leq t \leq t_n$, remain in its numerical domain of dependence on the region: $\mathbf{R} = \{(x,t)\colon x_i - ct_j \leq x \leq x_i + ct_j, 0 \leq t \leq t_j\}$.
Consider the following finite difference approximation to the wave equation equation(2.2),

$$u_{i,j+1} = \lambda^2 u_{i+1,j} + 2(1-\lambda^2)u_{ij} + \lambda^2 u_{i-1,j} - u_{i,j-1}$$

Now, to analyze the stability conditions of wave equation use the analytic solution of wave equation substitute $u_i^j = \omega_j e^{inhr}$ into (2.2) and we get:

$$e^{inrh}\omega_{j+1} = (\lambda^2 e^{inrh} + 2(1-\lambda^2 + \lambda^2 e^{inrh}))e^{inrh}\omega_j - e^{inrh}\omega_{j-1}$$

Cancelling terms and using the double angle formulae we get:

$$\begin{aligned} \omega_{j+1} &= 2(1+\lambda^2(\cos hr - 1)\omega_j - \omega_{j-1} \\ &= 2(1-2\lambda^2\sin^2(\frac{rh}{2}))\omega_j - \omega_{j-1} \end{aligned}$$

Assume that ω_j has the following exponential form $\omega_j = G^j$ then the above equation reduces to the following quadratic equation:
$G^2 - 2\gamma G + 1 = 0$, where, $\gamma = (1 - 2\lambda^2 \sin(\frac{rh}{2}))$
The solutions of this quadratic equation are given by $G = \gamma \pm \sqrt{\gamma^2 - 1}$.
Now, let G_1 and G_2 be the roots of this quadratic we may conclude that,

$$(G - G_1)(G - G_2) = G^2 - (G_1 + G_2)G + G_1G_2 = 0$$

Now, from the above two quadratic solution we see that $G_1G_2 = 1$.
However, for stability of solutions for the form $\omega_j = G^j$, we require that $\mid G_1 \mid \leq 1$ and $\mid G_2 \mid \leq 1$. If the solutions are to be stable, that is $\mid G_1 \mid = \mid G_2 \mid = 1$, which implies that $\mid \gamma \mid \leq 1$.
Thus, $\mid 1 - 2\lambda^2 \sin^2 \frac{rh}{2} \mid \leq 1$. We conclude that the condition for stability of the solution is $\lambda^2 \sin^2 \frac{rh}{2} \leq 1$.
Since the maximum value that $\sin^2 \frac{rh}{2}$ can achieve is $\lambda = (ck/h) \leq 1$, which imposes an upper bound on the time step.

2.1.3 Convergence

Definition 1. *A finite difference scheme approximating a partial differential equation is a convergent scheme if for any solution to the partial differential equation, $u(x,t)$, and solutions to the finite difference scheme, v_i^n , such that v_i^0 converges to $u_0(x)$ as $i\Delta x$ converges to x, then v_i^n converges to $u(x,t)$ as $(i\Delta x, n\Delta t)$ converges to (x,t) as $\Delta x, \Delta t$ converge to 0.*

A numerical method is said to converge if the solution of the discretized equations tends to the exact solution of the differential equation as the grid spacing tends to zero. The solution of the corresponding partial differential equation and that the approximation improves as the grid spacings, h and k, tend to zero.
Remark. *A finite difference method solution u_i converges to the exact solution of a partial differential equation U_i on $0 \leq t \leq T$ in a particular vector norm if $\|U_i - u_i\| \longrightarrow 0$ When $i \longrightarrow \infty$, $\triangle x \longrightarrow 0, \triangle t \longrightarrow 0$ and $n\triangle t \leq T$*

2.1.4 Dispersion and Dissipation property

Plane wave solutions are effective in determining the wave properties of the solution of the wave equation. Consider the form of plane wave solution of the form:

$$u(x,t) = e^{i(\bar{k}x - \bar{\omega}t)} \tag{2.4}$$

where, $i = \sqrt{-1}$ and k is the wave number such that $0 < k < \infty$. Now, substitute this equation in equation (2.1)and gives:

$$u_{tt} = c^2 u_{xx} \Rightarrow \bar{\omega}^2 = c^2 \bar{k}^2$$
$$\Rightarrow \bar{\omega} = \pm c\bar{k}$$

Now, with the above result we have the following properties:

