Numerical Solution of Acoustic Wave Equation Using Method of Lines *

Maheshwar Pathak¹, Pratibha Joshi²†

Department of Mathematics, School of Engineering, University of Petroleum and Energy Studies (UPES)

(Received May 25 2016, Accepted March 31 2018)

Abstract. In the present work, method of lines is applied to solve the acoustic wave equation to achieve better stability, accuracy and cost of computation than existing methods. The efficiency and strength of the method are demonstrated by testing it on three numerical examples and the computed numerical solutions are compared with the solutions obtained using standard finite difference method. All the computational work has been performed in MATLAB and the codes are provided in appendix.

Keywords: method of lines, partial differential equations, finite difference method

1 Introduction

Acoustic wave equation models propagation of sound waves in fluids. All fluids have characteristics of restoration, which occurs due to the ambient pressure change. Hence, an acoustic wave is the outcome of ambient pressure change in the fluid. A local pressure change causes immediate fluid to compress which in turn causes additional pressure changes that leads to the propagation of an acoustic wave [1–25].

Previously acoustic wave equation has been solved by many standard methods like finite difference method [8, 12, 13, 17, 20, 23] and finite element method [2, 4, 18]. Apart from the conventional methods, many researchers have proposed new and modified numerical schemes as well. In [3] a rapid expansion method is proposed which involves Chebyshev expansion to the governing wave equation written in operator form. In [15], 2.5 dimensional finite difference scheme is used to solve three dimensional wave equation for faster computation. In [5], to reduce the problems with local convergence of standard iterative methods, a homotopy inversion method is proposed for the inversion of acoustic wave equation. In [1], a MATLAB package is generated to solve a two dimensional visco-acoustic wave equation using frequency domain finite difference methods. In [2], an acoustic equation with loss operator is generalized to the concept of variable order derivative and then solved by Crank-Nicholson method. In [22], an explicit hybridizable discontinuous galerkin method is applied to solve acoustic wave equation which improves the convergence rate of the solution in comparison to other discontinuous galerkin methods. Many researchers have combined two concepts to solve acoustic wave equation. In [24], concept of plane wave theory and Taylor’s series expansion are coupled to develop a high order implicit staggered-grid finite difference method. In [14], analytic continuation of forward scattering solution of acoustic wave equation is performed by Pad approximation sequences. Another combined approach to solve acoustic wave equation is proposed in [21] where a mimetic finite difference method is developed by combining a second order tensor mimetic discretization in space and leap frog approximation in time.

The challenge in finding solution of an acoustic wave equation is requirement of highly accurate and stable method with less complexity. In the present work, we have applied method of lines to find the solution of the

* Authors would like to express their sincere thanks to the anonymous reviewers for their valuable suggestions and comments. This paper is part of the project [No.-UCS&T/R&D/PHY SC.-10/12-13/6180] funded by Uttarakhand State Council for Science and Technology, Dehradun, Uttarakhand, India.
† Corresponding author. E-mail address: pratibha.joshi@gmail.com

Published by World Academic Press, World Academic Union
acoustic wave equation. Method of lines\cite{6, 16, 25} is a combination of analytical and finite difference methods. This method converts a partial differential equation into a system of coupled ordinary differential equations. Then by decoupling these ordinary differential equations and applying the concept of analytical methods the required solution is obtained. Due to the combination of numerical and analytical methods, this method gives high accuracy as compared to standard numerical techniques. We have also developed a MATLAB package for solving acoustic wave equation using method of lines.

The rest of this paper is organized as follows: In Section 2, method of lines is briefly introduced. In Section 3, conventional finite different discretization of acoustic wave equation is discussed. In Section 4, accuracy and efficiency of our approach is demonstrated through some numerical examples. In Section 5, conclusion of our results is mentioned. The MATLAB programs of the computational work done in Section 4, are given in Appendix.

2 Theoretical concept of method of lines for acoustic wave equation

The method of lines is a special form of the standard finite difference method. In this method we discretize the given domain partially. It means we discretize the given domain in one direction only. Here, we are considering the case of two variables. These two variables can be both spatial variables or one spatial variable and one time variable. Now the accuracy of the method depends on the way of discretization of the domain. If both the variables are spatial variables\cite{19} then we can discretize the domain in any direction. But in the case of spatial and time variable, if we discretize the domain in time variable then the stability of the method may be lost because of the unbounded nature of the time variable. So the method is more stable if the discretization is to be done in spatial direction. Here, we are considering the case of one spatial and one time variable. To explain the concept of method of lines we take one dimensional acoustic wave equation in a homogeneous medium as given below:

$$\frac{\partial^2 p}{\partial x^2} = \frac{1}{v^2} \frac{\partial^2 p}{\partial t^2}$$  \(1\)

where \(p(x, t)\) is a scalar wave field and \(\nu\) is the velocity of the sound.

