Leon Wu
CID: 01190736
M3N10 Project 3

April 10, 2020

Please see the link for animations of the waves that accompany this report + references:
https://leonwu4951.github.io/comp-pdes/

Part 1

Discretisation formula

The one-dimensional wave equation for $u(x,t)$ is given by

$$u_{tt} = u_{xx}.$$

Letting $R = \frac{\Delta t}{\Delta x}$, the finite difference formula used for the interior of the domain is as follows:

$$U_j^{k+1} = R^2 \left[U_{j-1}^k - 2U_j^k + U_{j+1}^k\right] + 2U_j^k - U_j^{k-1} \tag{1}$$

First Time Step

For the first time step, the equation is slightly modified since the $(k-1)^{th}$ time step is unknown. Since we have the condition $u_t = 0$ at $t = 0$, we have $U_j^{-1} = U_j^1$, so that

$$U_j^1 = \frac{R^2}{2} \left[U_{j-1}^0 - 2U_j^0 + U_{j+1}^0\right] + U_j^0$$

The first time step for the boundary conditions need not be derived for this problem since the boundaries stay at 0 for a significant amount of time before the solution reaches them.

Open Boundary Conditions

The following formulas can be used to discretise the left and right boundary in order to let the waves pass through. The artificial boundary condition $u_x = +u_t$ can be approximated using

$$R(U_1^k - U_{-1}^k) = (U_0^{k+1} - U_0^{k-1}). \tag{2}$$

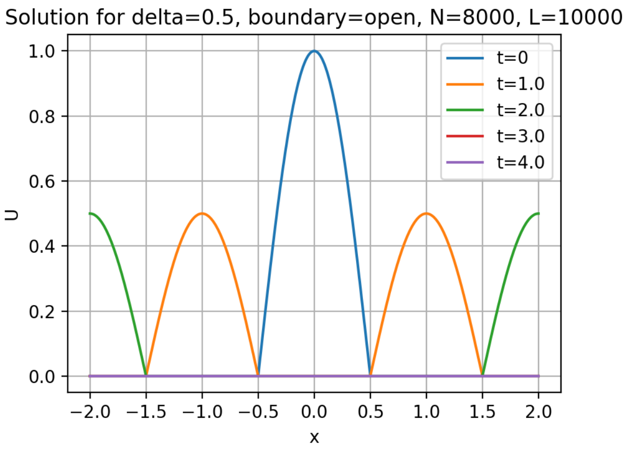
Equations (1) and (2) can be combined to eliminate U_{-1}^k. The final formula used to discretise the left boundary is

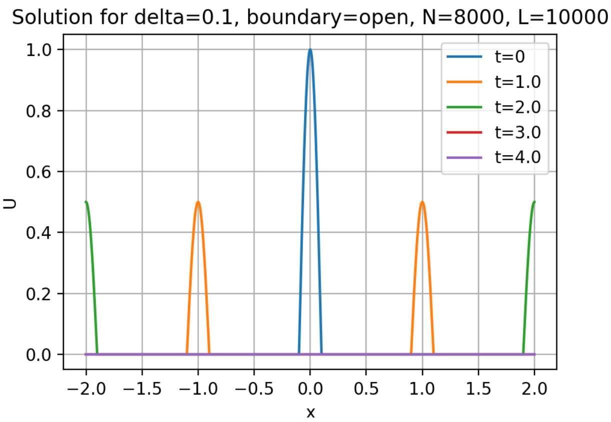
$$U_0^{k+1} = U_0^k + R(U_1^k - U_0^k)$$

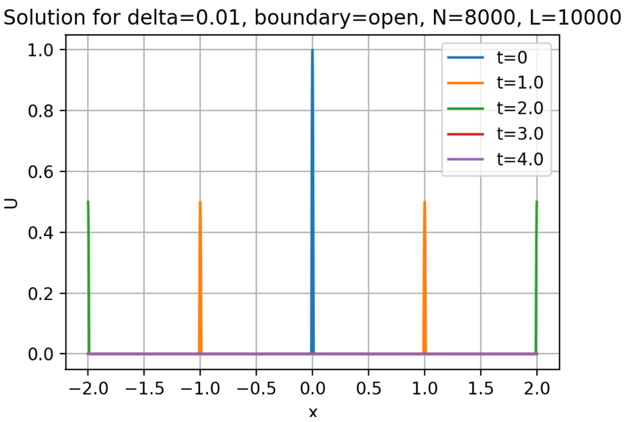
Similarly, the formula for the right boundary can be calculated as

$$U_{N-1}^{k+1} = U_{N-1}^{k} + R(U_{N-2}^{k} - U_{N-1}^{k})$$

These formulas allow for the waves to pass out the domain perfectly, and keeping 2nd order accuracy. The following figures show the results.