1. An equation is **stable** if $\bar{\omega}_i \leq 0$, $\forall \bar{\omega}_i$; otherwise it is unstable.
2. An equation it propagates with the speed $\upsilon_{ph} = \frac{\bar{\omega}}{k} = \pm c$, which is called **phase velocity**.
3. An equation is **dispersive** if υ_{ph} depends on $\bar{k}$; otherwise, it is non-dispersive.
4. An equation is **dissipative** if $\bar{\omega}_i$ is not identically zero; otherwise it is non-dissipative.
5. For non-dispersion wave equation there is a second derivative velocity that derived from the dispersion relation called υ_g and $\upsilon_g = \frac{d\bar{\omega}}{d\bar{k}}$.
   For the wave equation we found that $\upsilon_{ph} = \pm c$. Because υ_{ph} does not depend on k, it follows that the equation is non-dispersive.
   The plane waves introduced earlier provide an effective tool in determining the wave properties of the solution of the wave equation. Now consider the solution of the form:
   $$u(x,t) = e^{i(\bar{k}x - \bar{\omega}t)}$$
   Substituting this discrete plane wave into (2.3), one finds that
   $e^{-ik\bar{w}} = \lambda^2 e^{ih\bar{k}} + 2(1 - \lambda^2) + \lambda^2 e^{-ih\bar{k}} - e^{ik\bar{w}}$
   Combining the exponentials, and using the identity in which,
   $2\sin^2(\frac{\theta}{2}) = 1 - \cos(\theta)$, yields

$$\sin(\frac{\bar{w}k}{2}) = \pm\lambda \sin(\frac{\bar{k}h}{2}) \tag{2.5}$$

This is the numerical dispersion relation for the explicit method that differs from the actual dispersion relation for the wave equation which described above that $\bar{\omega} = \pm ck$

For a given k there are an infinite number of solutions of above (2.5) for $\bar{\omega}$. We are primarily interested in the case where h is small, and so it is assumed in what follows that $0 \leq kh \leq \pi$. Also, we confine our attention to the case $-\pi \leq \bar{\omega}k \leq \pi$. With this the above equation can be written as:

$$\bar{\omega} = \pm\frac{2}{k}\sin^{-1}[\lambda\sin(\frac{\bar{k}h}{2})] \tag{2.6}$$

This is a numerical plane wave dispersion properties and now we have the following numerical plane wave properties:

- For $\lambda > 1$ in (2.6) then there are values of k for which $\lambda\sin(\frac{\bar{k}h}{2}) > 1$, and in such cases $\bar{\omega}$ is complex-valued, $\bar{\omega}$ is real-valued if $\lambda \leq 1$ and for $\lambda = 1$ then $\bar{\omega} = \pm c\bar{k}$, exact dispersion for the wave equation.
- The numerical method is non-dissipative for $\bar{\omega}$ is real-valued for all k only if $\lambda \leq 1$, which is the stability region the numerical method has the same non-dissipative property as the wave equation.
- For numerical phase velocity, υ_{nph} which can be evaluated as:

$$\begin{aligned}\upsilon_{nph} &= \frac{\bar{w}}{\bar{k}} \\ &= \pm\frac{2}{k\bar{k}}\sin^{-1}(\lambda\sin(\frac{\bar{k}h}{2}))\end{aligned}$$

- Now, using the Taylor series expansion of the above equation for small h and $\bar{k}$ and λ we get:
  $\upsilon_{nph} \approx \pm c(1 - \frac{1}{12}(1-\lambda^2)(\bar{k}h)^2$
  Thu, the numerical method non-dispersive, independent on $\bar{k}$, if and only if $\lambda = 1$ otherwise it is dispersive.

Example 2. *Consider the problem $u_t + au_x = 0$, where a is a positive constant. Show the dispersion, dissipation, phase velocity properties for the given problem.*
In order to check the above properties consider that the problem has the solution of

the form $u(x,t) = e^{i(\bar{k}x - \bar{\omega}t)}$.

