

AE370 Group Project 2 (Team 14)

Vincent Kim*, Amruth Seetharaman†, Alec Wojcik‡, Max Jiang§
University of Illinois Urbana-Champaign

This paper presents a numerical framework for simulating wave propagation in thin aircraft skin panels, similar to laser-based inspection techniques for defect detection. Modeling the panel as a thin elastic membrane we derive the two-dimensional wave equation and solve it using a centered finite difference method and the Crank-Nicolson time stepping method. The solver is validated against an analytical solution and a high-resolution reference, confirming its second-order accuracy and stability. We use the model to study wave behavior in both ideal and defected materials demonstrating its utility in visualizing wave patterns and identifying defects. It is found that the two dimensional wave analysis converges to the analytical solution at the expected rate both temporally and spatially and that proportionately large defects or defect clusters may be detected with the wave with visual inspection.

I. Nomenclature

Latin symbols

A	= amplitude of initial Gaussian bump m
\mathbf{A}	= $2N \times 2N$ first-order system matrix $[\mathbf{0} \ \mathbf{I}; -K \ \mathbf{0}]$
B	= interior block of discrete Laplacian K in x -direction
C	= coupling block $-\frac{c^2}{\Delta y^2} I_x$ in K
c	= membrane wave speed $c = \sqrt{T/\rho}$ m s ⁻¹
D_x, D_y	= 1-D second-derivative finite-difference matrices
$E(t)$	= spatial energy norm $\ u\ _2^2 + \ u_t\ _2^2/c^2$
F_x, F_y	= net vertical tension forces on an element N
$f(x, y)$	= prescribed initial displacement (analytical test)
$g(x, y)$	= prescribed initial velocity (analytical test)
k	= flattened node index $k = jN_x + i$
K	= discrete 2-D Laplacian matrix ($N \times N$) scaled by c^2
L_x, L_y	= membrane side lengths m
m	= element mass $\rho \Delta x \Delta y$ kg
m, n	= modal indices in double-Fourier solution (context-dependent)
N_x, N_y	= number of interior nodes in x and y
N	= total spatial unknowns $N = N_x N_y$
$P(x)$	= local quadratic Lagrange interpolant
$R(z)$	= Crank–Nicolson amplification factor $(1 + \frac{z}{2})/(1 - \frac{z}{2})$
t, t^n	= continuous and discrete time s
T	= uniform in-plane tension N m ⁻¹
Δt	= time step s
$\Delta x, \Delta y$	= uniform grid spacing m

*Student, Department of Aerospace Engineering, University of Illinois Urbana-Champaign

†Student, Department of Aerospace Engineering, University of Illinois Urbana-Champaign

‡Student, Department of Aerospace Engineering, University of Illinois Urbana-Champaign

§Student, Department of Aerospace Engineering, University of Illinois Urbana-Champaign

$u(x, y, t)$	=	transverse displacement field m
\mathbf{u}	=	vector of nodal displacements ($N \times 1$)
\mathbf{v}	=	velocity vector $\dot{\mathbf{u}}$
\mathbf{w}	=	full state vector $[\mathbf{u}; \mathbf{v}]$
α	=	Gaussian width parameter m^{-2}
ρ	=	mass per unit area kg m^{-2}
λ, μ	=	separation constants (Sturm–Liouville)
ω_{mn}	=	modal circular frequency $\frac{c\pi}{L} \sqrt{m^2 + n^2} \text{ rad s}^{-1}$
Ω	=	computational domain $[0, L_x] \times [0, L_y]$
$\partial\Omega$	=	boundary of Ω
∇^2	=	continuous Laplacian operator

II. Introduction

Laser-generated inspection lets aircraft engineers probe an aircraft's skin without removing panels or adding downtime, but its usefulness depends on accurate models of how elastic waves travel through a panel and how those waves change when the material is damaged or has a defect. We treat a thin section of aircraft skin as an isotropic, tensioned membrane and derive the governing two-dimensional wave equation directly from Newton's law.

A centered finite-difference stencil in space and a Crank–Nicolson scheme in time give a fully implicit, second-order-accurate solver that remains stable for the millisecond time steps needed in practice.

After validating the solver code against both an analytical benchmark and a high-resolution reference run, we use it to explore a set of key question regarding the detection of defects and plot the propagation of vertical displacement in the range between the maximum positive and negative amplitude.

III. Engineering PDE

A. Physical System

In today's aerospace industry, safety and quality control are essential, particularly for commercial flights. Aircraft such as the Airbus A350 (shown in Figure 1) typically operate 4 to 5 flights per day. Due to this high frequency of use, these aircraft must be extremely reliable and safe. To meet these standards, manufacturers like Boeing and Airbus have developed fast and efficient methods for detecting material flaws and ensuring overall quality.



Fig. 1 Airbus A350 Passenger Plane

One such method involves directing a laser pulse at a small section of the aircraft's surface and analyzing how the resulting wave propagates through the material. Deviations in the wave's behavior can indicate structural flaws or

defects. To interpret these signals accurately, we need to predict how waves should ideally propagate in the material. This prediction can be achieved using the two-dimensional wave equation.

B. 2D Wave Equation

The wave equation describes how disturbances propagate in a medium. Now we will derive the 2D wave equation from first principles, using Newton's second law and considering physical forces due to tension and inertia.

1. Assumptions

To simplify the problem we have made some assumptions. The medium is a thin, elastic membrane lying in the xy -plane. The vertical displacement of the membrane at position (x, y) and time t is denoted by $u(x, y, t)$. The membrane has uniform mass per unit area ρ (units: kg/m^2). The membrane is under uniform tension T (units: N/m), assumed isotropic and acting tangentially in all directions. Displacements are small so linear approximations hold.

These assumptions greatly simplify the problem. Here, the elastic membrane will represent the wing. This assumption will not work for thicker surfaces or metals, but would be similar to a wing of a smaller aircraft or model plane.

2. Force Analysis

Consider a small rectangular element of the membrane with dimensions $\Delta x \times \Delta y$. Let us examine the net vertical force due to tension along the x - and y -directions. Tension causes a restoring force proportional to the slope of displacement. The net vertical force in the x -direction is given by:

$$F_x = T\Delta y \left[\frac{\partial u}{\partial x} \Big|_{x+\frac{\Delta x}{2}} - \frac{\partial u}{\partial x} \Big|_{x-\frac{\Delta x}{2}} \right] \approx T\Delta y \frac{\partial^2 u}{\partial x^2} \Delta x \quad (1)$$

Similarly, the net vertical force in the y -direction is:

$$F_y = T\Delta x \left[\frac{\partial u}{\partial y} \Big|_{y+\frac{\Delta y}{2}} - \frac{\partial u}{\partial y} \Big|_{y-\frac{\Delta y}{2}} \right] \approx T\Delta x \frac{\partial^2 u}{\partial y^2} \Delta y \quad (2)$$

Adding the two contributions:

$$F_{\text{total}} = F_x + F_y = T \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \Delta x \Delta y \quad (3)$$

The mass of the element is $m = \rho \Delta x \Delta y$, and the vertical acceleration is $\partial^2 u / \partial t^2$. Newton's second law gives:

$$\rho \Delta x \Delta y \frac{\partial^2 u}{\partial t^2} = T \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \Delta x \Delta y \quad (4)$$

Cancelling $\Delta x \Delta y$ from both sides:

$$\rho \frac{\partial^2 u}{\partial t^2} = T \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (5)$$

Dividing through by ρ :

$$\frac{\partial^2 u}{\partial t^2} = \frac{T}{\rho} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (6)$$

3. Wave Speed

We define the wave speed c as:

$$c = \sqrt{\frac{T}{\rho}} \quad (7)$$

Substituting this into the equation gives the standard form of the 2D wave equation:

$$\frac{\partial^2 u}{\partial t^2} = c^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (8)$$

4. Parameters

The function $u(x, y, t)$ represents the displacement of the membrane at a given position (x, y) and time t , measured in meters. The wave speed, denoted by c and measured in meters per second (m/s), is determined by the material properties of the membrane. The tension per unit length, T (in newtons per meter, N/m), provides the restoring force that drives the wave motion. Meanwhile, ρ , the mass per unit area (in kilograms per square meter, kg/m²), accounts for the membrane's inertia.