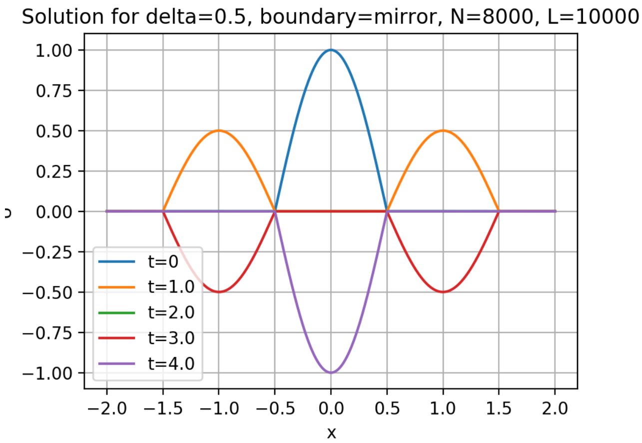


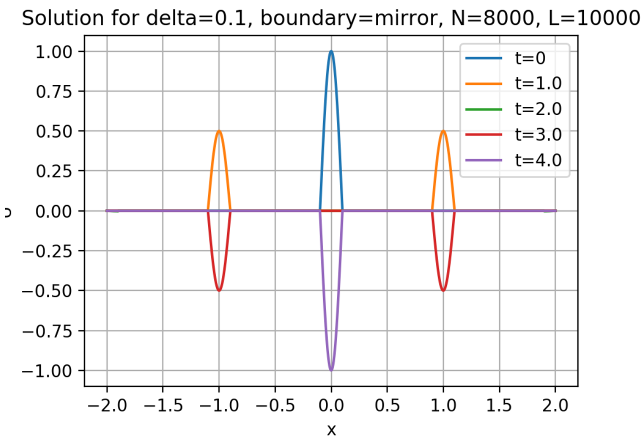


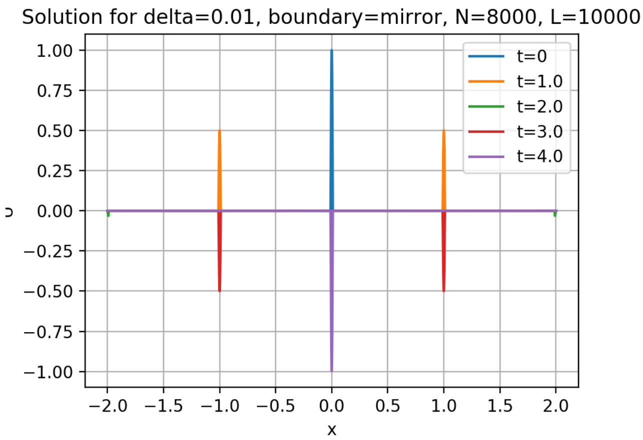


Mirror Condition

The Dirichlet mirror condition on the boundaries $u(-2,t) = u(2,t) = 0$ can easily be imposed in order to "mirror" the wave back from the boundaries.







Reflection Condition

The Neumann reflection condition $u_x = 0$ can be approximated at the left boundary using the centered difference

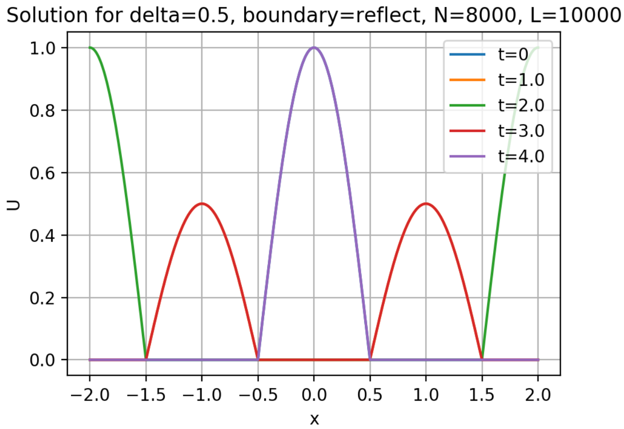
$$\frac{U_1^k - U_{-1}^k}{2\Delta x} = 0 \tag{3}$$

