

DOING PHYSICS WITH MATLAB

WAVE MOTION

THE [1D] SCALAR WAVE EQUATION THE FINITE DIFFERENCE TIME DOMAIN METHOD

Ian Cooper
School of Physics, University of Sydney
ian.cooper@sydney.edu.au

DOWNLOAD DIRECTORY FOR MATLAB SCRIPTS

The following mscripts are used to solve the scalar wave equation using the finite difference time development method. All the mscripts are essentially the same code except for differences in the initial conditions and boundary conditions. Graphical outputs and animations are produced for the solutions of the scalar wave equation.

em_swe_03.m

models the propagation of either a rectangular pulse or a Gaussian pulse

em_swe_04.m

models the superposition and interference of propagating waves

em_swe_05.m

models the propagation of traveling harmonic waves

em_swe_06.m

models standing waves

em_swe_07.m

models the reflection and transmission of a Gaussian pulse at an interface

Introduction

We will use the **finite difference time domain (FDTD) method** to find solutions of the most fundamental partial differential equation that describes wave motion, the **one-dimensional scalar wave equation**.

The [1D] scalar wave equation for waves propagating along the X axis can be expressed as

$$(1) \quad \frac{\partial^2 u(x,t)}{\partial t^2} = v^2 \frac{\partial^2 u(x,t)}{\partial x^2}$$

where $u(x,t)$ is the **wavefunction** and v is the speed of propagation of the waveform. For electromagnetic waves, the wave function represents the electric field or magnetic field; for sound waves, the wave function represents the pressure or particle displacement fluctuations; and for waves on strings, the wave function gives the string displacement.

Finite Difference time Development Method

The FDTD method can be used to solve the [1D] scalar wave equation. In the numerical solution, the wavefunction is approximated at discrete times and discrete grid positions. Let the execution time for a simulation be given by T . The time step is $\Delta t \equiv h_t$ and the number of time steps is N_t . The wavefunction $u(x,t)$ is calculate at N_x discrete positions from $x = 0$ to $x = L$ where the spacing between the grid points is $\Delta x \equiv h_x$

The n_t^{th} time interval is

$$t[n_t] = (n_t - 1)h_t \quad n_t = 1, 2, 3, \dots, N_t$$

The n_x^{th} grid position is

$$x[n_x] = (n_x - 1)h_x \quad n_x = 1, 2, 3, \dots, N_x$$

The wavefunction $u(x,t)$ at time $t[n_t]$ and grid position $x[n_x]$ is expressed in terms of the time index n_t and the spatial index n_x as $u[n_x, n_t]$

The second partial derivative of the wavefunction in space at time $t = t[n_t]$ for $2 \leq n_x \leq N_x - 1$ is approximated by the central difference formula

$$(2) \quad \frac{\partial^2 u[n_x, n_t]}{\partial x^2} = \frac{u[n_x + 1, n_t] - 2u[n_x, n_t] + u[n_x - 1, n_t]}{h_x^2}$$

The central difference approximation to the second partial time derivative of $u(x,t)$ for $2 \leq n_t \leq N_t - 1$ at the grid position $x = x[n_x]$ is

$$(3) \quad \frac{\partial^2 u[n_x, n_t]}{\partial t^2} = \frac{u[n_x, n_t + 1] - 2u[n_x, n_t] + u[n_x, n_t - 1]}{h_t^2}$$

Substituting the two central difference expressions of equations 2 and 3 into the [1D] scalar wave equation (1), we get for the latest value of the wavefunction at the grid point $x = x[n_x]$

$$(4) \quad u[n_x, n_t + 1] = 2u[n_x, n_t] - u[n_x, n_t - 1] + \left(\frac{v^2 h_t^2}{h_x^2} \right) (u[n_x + 1, n_t] - 2u[n_x, n_t] + u[n_x - 1, n_t])$$

This is a fully explicit second-order accurate expression for $u[n_x, n_t + 1]$ in that all wave quantities on the right hand side are known, that is, they were obtained during previous time steps, n_t and $n_t - 1$.

The wavefunction values have to be assigned at each grid point for the first two time steps to start the solution process given by equation 4

$$\begin{aligned}
 n_x &= 1, 2, 3, \dots, N_x \\
 n_t = 1 & \quad t = t[1] = 0 & \quad u[n_x, 1] \\
 n_t = 2 & \quad t = t[2] = h_t & \quad u[n_x, 2]
 \end{aligned}$$

Using the assigned values for the first two time steps and the specified boundary conditions, equation 4 is repeated used for all space points to calculate the solution of the scalar wave equation at all later times.

A fundamental assumption in the FDTD method is that in the calculation of the wavefunction at a given grid point we need to know the two adjacent values of the wavefunction, but at the ends of the grid, this is not possible. To solve the scalar wave equation, the wavefunction at the grid points $x[1]$ and $x[N_x]$ must be given at all times, that is, the **boundary conditions** must be specified. There are several different types of boundary conditions:

Dirichlet $u(\text{boundary}) = 0$ **Fixed end**

Neumann $\left. \frac{\partial u}{\partial x} \right|_{\text{boundary}} = 0$ **Free end**

Implementing this boundary condition at $x[1]$ and at $x[N_x]$ using a forward difference approximation for the derivative gives

$$u[1, n_t] = u[2, n_t] \quad u[N_x, n_t] = u[N_x - 1, n_t]$$

Absorbing boundary condition (ABC): This is an important type of boundary condition, for example, most electromagnetics problems involve unbounded regions, which cannot be modelled computationally. One option is to use one of the above boundary conditions and make the simulation region very large, and terminate the simulation before reflections from the boundary perturb the solution in the region of interest. The drawback of this approach is that the larger the simulation region, the greater the computational cost of the simulation. A better approach is to use a boundary condition that absorbs waves and reflects as little energy as possible.

Numerical Stability and the Magic Time Step

$$\Delta t = \frac{\Delta x}{v} \quad h_t = \frac{h_x}{v} \quad S = \frac{v \Delta t}{\Delta x} = \frac{v h_t}{h_x} = 1$$

The accuracy of the FDTD method is very dependent upon the time step h_t size and the size of the grid spacing h_x . The grid space resolution directly impact the numerical phase velocity. It is useful to define the numerical stability factor called the **Courant number** S where

$$(5) \quad S = \frac{v \Delta t}{\Delta x}$$

and the **grid sampling resolution** N_λ

$$(6) \quad N_\lambda = \frac{\lambda_0}{h_x}$$

λ_0 is the “real-world” wavelength and not the numerical wavelength in the FDTD grid. To have numerical stability the condition $S \leq 1$ must be satisfied.

For a particular choice of grid h_x the corresponding spatial sampling is

$$\lambda_{0,\min} = \frac{c}{f_{\max}} = 2v h_t$$

where f_{\max} is the maximum sampling rate and $\lambda_{0,\min}$ the minimum free-space wavelength that can be sampled without aliasing. Hence,

$$N_{\lambda,\min} = \frac{\lambda_{0,\min}}{h_x} = \frac{2v h_t}{h_x} = 2S$$

and since $S \leq 1$ then $N_{\lambda,\min} \leq 2$.

These conditions imply that in one time step h_t a wave can propagate at most one grid point h_x .

Of particular interest is the case where $S = 1$ or $\Delta t = \frac{\Delta x}{v}$ and the results of using this time step is so remarkable that it is called the **magic time step**. When the magic time step is used the numerical solution to the scalar wave equation is an **exact solution** and not an approximation. We can proceed any number of time steps of the FDTD algorithm, always calculating the exact propagating waveform.

Rectangular Pulse Propagation

We will give a number of example calculations of the propagation of pulses in a $N_x = 200$ grid point range of length $L = 100$ a.u. and grid spacing $h_x = 0.500$ a.u. The free space propagation velocity is set to $v = 10$ a.u.

Running the simulations for different Courant numbers S for a rectangular pulse, clearly demonstrates, that in the solving the [1D] scalar wave equation it is best to use the magic time step where $S = 1$.

Rectangular pulse $S = 1$ magic time-step

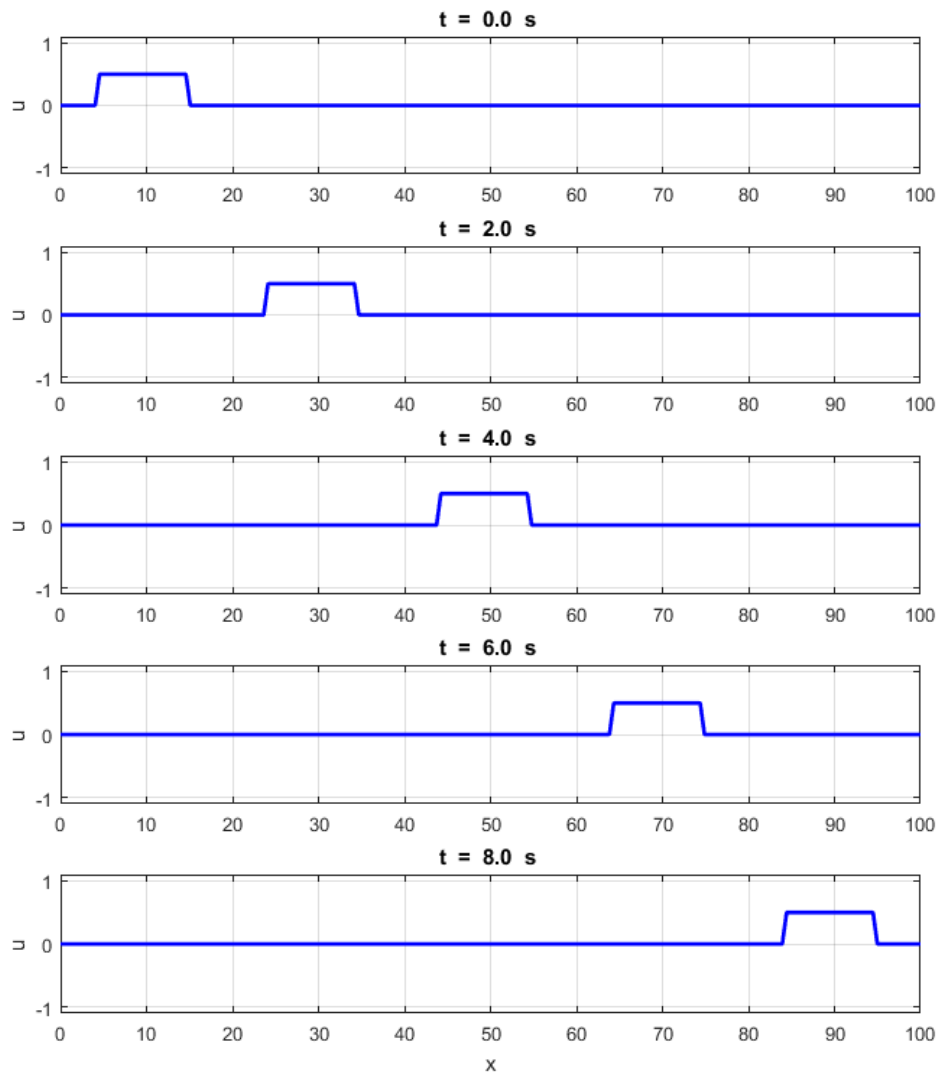


Fig. 1. Position of the rectangular pulse at fixed time intervals for $S = 1$ (magic time-step). $N_t = 160$

em_swe_03.m

Figure 1 shows the position of the pulse after fixed time intervals when the time step is set equal to the magic time step where the Courant number is $S = 1$. The shape and spatial width of the pulse are completely preserved as it propagates along the X axis. The free space velocity is $v = 10.00$ a.u. The numerical velocity v_n can be measured from the displacement of the pulse dx from $t = 0$ a.u. and at the end of the execution time $dt = 8.00$ a.u. The uncertainty in the measurement of the displacement is $h_x / 2 = 0.25$.

$$v_n = (9.99 \pm 0.02) \text{ a.u.} \quad \text{measured using Matlab Data Cursor tool}$$

The free space velocity and the numerical velocity of propagation are the same when $S = 1$ (the time step is set equal to the magic time step).