Then substituting in the given problem we get:

$u_t + au_x = 0 \Rightarrow (-i\bar{\omega})u(x,t) + a(-i\bar{k})u(x,t) = 0$

$(-i\bar{\omega}) + a(-i\bar{k}) = 0$

$\bar{\omega} = a\bar{k}$

Since $\bar{\omega}$ *is real-valued and non-zero, the given problem is non-dissipative and also,*

$$\begin{aligned} \upsilon_{ph} &= \frac{\bar{\omega}}{\bar{k}} \\ &= \frac{a\bar{k}}{\bar{k}} = a \end{aligned}$$

Since υ_{ph} *doesn't depend on* k *then the problem is non-dispersive.*

Chapter 3

Matlab implementation program

Now, we see Matlab for the explicit finite difference scheme of wave equation to compare the exact solution of the following wave problem:

3.1 Test problem 1

Consider the initial and boundary problem of (1.1):

$$\begin{cases} u_{tt} = c^2 u_{xx}, 0 < x < L, t > 0 \\ u(0,t) = u(L,t) = 0 \\ u(x,0) = f(x), u_t(x,0) = g(x), 0 < x < L. \end{cases}$$

To solve the analytic solution use method of separation of variables. The exact analytical solution of the above wave equation is:

$$u(x,t) = \sum_{n=1}^{\infty} \sin \frac{n\pi}{L} x (a_n \cos \lambda_n t + b_n \sin \lambda_n t) \tag{3.1}$$

where

$$\begin{aligned} a_n &= \frac{2}{L} \int_0^L f(x) \sin \frac{n\pi}{L} x dx \\ b_n &= \frac{2}{cn\pi} \int_0^L g(x) \sin \frac{n\pi}{L} x dx \\ \lambda_n &= c \frac{n\pi}{L}, n = 1, 2, ... \end{aligned}$$

and use the explicit finite difference scheme (2.3) using the following data to simulate the following example.

$c = 1 = L,\ f(x) = \sin\frac{2\pi x}{1}$ and $g(x) = 0$

Using the above data and by (3.1) the exact solution of the wave equation is:

$\Rightarrow u(x,t) = \sin 2\pi x \cos 2\pi t$.

Implementation

The explicit finite difference scheme (2.4) is implemented in the Matlab function wave as follows:

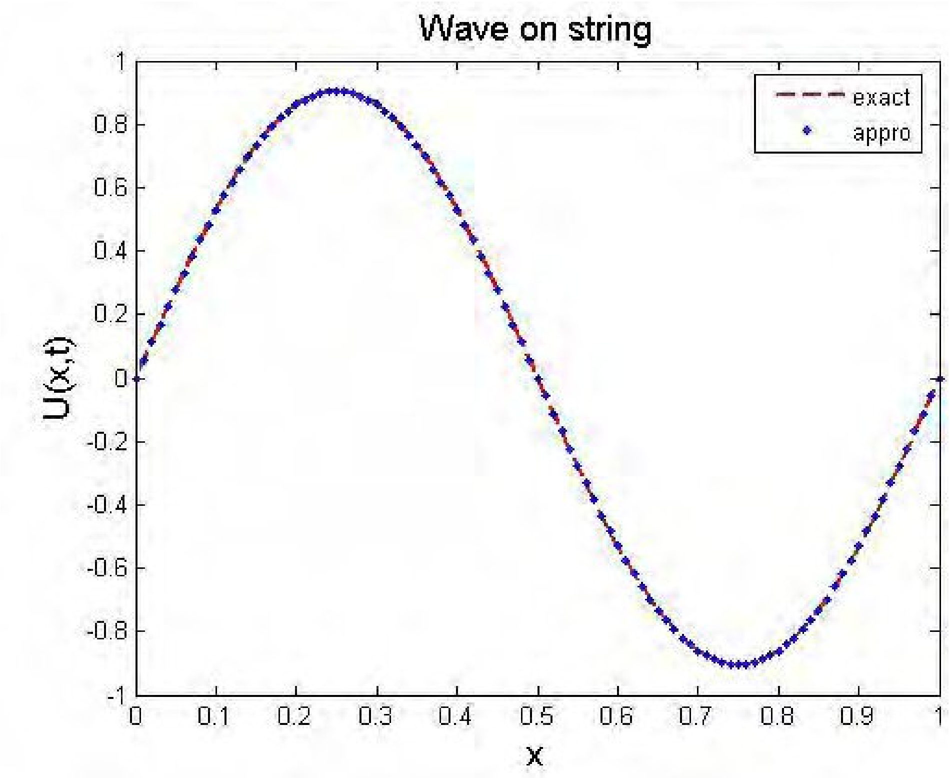
Matlab Code

```
function [x0,t0,u0]=wave(nt,nx,alpha,L,tmax)
%wave solve 1D wave equation with finite difference scheme
%wave (nt)
%wave(nx,nt)
%wave(nx,nt,alpha,L)
%input: nt=number of steps;
%nx=number of mesh(Grid) pointsin x-direction:
%alpha=speed
%L=length of the domain
%tmax = maximum time for the simulation
%output: x=location of the finite difference nodes
%t=value of time at which solution is obtained (time nodes)
%U=matrix of the solution: U(i,j) is U(xi) at t=t(j)
if nargin <1,nt=101;end
if nargin<2,nx=101;end
if nargin <3,alpha=1;end
if nargin <4,L=1;end
if nargin <5,tmax =0.5;end
```