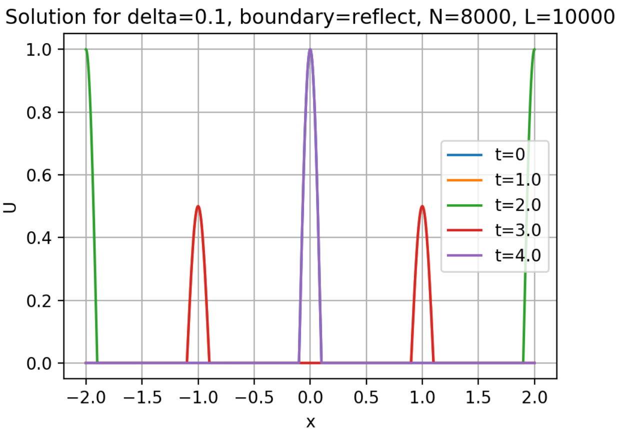
Again, equations (1) and (3) can be combined to eliminate U_{-1}^k. The final formula used to discretise the left boundary is

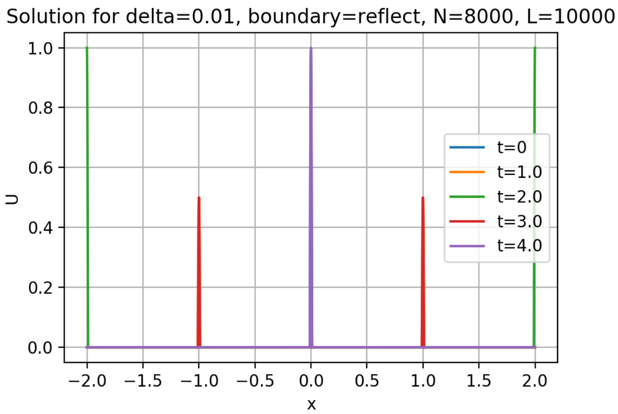
$$U_0^{k+1} = -U_0^{k-1} + 2U_0^k + 2q^2(U_1^k - U_0^k)$$

Similarly, the formula for the right boundary can be calculated as

$$U_{N-1}^{k+1} = -U_{N-1}^{k-1} + 2U_{N-1}^{k} + 2R^2(U_{N-2}^{k} - U_{N-1}^{k})$$



