1 Nonhomogenous Wave Equation

We present a numerical investigation of the linear wave equation with a nonhomogeneous source term using a space–time finite element method (FEM) in $1+1$ dimensions. The second–order wave equation is reduced to a first–order system by introducing an auxiliary variable. The resulting weak forms are discretized using rectangular finite elements and solved within PETSc's SNES framework. We discuss implementation aspects including residual and Jacobian assembly and enforcement of initial and boundary conditions. A convergence study is reported by measuring the discrete L_2 error norm. The ultimate goal here is to accurately reproduce the results found in [1] using PETSc's SNES framework for nonlinear PDE's. We also use their DMDA framework (structured grid) for feasible parallel implementation.

1.1 Introduction

The wave equation

$$u_{tt} - u_{xx} = f(x,t) \tag{1}$$

with wave speed $c = 1$. In this work, we consider a case where the source term is given by

$$f(x,t) = -2\cos(t)\,\exp\Big(-\big(x-\cos(t)\big)^2\Big)\,\big(2\cos^3(t) - 4x\cos^2(t) + 2x^2\cos(t) - 2\cos(t) + x\big)\,,$$

over the space–time domain $\Omega \times [0,T]$ with $\Omega = [-5,5]$ and $T = 10$. The manufactured solution,

$$u(x,t) = \exp\Big(-\big(x-\cos(t)\big)^2\Big),$$

is used both to specify the initial condition and to assess the numerical accuracy. As this problem is linear, we expect to see SNES convergence in 1 iteration.

1.2 Problem Setup and Formulation

To recast (1) into a first–order system, we introduce the auxiliary variable

$$v = u_t,$$

leading to the system

$$v_t - u_{xx} = f(x,t), \tag{2}$$

$$-u_t + v = 0. \tag{3}$$

The initial conditions are

$$u(x,0) = \exp\Big(-\big(x-1\big)^2\Big), \quad v(x,0) = 0,$$

(which is consistent with the exact solution since $\cos(0) = 1$), and homogeneous Neumann boundary conditions (i.e., $u_x = 0$ and $v_x = 0$) are imposed for $t > 0$.

The weak forms for the two equations are obtained by multiplying by a test function $\phi(x,t)$ and integrating over the space–time domain $Q = \Omega \times (0,T)$:

$$M(u,v,\phi) = \int_Q (v_t\,\phi + u_x\,\phi_x - f\,\phi)\,dx\,dt = 0, \tag{4}$$

$$N(u,v,\phi) = \int_Q (-u_t\,\phi + v\,\phi)\,dx\,dt = 0.$$

The global unknown vector is taken as

$$\mathbf{U} = \begin{bmatrix} u \\ v \end{bmatrix},$$

with the u-components stored at even indices and the v-components at odd indices.

You may notice that in going from the strong from, Eq. 2, to the weak form, Eq. 4, we integrated by parts spatially. This moved a derivative over to the test function, and since there is nothing at the spatial boundary, we are able to drop those terms.

1.3 Numerical Method and Implementation

We discretize the problem using a space–time finite element method with rectangular (bilinear) elements. With $\xi = x/hx$ and $\tau = t/ht$, we define the rectangular basis functions:

$$\phi_1(\xi,\tau) = (1-\xi)(1-\tau),$$

$$\phi_2(\xi,\tau) = \xi(1-\tau),$$

$$\phi_3(\xi,\tau) = \xi\tau,$$

$$\phi_4(\xi,\tau) = (1-\xi)\tau.$$

In our implementation, the residuals for the two equations are assembled elementwise. The contributions computed in each element are then scattered into the global residual vector. Similarly, the Jacobian matrix is assembled from local element matrices. We solve Equations M and N on each element (K_e) using the aforementioned basis functions. We compute symbolically all terms that do not pertain to the forcing function f. We handle the forcing function in a different routine that uses Gauss quadrature to integrate over each element. Thus, to find the symbolic 4x4 matrices we solve

$$M_{ij}^{(e)} = \int_{K_e} \left(\frac{\partial \phi_i}{\partial t}\phi_j + \frac{\partial \phi_i}{\partial x}\frac{\partial \phi_j}{\partial x} \right) dx\,dt$$

$$N_{ij}^{(e)} = \int_{K_e} \left(\frac{\partial \phi_i}{\partial t}\phi_j + \phi_i\phi_j \right) dx\,dt.$$

This requires the computation of 3 different element stiffness matrices that we refer to as

1. The time term: $\frac{\partial \phi_j}{\partial t}\phi_i$
2. The space term: $\frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x}$
3. and the mass/standard term: $\phi_j \phi_i$

we symbolically compute the element stiffness matrices utilizing Python's SymPy.

$$\text{Time} = \frac{h_x}{12}\begin{bmatrix} -2 & -1 & 1 & 2 \\ -1 & -2 & 2 & 1 \\ -1 & -2 & 2 & 1 \\ -2 & -1 & 1 & 2 \end{bmatrix},$$

$$\text{Space} = \frac{h_t}{6h_x}\begin{bmatrix} 2 & -2 & -1 & 1 \\ -2 & 2 & 1 & -1 \\ -1 & 1 & 2 & -2 \\ 1 & -1 & -2 & 2 \end{bmatrix},$$

$$\text{Mass} = \frac{h_t h_x}{36}\begin{bmatrix} 9 & 18 & 36 & 18 \\ 18 & 9 & 18 & 36 \\ 36 & 18 & 9 & 18 \\ 18 & 36 & 18 & 9 \end{bmatrix}.$$

The assembly routines in our PETSc code include:

- **FormResidual:** Local contributions for the two equations are computed. In our implementation, the residual for the equation

$$v_t - u_{xx} - f = 0$$

  is denoted as `eqn1-r-local` and is scattered into the u-component. Similarly, the residual for

$$-u_t + v = 0$$

  is denoted as `eqn2-r-local` and scattered into the v-component.