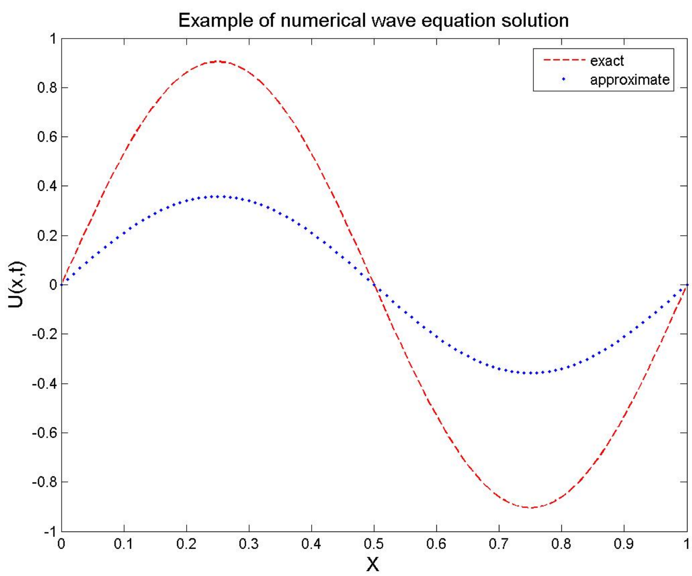
```
% Compute mesh spacing and time step
dx=L/(nx-1);
dt=tmax/(nt-1);
r=alpha^2*dt^2/dx^2;
r2=2-2*r;
%Create arrays to save data for export
x=linspace(0,L,nx)';
t=linspace(0,tmax,nt);
%Set I.C and B.C
U(:,1)=sin(2*pi*x/L);
U(:,2)=sin(2*pi*x/L)-2*(pi/L)^2*dt^2*sin(2*pi*x/L);%implies
    u0=0;uL=0;
u0=0;uL=0;
%Apply B.C inside time step loop
%Loop over time steps
for m=3:nt
      for i=2:nx-1
 U(i,m)=r2*U(i,m-1)+r*U(i+1,m-1)+r*U(i-1,m-1)-U(i,m-2);
      end
end
%Compare the exact solution at end of simulation
ue=sin(2*pi*x/L)*cos(2*alpha*pi*t/L);
plot(x,ue(:,15),'r--',x,U(:,15),'.','LineWidth',1);
title('Example of numerical wave equation solution','FontSize
    ',14);
xlabel('X','FontSize',14);
ylabel('U(x,t)','FontSize',14);
Legend('exact','approximate');
grid
```

Running wave function with the above parameters gives:

Case 1. *For* $\lambda = \frac{ck}{h} < 1$. *Let* $\lambda = \frac{1}{4}$, *then the exact and approximate solutions are shown below*



Case 2. *For* $\lambda = \frac{ck}{h} > 1$*. Let* $\lambda = 4$*, then the exact and approximate solutions are shown below*



We can see from the two Figures that the numerical solution is more stable when $\lambda \ll 1$ and is unstable when $\lambda > 1$. The explicit scheme (2.4) is conditional stable.

3.2 Test problem 2

Consider the wave equation where $c^2 = 4$. The string at rest has length $L = 1$. Assume that the initial condition is:

- $u(x, 0) = f(x) = \sin(\pi x) + \sin(2\pi x)$
- $u_t(x, 0) = g(x) = 0$.

Use the finite difference method to solve the wave equation over the rectangle $\mathbf{R} = \{(x, t) : 0 \leq x \leq 1, 0 \leq t \leq 1\}$. To solve the analytic solution use method of separation of variables. The exact analytical solution of the above wave equation is:

$$u(x,t) = \sum_{n=1}^{\infty}(c_n \cos \lambda_n t + d_n \sin \lambda_n t) \sin \frac{n\pi}{L} x \tag{3.2}$$

where

$$\begin{aligned} c_n &= \frac{2}{L}\int_0^L f(x) \sin \frac{n\pi}{L} x dx \\ d_n &= \frac{2}{cn\pi}\int_0^L g(x) \sin \frac{n\pi}{L} x dx \end{aligned}$$

solution 2. *Compare the solution with the exact solution:* $u(x,t) = \sin(\pi x)\cos(\pi t) + \sin(2\pi x)\cos(4\pi t)$. *Using c=2,h=0.1,k = 0.05. This becomes* $\lambda = \frac{ck}{h} = 1$

```
a = 1.0;
b = 1.0;
c = 2.0;
n = 11;
m = 21;
F[x_] = Sin[π x] + Sin[2 π x];
G[x_] = 0.0;
h = a/(n - 1);
k = b/(m - 1);
f[i_] = F[h (i - 1)];
g[j_] = G[k (j - 1)];
```