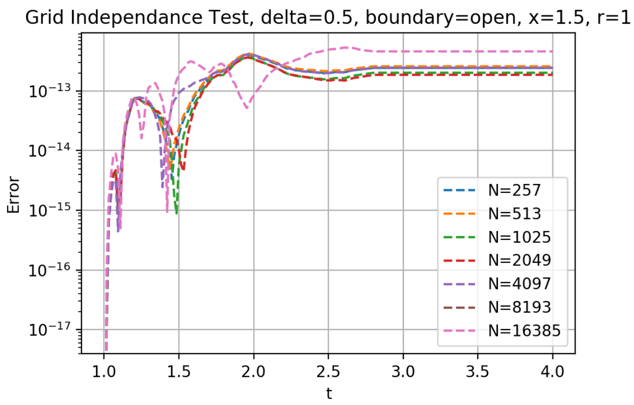


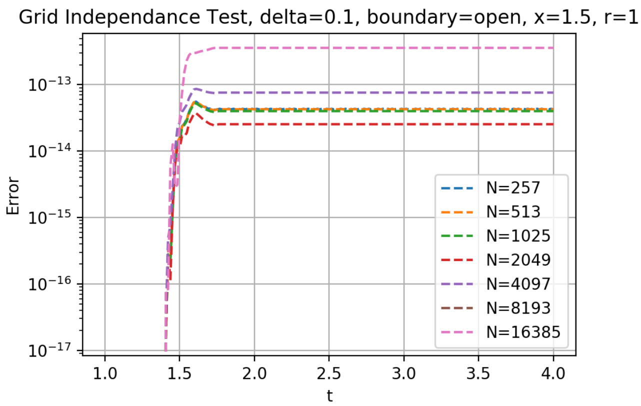


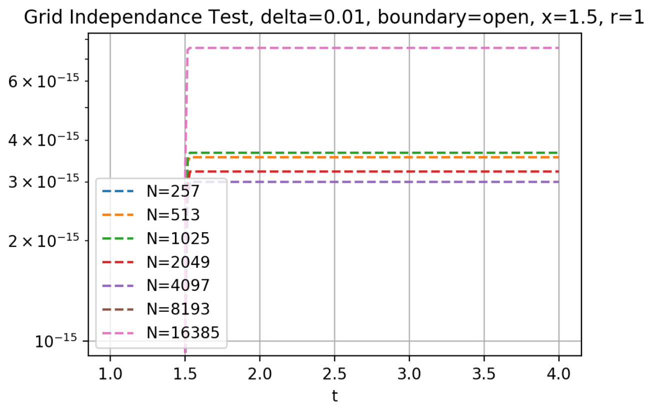
Grid Independence

- Grid Independence were performed for all 3 different boundary conditions and the 3 relevant values of δ.
- The solutions at $x = 0$, $x = 0.5$, $x = 1$ and $x = 1.5$ were compared to a reference solution (very large N), with the error calculated at each timestep. The results for 1 value of δ and 1 boundary are shown below.
- As an example, the plots below show the solution error for $x = 1.5$ over time with open boundary conditions (compared to a reference solution with large N). The plots show that all grids have an error less than 1e-12 for all values of δ being tested, and the values of N tested don't change the solution, so as long as $r = 1$ we can use any reasonable grid within the range of N tested.

- For $r > 1$, the solution blows up since the scheme is unstable.
- The grid must be fine enough so that it captures the initial condition with enough accuracy, which depends on the value of δ. A smaller value of δ means the grid needs to be finer to capture the detail.







Modified Equation and Dissipation-Dispersion Analysis

- Using Taylor Series to expand the terms in the discretization formula, we get for the modified equation:

$$
\begin{aligned}
&\frac{u_j^{k+1} - 2u_j^k + u_j^{k-1}}{(\Delta t)^2} = \frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{(\Delta x)^2} \\
\implies &\frac{1}{(\Delta t)^2}\Big([u + (\Delta t)u_t + \frac{(\Delta t)^2}{2}u_{tt} + \frac{(\Delta t)^3}{6}u_{ttt} + \frac{(\Delta t)^4}{24}u_{tttt} + ...]_j^k \\
&+ [u - (\Delta t)u_t + \frac{(\Delta t)^2}{2}u_{tt} - \frac{(\Delta t)^3}{6}u_{ttt} + \frac{(\Delta t)^4}{24}u_{tttt} + ...]_j^k - 2u\Big) \\
=&\frac{1}{(\Delta x)^2}\Big([u + (\Delta x)u_x + \frac{(\Delta x)^2}{2}u_{xx} + \frac{(\Delta x)^3}{6}u_{xxx} + \frac{(\Delta x)^4}{24}u_{xxxx} + ...]_j^k \\
&+ [u - (\Delta x)u_x + \frac{(\Delta x)^2}{2}u_{xx} + \frac{(\Delta x)^3}{6}u_{xxx} + \frac{(\Delta x)^4}{24}u_{xxxx} + ...]_j^k - 2u\Big) \\
\implies &u_{tt} + \frac{(\Delta t)^2}{12}u_{tttt} + ... = u_{xx} + \frac{(\Delta x)^2}{12}u_{xxxx} + ...
\end{aligned}
$$

  Since we have $u_{tttt} = u_{xxtt} = u_{xxxx}$, the discretization error can be written as

$$
\frac{1}{12}\Big[(\Delta t)^2 - (\Delta x)^2\Big]u_{xxxx} + ...
$$

- The even terms contribute to dissipation in the solution.

- Here we can see that if $(\Delta t) = (\Delta x)$, there is no dissipation.

- Plugging in the wave $e^{i(kx-\omega t)}$ we get the following equation:

$$
\frac{4}{(\Delta t)^2}sin^2(\frac{\omega\Delta t}{2}) = \frac{4}{(\Delta x)^2}sin^2(\frac{k\Delta x}{2}).
$$

  The RHS is 1 when

$$
\frac{k}{2} = m\frac{\pi}{2}
$$

  for integer a. $a = 1$ gives the smallest wavelength that can be calculated on the grid: $2\Delta x$. This is why a finer minimum grid is needed for lower values of δ, since the initial condition has a wavelength of 4δ.

- We then get the numerical dispersion relation

$$
\omega = \frac{2}{\Delta t}sin^{-1}(sin(\frac{k\Delta x}{2})).
$$

- Analytically, there is no dispersion, but there is dispersion in the numerical scheme.