C. Initial and Boundary Conditions

1. Initial Conditions

We prescribe initial conditions for the displacement and velocity fields. Let $\mathbf{u}(x, y, t)$ denote the vertical displacement of the membrane at position (x, y) and time t . The wave equation requires both the initial displacement $\mathbf{u}(x, y, 0)$ and initial velocity $\frac{\partial \mathbf{u}}{\partial t}$ to be specified.

Gaussian Bump

A smooth, localized initial disturbance is introduced using a two-dimensional Gaussian bump centered within the domain. The general form is given by:

$$\mathbf{u}(x, y, 0) = A e^{-\alpha[(x-x_0)^2 + (y-y_0)^2]}, \quad (9)$$

where:

- A is the amplitude of the initial displacement.
- α controls the width of the bump (higher α yields a narrower peak).
- (x_0, y_0) is the center of the bump, typically chosen to be the center of the domain.

This form ensures the displacement is smooth and rapidly decaying away from the center, mimicking a localized excitation similar to an impulse from a laser pulse.

Zero Velocity

The initial velocity is assumed to be zero across the entire membrane, indicating that the membrane starts from rest. This is written as:

$$\frac{\partial \mathbf{u}}{\partial t} = 0. \quad (10)$$

This condition models an impulsive start where the shape is displaced but has no initial momentum.

2. Boundary Conditions

Our membrane is held fixed at the edges of the wing section. To add this to our model, we impose Dirichlet boundary conditions, which physically correspond to a membrane with edges held fixed. Mathematically, this means that the displacement is zero along the boundary of the spatial domain for all time.

Let the spatial domain be defined as:

$$\Omega = [0, L_x] \times [0, L_y]$$

and the boundary of the domain as:

$$\partial\Omega = \{(x, y) \in \Omega \mid x = 0, x = L_x, y = 0, \text{ or } y = L_y\}$$

The boundary conditions are imposed as:

$$u(x, y, t) = 0 \quad \text{for all } (x, y) \in \partial\Omega, t \geq 0 \quad (11)$$

This enforces zero displacement at every boundary point and for all time.

D. Key Questions

Here are some of the key questions that we will seek to answer in this report relating to the physical system:
We seek to answer the following key questions about the physical system:

- 1) What does wave propagation look like in an ideal, defect-free material?
- 2) How can material defects be modeled as local variations in wave speed c ?
- 3) How can these observations be used to characterize the presence of a defect?

Regarding the numerical method, we aim to address:

- 1) What are the stability properties of the Crank–Nicolson time-stepping method?
- 2) How does the method's error converge with spatial resolution?
- 3) How does the method's error converge with temporal resolution?

IV. Numerical Method

A. Method of Lines

In order to approximate a solution to the 2D wave equation, we will use the method of lines (MOL). The method of lines is an approach to solving partial differential equations (PDEs). This method works by first discretizing the spatial domain. In doing so, the PDE is transformed into a system of ordinary differential equations (ODEs). This system of ODEs can then be solved using a time-stepping integration method.

B. Spatial Discretization using Finite Difference Method

We begin with the 2D wave equation as defined in the previous section:

$$\frac{\partial^2 u(x, y, t)}{\partial t^2} = c^2 \left(\frac{\partial^2 u(x, y, t)}{\partial x^2} + \frac{\partial^2 u(x, y, t)}{\partial y^2} \right) \quad (12)$$

1. Lagrange Polynomials

To spatially discretize the 2D wave equation, we use a finite difference method. Finite difference methods are attractive for their simplicity, especially when working with structured, rectangular grids. They approximate derivatives using algebraic expressions involving function values at discrete grid points.

In the context of this problem, we assume the spatial domain is a rectangular region $\Omega = [0, L_x] \times [0, L_y]$, and we discretize it using a uniform grid spacing Δx and Δy in the x - and y -directions, respectively. Let $u_{i,j}(t)$ denote the numerical approximation to the true solution $u(x_i, y_j, t)$ at grid point (x_i, y_j) and time t .

Let us first focus on approximating the second partial derivative with respect to x . We consider a fixed row j , and construct a local interpolating polynomial $P(x)$ to $u(x, y_j, t)$ near x_i using three points: x_{i-1} , x_i , and x_{i+1} .

Define the stencil as:

$$x_{i+k} = x_i + k\Delta x \quad \text{for } k = -1, 0, 1.$$

We interpolate the solution using a quadratic Lagrange polynomial:

$$P(x) = \sum_{k=-1}^1 u_{i+k,j} \ell_k(x), \quad (13)$$

where $\ell_k(x)$ is the Lagrange basis polynomial associated with node x_{i+k} , defined as:

$$\ell_k(x) = \prod_{\substack{m=-1 \\ m \neq k}}^1 \frac{x - x_{i+m}}{x_{i+k} - x_{i+m}}. \quad (14)$$

The second derivative of u at x_i is approximated by the second derivative of the interpolant:

$$\frac{\partial^2 u}{\partial x^2} \Big|_{(x_i, y_j)} \approx P''(x_i) = \sum_{k=-1}^1 u_{i+k,j} \ell_k''(x_i). \quad (15)$$

Computing the second derivatives of the Lagrange basis functions at x_i , we obtain:

$$\ell_{-1}''(x_i) = \frac{1}{\Delta x^2}, \quad (16)$$

$$\ell_0''(x_i) = -\frac{2}{\Delta x^2}, \quad (17)$$

$$\ell_1''(x_i) = \frac{1}{\Delta x^2}. \quad (18)$$

Therefore, the finite difference approximation becomes:

$$\frac{\partial^2 u}{\partial x^2}(x_i, y_j) \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2}. \quad (19)$$

A similar procedure is applied in the y -direction. Fixing i , we construct a local quadratic interpolant in the y -direction over the points y_{j-1} , y_j , and y_{j+1} .

The finite difference approximation of the second derivative with respect to y is:

$$\frac{\partial^2 u}{\partial y^2}(x_i, y_j) \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2}. \quad (20)$$

Adding the approximations in both directions, we approximate the Laplacian $\nabla^2 u$ at a grid point (x_i, y_j) as:

$$\nabla^2 u(x_i, y_j, t) \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2}. \quad (21)$$

Substituting the finite difference approximations into the PDE, we obtain the semi-discrete system (ODEs in time):

$$\frac{d^2 u_{i,j}(t)}{dt^2} = c^2 \left(\frac{u_{i+1,j}(t) - 2u_{i,j}(t) + u_{i-1,j}(t)}{\Delta x^2} + \frac{u_{i,j+1}(t) - 2u_{i,j}(t) + u_{i,j-1}(t)}{\Delta y^2} \right) \quad (22)$$

This equation holds for interior grid points. Boundary values are set explicitly according to Dirichlet boundary conditions.