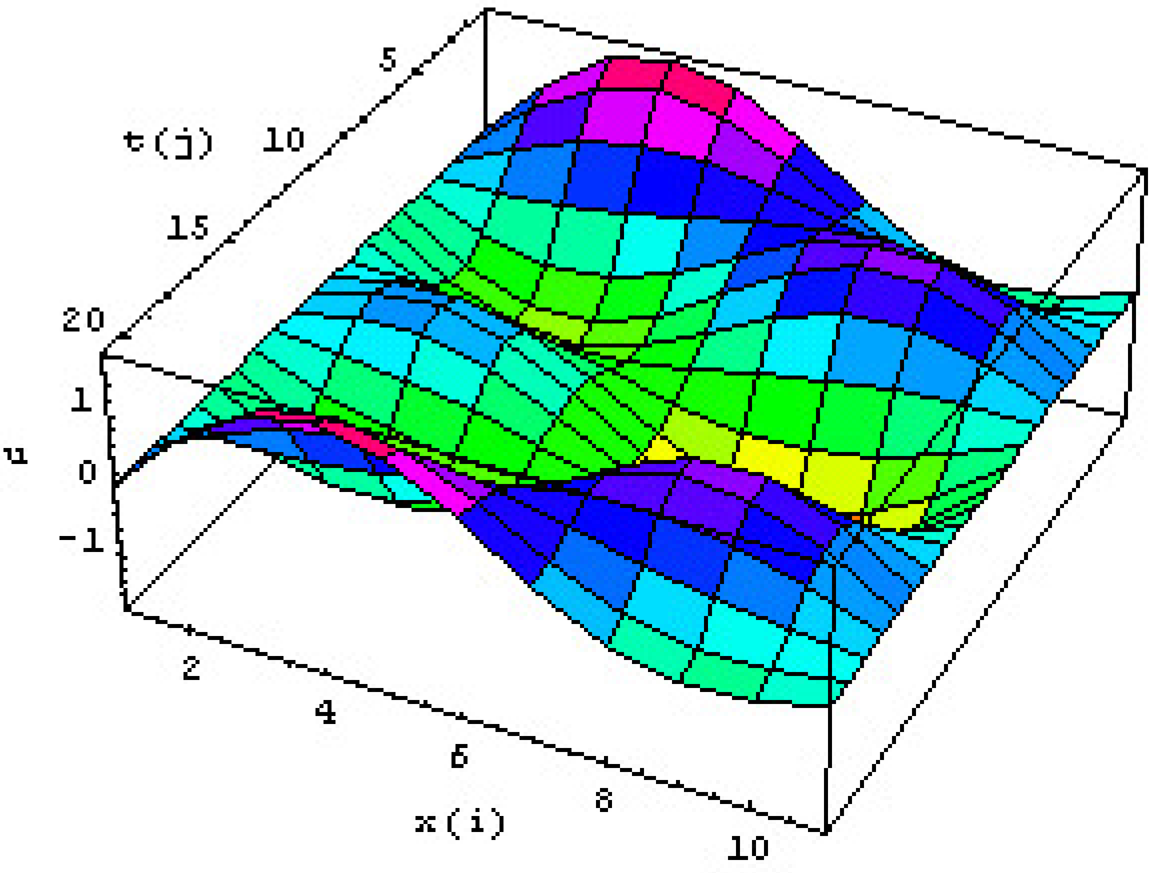
Now set up the table of solutions, then numerical Solution graph becomes:

```
r = c k/h;
FDgrid[n, m];
FDsolve[n, m];
ListPlot3D[u, AxesLabel → {"t(j)", "x(i)", "u"}, ViewPoint → {4, 2, 3}, ColorFunction → Hue];
Print["The numerical solution to the P.D.E."];
Print["utt(x,t) = 4 uxx(x,t)"];
Print[" u(x,0) = f(x) = ", F[x]];
Print["ut(0,x) = g(x) = ", G[x]];
Print["c = ", c];
Print["h = ", h];
Print["k = ", k];
Print[" ck/h = ", c k/h];
```