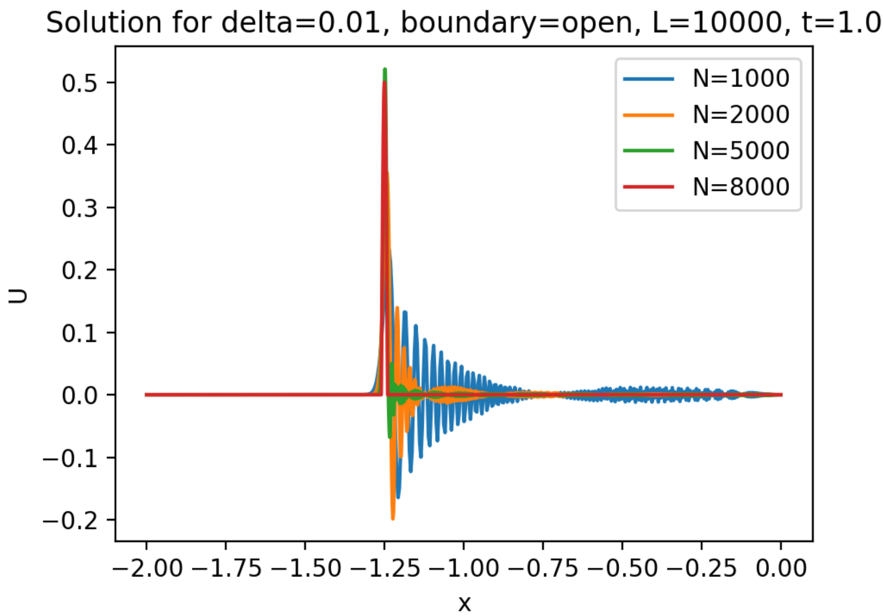
- For the case when $\Delta t = \Delta x$, we get

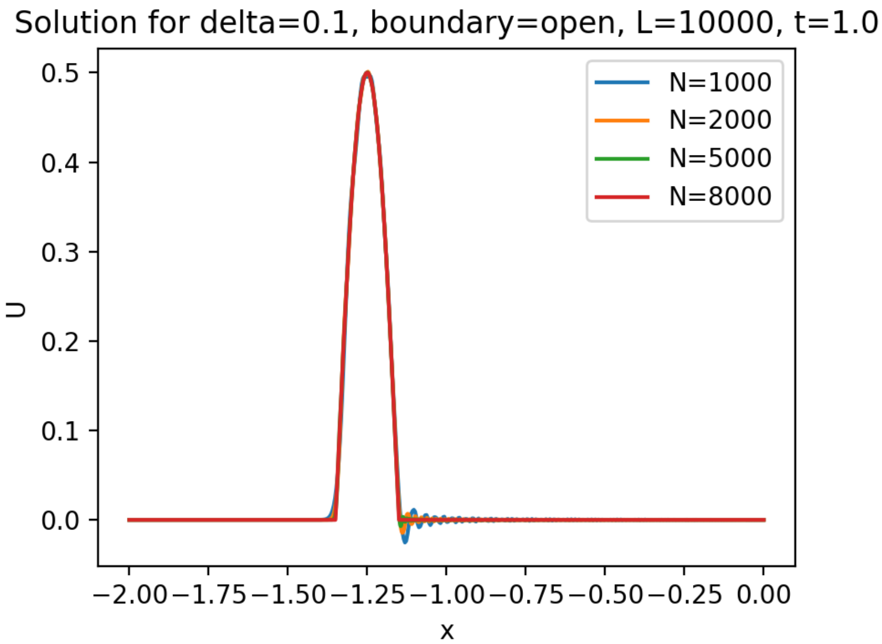
$$\omega = \frac{2}{\Delta t}\frac{k\Delta x}{2} = k,$$