Let us vectorize the system by flattening the 2D array (i, j) into a single vector of length $N = N_x \times N_y$, and write the system as:

$$\frac{d^2 \mathbf{u}(t)}{dt^2} = -K \mathbf{u}(t) \quad (23)$$

where K is the discrete Laplacian matrix scaled by c^2 , built from Kronecker products of 1D second-derivative matrices in x and y .

$$K = -c^2 \left(\frac{I_y \otimes D_x}{\Delta x^2} + \frac{D_y \otimes I_x}{\Delta y^2} \right) \quad (24)$$

Here:

- I_x, I_y are identity matrices of size N_x and N_y
- D_x, D_y are the standard second derivative finite difference matrices for x and y

2. Matrix Form of K

Let us now look at what the entries in K look like. We know that D_y and D_x are the standard second-order finite difference matrices:

$$D_x = D_y = \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & 1 & \ddots & \ddots & \\ & & \ddots & -2 & 1 \\ & & & 1 & -2 \end{bmatrix} \in \mathbb{R}^{N \times N}$$

We now write out the full matrix $K \in \mathbb{R}^{N_x N_y \times N_x N_y}$, where each entry K_{mn} corresponds to the interaction between flattened grid points m and n , defined by their 2D coordinates:

Let each 2D grid point (i, j) be mapped to a 1D index

$$k = j \cdot N_x + i, \quad \text{where } 0 \leq i < N_x, 0 \leq j < N_y.$$

Then, the matrix entries of K are given by:

$$\begin{aligned} K_{k,k} &= \frac{2c^2}{\Delta x^2} + \frac{2c^2}{\Delta y^2}, \\ K_{k,k \pm 1} &= -\frac{c^2}{\Delta x^2}, \quad (\text{only if } i \pm 1 \text{ is in bounds}), \\ K_{k,k \pm N_x} &= -\frac{c^2}{\Delta y^2}, \quad (\text{only if } j \pm 1 \text{ is in bounds}). \end{aligned}$$

All other entries are zero. The resulting matrix is block tridiagonal, with each block being tridiagonal. The structure is:

$$K = \begin{bmatrix} B & C & & & \\ C & B & C & & \\ & C & B & \ddots & \\ & & \ddots & \ddots & C \\ & & & C & B \end{bmatrix}, \quad (25)$$

where each $B \in \mathbb{R}^{N_x \times N_x}$ is the 1D discrete Laplacian in x -direction:

$$B = \frac{c^2}{\Delta x^2} \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & \ddots & \ddots & \\ & & \ddots & 2 & -1 \\ & & & -1 & 2 \end{bmatrix},$$

and $C = -\frac{c^2}{\Delta y^2} I_x$ represents coupling between rows in the y -direction.

Thus, K has a five-point stencil structure: each interior point is coupled to its four immediate neighbors (left, right, up, down) and itself.

3. Converting to a First-Order System

To convert this second-order system into a first-order system suitable for numerical time integration, we define a new variable — the velocity vector:

$$\mathbf{v}(t) = \frac{d\mathbf{u}}{dt}. \quad (26)$$

The second-order equation then becomes a coupled system of two first-order equations:

$$\frac{d\mathbf{u}}{dt} = \mathbf{v}(t), \quad (27)$$

$$\frac{d\mathbf{v}}{dt} = -K\mathbf{u}(t). \quad (28)$$

We now define the full state vector:

$$\mathbf{w}(t) = \begin{bmatrix} \mathbf{u}(t) \\ \mathbf{v}(t) \end{bmatrix} \in \mathbb{R}^{2N}, \quad (29)$$

so the system becomes:

$$\frac{d\mathbf{w}}{dt} = \mathbf{A}\mathbf{w}(t), \quad (30)$$

with the system matrix $\mathbf{A} \in \mathbb{R}^{2N \times 2N}$ given by:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -K & \mathbf{0} \end{bmatrix}, \quad (31)$$

where:

- $\mathbf{0} \in \mathbb{R}^{N \times N}$ is the zero matrix.
- $\mathbf{I} \in \mathbb{R}^{N \times N}$ is the identity matrix.
- $K \in \mathbb{R}^{N \times N}$ is the discrete Laplacian as defined above.

C. Crank–Nicolson Time Stepping

Let time be discretized as $t^n = n\Delta t$. Define $\mathbf{u}^n \approx \mathbf{u}(t^n)$ and $\mathbf{v}^n \approx \mathbf{v}(t^n)$. The Crank–Nicolson scheme applied to each ODE is:

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \frac{\Delta t}{2} (\mathbf{v}^{n+1} + \mathbf{v}^n) \quad (32)$$

$$\mathbf{v}^{n+1} = \mathbf{v}^n - \frac{\Delta t}{2} K (\mathbf{u}^{n+1} + \mathbf{u}^n) \quad (33)$$

This is an implicit system. To solve, eliminate one variable.

Substitute the first equation into the second:

$$\begin{aligned} \mathbf{v}^{n+1} &= \mathbf{v}^n - \frac{\Delta t}{2} K (\mathbf{u}^n + \mathbf{u}^{n+1}) \\ &= \mathbf{v}^n - \frac{\Delta t}{2} K \left(\mathbf{u}^n + \mathbf{u}^n + \frac{\Delta t}{2} (\mathbf{v}^{n+1} + \mathbf{v}^n) \right) \\ &= \mathbf{v}^n - \frac{\Delta t}{2} K \left(2\mathbf{u}^n + \frac{\Delta t}{2} (\mathbf{v}^{n+1} + \mathbf{v}^n) \right) \end{aligned}$$

Rearrange and solve for \mathbf{v}^{n+1} :

$$\left(I + \frac{\Delta t^2}{4} K \right) \mathbf{v}^{n+1} = \left(I - \frac{\Delta t^2}{4} K \right) \mathbf{v}^n - \Delta t K \mathbf{u}^n \quad (34)$$

Once \mathbf{v}^{n+1} is found, compute \mathbf{u}^{n+1} from:

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \frac{\Delta t}{2} (\mathbf{v}^n + \mathbf{v}^{n+1}) \quad (35)$$

V. Error and Convergence

A. Spatial Error

1. Deriving Expected Spatial Error

We begin with the two-dimensional wave equation defined on a rectangular domain with sufficiently smooth initial and boundary data. To approximate this equation numerically, we discretize the spatial variables using a uniform Cartesian grid with spacing Δx and Δy in the x - and y -directions, respectively.

Let $u_{i,j}(t) \approx u(x_i, y_j, t)$, where $x_i = i\Delta x$ and $y_j = j\Delta y$. We apply second-order centered finite difference approximations for the spatial second derivatives:

$$\frac{\partial^2 u}{\partial x^2}(x_i, y_j) \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2}, \quad (36)$$

$$\frac{\partial^2 u}{\partial y^2}(x_i, y_j) \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2}. \quad (37)$$

We now analyze the local truncation error of each spatial discretization. Consider the **Taylor expansions** of u in the x -direction about the point (x_i, y_j) :

$$u(x_{i+1}, y_j) = u(x_i, y_j) + \Delta x \frac{\partial u}{\partial x} + \frac{\Delta x^2}{2} \frac{\partial^2 u}{\partial x^2} + \frac{\Delta x^3}{6} \frac{\partial^3 u}{\partial x^3} + \frac{\Delta x^4}{24} \frac{\partial^4 u}{\partial x^4} + O(\Delta x^5), \quad (38)$$

$$u(x_{i-1}, y_j) = u(x_i, y_j) - \Delta x \frac{\partial u}{\partial x} + \frac{\Delta x^2}{2} \frac{\partial^2 u}{\partial x^2} - \frac{\Delta x^3}{6} \frac{\partial^3 u}{\partial x^3} + \frac{\Delta x^4}{24} \frac{\partial^4 u}{\partial x^4} + O(\Delta x^5). \quad (39)$$

Adding the two expansions and simplifying yields:

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} = \frac{\partial^2 u}{\partial x^2}(x_i, y_j) + \frac{\Delta x^2}{12} \frac{\partial^4 u}{\partial x^4}(x_i, y_j) + O(\Delta x^4). \quad (40)$$

A similar expansion in the y -direction gives:

$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} = \frac{\partial^2 u}{\partial y^2}(x_i, y_j) + \frac{\Delta y^2}{12} \frac{\partial^4 u}{\partial y^4}(x_i, y_j) + O(\Delta y^4). \quad (41)$$

Substituting these into the semi-discrete form of the wave equation:

$$\frac{d^2 u_{i,j}}{dt^2} = c^2 \left(\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} \right), \quad (42)$$

we find the local truncation error to be:

$$\tau_{i,j} = \frac{d^2 u_{i,j}}{dt^2} - c^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = -\frac{c^2 \Delta x^2}{12} \frac{\partial^4 u}{\partial x^4} - \frac{c^2 \Delta y^2}{12} \frac{\partial^4 u}{\partial y^4} + O(\Delta x^4 + \Delta y^4). \quad (43)$$

This shows that the spatial discretization introduces a local truncation error of **second order** in both Δx and Δy .