Rectangular pulse $S = 0.95$

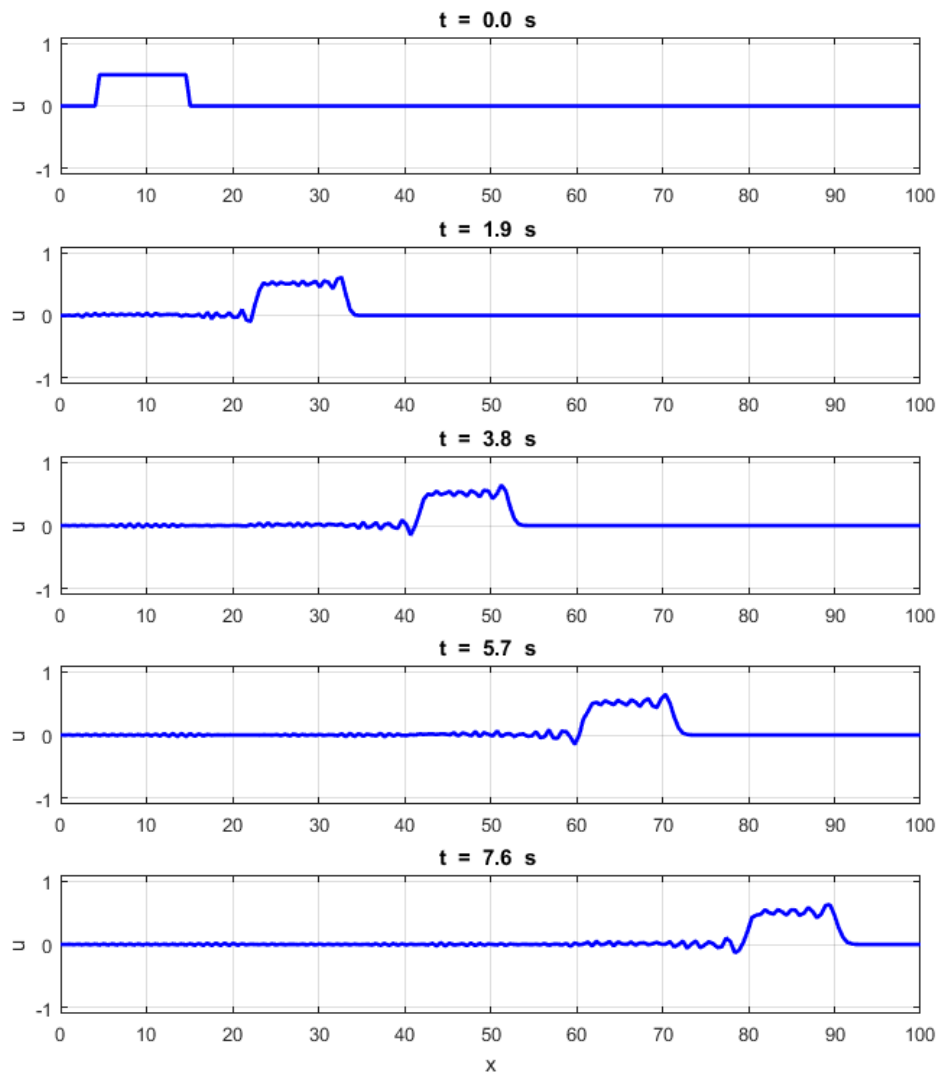


Fig. 2. Position of the rectangular pulse at fixed time intervals for $S = 0.95$. $N_t = 168$ [em_swe_03.m](#)

Figure 2 shows the propagation of the pulse when $S = 0.95 < 1$. There is a ringing effect at the leading and trailing edges of the pulse. The numerical velocity is $v_n = (9.93 \pm 0.02) < v$. The ringing represents time retarded propagation of the sparsely, high frequency spectral components of the discontinuities at the steps of the rectangular pulse.

Rectangular pulse $S = 0.70$

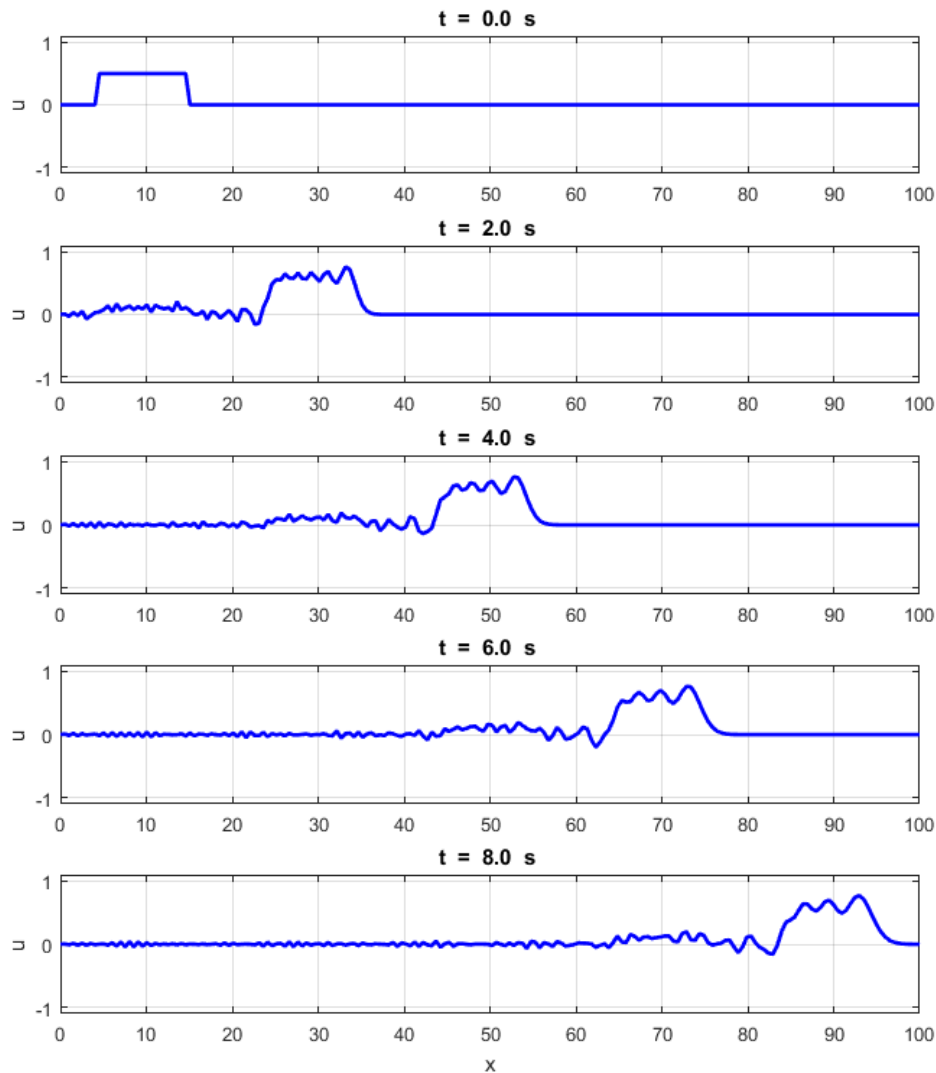


Fig. 3. Position of the rectangular pulse at fixed time intervals for $S = 0.70$. $N_t = 230$ [em_swe_03.m](#)

If you carefully compare figures 1 and 3, there is a weak **superluminal response** ($v_n > v$) that propagates just ahead of the leading edge of the rectangular pulse. After the time interval of $dt = 8.00$ a.u. the leading edge of the pulse for $S = 1.00$ is at $x \approx 95$ a.u. whereas for the pulse for $S = 0.7$, the position of the leading edge is at $x \approx 100$ a.u.

In the numerical solution for $S < 1$ some of the spectral components of the pulse travel at a speed slower than v whilst other spectral components travel faster than v . As well as the ringing effect, there is broadening of the pulse as it propagates.

Gaussian pulse propagation

We can study the propagation of pulses which have an initial Gaussian spatial profile using the magic time step. For the magic time step, the pulse advances one grid space in one time step.

The Gaussian pulse is specified in the Matlab mscript as a function of the spatial index n_x . The input parameters for the Gaussian pulse are

```
% Gaussian pulse
A = 0.8;           % pulse amplitude
s = 5;            % pulse width
np = 20;          % pulse starting point given by index np
```

Gaussian profile is specified at time steps 1 and 2

```
if flag == 1      % Gaussian pulse: 1st & 2nd time steps
  for nx = 2 : Nx-1 % time step nt = 1
    u(nx,1) = A .* exp(-0.5 .* ((nx-np)./s).^2);
  end

  for nx = 2 : Nx - 1 % time step 2
    u(nx,2) = A .* exp(-0.5 .* ((nx-np-1)./s).^2);
  end
end
```

When the magic time step is used the Gaussian profile and speed of propagation are preserved as shown in figure 4.

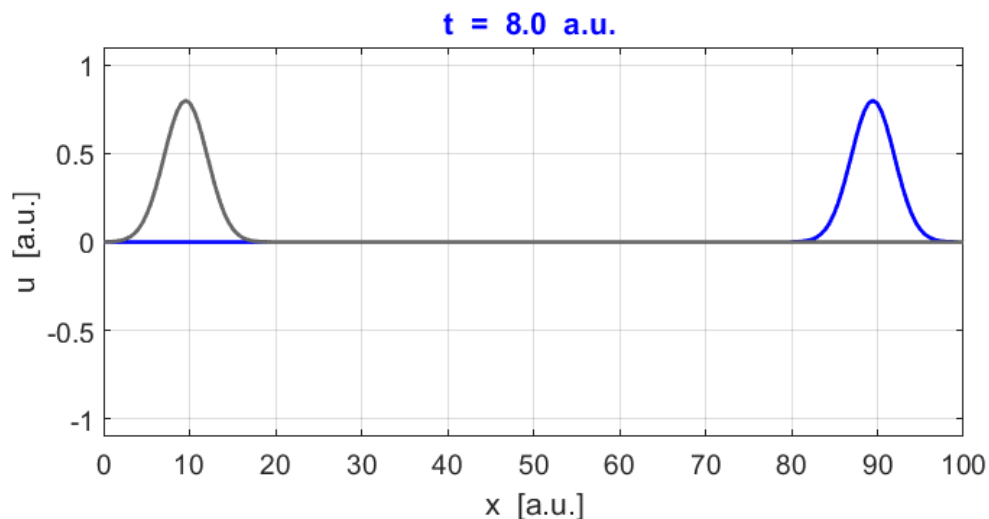


Fig. 4. Propagation of a Gaussian pulse. The pulse starts at position $x = 10$ a.u. at time $t = 0$ and finishes at position $x = 90$ a.u. after a simulation time of $t = 8.0$ a.u.

[em_swe_03.m](#)

The numerical speed of the pulse can be calculated in the Matlab Command Window using logical functions

$$v_n = \frac{dx}{dt} \quad dx = \text{distance travelled in the time interval } dt$$

```
>> (x(u(:,end) == max(u(:,end))) - x(u(:,1) == max(u(:,1))))/8
ans = 9.9874
```

The free space wave speed is $v = 10$ a.u. and its value agrees with the numerical wave speed $v_n = (9.99 \pm 0.02)$ a.u.

Boundary Conditions

We can use the Gaussian pulse to investigate the behaviour of the pulse for different boundary conditions. The same boundary conditions apply at each time step for $n_t = 1$ to $n_t = N_t - 1$

Dirichlet boundary conditions $u(\text{boundary}) = 0$

```
u(1, nt+1) = 0; % Fixed end
u(Nx, nt+1) = 0; % Fixed end
```

Neumann boundary conditions $\left. \frac{\partial u}{\partial x} \right|_{\text{boundary}} = 0$

```
u(Nx, nt+1) = u(Nx-1, nt+1); % Free end
u(1, nt+1) = u(2, nt+1); % Free end
```

Absorbing boundary conditions ABC

Using the magic time step, it takes one time step for the wave to propagate from one grid position to the next. Hence, to apply absorbing boundary conditions at the ends of the simulation space, the values of the wavefunction at the boundaries are set to the adjacent wavefunction values one time step earlier.

```
u(Nx, nt+1) = u(Nx-1, nt); % ABC
u(1, nt+1) = u(2, nt); % ABC
```

Figures 5 and 6 show the graphical output for a number of simulations of a pulse being reflected from the boundaries of the simulation region.