Now we discretize Eq. (1) along spatial variable \(x\) as shown in Fig. 1. Let the region is divided into strips by \(n\) dividing straight lines parallel to the time axis (i.e. in a perpendicular direction from the direction of variable \(x\)). We apply the central difference scheme for spatial discretization as

$$\left( \frac{\partial^2 p}{\partial x^2} \right)_i = \frac{p_{i+1} - 2p_i + p_{i-1}}{h^2}$$  \(2\)

where \(h\) is the spacing between two consecutive lines.

i.e.

\(h = \frac{a}{n + 1}, \quad p_i = p_i(t) = p(x_i, t) \& x_i = i.h, i = 1, 2, 3, \ldots, n.\)

Therefore, using Eq. (2) in Eq. (1), we get

$$\left[ \frac{p_{i+1} - 2p_i + p_{i-1}}{h^2} \right] = \frac{1}{v^2} \frac{\partial^2 p_i}{\partial t^2}$$  \(3\)

Thus the wave field in Eq. (1) can be replaced by a vector of size \(n\), namely

$$[P] = [p_1, p_2, \ldots, p_n]^t$$  \(4\)
To obtain this, we define the transformed wave field analytically, we need to obtain a system of uncoupled ordinary differential equations from the coupled Eq. (6).

The next step is to solve the resulting equations analytically along the \( t \) coordinate. To solve Eq. (6) analytically, we need to obtain a system of uncoupled ordinary differential equations from the coupled Eq. (6). To obtain this, we define the transformed wave field \( \bar{P} \) as given below:

\[
[P] = [T][\bar{P}]
\]

and requiring that

\[
[T]^t [Q] [T] = [\lambda^2]
\]

where \([\lambda^2]\) is a diagonal matrix and is the transpose of \([T]^t\). \([\lambda^2]\) and \([T]^t\) are eigenvalue and eigenvector matrices belonging to \([Q]\). The transformation matrix \([T]\) and the eigenvalue matrix \([\lambda^2]\) depend on the boundary conditions and are given in Table I for various combinations of boundaries\(^{19}\).

It should be noted that the eigenvector matrix \([T]\) has the following properties:

\[
[T]^{-1} = [T]^t, \quad [T][T]^t = [T]^t [T] = [I].
\]

where \([I]\) is an identity matrix.

Substituting Eq. (8) into Eq. (6) gives

\[
\frac{1}{v^2} \frac{\partial^2 [T][\bar{P}]}{\partial t^2} + \frac{1}{h^2} [Q][T][\bar{P}] = 0
\]

multiplying by \([T]^{-1} = [T]^t\) gives

\[
- \frac{1}{h^2} [Q][P(t)] = \frac{1}{v^2} \frac{\partial^2 [P(t)]}{\partial t^2},
\]

where \([Q]\) is an \( n \times n \) tridiagonal matrix representing the discretized form of second derivative with respect to \( x \),

\[
[Q] = \begin{bmatrix}
    a_l & -1 & 0 & \cdots & 0 \\
    -1 & 2 & -1 & \cdots & 0 \\
    \vdots & \ddots & \ddots & \ddots & \ddots \\
    0 & \cdots & \cdots & -1 & 2 \\
    0 & \cdots & \cdots & 0 & -1 & a_r 
\end{bmatrix}.
\]

All the elements of matrix \([Q]\) are zeros except the tridiagonal terms. Elements of the first and the last row of \([Q]\) depends on the boundary conditions at \( x = 0 \) and \( x = a \). \( a_l = 2 \) for the Dirichlet boundary condition and \( a_l = 1 \) for the Neumann boundary condition. The same is true for \( a_r \).
Table 1: Elements of transformation matrix \([T]\) and eigenvalues\(^*\)