The numerical solution to the P.D.E. was computed on a *"grid"* in a matrix. Hence, we have *"lost"* the connection between the *"x"* and *"t"* variables when plotting the solution.

```
Print["The numerical solution to the P.D.E."];
Print["utt(x,t) = 4 uxx(x,t)", "\n"];
Print[NumberForm[TableForm[Transpose[Chop[u]]], 3]];
```

0	0.897	1.54	1.76	1.54	1.	0.363	-0.142	-0.363	-0.279	0
0	0.769	1.33	1.54	1.38	0.951	0.429	0	-0.21	-0.182	0
0	0.432	0.769	0.948	0.951	0.809	0.588	0.361	0.182	0.0684	0
0	0	0.0516	0.182	0.377	0.588	0.741	0.769	0.639	0.363	0
0	-0.38	-0.588	-0.519	-0.182	0.309	0.769	1.02	0.951	0.571	0
0	-0.588	-0.951	-0.951	-0.588	0	0.588	0.951	0.951	0.588	0
0	-0.571	-0.951	-1.02	-0.769	-0.309	0.182	0.519	0.588	0.38	0
0	-0.363	-0.639	-0.769	-0.741	-0.588	-0.377	-0.182	-0.0516	0	0
0	-0.0684	-0.182	-0.361	-0.588	-0.809	-0.951	-0.948	-0.769	-0.432	0
0	0.182	0.21	0	-0.429	-0.951	-1.38	-1.54	-1.33	-0.769	0
0	0.279	0.363	0.142	-0.363	-1.	-1.54	-1.76	-1.54	-0.897	0
0	0.182	0.21	0	-0.429	-0.951	-1.38	-1.54	-1.33	-0.769	0
0	-0.0684	-0.182	-0.361	-0.588	-0.809	-0.951	-0.948	-0.769	-0.432	0
0	-0.363	-0.639	-0.769	-0.741	-0.588	-0.377	-0.182	-0.0516	0	0
0	-0.571	-0.951	-1.02	-0.769	-0.309	0.182	0.519	0.588	0.38	0
0	-0.588	-0.951	-0.951	-0.588	0	0.588	0.951	0.951	0.588	0
0	-0.38	-0.588	-0.519	-0.182	0.309	0.769	1.02	0.951	0.571	0
0	0	0.0516	0.182	0.377	0.588	0.741	0.769	0.639	0.363	0
0	0.432	0.769	0.948	0.951	0.809	0.588	0.361	0.182	0.0684	0
0	0.769	1.33	1.54	1.38	0.951	0.429	0	-0.21	-0.182	0
0	0.897	1.54	1.76	1.54	1.	0.363	-0.142	-0.363	-0.279	0



Summary

Using an explicit finite difference numerical scheme for the second order time and space derivatives we derived a difference equation equivalent to the wave equation up to an error of order (Δx^2), which lead to. Analyzing the stability of this scheme we arrived at the condition $c\Delta t \leq \Delta x$, where c is the wave speed. This condition means that the wave speed of the numeric scheme must be at least as large as the wave speed of the exact equation. We also observed that the derived scheme uses two previous time steps to compute the values of the numerical solution at a particular grid point, thus one needs the values of two initial times steps to run the scheme. These, we were able to find from the initial-boundary conditions by explicit finite difference method, which does not add smaller order errors to the numerical scheme.

Bibliography

[1] J.W. Thomas Numerical partial differential equations:Finite difference method,Springer,Verlag,Inc,New York,1995.

[2] Lawrence.C.Evans, *Partial Differential Equation Graduate Studies in Mathematics*, American Mathematical Socity,Volume 19, August 1997.

[3] Vladimir G.Marz'ya, *Boundary Behavior of Solution to Elliptic Equations in General Domains*, European Mathematical Society, 2010.

[4] Dennis G.Zill, *Differential Equations with Boundary Value Problems*, Brooks/-CoLE,Canada, 2009.

[5] Mark H .Holmes, *Introduction to Numerical Methods in Differential Equations*, Academic Science and Engineering ,Rrnsselaer Polytechnic institute, 2007.

[6] Yehuda Pinchover and Jacob Rubinstein, *An Introduction to Partial Differential Equations*, cambridge university Press,New York, 2005.