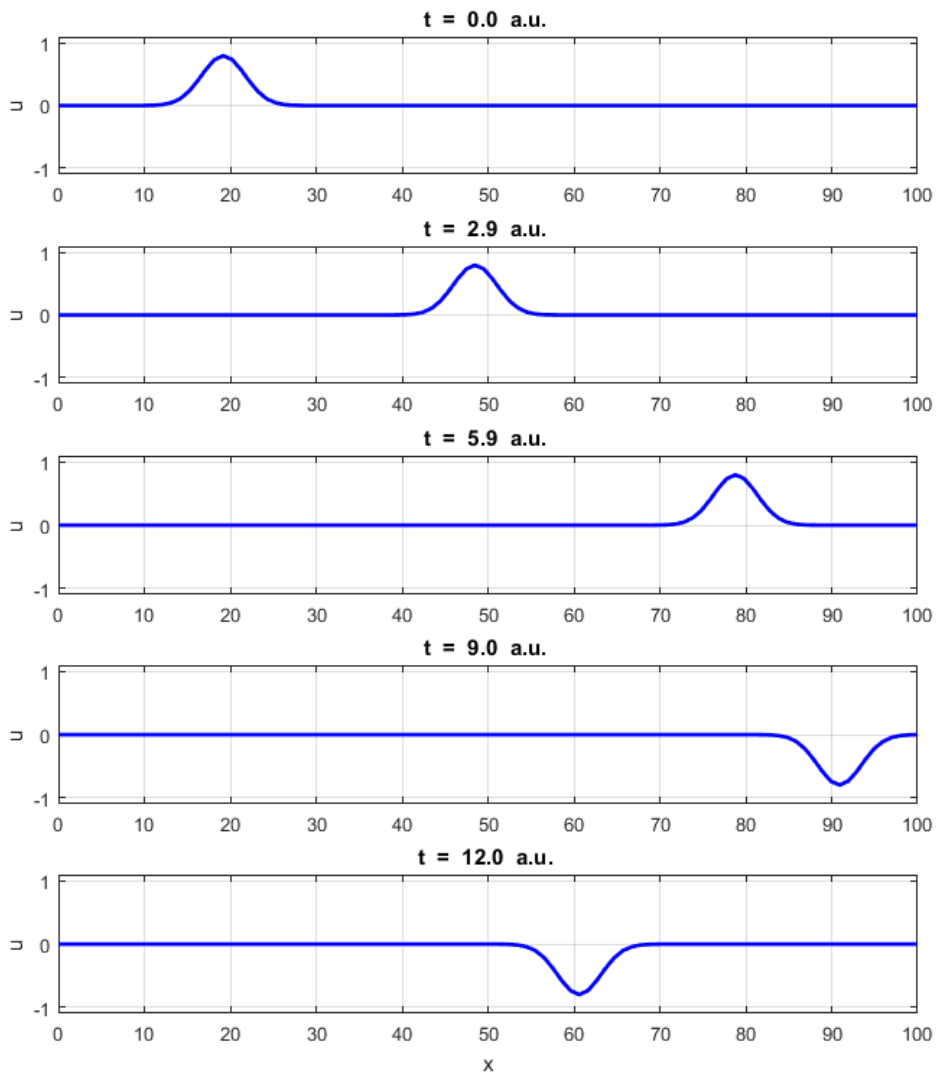


Fig. 5. A Gaussian pulse reflected a fixed end termination $u(\text{boundary}) = 0$. The reflected pulse is inverted because of a π change in phase upon reflection. [em_swe_03.m](#)

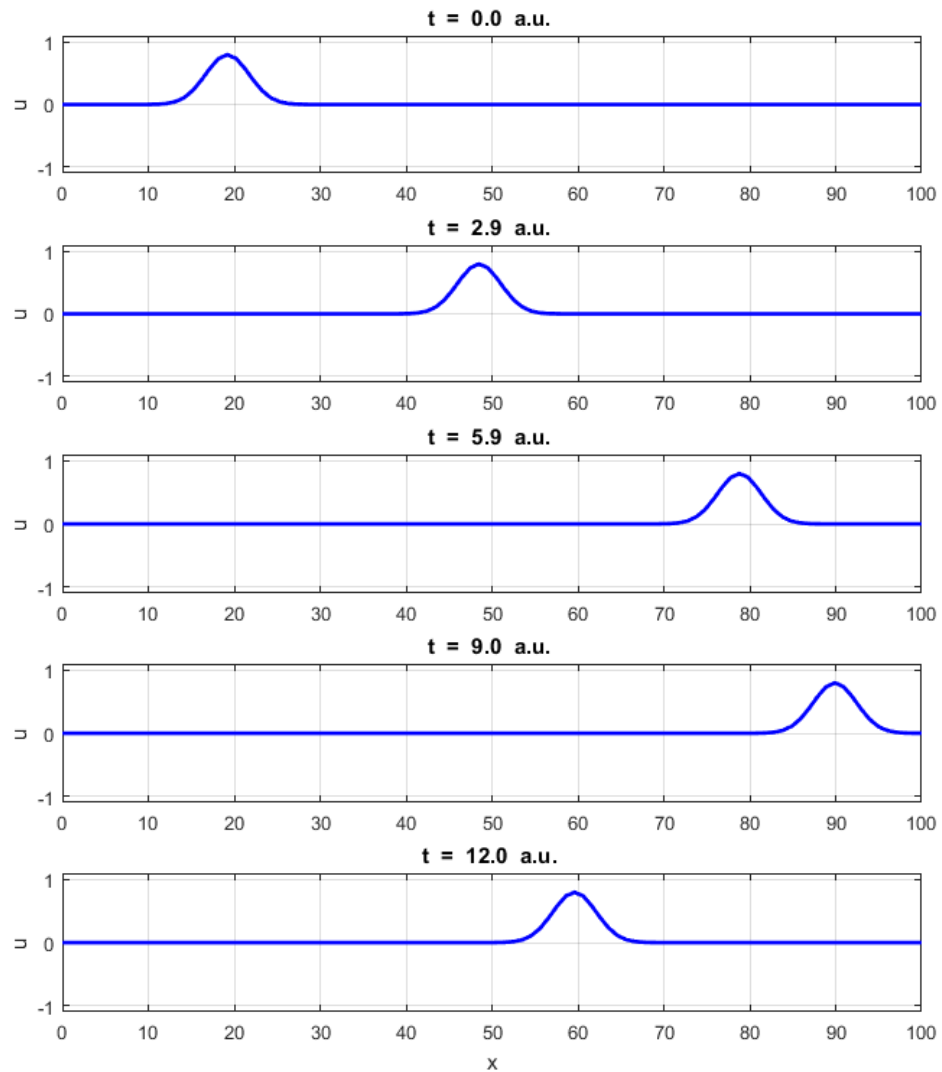


Fig. 6. A Gaussian pulse reflected a free end termination

$\left. \frac{\partial u}{\partial x} \right|_{\text{boundary}} = 0$. The reflected pulse is upright because of a zero change in phase upon reflection. [em_swe_03.m](#)

You can view a simulation of a pulse of being reflected from the end $x = L$ which is a free end and from the end $x = 0$ which is a fixed end.

[view animation](#)

Superposition Principle: Interference

We can model the superposition of waves by considering two pulses moving in opposite directions. If one pulse is given by the wavefunction $u_1(x,t)$ and the other pulse by $u_2(x,t)$ and both wavefunctions are solutions of the scalar wave equation (equation 1) then using the property of **linearity**, the sum of any number of solutions is also a solution. This fact is called the **superposition principle**. Hence, when two or waves combine, the resultant wave is the algebraic sum of the individual waves.

The total function $u(x,t)$ for our two pulses is simply the sum of the two individual wavefunctions

$$u(x,t) = u_1(x,t) + u_2(x,t)$$

The superposition principle vastly increases the number of waveforms that waves can take and gives rise to a rich palette of patterns called **interference**. When the two pulses add to give a maximum amplitude of the result waveform, the waves are said to **interfere constructively**. **Destructive interference** occurs when the amplitude of the waves cancel each other giving a minimum amplitude of the resultant waveform.

Figure 7 shows the two pulses interfering constructively as they pass through each other and figure 8 shows destructive interference. Notice that the pulse simply pass through each other.

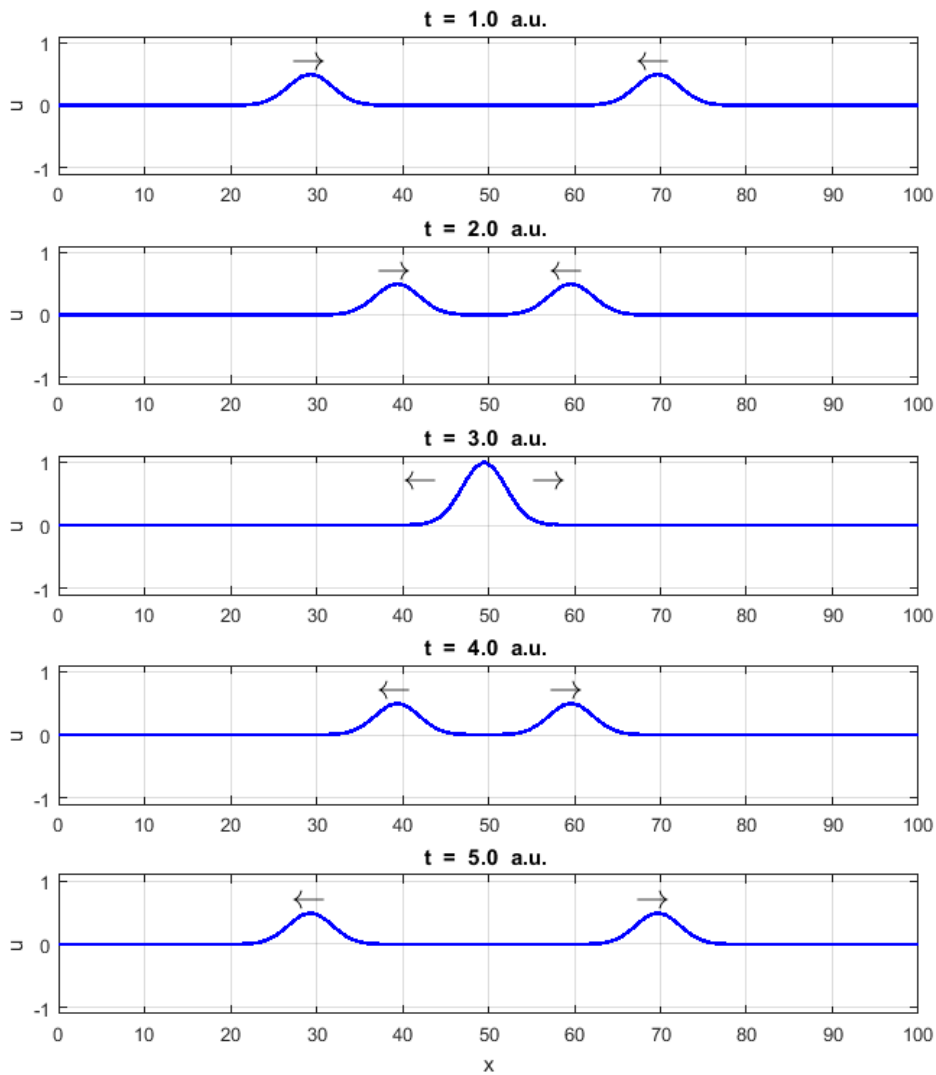


Fig. 7. Two pulses traveling in opposite directions interfere with each other. At time $t = 3.0$ a.u. the pulses interfere constructively, the resultant waveform has a maximum amplitude. [em_swe_04.m](#)