<table>
<thead>
<tr>
<th>Left boundary condition</th>
<th>Right boundary condition</th>
<th>(T_{ij})</th>
<th>(\lambda_i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dirichlet</td>
<td>Dirichlet</td>
<td>(\sqrt{\frac{2}{n+1}} \sin \frac{j \pi}{n+1}, [T_{DD}])</td>
<td>(2\sin \frac{i \pi}{2(n+1)})</td>
</tr>
<tr>
<td>Dirichlet</td>
<td>Neumann</td>
<td>(\sqrt{\frac{2}{n+1}} \sin \frac{i(j-0.5) \pi}{n+0.5}, [T_{DN}])</td>
<td>(2\sin \frac{i-0.5 \pi}{2n+1})</td>
</tr>
<tr>
<td>Neumann</td>
<td>Dirichlet</td>
<td>(\sqrt{\frac{2}{n+0.5}} \cos \frac{(i-0.5)(j-0.5) \pi}{n+0.5}, [T_{ND}])</td>
<td>(2\sin \frac{i-0.5 \pi}{2n+1})</td>
</tr>
<tr>
<td>Neumann</td>
<td>Neumann</td>
<td>(\sqrt{\frac{2}{n+0.5}} \cos \frac{(i-0.5)(j-1) \pi}{n+0.5}, j &gt; 1, [T_{NN}])</td>
<td>(\frac{1}{\sqrt{n}}, j = 1)</td>
</tr>
</tbody>
</table>

\(^{i, j = 1, 2, \ldots, n}\) and subscripts \(D\) and \(N\) denote Dirichlet and Neumann conditions, respectively.

\[
\frac{1}{v^2} \frac{\partial^2 p}{\partial t^2} + \frac{1}{h^2} \left[ \lambda^2 \right] [\bar{P}] = 0,
\]

\[
\left( \frac{1}{v^2} \frac{\partial^2}{\partial t^2} + \frac{1}{h^2} \left[ \lambda^2 \right] \right) [\bar{P}] = 0,
\]

\[
\left( \frac{\partial^2}{\partial t^2} + \alpha_i^2 \right) [\bar{P}] = 0,
\]

\[
\left( \frac{d^2}{dt^2} + \alpha_i^2 \right) [\bar{P}] = 0
\]

\[(12)\]

where \(\alpha_i = \frac{v \lambda_i}{h}\).

This is an ordinary differential equation with solution

\[
\bar{P}_i = A_i \cos \alpha_i t + B_i \sin \alpha_i t
\]

\[(13)\]

where \(\alpha_i = \frac{v \lambda_i}{h}\).

Thus, Acoustic wave Eq. (1) is solved numerically using a finite difference method in the \(x\) direction and analytically in the \(t\) direction.

### 3 Standard finite difference discretization of the acoustic wave equation

Here, we have given the standard finite difference discretization of a uniform grid for the acoustic wave equation. Using central difference scheme for both directions, the discretization is given below:

\[
\left( \frac{\partial^2 p}{\partial x^2} \right)_{(i,j)} = \frac{p_{i+1,j} - 2p_{i,j} + p_{i-1,j}}{h^2}
\]

\[(14)\]

\[
\left( \frac{\partial^2 p}{\partial t^2} \right)_{(i,j)} = \frac{p_{i,j+1} - 2p_{i,j} + p_{i,j-1}}{k^2}
\]

\[(15)\]

where \(p_{i,j} = p(x_i, t_j)\).

Now using Eqs. (14) and (15) in Eq. (1), we get the finitely discretize acoustic wave equation as given below:

\[
\frac{p_{i+1,j} - 2p_{i,j} + p_{i-1,j}}{h^2} = \frac{1}{v^2} \left[ p_{i,j+1} - 2p_{i,j} + p_{i,j-1} \right].
\]

**WJMS email for contribution**: submit@wjms.org.uk
Now using equations (14) and (15) in equation (1), we get the finitely discretize acoustic wave equation as given below:

\[
\begin{align*}
    p(i, j + 1) &= \alpha^2 p(i + 1, j) - 2 (\alpha^2 - 1) p(i, j) + \alpha^2 p(i - 1, j) - p(i, j - 1) \\
    \alpha &= \frac{k v}{h}.
\end{align*}
\]

Eq. (16) provides solution of Eq. (1) at future time level if the solutions at present and past time levels are known.

4 Numerical illustration

In this section we will show the efficiency of the method of lines by taking three real life problems. All computational work has been performed on a TOSHIBA laptop with intel core i3 processor and 4 GB RAM.