2. Computing Error

To test the spatial convergence, we decrease the spatial grid resolution by refining Δx and Δy , while holding Δt at a small value. As the grid resolution increases (i.e., as Δx and Δy decrease), the numerical solution should converge toward the exact solution. For second-order accurate finite difference methods, we expect the error to scale with $O(\Delta x^2)$.

To evaluate the spatial error, we compute the **grid 2 norm** at several different spatial resolutions:

$$E_{\text{grid}}(\Delta x, \Delta y) = \sqrt{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} (u_{\text{exact}}(x_i, y_j) - u_{\text{num}}(x_i, y_j))^2}$$

An important thing to note here, is that the grids will not have the same number of points or resolution. Because of this it becomes difficult to compute. To get around this we will interpolate the girds such that their nodes align properly.

B. Temporal Error

The temporal error arises from the discretization of the time derivative in the time-stepping method. Since we are using the Crank-Nicolson method, which is an implicit method with second-order accuracy in time, the temporal error is expected to decrease quadratically as the time step Δt becomes smaller. Specifically, the temporal error is of order $O(\Delta t^2)$.

1. Deriving Expected Temporal Error

To determine the order of accuracy, we analyze the local truncation error, which is defined as the error incurred in one time step assuming previous values are exact. The local truncation error (LTE) is given by

$$\tau^{n+1} = \frac{y(t_{n+1}) - y(t_n)}{\Delta t} - \frac{1}{2} (f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))). \quad (44)$$

We expand $y(t_{n+1})$ in a **Taylor series** about t_n :

$$y(t_{n+1}) = y(t_n) + \Delta t y'(t_n) + \frac{\Delta t^2}{2} y''(t_n) + \frac{\Delta t^3}{6} y^{(3)}(t_n) + O(\Delta t^4). \quad (45)$$

Since $f(t, y(t)) = y'(t)$, we also expand $f(t_{n+1}, y(t_{n+1})) = y'(t_{n+1})$ as:

$$y'(t_{n+1}) = y'(t_n) + \Delta t y''(t_n) + \frac{\Delta t^2}{2} y^{(3)}(t_n) + O(\Delta t^3). \quad (46)$$

The right-hand side of the Crank-Nicolson formula becomes:

$$\frac{1}{2} (y'(t_n) + y'(t_{n+1})) = y'(t_n) + \frac{\Delta t}{2} y''(t_n) + \frac{\Delta t^2}{4} y^{(3)}(t_n) + O(\Delta t^3). \quad (47)$$

The left-hand side is:

$$\frac{y(t_{n+1}) - y(t_n)}{\Delta t} = y'(t_n) + \frac{\Delta t}{2} y''(t_n) + \frac{\Delta t^2}{6} y^{(3)}(t_n) + O(\Delta t^3). \quad (48)$$

Subtracting both sides yields:

$$\tau^{n+1} = \left(y'(t_n) + \frac{\Delta t}{2} y''(t_n) + \frac{\Delta t^2}{6} y^{(3)}(t_n) \right) - \left(y'(t_n) + \frac{\Delta t}{2} y''(t_n) + \frac{\Delta t^2}{4} y^{(3)}(t_n) \right) + O(\Delta t^3), \quad (49)$$

$$\tau^{n+1} = -\frac{\Delta t^2}{12} y^{(3)}(t_n) + O(\Delta t^3). \quad (50)$$

Therefore, the local truncation error is of order $O(\Delta t^2)$, which confirms that the Crank-Nicolson method is second-order accurate in time.

2. Computing Error

We can test temporal convergence by decreasing the time step Δt , while keeping Δx and Δy very small. As Δt decreases, the numerical solution should converge to the exact solution in time. For second-order time-stepping methods, we expect the temporal error to scale with $O(\Delta t^2)$.

To evaluate the temporal error, we compute the **grid 2** norm at each time step and refine Δt accordingly:

$$E_{\text{grid}}(\Delta t) = \sqrt{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} (u_{\text{exact}}(x_i, y_j, t_n) - u_{\text{num}}(x_i, y_j, t_n))^2}$$

where t_n represents the time at which the solution is evaluated.

3. Stability Region

As mentioned in the MOL derivation, the Crank-Nicolson method was chosen because it is **implicit**. This means that it will be more stable than an explicit method, such as RK4, especially for stiff problems such as this one. The stability relates to how the error accumulates over the entire numerical integration processes. We want the error to converge to zero as Δt goes to zero. We will compare the stability region of RK4 to that of Crank-Nicolson to visually show the higher stability of Crank-Nicolson.

To find the stability region of the Crank–Nicolson method, we apply it to the standard linear test equation:

$$y' = \lambda y, \quad (51)$$

where $\lambda \in \mathbb{C}$. The Crank–Nicolson method is given by:

$$y_{n+1} = y_n + \frac{\Delta t}{2} (f(t_n, y_n) + f(t_{n+1}, y_{n+1})), \quad (52)$$

where Δt is the time step. Applying the method to the test equation, with $f(t, y) = \lambda y$, gives:

$$y_{n+1} = y_n + \frac{\Delta t}{2} (\lambda y_n + \lambda y_{n+1}). \quad (53)$$

Rearranging terms:

$$y_{n+1} - \frac{\Delta t}{2} \lambda y_{n+1} = y_n + \frac{\Delta t}{2} \lambda y_n, \quad (54)$$

$$\left(1 - \frac{\Delta t}{2} \lambda\right) y_{n+1} = \left(1 + \frac{\Delta t}{2} \lambda\right) y_n. \quad (55)$$

Define the stability variable $z = \Delta t \lambda$. The equation becomes:

$$\left(1 - \frac{z}{2}\right) y_{n+1} = \left(1 + \frac{z}{2}\right) y_n, \quad (56)$$

and the amplification factor $R(z)$ is given by:

$$R(z) = \frac{y_{n+1}}{y_n} = \frac{1 + \frac{z}{2}}{1 - \frac{z}{2}}. \quad (57)$$

The stability region is defined as the set of all $z \in \mathbb{C}$ such that the method is stable, i.e.,

$$|R(z)| = \left| \frac{1 + \frac{z}{2}}{1 - \frac{z}{2}} \right| \leq 1. \quad (58)$$

This inequality holds if and only if $\text{Re}(z) \leq 0$. Thus, the Crank–Nicolson method is A-stable, meaning its stability region includes the entire left half of the complex plane as seen in Figure 2.

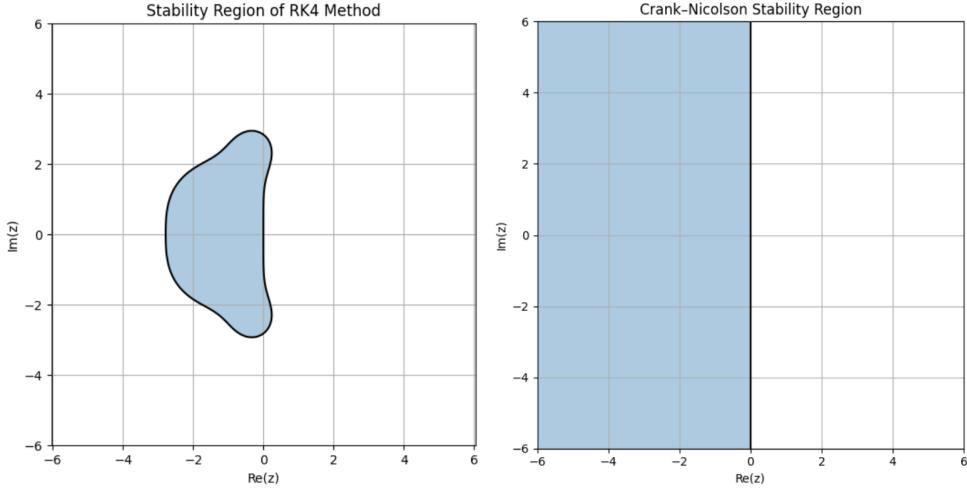


Fig. 2 Stability Region of RK4 vs Crank-Nicolson

Figure 2 is also a great visualization of the increased stability that implicit methods offer.