- **FormJacobian:** The local stiffness matrices (denoted by A_{time}, A_{space}, and A_{standard}) are hard-coded. The local Jacobian is then assembled using these matrices.

1.4 Results

We conducted a convergence study by varying the number of spatial and temporal nodes simultaneously. The following table summarizes the mesh sizes, grid spacings, the computed L_2 error norms, and the convergence rates.

Table 1: Convergence Study Results

Mesh ($nx \times nt$)	$hx = ht$	L_2 Error Norm	Rate
100×100	0.1010	1.630×10^{-02}	-
200×200	0.0503	3.971×10^{-3}	2.04
400×400	0.0251	9.823×10^{-4}	2.02
800×800	0.0125	2.445×10^{-4}	2.01

To find the rate, given two successively refined grids (h_i, E_i) and (h_{i+1}, E_{i+1}), we find the order of convergence,

$$p_i = \frac{\log\big(E_i/E_{i+1}\big)}{\log\big(h_i/h_{i+1}\big)}.$$

With finer refinements, we see the order of convergence approach 2nd order.

The residual norms obtained during the SNES iterations were very small after the first iteration (typically on the order of 10^{-13} to 10^{-14}), and the SNES convergence reason indicated successful termination after 1 iteration, which is to be expected for a linear PDE solve.

1.5 Conclusions

We have formulated and implemented a space–time FEM for solving a linear wave equation with a nonhomogeneous source term. The problem was reduced to a first–order system, and both the residual and Jacobian were assembled within PETSC's SNES framework. The convergence study shows that as the mesh is refined, the L_2 error norm decreases with an observed 2nd order convergence rate.

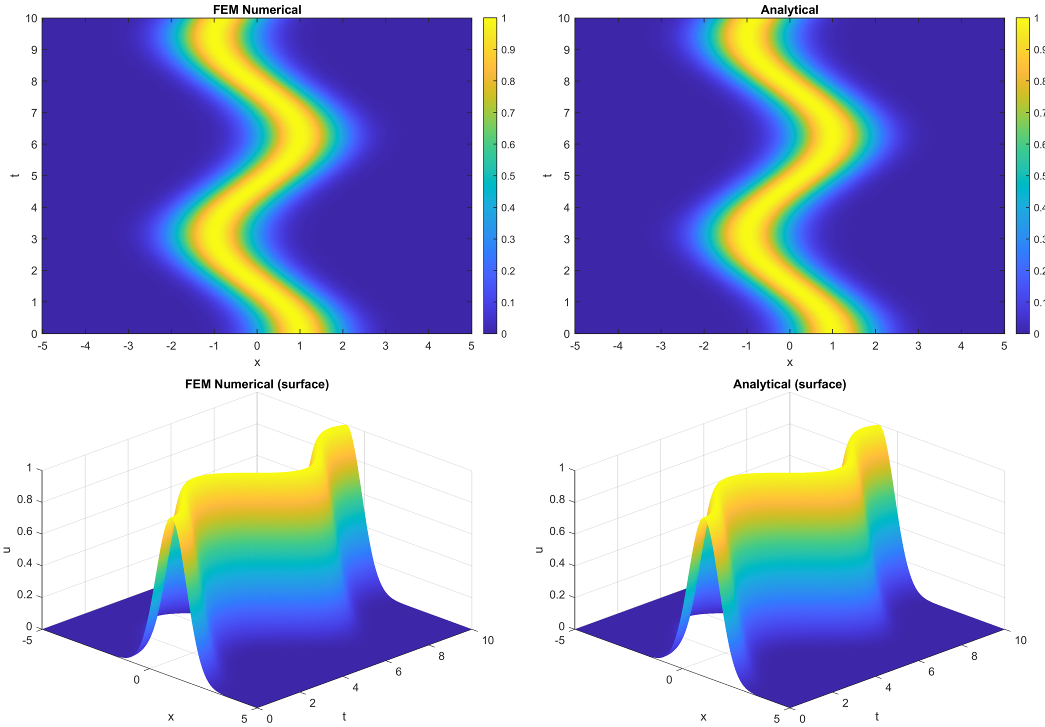

Figure 1: Visual comparison between numerical and analytical solutions of the wave equation for $nx = nt = 400$. Left: FEM solution. Right: Exact solution $u(x,t) = exp(-(x - cos(t))^2$.

References

Matthew Anderson and Jung-Han Kimn. A numerical approach to space-time finite elements for the wave equation. *Journal of Computational Physics*, 226(1):466–476, 2007.