**Problem 1:** Let the one dimensional acoustic wave equation in a homogeneous medium is

\[
\frac{\partial^2 p}{\partial x^2} = \frac{1}{v^2} \frac{\partial^2 p}{\partial t^2}
\]

where, \( p = p(x, t) \) is a scalar wave field, and \( v \) is the velocity. We have considered the case where the above equation satisfies Dirichlet’s boundary conditions with initial conditions as given below:

\[
\begin{align*}
    p(x, 0) &= 3 \sin \pi x, \quad 0 \leq x \leq 1 \\
    \frac{\partial p(x, t)}{\partial t} &= 0, \quad \text{for} \quad t = 0 \quad \text{and} \quad 0 \leq x \leq 1.
\end{align*}
\]

This problem has exact solution

\[
\begin{align*}
    p(x, t) &= 3 \cos (\pi vt) \cdot \sin (\pi x).
\end{align*}
\]

We applied the method of lines scheme on it by developing a MATLAB code for \( v = 300 m/s \) [Appendix I]. The error analysis of the problem 1, for 0.1 seconds for by method of lines scheme is described in Tab. 2, which shows nearly fourth order accuracy.

\[\text{WJMS email for subscription: info@wjms.org.uk}\]
Table 2: Error analysis of problem 1 for 0.1 seconds by method of lines scheme

| Number of subintervals in spatial direction ($n$) | $||e_{\infty}||_\infty$ | Ratio ($r$) | Order of convergence ($\eta$) (approx.) |
|-----------------------------------------------|--------------------------|-------------|--------------------------------------|
| 40                                            | 8.799993448711518e-004   | 4           | 4                                    |
| 80                                            | 5.50084199571843e-005    | 15.9974     | 4                                    |
| 160                                           | 3.438161890745306e-006   | 15.9994     | 4                                    |
| 320                                           | 2.148867119800002e-007   | 15.9998     | 4                                    |
| 640                                           | 1.343044919721592e-008   | 15.9999     | 4                                    |

Table 3: Error analysis of problem 1 for 0.1 seconds by standard finite difference scheme

| Number of subintervals in spatial direction ($n$) | $||e_{\infty}||_\infty$ | Ratio ($r$) | Order of convergence ($\eta$) (approx.) |
|-----------------------------------------------|--------------------------|-------------|--------------------------------------|
| 40                                            | 2.131628207280301e-014    | 58.02       | 5.85                                 |
| 80                                            | 3.673940397442059e-016    | 1.00        | 0.0                                  |
| 160                                           | 3.673940397442059e-016    | 1.00        | 0.0                                  |
| 320                                           | 3.673940397442059e-016    | 1.00        | 0.0                                  |

We also applied standard finite difference scheme on the problem 1 by developing a MATLAB code for $v = 300 \text{m/s}$ [Appendix II]. The error analysis of the problem 1, for 0.1 seconds and $k = 1200$. (number of subintervals in time direction) by standard finite difference scheme is described in Table III, which shows instability of the method.

Hence from the above two Tables 2 and 3, we can conclude that method of lines is more effective and stable compared to standard finite difference method. Although standard finite difference method gives better and stable results for the condition $k/h = 1/v$, it is unstable otherwise. Thus the standard finite difference method is conditionally convergent, while method of lines remains unconditionally convergent. Hence for the above type of problem method of lines scheme is more suitable and effective method.

The following Fig. 3 to Fig. 6 display the numerical as well as exact solution of problem 1 in 40 and 80 subintervals by method of line scheme and standard finite difference scheme. It can be easily seen that numerical solution closely matches with exact solution for method of line scheme in both cases (Figs. 3 and 5) while it becomes unstable for standard finite difference method in second case (Fig. 6).

![Fig. 3](image-url)

Fig. 3: Numerical and exact solution of problem 1, at $t = 0.1$ seconds for 40 subintervals by method of lines scheme

In Figs. 7(a) and 7(b) errors in numerical solutions of problem 1 at $t = 0.1$ seconds for 40 subintervals by method of lines scheme and by standard finite difference method ($k = 1200$) have been given.