C. Solution Comparison

1. High Resolution Solution

To use the grid-2 norm, as defined above, we need a true solution. However, there is no closed form analytical solution to the two-dimensional wave equation when the initial condition is a Gaussian bump. Instead we can assume that at very small Δt , Δx , and Δy , the numerical solution is the "true solution". We can then compute both spatial and temporal error convergence. This will not prove that our solution converges to the correct physical value, but will show us the rate at which both converge. If both the finite difference and Crank-Nicolson methods were implemented correctly, we would expect to see second-order convergence ($O(\Delta t^2)$, $O(\Delta x^2)$, $O(\Delta y^2)$).

The high-resolution solution uses the following discretization parameters: a time step of $\Delta t = 0.0001$ seconds, and spatial step sizes of $\Delta x = 0.002$ meters and $\Delta y = 0.002$ meters.

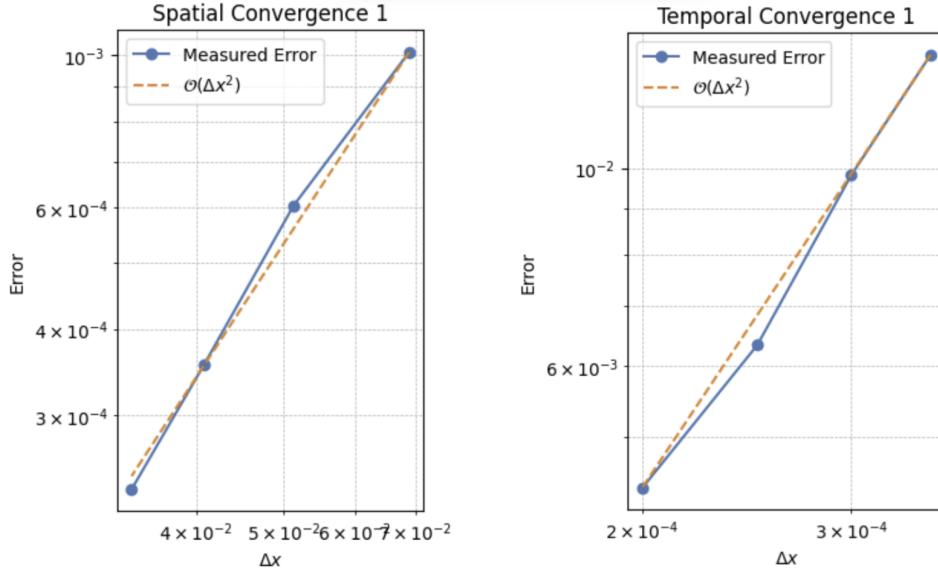


Fig. 3 Temporal and Spatial Convergence with High Res Solution

As seen in Figure 2 both the spatial and temporal error have second-order convergence. This partially proves that our method is implemented correctly. It proves that it converges at the correct rate but not necessarily to the physically accurate result.

2. Analytical Solution

As mentioned above, there is no analytical solution for the wave equation with the initial condition being a Gaussian bump. However, for homogeneous Dirichlet boundary conditions, there can still be an analytical solution if we change the initial conditions.

We consider the two-dimensional wave equation on a square domain of side length L , with homogeneous Dirichlet boundary conditions. Our goal is to find an analytical solution $u(x, y, t)$ and determine the corresponding initial conditions:

- The initial displacement: $u(x, y, 0) = f(x, y)$

- The initial velocity: $\frac{\partial u}{\partial t} \Big|_{t=0} = g(x, y)$

We begin by assuming a separable solution of the form:

$$u(x, y, t) = X(x)Y(y)T(t) \quad (59)$$

Substituting this into the wave equation:

$$X(x)Y(y) \frac{d^2T}{dt^2} = c^2 \left(Y(y)T(t) \frac{d^2X}{dx^2} + X(x)T(t) \frac{d^2Y}{dy^2} \right) \quad (60)$$

Divide both sides by $X(x)Y(y)T(t)$:

$$\frac{1}{T(t)} \frac{d^2T}{dt^2} = c^2 \left(\frac{1}{X(x)} \frac{d^2X}{dx^2} + \frac{1}{Y(y)} \frac{d^2Y}{dy^2} \right) \quad (61)$$

Each side must equal a constant. Let:

$$\frac{1}{T(t)} \frac{d^2T}{dt^2} = -\lambda, \quad \frac{1}{X(x)} \frac{d^2X}{dx^2} + \frac{1}{Y(y)} \frac{d^2Y}{dy^2} = -\frac{\lambda}{c^2} \quad (62)$$

We further separate the spatial part:

$$\frac{1}{X(x)} \frac{d^2X}{dx^2} = -\mu \quad (63)$$

$$\frac{1}{Y(y)} \frac{d^2Y}{dy^2} = -\left(\frac{\lambda}{c^2} - \mu\right) \quad (64)$$

This gives us three ordinary differential equations:

$$\frac{d^2X}{dx^2} + \mu X = 0 \quad (65)$$

$$\frac{d^2Y}{dy^2} + \left(\frac{\lambda}{c^2} - \mu\right) Y = 0 \quad (66)$$

$$\frac{d^2T}{dt^2} + \lambda T = 0 \quad (67)$$

We apply the boundary conditions. Since $u = 0$ on the boundary, and the solution is separable:

$$X(0) = X(L) = 0, \quad Y(0) = Y(L) = 0 \quad (68)$$

The solutions to these Sturm-Liouville problems are:

$$X_n(x) = \sin\left(\frac{n\pi x}{L}\right), \quad \mu_n = \left(\frac{n\pi}{L}\right)^2 \quad (69)$$

$$Y_m(y) = \sin\left(\frac{m\pi y}{L}\right), \quad \nu_m = \left(\frac{m\pi}{L}\right)^2 \quad (70)$$

Hence, the separation constant is:

$$\lambda_{mn} = c^2 (\mu_n + \nu_m) = c^2 \pi^2 \left(\frac{n^2 + m^2}{L^2} \right) \quad (71)$$

The time-dependent equation becomes:

$$\frac{d^2 T_{mn}}{dt^2} + \lambda_{mn} T_{mn} = 0 \quad (72)$$

with general solution:

$$T_{mn}(t) = A_{mn} \cos(\omega_{mn} t) + B_{mn} \sin(\omega_{mn} t), \quad \omega_{mn} = \frac{c\pi}{L} \sqrt{m^2 + n^2} \quad (73)$$

The full analytical solution is a double Fourier series:

$$u(x, y, t) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} [A_{mn} \cos(\omega_{mn} t) + B_{mn} \sin(\omega_{mn} t)] \sin\left(\frac{m\pi x}{L}\right) \sin\left(\frac{n\pi y}{L}\right) \quad (74)$$

To construct an example with a closed-form analytical solution, we choose only a single mode, say $m = n = 1$:

$$u(x, y, t) = \sin\left(\frac{\pi x}{L}\right) \sin\left(\frac{\pi y}{L}\right) \cos(\omega_{11} t), \quad \omega_{11} = \frac{c\pi\sqrt{2}}{L} \quad (75)$$

Initial displacement:

$$f(x, y) = u(x, y, 0) = \sin\left(\frac{\pi x}{L}\right) \sin\left(\frac{\pi y}{L}\right) \quad (76)$$

Initial velocity:

$$g(x, y) = \frac{\partial u}{\partial t} \Big|_{t=0} = -\omega_{11} \sin\left(\frac{\pi x}{L}\right) \sin\left(\frac{\pi y}{L}\right) \cdot \sin(0) = 0 \quad (77)$$

Thus, we have found an analytical solution to the 2D wave equation on a square domain with Dirichlet boundary conditions:

$$u(x, y, t) = \sin\left(\frac{\pi x}{L}\right) \sin\left(\frac{\pi y}{L}\right) \cos\left(\frac{c\pi\sqrt{2}}{L} t\right) \quad (78)$$

with corresponding initial conditions:

$$f(x, y) = \sin\left(\frac{\pi x}{L}\right) \sin\left(\frac{\pi y}{L}\right) \quad (79)$$

$$g(x, y) = 0 \quad (80)$$

This solution satisfies the wave equation, the Dirichlet boundary conditions, and the specified initial conditions.