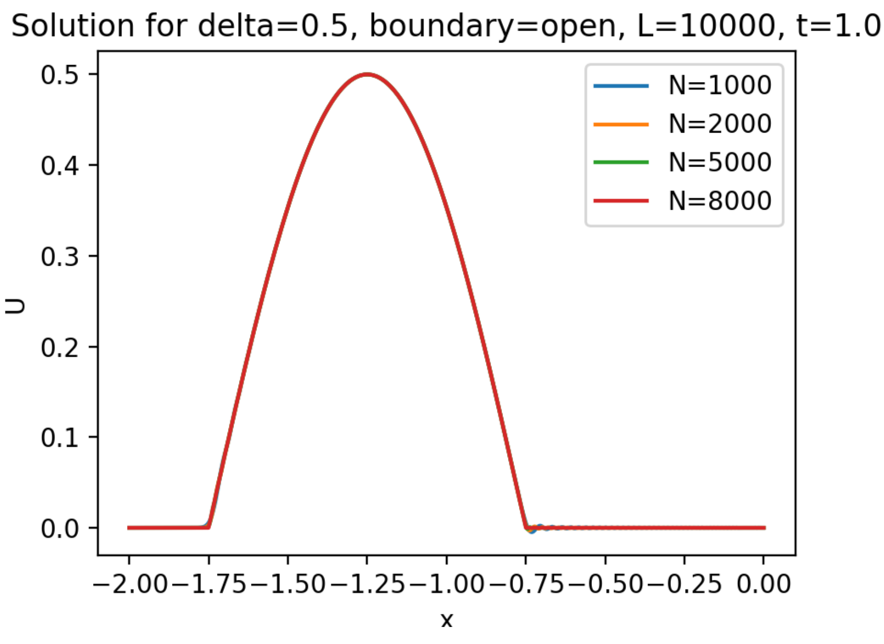
which is the analytical dispersion relation, ie there is no dispersion.

- The figures below show a solution at time $t = 1$. $L = 10000$ and $T = 5$, so that $\Delta x = \Delta t$ when $N = 8000$. For this case, we see that this is no dispersion of dissipation in the solution as was suggested by the analysis above. For values less than $N = 8000$, ie $\Delta x > \Delta t$, we can see dissipation on the left of the wave as the solution rises before the exact solution given by $N = 8000$. On the right, we can see dispersion as the solution oscillates about the exact solution before settling at $U = 0$.

- We can see the severity of the dissipation and dispersion depends on δ. A lower value of δ gives high dissipation and dispersion since the wavelength is lower.




Solution for delta=0.1, boundary=open, L=10000, t=1.0
N=1000
N=2000
N=5000
N=8000
U
x


Solution for delta=0.5, boundary=open, L=10000, t=1.0
N=1000
N=2000
N=5000
N=8000
U
x

Part 2

Discretisation formula

The 2D wave equation for $u(x, y, t)$ is given by

$$u_{tt} = u_{xx} + qu_{yy}.$$

Letting $R_x = \frac{\Delta t}{\Delta x}$ and $R_y = \frac{\Delta t}{\Delta y}$, the finite difference formula used for the interior of the domain can be derived as follows:

$$\frac{U_{mn}^{k-1} - 2U_{mn}^{k} + U_{mn}^{k+1}}{(\Delta t)^2} = \frac{U_{m-1n}^{k} - 2U_{mn}^{k} + U_{m+1n}^{k}}{(\Delta x)^2} + q\frac{U_{mn-1}^{k} - 2U_{mn}^{k} + U_{mn+1}^{k}}{(\Delta y)^2}$$
$$\implies U_{mn}^{k+1} = R_x^2 \left[U_{m-1n}^{k} - U_{mn}^{k} + U_{m+1n}^{k}\right] + qR_y^2 \left[U_{mn-1}^{k} - U_{mn}^{k} + U_{mn+1}^{k}\right] + 2U_{mn}^{k} - U_{mn}^{k-1}$$

Again, for the first time step, the equation is slightly modified. Since we have the condition $u_t = 0$ at $t = 0$, we have $U_{mn}^{-1} = U_{mn}^{1}$, so that

$$U_{mn}^{1} = \frac{R_x^2}{2} \left[U_{m-1n}^{0} - U_{mn}^{0} + U_{m+1n}^{0}\right] + q\frac{R_y^2}{2} \left[U_{mn-1}^{0} - U_{mn}^{0} + U_{mn+1}^{0}\right] + U_{mn}^{0}$$