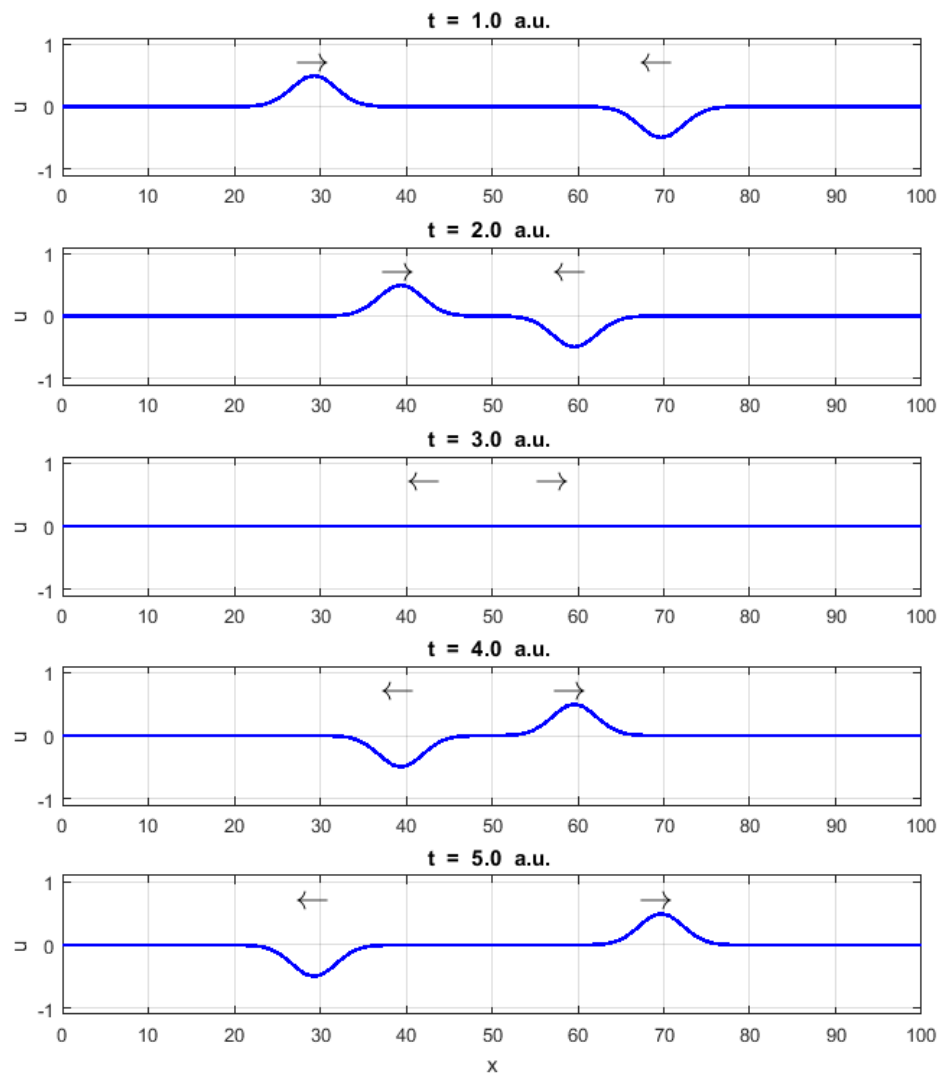


Fig. 8. Two pulses travelling in opposite directions interfere with each other. At time $t = 3.0$ a.u. the pulses interfere destructively and the resultant waveform has a minimum amplitude. [em_swe_04.m](#)

The wavefunction $u(x,t)$ is a function of both position and time. Figures 7 and 8 shows the position at the pulse at fixed times. However, Matlab can be used to create animated gifs. You can view an animation of colliding pulses which exhibit both constructive and destructive interference. The animation gives you a better picture of the interference phenomenon.

[view animation](#)

Traveling waves

We can model harmonic waves propagating along the simulation space. At $x = 0$, a source point is used to set up the traveling wave which moves from the left to the right

$$u(x=0,t) = A \sin\left(\frac{2\pi t}{T}\right) \quad \text{sinusoidal source at } x = 0$$

At the end of the simulation region, $x = L$, we need to employ absorbing boundary conditions, otherwise, a reflected wave traveling from right to left would interfere with the incident waveform traveling left to right. Once the sinusoidal wave has been established, each grid point moves up and down with simple harmonic motion and the speed of the wave is the free space speed v .

Figure 9 shows the waveform at fixed time intervals for a wave with the source parameters $A = 1$, $T = 2.5$ a.u. and $v = 10$ a.u. The theoretical wavelength of the traveling wave is $\lambda = vT = 25$ a.u. From figure 9 you can estimate the speed of the wave, the wavelength and then calculate the period of the wave using the Matlab Data Cursor tool

Numerical velocity	$v_n = (10.1 \pm 0.1)$ a.u.
Numerical wavelength	$\lambda_n = (25.3 \pm 0.5)$ a.u.
Numerical period	$T_n = (2.51 \pm 0.05)$ a.u.

The numerical values agree well with the specified values within the uncertainties in making the measurements on the course grid.

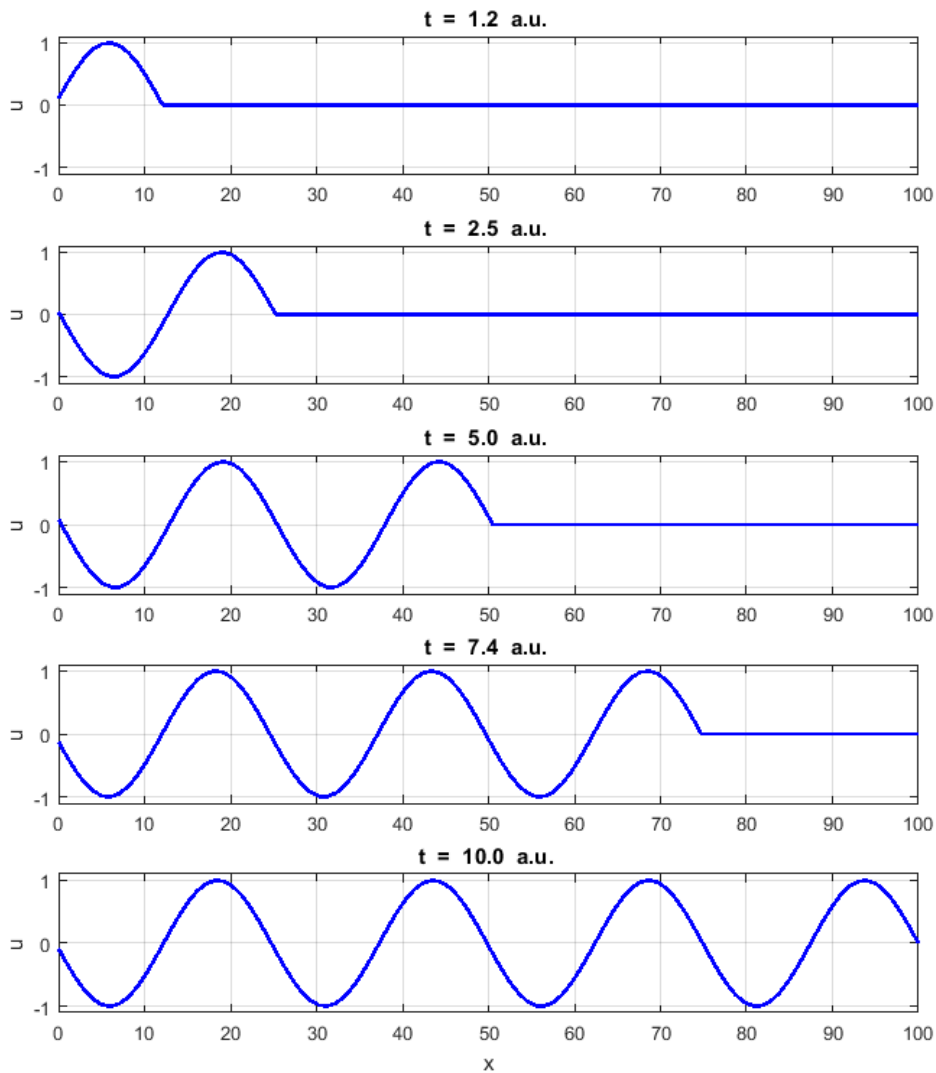


Fig. 9. Harmonic wave at fixed time intervals due to a sinusoidal point source located at $x = 0$. Absorbing boundary conditions are employed at $x = L$ so that there is no reflected wave at the boundary. [em_swe_05.m](#)

[view animation](#)

Standing Waves

We can model the behaviour of waves confined in [1D] space by finding the wavefunction that is a solution to the scalar wave equation. In our model no energy is dissipated from the vibrating system. We start with the wavefunction being zero at all grid point $u(x,0) = 0$ and a small amplitude sinusoidal point source applied to the boundary at $x = 0$ (origin). The wavefunction is set to zero at the boundary $x = L$.

Boundary conditions

$$u(L,t) = 0.$$

$$u(0,t) = A \sin\left(\frac{2\pi t}{T}\right) \quad \text{sinusoidal source at } x = 0$$

A node occurs when the value of the wavefunction is zero at all times. A true node exists in our model at $x = L$ and provided the amplitude A is small than at $x = 0$ we can assume the boundary condition approximates a node and behaves as a fixed end. Therefore, the wave will be confined to the region from $x = 0$ to $x = L$ since reflections occur at both boundaries.

For our model we will use the input parameters (all units are arbitrary)

Length of confined space $L = 100$

Number of grid points $N_x = 200$

Number of time steps N_t variable

Speed of wave $v = 10$

Courant number $S = 1$ (use magic time step)

Sinusoidal point source located at the origin

Amplitude $A = 0.1$

Period T variable

Boundary condition at $x = L$ is $u(L,t) = 0$ (fixed end)

Wavelength $\lambda = v / T$

The sinusoidal vibration of the source at the origin initiates a disturbance that travels at a speed v from the left to the right towards the boundary at $x = L$. When the traveling sinusoidal wave reaches this boundary, the wave is reflected. The wave is then reflected from the boundary at the origin. The waves are now confined in space with a node at each end and travel left to right and right to left interfering with each other according to the superposition principle. The sinusoidal source at $x = 0$ is continually adding energy to the wave and energy is returned to the source from the wave moving from right to left. A wave is built-up that vibrates at the frequency of the source. At most frequencies very little energy is added from the source to the wave.

However, at certain frequencies of vibration of the source, energy is very efficiently added to the wave and large amplitude vibrations are created and give a stationary vibration pattern called a **stationary wave** or **standing wave**. The frequencies at which the confined wave can vibrate are called the **natural frequencies** of vibration. When a system is disturbed, it tends to vibrate in a mode that is a mixture of its natural frequencies of vibration. The source frequency is called the **driving frequency**. When the driving frequency matches a natural frequency, large amplitude vibrations can occur. This phenomenon is called **resonance** and the frequencies are often referred to as the **resonance frequency**.

The time taken for the disturbance to travel from the origin and return is $T_1 = 2L / v$. Just as the disturbance returns to the origin, the source produces a disturbance that reinforces the wave and energy is therefore efficiently transferred from the source to the wave. If the source vibrates sinusoidally with a period equal to $2L / v$ then energy will be continually added to the wave building up its amplitude. The natural frequency for this mode of vibration is called the **first harmonic** or the **fundamental frequency**.

$$f_1 = \frac{1}{T_1} = \frac{v}{2L}$$

If the source driving frequency $f_n = n f_1$ $n = 1, 2, 3, \dots$ then the rate at which energy is transferred from the source to the wave is maximized and large amplitude vibrations will occur. n is referred to as the **mode number**, for example, if $n = 2$, then f_2 is called the **2nd harmonic**.