Now that we have this, we can use this as our "true solution" when computing error. Shown below are the temporal and spatial convergence rates in comparison to Crank-Nicolson's expected convergence rate and the centered finite-difference method's convergence rate.

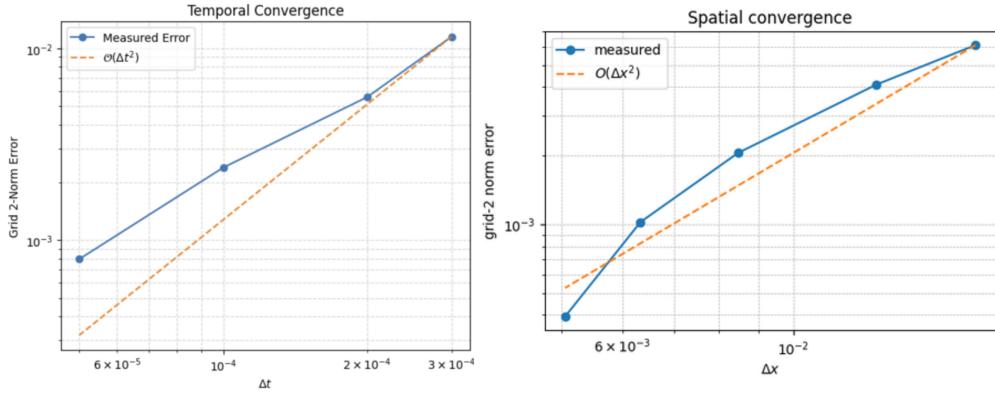


Fig. 4 Temporal and Spatial Convergence Rates with Expected Rates of Crank-Nicolson

We see that the spatial grid-2 norm error converges rapidly towards its expected rate of second order for Crank-Nicolson, exhibiting similar behaviors shown above in our comparison with a high resolution solution. We see a slightly slower convergence rate for the temporal solution and that is attributed to the larger Δt values that were used to compute the error. It was observed that any timestep that was smaller than around $100\mu\text{s}$ resulted in an exponentially longer runtime.

VI. Results

A. Ideal Propagation

We first establish a defect-free baseline by allowing the Gaussian pulse to propagate on a square membrane whose side lengths are $L_x = L_y = 0.5$ m. The material is assigned a uniform phase speed of $c = 30 \text{ m s}^{-1}$, and the membrane is discretized with $N = 200$ interior nodes in each coordinate direction. Time integration is carried out up to $t_{\text{final}} = 0.1$ s using an implicit Crank–Nicolson step size of $\Delta t = 1$ ms.

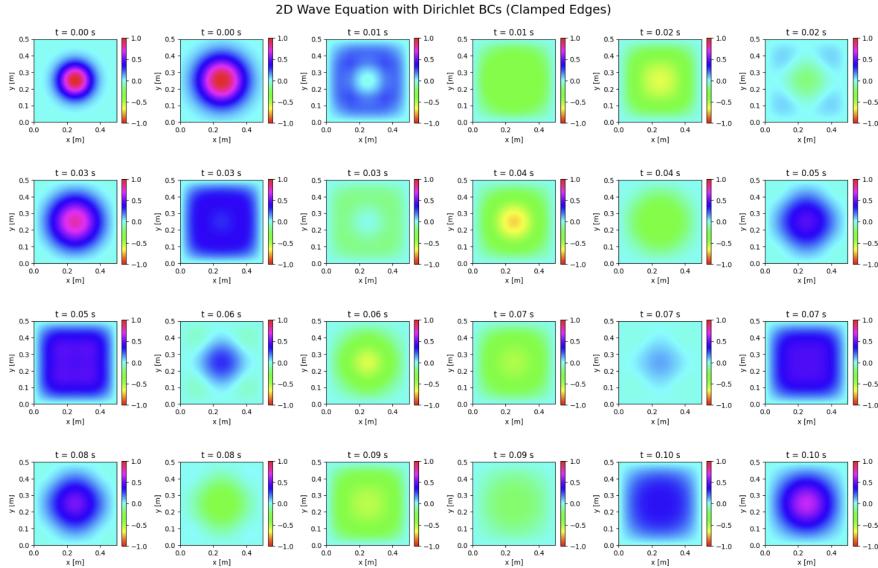


Fig. 5 Ideal Wave Propagation at Different Times

As seen in Figure 5, the wave originating from the initial Gaussian bump at $t = 0$ propagates symmetrically and smoothly outward in all directions, as expected in a homogeneous and defect-free medium. Due to the boundary

constraints, the wave reflects back continuously, resulting in the formation of a circular standing wave pattern. This radially symmetric behavior acts as a baseline to compare how wave propagation changes in the presence of defects or different material properties in later cases.

B. Propagation with Defects

In an ideal, defect-free material, wave propagation is smooth and radially symmetric when initiated by a localized disturbance such as a Gaussian pulse. This behavior arises from the uniform wave speed c , which ensures that the disturbance expands evenly in all directions. However, real materials often contain imperfections or inclusions that locally alter mechanical properties, resulting in non-uniform propagation behavior. In this simulation, material defects are modeled as spatially localized regions with reduced wave speed—implemented as circular zones where the value of c is scaled down within a predefined radius. This creates a heterogeneous wave speed field that directly modifies the system's discrete wave equation through a spatially varying coefficient $c(x, y)^2$ in the Laplacian term. The finite difference method applied here, using an implicit time integrator and sparse matrix formulation, enables stable and efficient time evolution of the wave field even under these complex conditions.

By running simulations with multiple randomized defect maps, the method reveals how wavefronts become distorted: bending, scattering, or slowing as they encounter low-speed regions. These distortions are clear indicators of internal structure and can be systematically analyzed across time snapshots. This numerical approach allows us to not only predict how defects affect wave motion but also suggests an inverse application: by observing anomalies in wave propagation patterns—such as asymmetries, delays, or localized attenuations—one can infer the presence, location, and possibly the severity of material defects. Thus, the numerical framework is both a forward modeling tool and a diagnostic mechanism for nondestructive evaluation of materials.

1. Material Defect Seeds

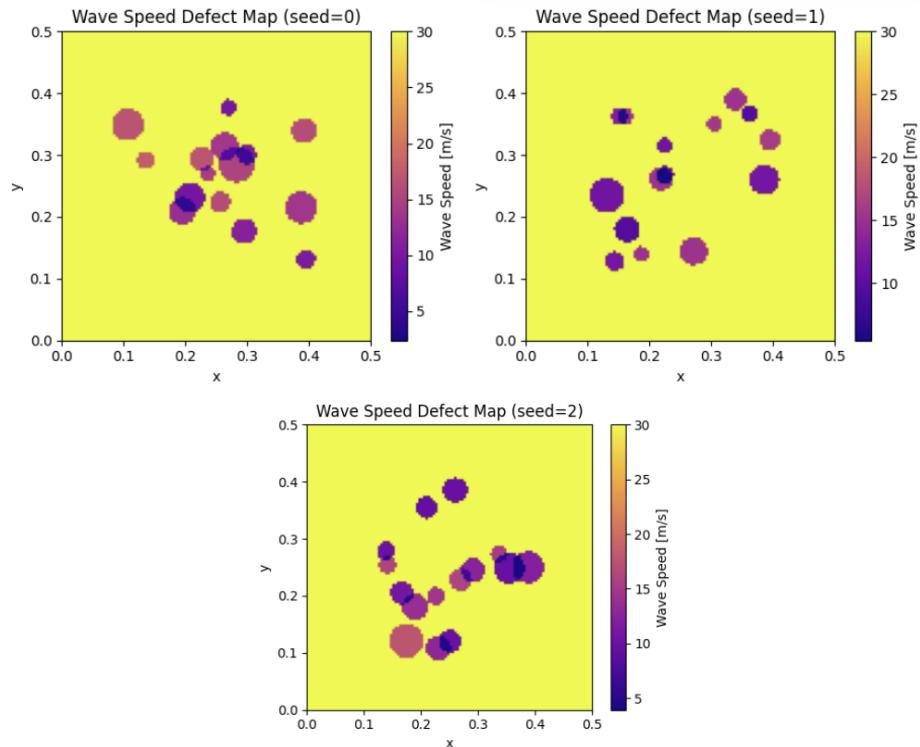


Fig. 6 Three Examples of Defect Seed Maps

Three generated materials with randomly placed and sized defects can be seen in Figure 6. The yellow region represents a wave speed of 30[m/s]. This is the same wave speed as seen in the ideal case from Figure 5. As discussed