Open Boundary Conditions

An approximate boundary condition can be implemented in order to greatly reduce reflections from the boundary. This must be done for a general q, ensuring accuracy across all values of q. As in the notes, taking Fourier transofrms, the general solution can be written as a superposition of the waves $exp(ilx + imy + i\omega t)$, where l, m and ω are related by

$$l^2 + qm^2 = \omega^2$$

. For $y = -2$, we must impose $m > 0$, so that

$$m = +\sqrt{\frac{\omega^2 - l^2}{q}}$$

Expanding to the next order, we have

$$mc = \frac{\omega}{\sqrt{q}}(1 - \frac{l^2}{\omega^2})^{\frac{1}{2}}$$
$$\implies m\sqrt{q} = \omega - \frac{l^2}{2\omega}$$

which is equivalent to the boundary condition

$$u_{yt}\sqrt{q} = u_{tt} - \frac{1}{2}u_{xx}, \qquad y = -2$$

Similarly, boundary conditions for the other 3 sides can be obtained:

$$\begin{aligned} u_{yt}\sqrt{q} &= \frac{1}{2}u_{xx} - u_{tt}, \qquad y = 2 \\ u_{xt} &= u_{tt} - \frac{1}{2}qu_{yy}, \qquad x = -2 \\ u_{xt} &= \frac{1}{2}qu_{yy} - u_{tt}, \qquad x = 2 \end{aligned}$$

For the finite difference formula, the mixed derivative terms can be approximated on the boundaries $n = 0$ using the formulas

$$\begin{aligned} u_{yt} &\approx \frac{1}{2\Delta t\Delta y}(U_{mn+1}^{k+1} + U_{mn}^{k-1} - U_{mn}^{k+1} - U_{mn+1}^{k-1}), \qquad y = -2 \\ u_{yt} &\approx \frac{1}{2\Delta t\Delta y}(U_{mn}^{k+1} + U_{mn-1}^{k-1} - U_{mn-1}^{k+1} - U_{mn}^{k-1}), \qquad y = 2 \\ u_{xt} &\approx \frac{1}{2\Delta t\Delta x}(U_{m+1n}^{k+1} + U_{mn}^{k-1} - U_{mn}^{k+1} - U_{m+1n}^{k-1}), \qquad x = -2 \\ u_{xt} &\approx \frac{1}{2\Delta t\Delta x}(U_{mn}^{k+1} + U_{m-1n}^{k-1} - U_{m-1n}^{k+1} - U_{mn}^{k-1}), \qquad x = 2 \end{aligned}$$

where the centered difference was taken in both the t direction, then the forward difference was taken in the y direction. For the scheme for the boundary conditions, taking the centered differences for the terms u_{xx} and u_{tt} also, we get for $n = 0$,

$$\begin{aligned} &\frac{\sqrt{q}}{2\Delta t\Delta y}(U_{mn+1}^{k+1} + U_{mn}^{k-1} - U_{mn}^{k+1} - U_{mn+1}^{k-1}) = \frac{U_{mn}^{k+1} - 2U_{mn}^{k} + U_{mn}^{k-1}}{(\Delta t)^2} - \frac{U_{m+1n}^{k} - 2U_{mn}^{k} + U_{m-1n}^{k}}{2(\Delta x)^2} \\ &\implies U_{mn}^{k+1} = \frac{-1}{(\frac{2\Delta y}{\sqrt{q}\Delta t} + 1)}\left[2\Delta y\frac{-2U_{mn}^{k} + U_{mn}^{k-1}}{\sqrt{q}\Delta t} - \Delta y\Delta t\frac{U_{m+1n}^{k} - 2U_{mn}^{k} + U_{m-1n}^{k}}{\sqrt{q}(\Delta x)^2} - U_{mn}^{k-1} + U_{mn+1}^{k-1} - U_{mn+1}^{k+1}\right. \end{aligned}$$

Similarly, for $n = N - 1$, $m = 0$ and $m = M - 1$ respectively, the following formulas can be derived. The direction (forward of backward difference) of the mixed derivative approximation is changed accordingly (eg. for $m = M - 1$ the backward difference is used in the x direction to approximate u_{xt}.