When a stationary waveform pattern is established in the confined space with a node at each end $u(0,t) = u(L,t) = 0$, the waveform is characterised by a sequence of equally spaced nodes ($u = 0$ zero amplitude) and antinodes ($u = u_{\max}$ maximum amplitude) at fixed positions in the confined space. The positions of the nodes and antinodes do not vary with time. The spacing between adjacent nodes or adjacent antinodes is equal half-wavelength

$$\lambda_n = \frac{2L}{n} \quad n = 1, 2, 3, \dots$$

For our specified parameters the resonances modes for the wave confined in space for the first five harmonics are

$$\begin{array}{llll} n = 1 & T_1 = 20 & f_1 = 1 / 20 & \lambda_1 = 200 \\ n = 2 & T_2 = 10 & f_2 = 2 / 20 & \lambda_2 = 100 \\ n = 3 & T_3 = 6.7 & f_3 = 3 / 20 & \lambda_3 = 66.7 \\ n = 4 & T_4 = 5 & f_4 = 4 / 20 & \lambda_4 = 50 \\ n = 5 & T_5 = 2.5 & f_5 = 5 / 20 & \lambda_5 = 40 \end{array}$$

Figure 10 shows the time evolution of the wavefunction when the source frequency is not equal to a natural frequency. Only a small amplitude wave is created and there are no nodes or antinodes.

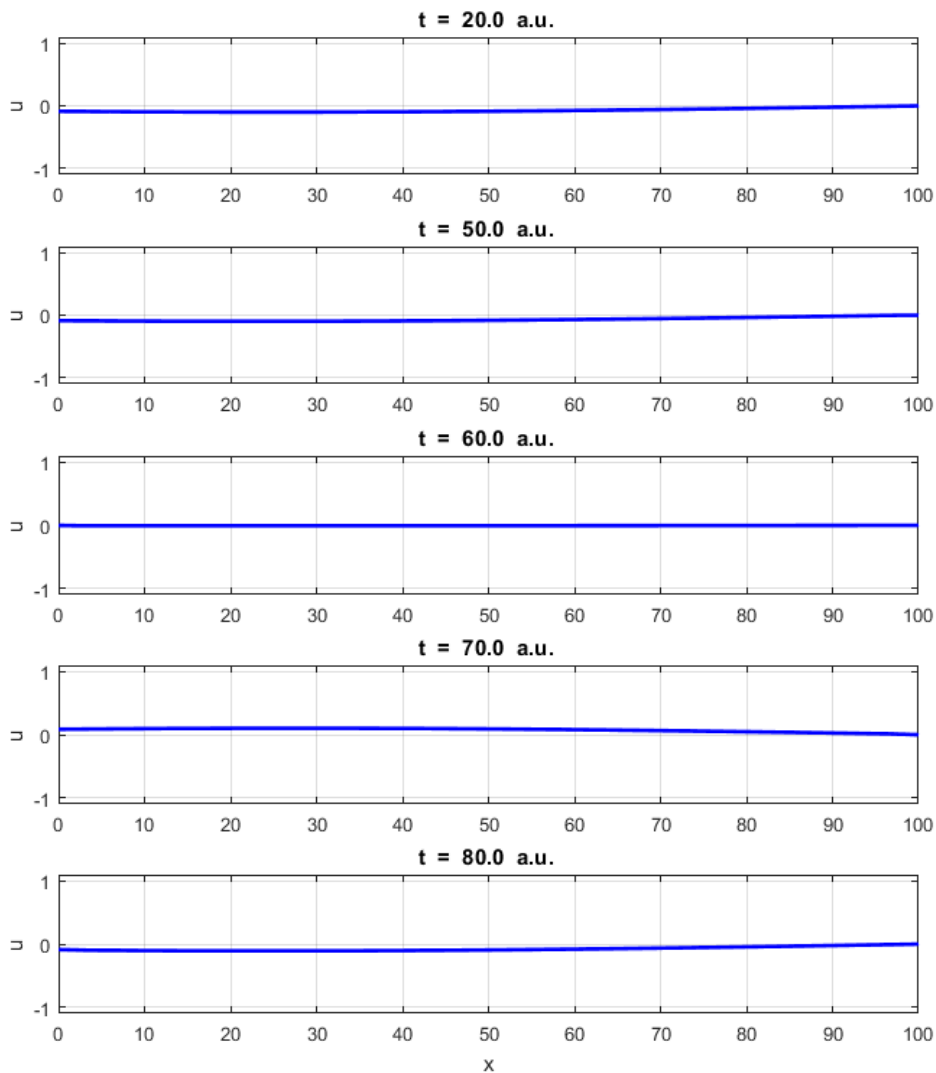


Fig. 10. The driving frequency of the source is not equal to a natural frequency. Only a small amplitude wave is created that oscillates at the driving frequency. [em_swe_06.m](#)

Figures 11 to 15 shows the time evolution of the wavefunction as a function of position in the confined space for the first five harmonics. The amplitude of the wave increases with time as energy is continually transferred from the source to the wave. For each harmonic that is excited, a standing wave pattern is established with the sequence node, antinode, node, ... , antinode, node. The location of the nodes and antinodes varies slightly with time because there is not a true node at the origin ($x = 0$).

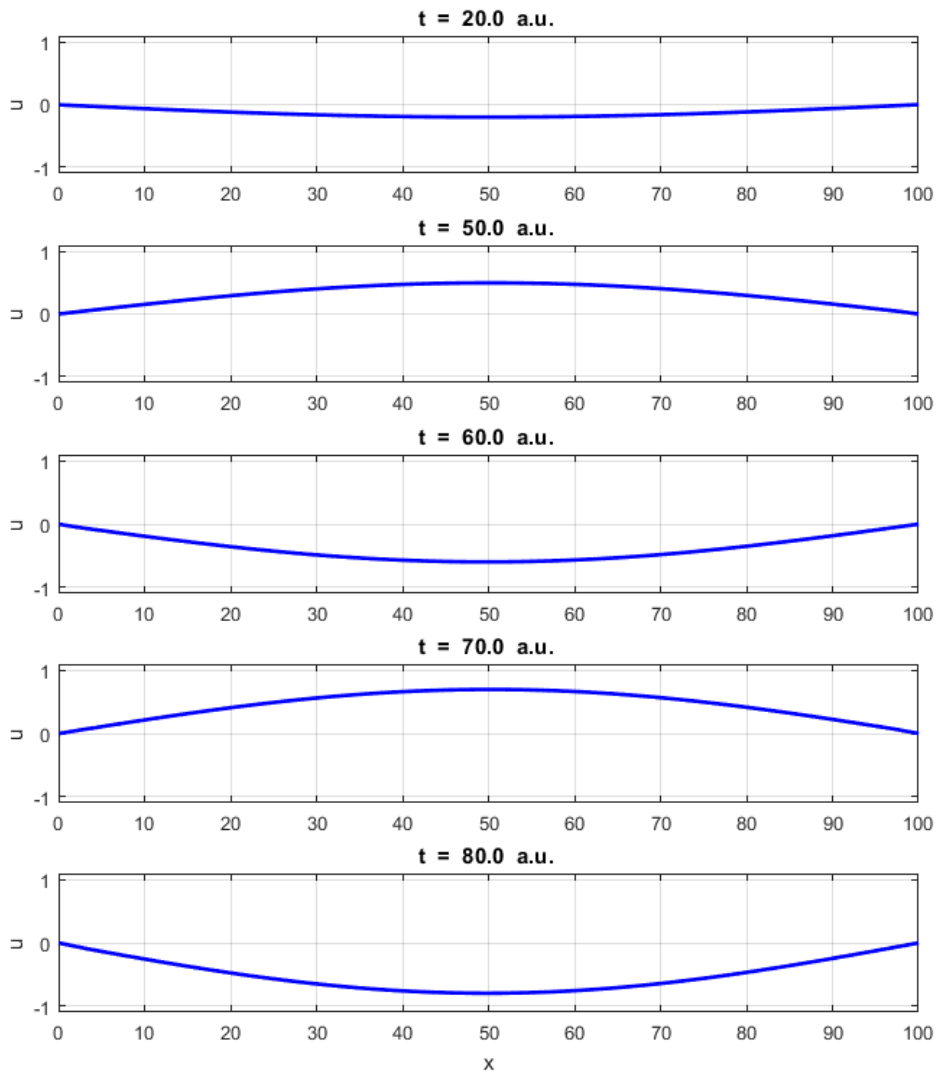


Fig. 11. Fundamental or 1st harmonic mode.

$$n = 1 \quad T_1 = 20 \quad f_1 = 1/20 \quad \lambda_1 = 200 \quad \text{em_swe_06.m}$$

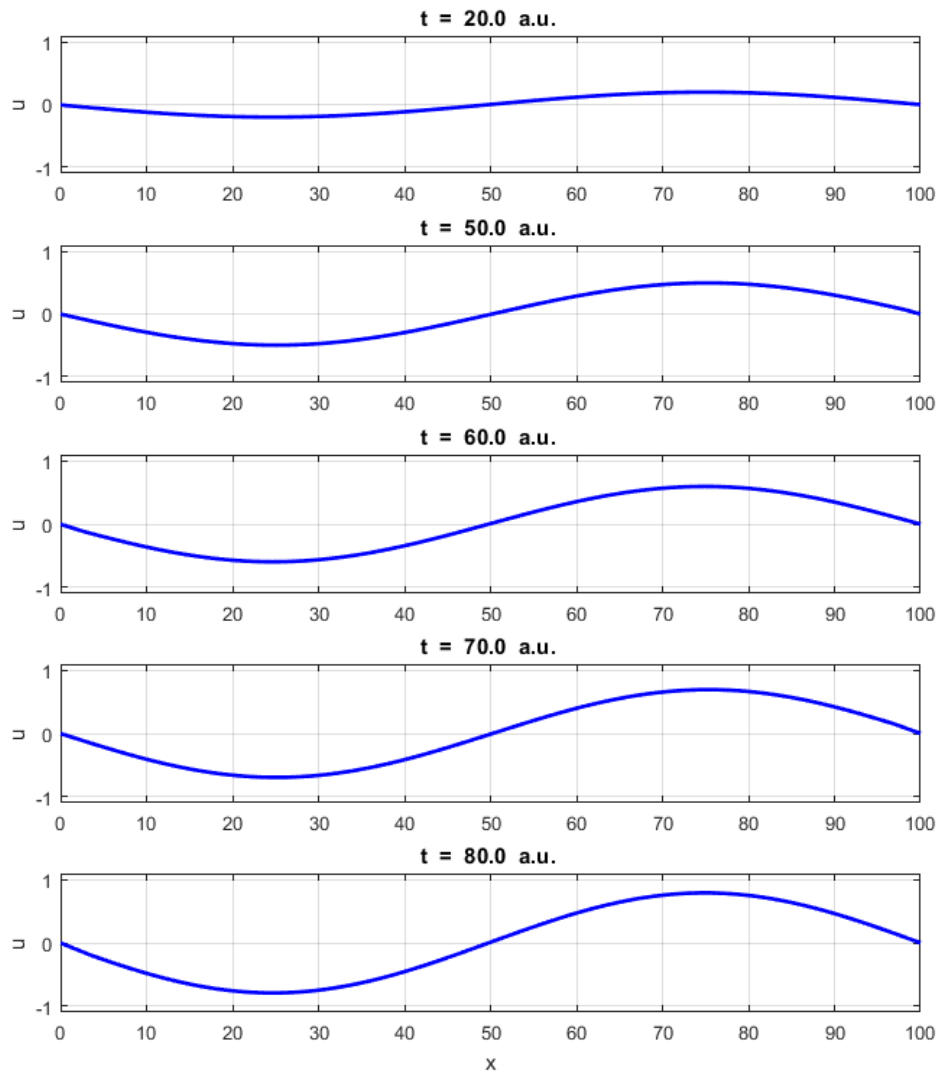


Fig. 12. 2nd harmonic mode.