above, the defects are modeled as spots where the wave speed, c , is locally reduced by a random factor. These defects appear a darker shade of purple on the plots.

2. Propagation with Different Seeds

Now we will run our numerical solver with the sections that have the random defects. This will allow us to qualitatively determine whether the wave propagation has changed due to the added material defects. Ideally, these defects should significantly alter the behavior of the wave such that it is visually obvious that there is a defect. This will make the defects easier to catch and will decrease the likelihood that defects will pass undetected by safety and quantity inspectors.

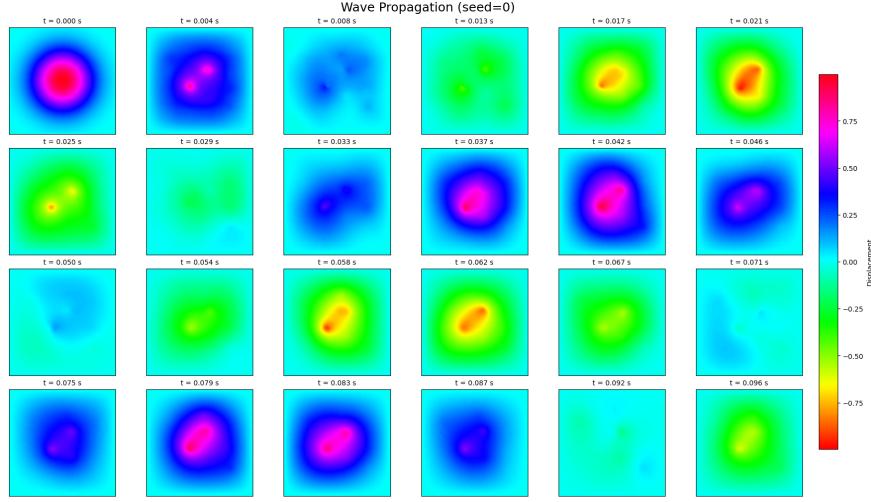


Fig. 7 Wave Propagation for Defect Seed 0

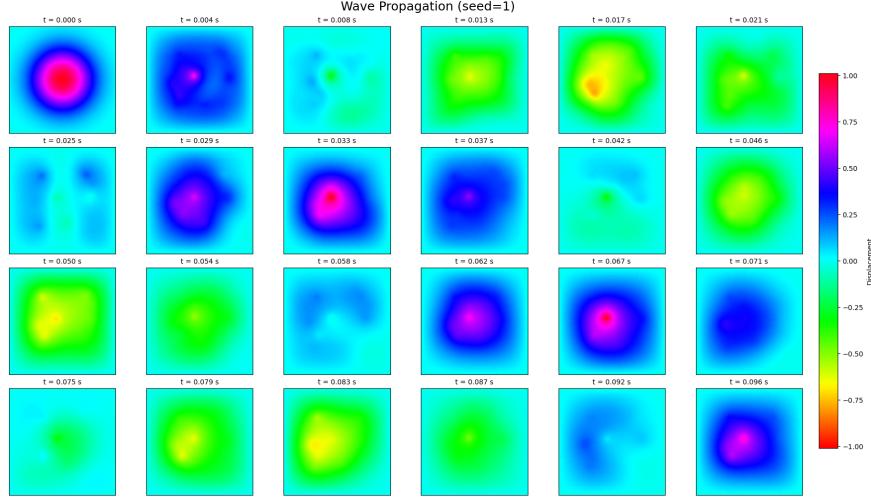


Fig. 8 Wave Propagation for Defect Seed 1

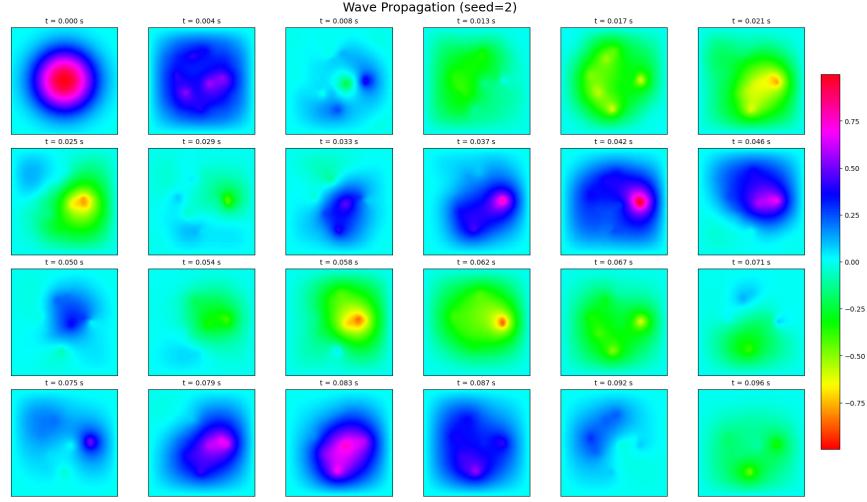


Fig. 9 Wave Propagation for Defect Seed 2

As seen in Figure 7, Figure 8, and Figure 9, from $t = 0$ to $t = 0.1$, the wave behavior changes based on the defects and their positions. More importantly, none of their standing waves exhibit full radial symmetry as seen in the ideal material with no defects due to defects altering propagation speed in a specific direction. This indicates the presence of defects with different properties that pose a risk to the normal function of the component.