**Problem 2:** Let the one dimensional acoustic wave equation in a homogeneous medium is

$$\frac{\partial^2 p}{\partial x^2} = \frac{1}{v^2} \frac{\partial^2 p}{\partial t^2}$$

\[ (21) \]
where, \( p = p(x, t) \) is a scalar wave field, and \( v \) is the velocity. We have considered the case where the above equation satisfies Dirichlet’s boundary conditions with initial conditions\(^\text{[10]}\), as given below:

\[
p(x, 0) = (x - x_0)(a - x)e^{-\frac{r^2}{4}(x-x_0)^2}, \quad 0 \leq x \leq a
\]

(22)

\[
\frac{\partial p(x, t)}{\partial t} = 0, \quad \text{for} \ t = 0 \ \text{and} \ 0 \leq x \leq a
\]

(23)

where \( x_0 \) is the location of the center of the source, and \( r^2 \) is an attenuation coefficient.

The numerical solutions of the above problem by method of lines and by standard finite difference method are given in Table 5.

Taking \( v = 300 \text{m/s}, \ a = 5 \text{m}, \ r^2 = 1 \) and \( x_0 = 0 \), we get the wave field as given below in Table 5:

**Problem 3**: Let us consider the wave equation
Table 4: Error analysis of problem 1 for 0.1 seconds by standard finite difference scheme

Table 5: Numerical solutions of problem 2 at 0.1 seconds by method of lines and by standard finite difference scheme

\[
\frac{\partial^2 p}{\partial x^2} = \frac{1}{v^2} \frac{\partial^2 p}{\partial t^2}
\]

(24)

where, \( p = p(x, t) \) represents pressure field and \( v \) represents velocity of sound in air (at 20°C, can be taken 340 m/s). We have considered the case where the above equation satisfies Dirichlet’s boundary conditions with initial conditions as given below:

\[
p(x, 0) = x^2 (a - x), \quad 0 \leq x \leq a
\]

(25)

\[
\frac{\partial p(x, t)}{\partial t} = 0, \text{ for } t = 0 \text{ and } 0 \leq x \leq a
\]

(26)

The numerical solutions of the above problem by method of lines and by standard finite difference method are given in Table 6.

From Tables 5 and 6, we can see that numerical solutions of problem 2 and 3 obtained by method of lines are in good agreement with the numerical solutions obtained by standard finite difference method. But, if we increase the number of subintervals in spatial direction then the second scheme becomes unstable while first one remains stable.
Table 6: Numerical solutions of problem 3 at 0.1 seconds by method of lines and by standard finite difference scheme

<table>
<thead>
<tr>
<th>$x$</th>
<th>Wave field $p$ at $t = 0.1$ seconds by method of line scheme</th>
<th>Wave field $p$ at $t = 0.1$ seconds by standard finite difference method ($k = 600$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.8333</td>
<td>0.2011</td>
<td>0.2085</td>
</tr>
<tr>
<td>1.6666</td>
<td>-3.7024</td>
<td>-3.7375</td>
</tr>
<tr>
<td>2.5</td>
<td>-10.6919</td>
<td>-10.6921</td>
</tr>
<tr>
<td>3.3333</td>
<td>-14.1315</td>
<td>-14.1222</td>
</tr>
<tr>
<td>4.1666</td>
<td>-10.1858</td>
<td>-10.2288</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

5 Conclusion

In this work, acoustic wave equation is solved by method of lines which is a combination of analytic and numerical methods. The results by method of lines have been compared with results by standard finite difference method. It was observed that for solving acoustic wave equation, standard finite difference method is very sharp method in reference to convergence but lacks on the account of stability i.e. some additional conditions have to be imposed on the style of discretization of the domain to prevent it from becoming highly unstable. On the other hand method of lines scheme remains stable for every type of discretization of given domain. The comparison made, clearly indicates that method of lines is more useful and effective than the standard finite difference method to solve acoustic wave equation.

References

Appendix I: MATLAB code of method of lines scheme for problem 1:

WJMS email for contribution: submit@wjms.org.uk
clc;%code for method of lines scheme clear all;
q=0;
q=cputime;
format long;
n=input('number of interior points');
t=input('desire time');
a=1;
x=linspace(0,a,n+2);
h=a/(n+1);
v=300;

T=[];
P=[];
R=zeros(n,1);
Pbar=[];
lambda=[];
alpha=[];
A=[];
B=[];

%Transformation matrix[TDD]
for i=1:n
    for j=1:n
        T(i,j)=sqrt(2/(n+1))*sin((i*j*pi)/(n+1));
    end
end