$$n = 2 \quad T_2 = 10 \quad f_2 = 2/20 \quad \lambda_2 = 100$$

em_swe_06.m

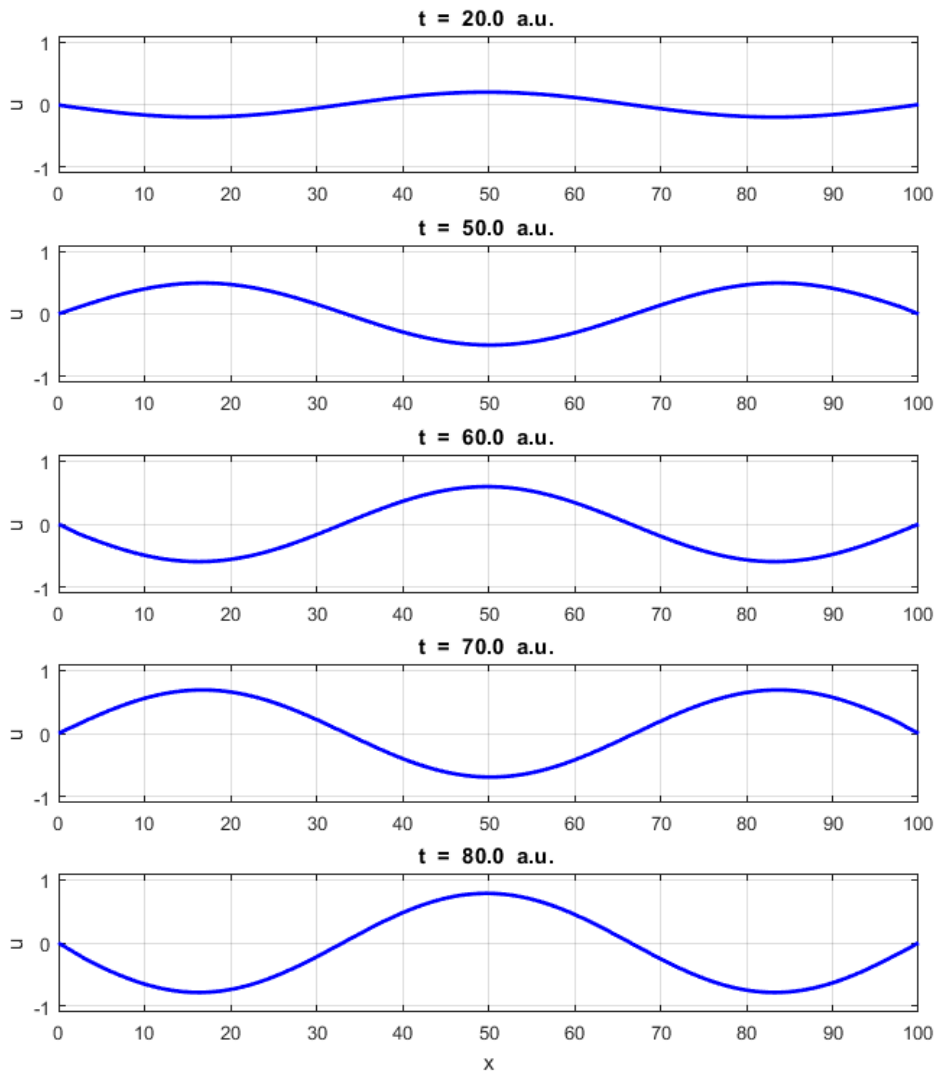


Fig. 13. 3rd harmonic mode.

$$n = 3 \quad T_3 = 6.7 \quad f_3 = 3/20 \quad \lambda_3 = 66.7 \quad \text{em_swe_06.m}$$

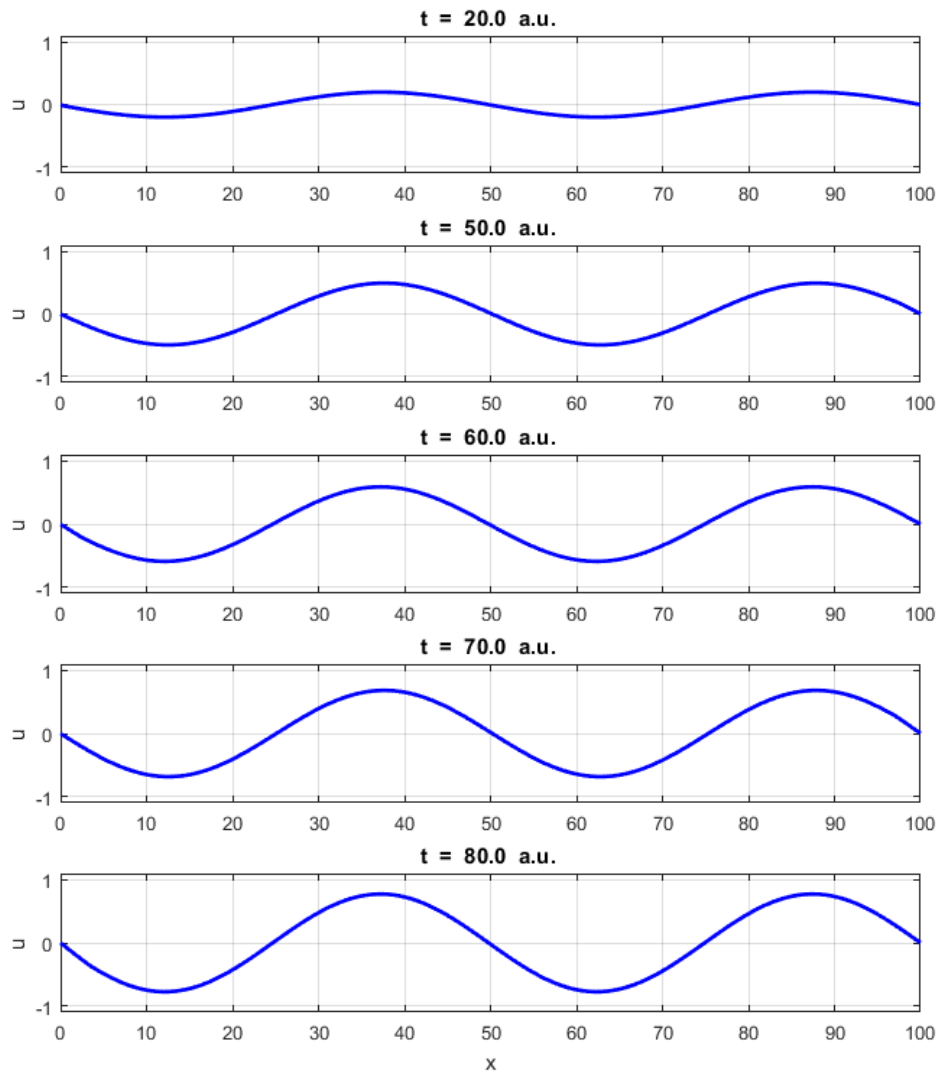


Fig. 14. 4th harmonic mode.

$$n = 4 \quad T_4 = 5 \quad f_4 = 4 / 20 \quad \lambda_4 = 50 \quad \text{em_swe_06.m}$$

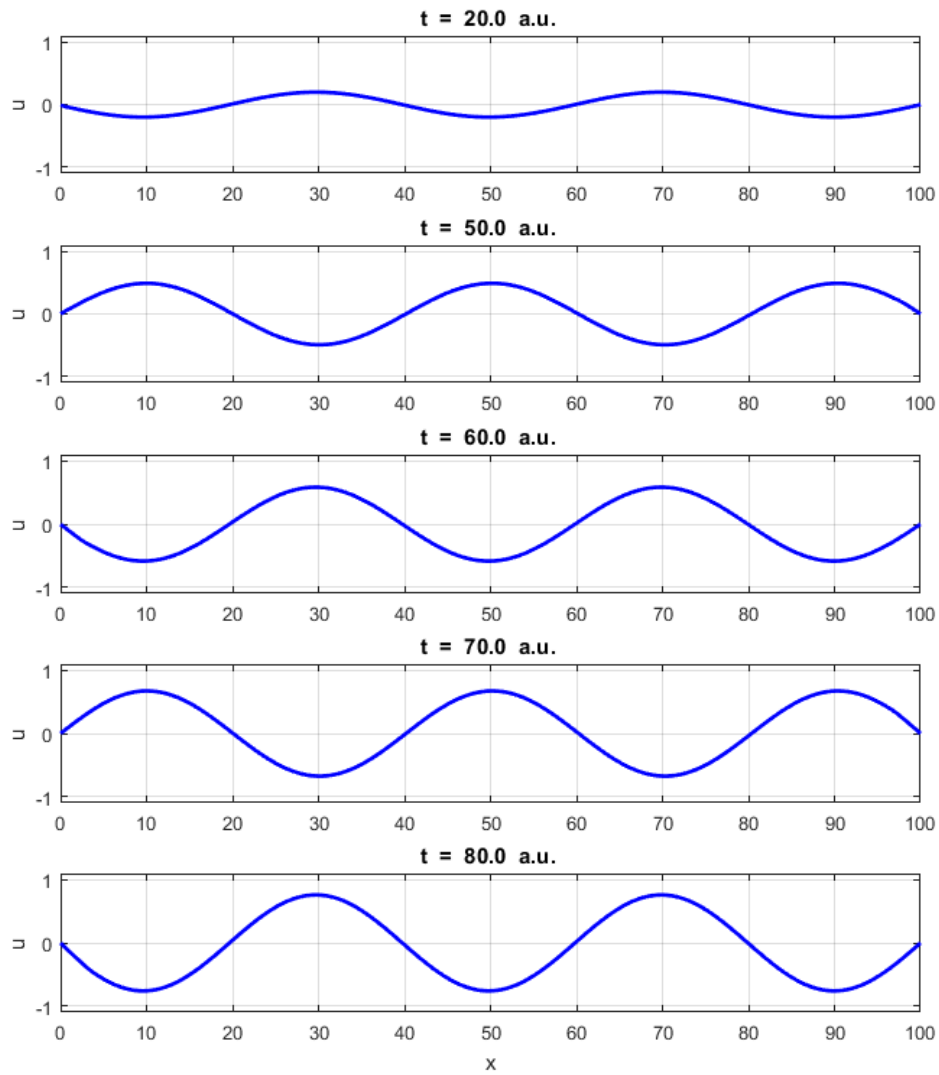


Fig. 15. 5th harmonic mode.

$$n = 5 \quad T_5 = 2.5 \quad f_5 = 5 / 20 \quad \lambda_5 = 40 \quad \text{em_swe_06.m}$$

We can change the boundary conditions so there is always an antinode at $x = L$. The wave is still confined in the region from $x = 0$ to $x = L$ because of the reflections at the boundaries. When the source frequency matches a natural frequency, large amplitude oscillations are created.