3. Propagation with Scaled Defect Speeds

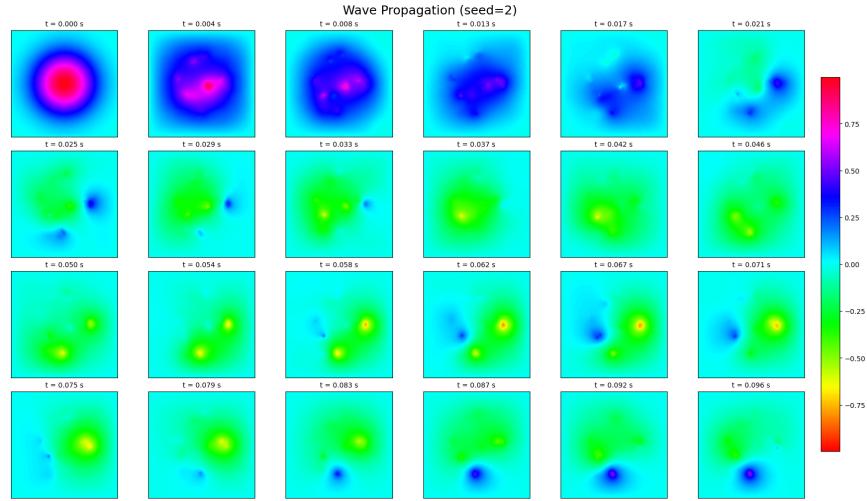


Fig. 10 Wave Propagation for 0.5 Scaled Defect Seed 2

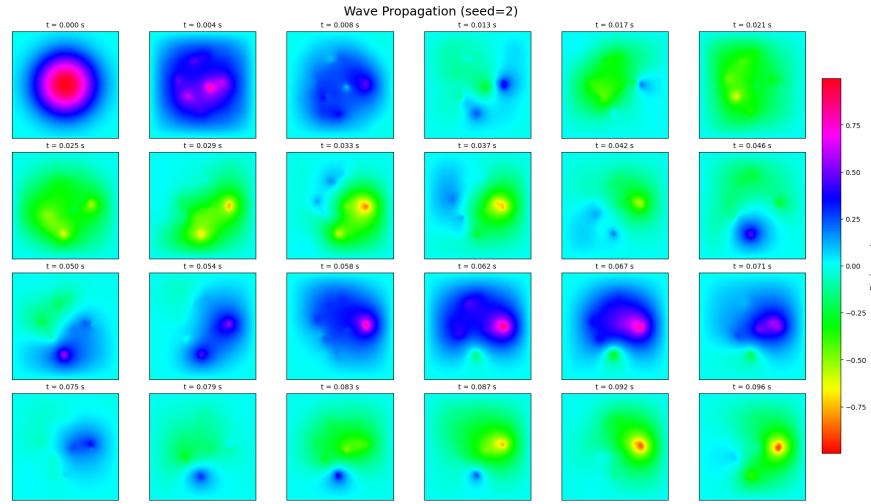


Fig. 11 Wave Propagation for 0.75 Scaled Defect Seed 2

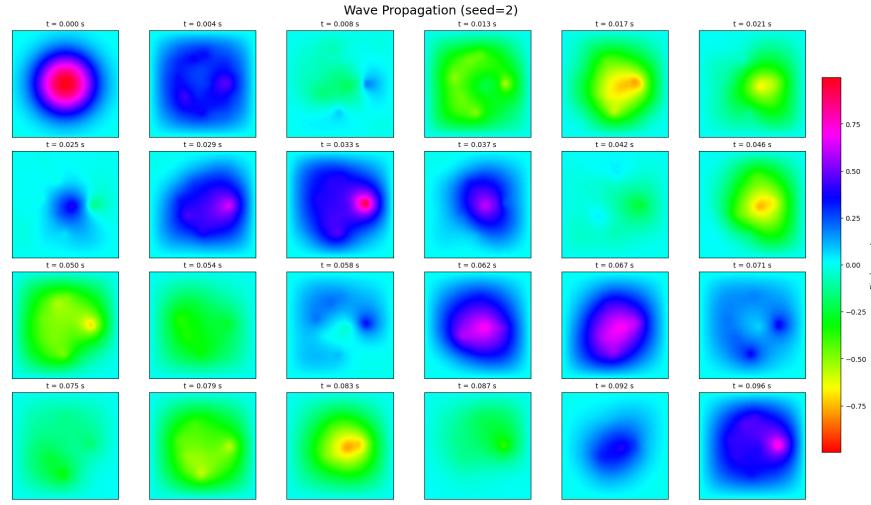


Fig. 12 Wave Propagation for 1.25 Scaled Defect Seed 2

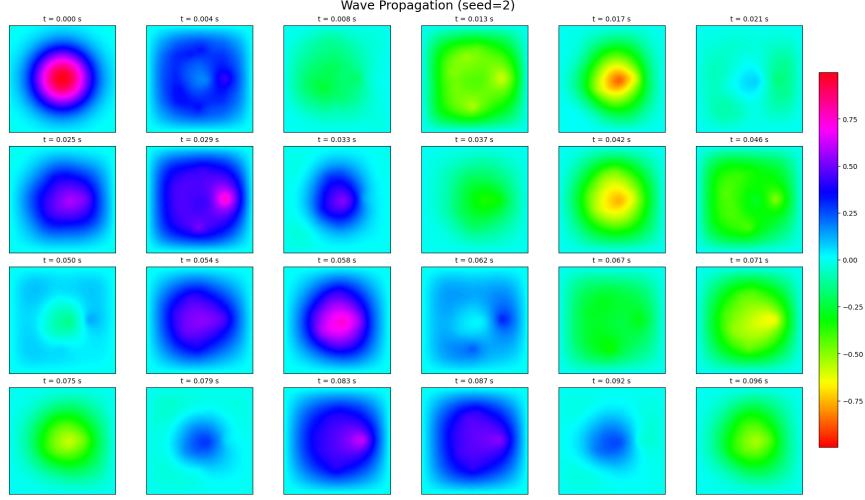


Fig. 13 Wave Propagation for 1.5 Scaled Defect Seed 2

As seen in Figure 10, Figure 11, Figure 12, and Figure 13, the speed of propagation is visible based on the change in the displacement plot for the same time-frame. For instance, the initial Gaussian bump displacement for a higher defect speed scale of 1.25 fades away in less than 0.008 s, compared to the 0.037 s for a scale speed of 0.5. Additionally for higher speeds in the defect regions, the propagation is similar to our analysis with no defects. This proves useful in further characterizing the nature of a defect and analyzing the risk it poses to the component in question. In future analysis, we can quantitatively determine the difference between the ideal and defect cases using the grid 2 norm or a more advanced technique such as a structural similarity index.

VII. Conclusion

We have developed and validated a fully implicit, second-order finite-difference solver for the two-dimensional membrane wave equation and used it to clarify how waves propagate within materials that have and don't have defects. Numerical convergence tests against an analytical single-mode solution and a high-resolution reference run confirmed the expected $O(\Delta x^2)$ spatial and $O(\Delta t^2)$ temporal accuracy, while a stability analysis demonstrated the large time-step margin offered by the Crank–Nicolson integrator.

Additionally, we have validated that this two-dimensional membrane wave can effectively detect large defects or defect clusters and that our numerical method is able to capture the impact of defects as the wave propagates in the membrane.

Although the present model neglects plate bending, anisotropy, and distributed damping, it establishes a computational framework that can be extended to more realistic shell kinematics and calibrated against real vibrometer data. In the future, we can also create a numerical model that compares the ideal and defected material waves and can pinpoint where the actual defects are.

VIII. Contributions

Alec (25%): I was responsible for writing out the problem statement and defining and deriving the 2D wave PDE. I also was responsible for writing up the numerical method and making sure that the method aligned with the one that we learned in class. I also worked alongside Vincent to create the spatial and temporal error convergence plots. I also worked on creating the displacement plot for the wing with no material defects. I also spent a very significant amount of time on our previous idea for this project, 2D Navier-Stokes to simulate nozzle flow. This ended up being too complicated and out of the scope of this project and class.

Amruth (25%): I was responsible for helping with generating meaningful questions about the system so that we could generate a more precise analysis of our numerical method as applied to the physical model. I spent time creating the defect method, considering multiple different methods of introducing defects into the study. I also spent a large portion of time attempting to develop 2 other project ideas: a nozzle louver simulation with compressible flow, and an

Euler-Bernoulli model of the SpaceX chopsticks. Part of this involved creating an Ansys simulation to compare to our method as a relative truth to generate meaningful error plots. Largely, I helped with project ideation so that we met the project requirements in a meaningful and applicable way.

Vincent (25%): I worked alongside Alec to get the correct convergence rates for spatial and temporal errors when comparing numerical versus analytical solutions. I also did some debugging to obtain correct displacement plots, as some of the initial conditions in the code were not getting enforced.

Max (25%): I worked on defining key questions and boundary conditions on multiple project ideas, and tested code on a previous project idea of simulating the Super Heavy booster catch as the bending of a Euler-Bernoulli beam. On the final project idea, I analyzed the wave propagation on a 2D material with defects. I also helped with debugging code to enforce boundary conditions, and wrote the nomenclature, introduction, and conclusion of the final report.

IX. References

[1] OpenAI, ChatGPT: GPT-4 Language Model, OpenAI, San Francisco, CA, 2025. [online], URL: <https://chat.openai.com> [retrieved 14 May 2025].

Github Link: AE370GP2_14 Github Link