%WJMS email for subscription: info@wjms.org.uk
for i=1:n
alpha(i)=(v*lambda(i))/h;
end
%————————————————————————–
% define P at initial time
% r² represent attenuation coefficient
% x0 represent location of the centre of source
r=1;
x0=0;
for i=1:n
% P(i) = (x(i + 1)− x0) * (a − x(i + 1)) * exp((r²/4 * h²) * (x(i + 1)− x0)²);
P(i)=3*sin(pi*x(i+1));
end
P=P';
%————————————————————————–
A=inv(T)*P;
B=inv(T)*R;
%————————————————————————–
for i=1:n
Pbar(i)=A(i)*cos(alpha(i)*t)+B(i)*sin(alpha(i)*t);
end
Pbar=Pbar';
%————————————————————————–
P=T*Pbar;
P=P';
nums=[0 P 0];
exs=[];
for i=1:n+2
exs(i)=3*cos(pi*v*t)*sin(pi*x(i));
end
aberror=[];
for i=1:n+2
aberror=[aberror; abs(exs(i)-nums(i))];
end
merror=max(aberror)
%————————————————————————–
% graph
plot(x,nums,'b',x,exs,'*');
plot(x,aberror);
CT=cputime-q;
Appendix II: MATLAB code of standard finite difference method for problem 1:

clc;%code for standard finite difference method
clear all
q=0;
q=cputime;
format long;
n=input('enter number of subintervals in spatial direction');
m=input('enter number of subintervals in time direction');
t=input('desire time');
a=1;
x=linspace(0,a,n+1);
h=a/n;
k=t/m;
v=300;
g=[];
lb=zeros(m+1,1);%left boundary
rb=lb;%right boundary
lob=[];%lower boundary
A=[];%left coefficient matrix
B=[];%right coefficient matrix
%————————————————————

r=1;
x0=0;
for j=1:n-1
    lob(j) = (x(j + 1) - x0) * (a - x(j + 1)) * exp((r^2/4 * h^2) * (x(j + 1) - x0)^2);
    lob(j)=3*sin(pi*x(j+1));
end
%————————————————————

s=k/h;
alpha=(k*v)/h;
B(1, 1) = ((alpha)^2/2) * lb(1) - ((alpha)^2 - 1) * lob(1) + ((alpha)^2/2) * lob(2);
for j=2:n-2
    B(1, j) = ((alpha)^2/2) * lob(j - 1) - ((alpha)^2 - 1) * lob(j) + ((alpha)^2/2) * lob(j + 1);
end
B(1, n - 1) = ((alpha)^2/2) * lob(n - 2) - ((alpha)^2 - 1) * lob(n - 1) + ((alpha)^2/2) * rb(1);
%————————————————————
B(2, 1) = (alpha)^2 * lb(2) - 2 * ((alpha)^2 - 1) * B(1, 1) + (alpha)^2 * B(1, 2) - lob(1);
for j=2:n-2
    B(2, j) = (alpha)^2 * B(1, j - 1) - 2 * ((alpha)^2 - 1) * B(1, j) + (alpha)^2 * B(1, j + 1) - lob(j);
end
B(2, n - 1) = (alpha)^2 * B(1, n - 2) - 2 * ((alpha)^2 - 1) * B(1, n - 1) + (alpha)^2 * rb(2) - lob(n - 1);
%————————————————————
for i=3:m
for j=2:n-2
B(i, 1) = (alpha)^2 * lb(i) - 2 * ((alpha)^2 - 1) * B(i - 1, 1) + (alpha)^2 * B(i - 1, 2) - B(i - 2, 1);
B(i, j) = (alpha)^2 * B(i - 1, j - 1) - 2 * ((alpha)^2 - 1) * B(i - 1, j) + (alpha)^2 * B(i - 1, j + 1) - B(i - 2, j);
B(i, n - 1) = (alpha)^2 * B(i - 1, n - 2) - 2 * ((alpha)^2 - 1) * B(i - 1, n - 1) + (alpha)^2 * rb(i) - B(i - 2, n - 1);
end
end

%————————————————————————–

nums=[];
for i=1:n-1
nums(i)=B(m,i);
end
nums=[0 nums 0];
exs=[];
for i=1:n+1
exs(i)=3*cos(pi*v*t)*sin(pi*x(i));
end
aberror=[];
for i=1:n+1
aberror=[aberror;abs(exs(i)-nums(i))];
end
merror=max(aberror)
%————————————————————————–

%graph
plot(x,nums,'b',x,exs,'*');
plot(x,aberror);
CT=cputime-q;