Boundary conditions

$$x = 0 \quad u(0, t) = A \sin\left(\frac{2\pi t}{T}\right) \quad A \text{ small} \rightarrow u(0, t) \approx 0 \Rightarrow \text{node}$$

$$\left.\frac{\partial u}{\partial x}\right|_{x=L} = 0 \Rightarrow \text{antinode}$$

Normal modes $n = 1, 2, 3, \dots$

$$\lambda_{2n-1} = \frac{4L}{2n-1} \quad f_{2n-1} = \frac{(2n-1)v}{4L} \quad T_{2n-1} = \frac{4L}{(2n-1)v}$$

$n = 1$ Fundamental or 1st harmonic

Only the odd harmonics are natural frequencies f_1, f_3, f_5, \dots

$$f_{2n-1} = (2n-1) f_1$$

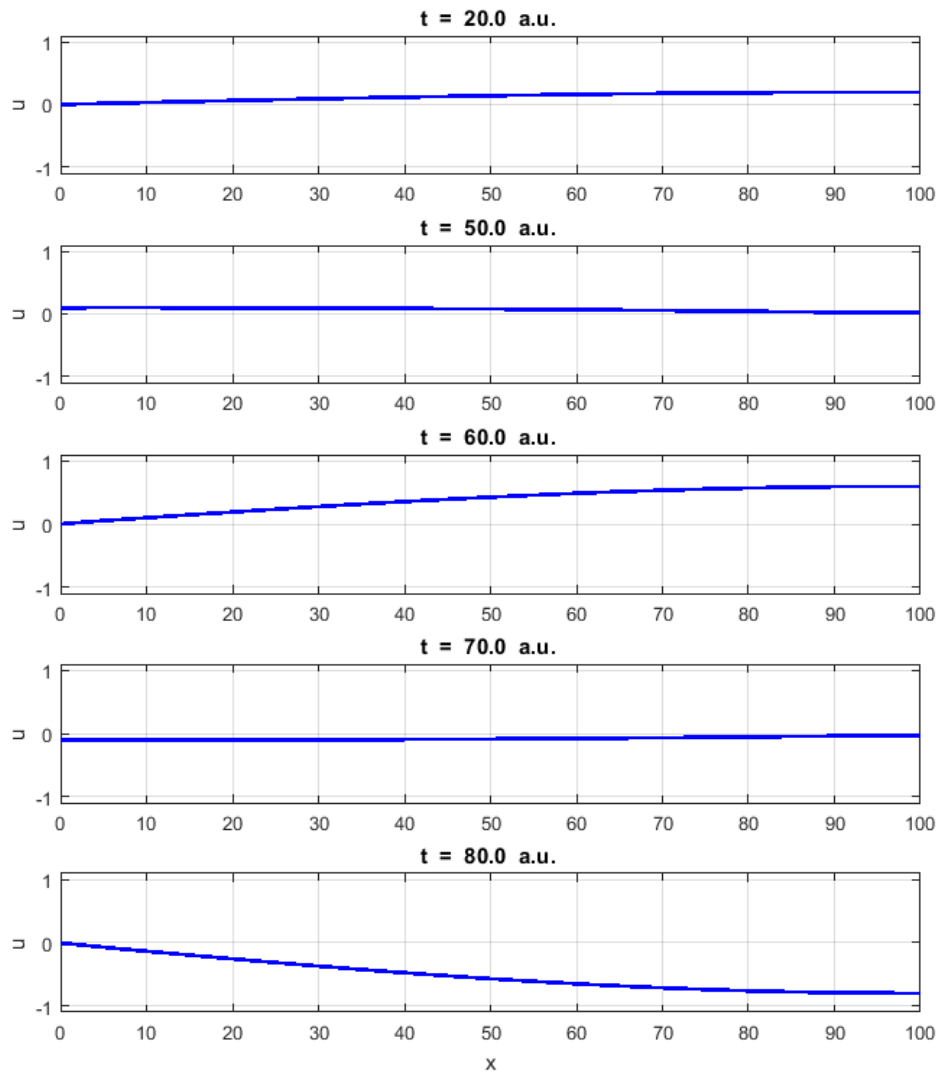


Fig. 16. Fundamental or 1st harmonic

$$n = 1 \quad T_1 = 40 \quad f_1 = 1/40 \quad \lambda_1 = 400 \quad \text{em_swe_06.m}$$

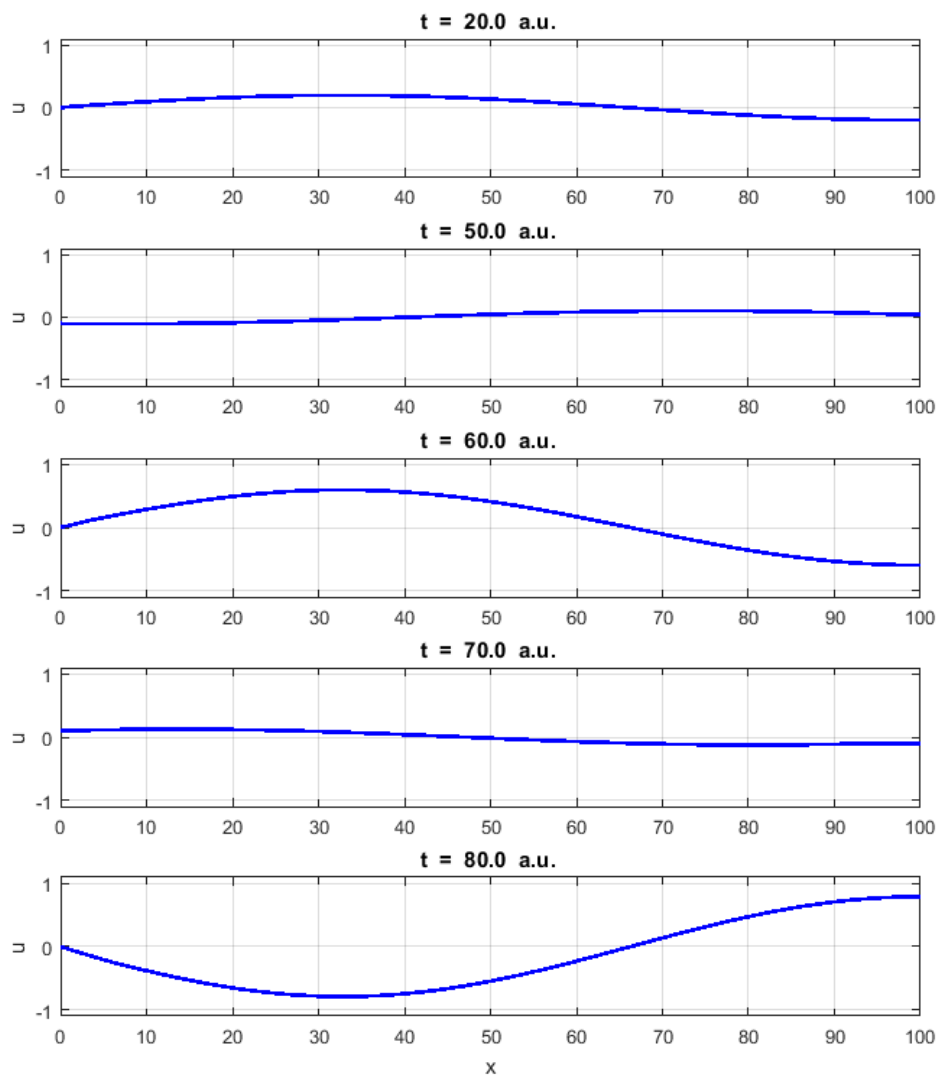


Fig. 17. 3rd harmonic

$$n = 2 \quad T_3 = 40/3 = 13.33 \quad f_3 = 3/40 = 0.0750 \quad \lambda_3 = 133.33$$

em_swe_06.m

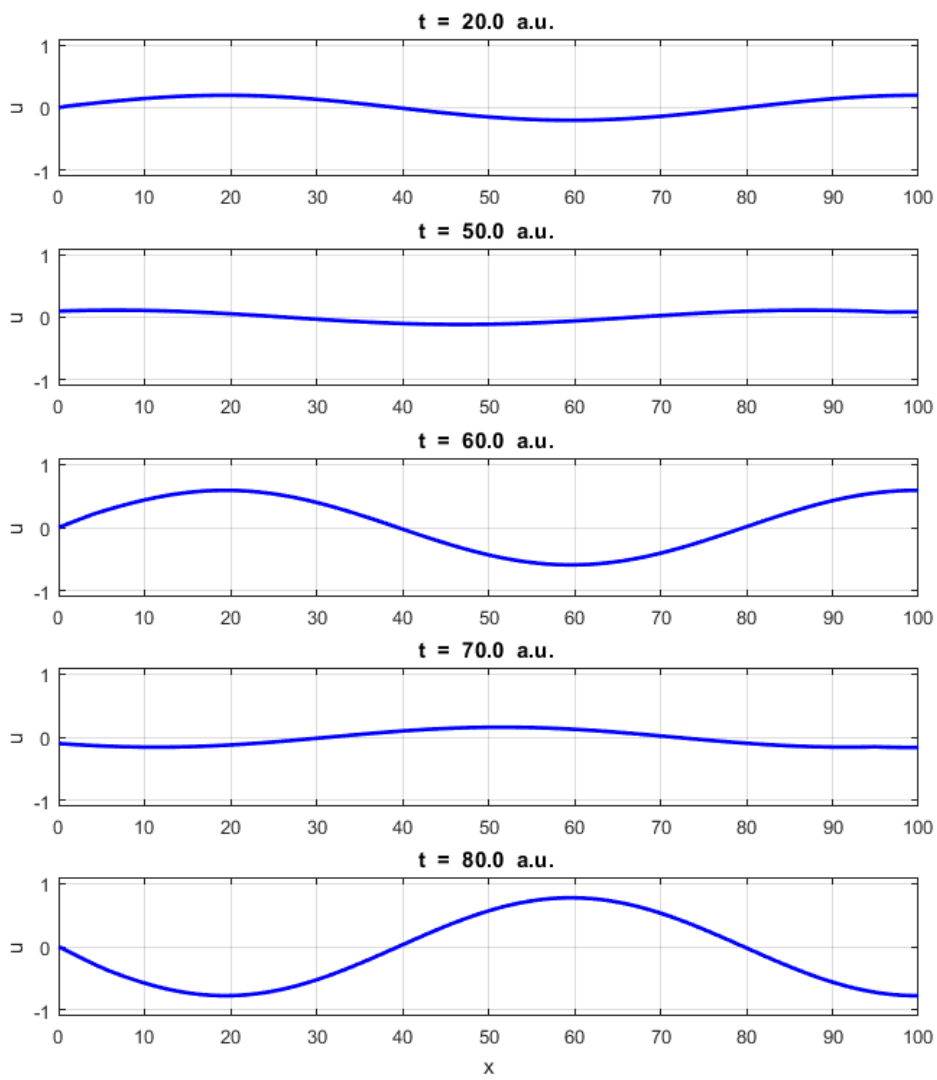


Fig. 18. 5th harmonic

$$n = 3 \quad T_5 = 40 / 5 = 8.000 \quad f_5 = 5 / 40 = 0.1250 \quad \lambda_5 = 80.00$$

em_swe_06.m

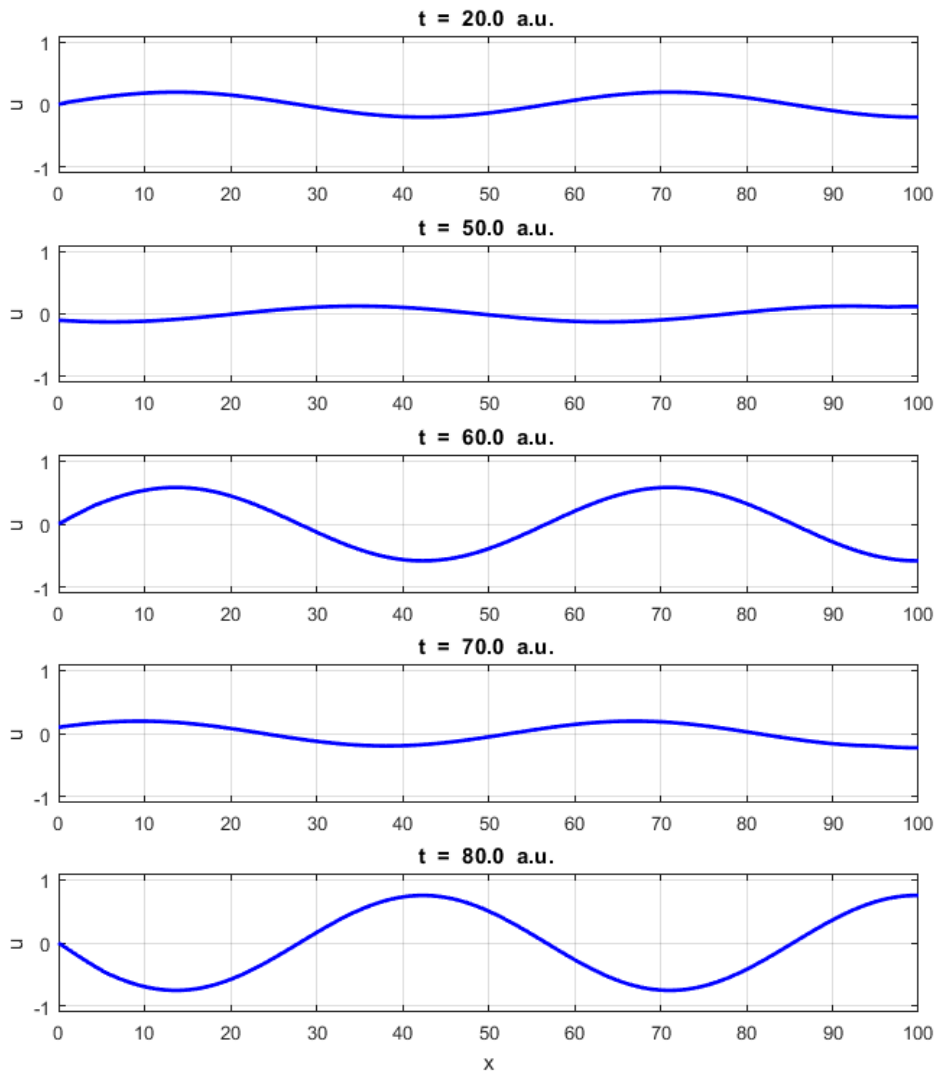


Fig. 19. 7th harmonic

$$n = 4 \quad T_7 = 40 / 7 = 5.714 \quad f_7 = 7 / 40 = 0.1750 \quad \lambda_7 = 57.14$$

em_swe_06.m

[view animations of standing waves](#)

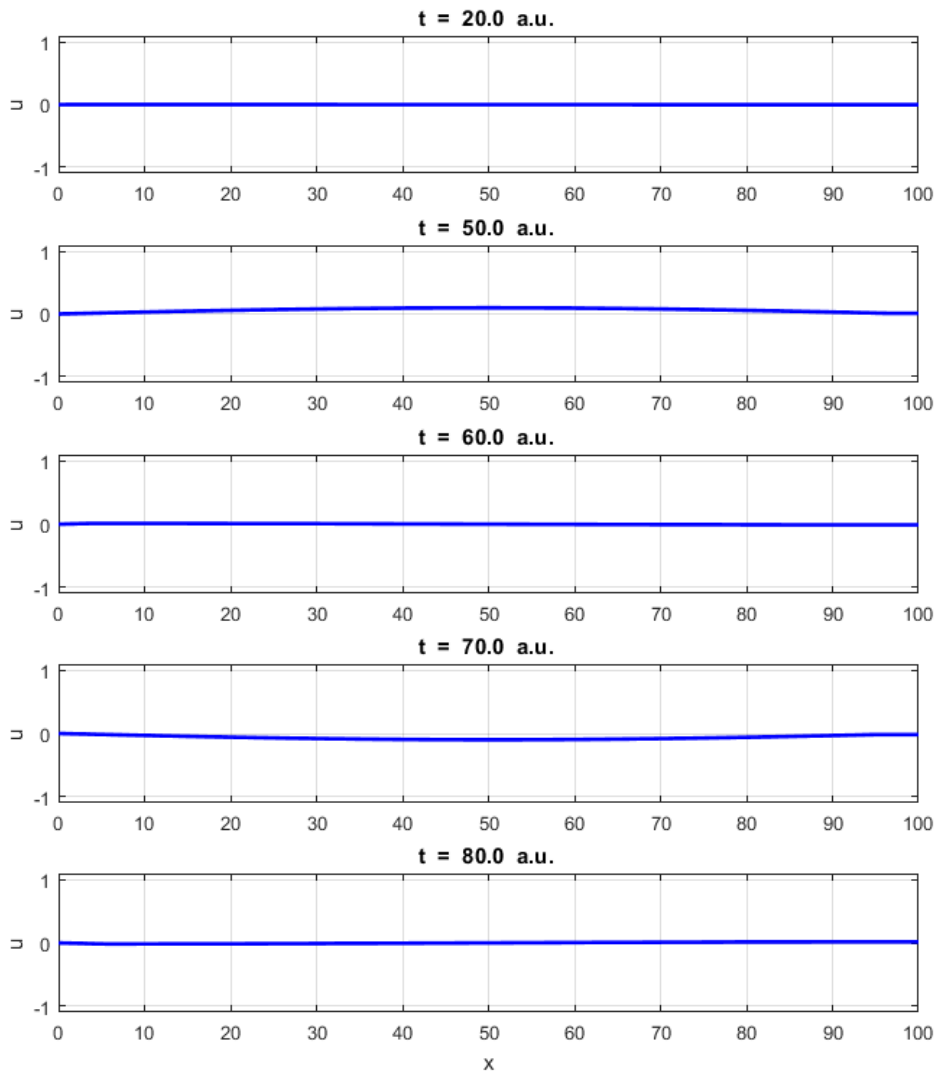


Fig. 20. Source driving frequency is equal to the frequency of the second harmonic $f_{source} = 2f_1$ and this does not equal one of the natural frequencies. A large amplitude standing wave is not created.

em_swe_06.m

Reflection and Transmission at an interface

Waves experiences partial transmission and partial reflection at an interface when there is a difference in velocity of the wave on either side of the interface. Reflection is a wave phenomenon that changes the direction of a wavefront at the interface. Transmission occurs when some of the energy of the incident wave is transferred across the interface.

We can define the reflection coefficient r and the transmission coefficient t as

$$r = \frac{A_R}{A_I} = \frac{v_2 - v_1}{v_1 + v_2} \quad t = \frac{A_T}{A_I} = \frac{2v_1 v_2}{v_1 + v_2}$$

where A_I is the amplitude of the incident wave and v_1 is the speed of the incident wave, A_R is the amplitude of the reflected wave travelling at speed v_1 and A_T is the amplitude of the transmitted wave traveling at a speed v_2 .

We can model the reflection and transmission of waves at an interface by solving the scalar wave equation in which the wave speed to the left of the interface is v_1 and v_2 to the right of the interface. If the interface position is x_I then

$$x \leq x_I \quad \Rightarrow \quad v = v_1 \quad x > x_I \quad \Rightarrow \quad v = v_2$$

Figure 21 shows the propagation of a Gaussian pulse striking the interface where $v_2 = v_1 / 4$. The incident pulse has an amplitude $A_I = 1$ traveling from left to right and then striking the interface. There is excellent agreement between the theoretical and simulation values for the amplitudes of the reflected and transmitted pulses:

Theory	$A_R = -0.6000$	$A_T = 0.4000$
--------	-----------------	----------------

Simulation	$A_R = -0.6008$	$A_T = 0.3993$
------------	-----------------	----------------

The incident pulse has a width of $58h_x$ between the $1/e$ points. The width of the reflected pulse is $56h_x$ which is almost the same as the width of the incident pulse. The spatial width of the transmitted pulse is $14h_x$ which is about $\frac{1}{4}$ of the incident pulse width as predicted by the analytical theory.

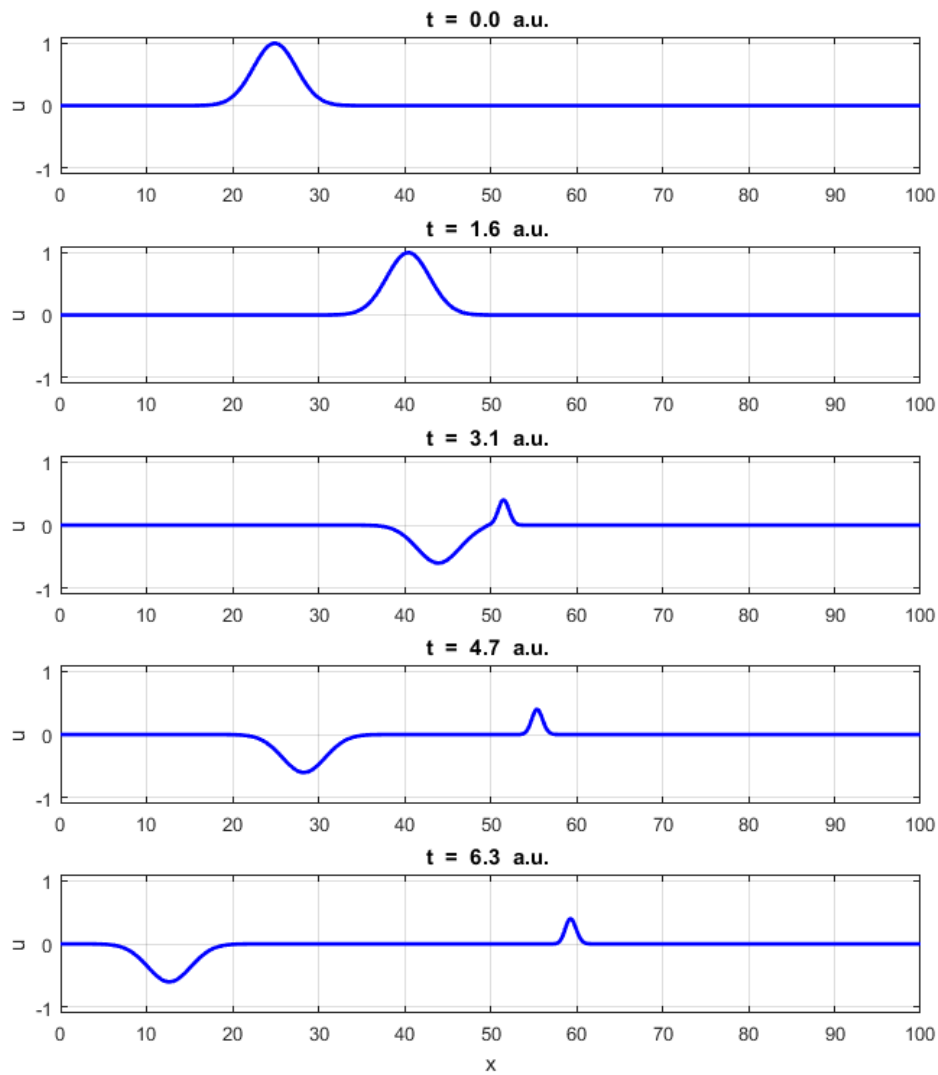


Fig. 21. Time evolution of a Gaussian pulse striking an interface at $x = 50$ a.u.

$$x \leq x_I \Rightarrow v = v_1 \quad x > x_I \Rightarrow v = v_2$$

[view animation](#)

Both regions of the grid are modelled with the same values for h_x and h_t but the Courant numbers are different $S_1 = 1$ and $S_2 = 1/4$. We are no longer using the magic time step, therefore, our solution is only an approximation. In figure 21 the number of space grid point is $N_x = 800$. If we run the simulation with $N_x = 200$ as shown in figure 22, then the results of the simulation are not accurate. Generally, the smaller the value assigned for h_x , the more accurate the numerical predictions.

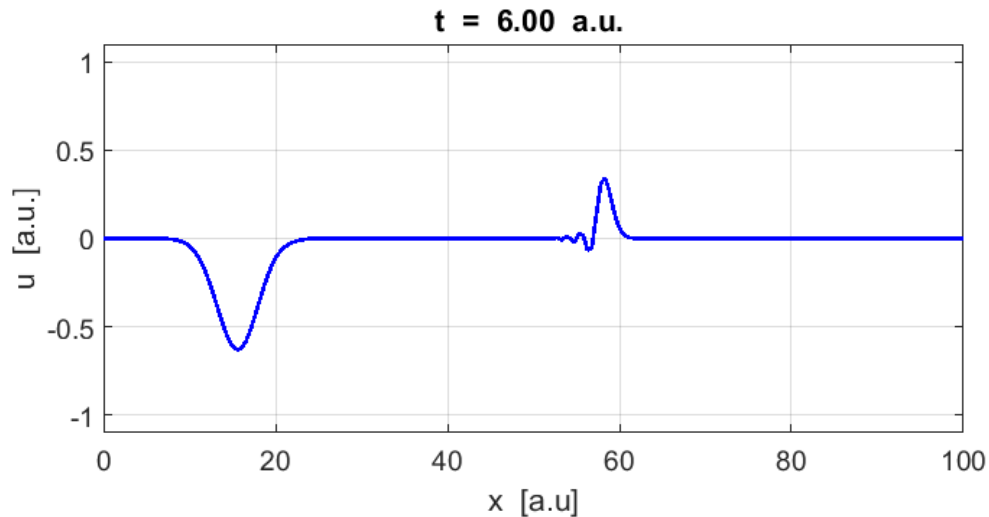


Fig. 22. Simulation showing a numerical instability in the solution. Generally, the smaller the values for the spatial and time increments the better the numerical stability and the more accurate predictions.