$$U_{mn}^{k+1} = \frac{-1}{(\frac{2\Delta y}{\sqrt{q}\Delta t}+1)}\left[2\Delta y\frac{-2U_{mn}^{k}+U_{mn}^{k-1}}{\sqrt{q}\Delta t} - \Delta y\Delta t\frac{U_{m+1n}^{k}-2U_{mn}^{k}+U_{m-1n}^{k}}{\sqrt{q}(\Delta x)^2} - U_{mn}^{k-1} + U_{mn-1}^{k-1} - U_{mn-1}^{k+1}\right]$$

$$U_{mn}^{k+1} = \frac{-1}{(\frac{2\Delta x}{\Delta t}+1)}\left[2\Delta x\frac{-2U_{mn}^{k}+U_{mn}^{k-1}}{\Delta t} - q\Delta x\Delta t\frac{U_{mn+1}^{k}-2U_{mn}^{k}+U_{mn-1}^{k}}{(\Delta y)^2} - U_{mn}^{k-1} + U_{m+1n}^{k-1} - U_{m+1n}^{k+1}\right]$$

$$U_{mn}^{k+1} = \frac{-1}{(\frac{2\Delta x}{\Delta t}+1)}\left[2\Delta x\frac{-2U_{mn}^{k}+U_{mn}^{k-1}}{\Delta t} - q\Delta x\Delta t\frac{U_{mn+1}^{k}-2U_{mn}^{k}+U_{mn-1}^{k}}{(\Delta y)^2} - U_{mn}^{k-1} + U_{m-1n}^{k-1} - U_{m-1n}^{k+1}\right]$$

Note the the $k+1$ terms on the RHS are known since the interior of the domain has already been calculated. Note also the corners are calculated using points from the current timestep on the boundary, so care must be taken to implement this correctly. For the 4 corners, we can modify the first two equations (for $y=-2,2$) to use sided differences, so that at the corners $(m=0,n=0),(m=M-1,n=0),(m=0,n=N-1),(m=M-1,n=N-1)$ respectively, we get

$$U_{mn}^{k+1} = \frac{-1}{(\frac{2\Delta y}{\sqrt{q}\Delta t}+1)}\left[2\Delta y\frac{-2U_{mn}^{k}+U_{mn}^{k-1}}{\sqrt{q}\Delta t} - \Delta y\Delta t\frac{U_{m+2n}^{k}-2U_{m+1n}^{k}+U_{mn}^{k}}{\sqrt{q}(\Delta x)^2} - U_{mn}^{k-1} + U_{mn+1}^{k-1} - U_{mn+1}^{k+1}\right]$$

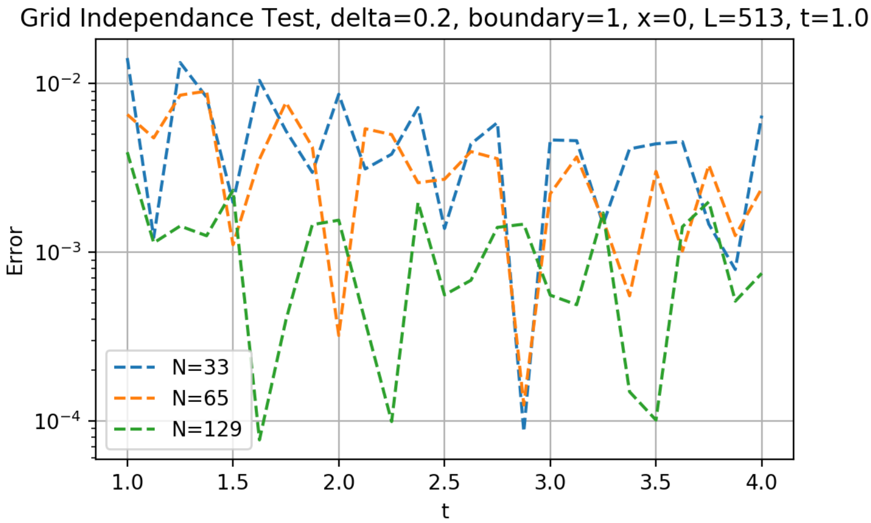
$$U_{mn}^{k+1} = \frac{-1}{(\frac{2\Delta y}{\sqrt{q}\Delta t}+1)}\left[2\Delta y\frac{-2U_{mn}^{k}+U_{mn}^{k-1}}{\sqrt{q}\Delta t} - \Delta y\Delta t\frac{U_{mn}^{k}-2U_{m-1n}^{k}+U_{m-2n}^{k}}{\sqrt{q}(\Delta x)^2} - U_{mn}^{k-1} + U_{mn+1}^{k-1} - U_{mn+1}^{k+1}\right]$$

$$U_{mn}^{k+1} = \frac{-1}{(\frac{2\Delta y}{\sqrt{q}\Delta t}+1)}\left[2\Delta y\frac{-2U_{mn}^{k}+U_{mn}^{k-1}}{\sqrt{q}\Delta t} - \Delta y\Delta t\frac{U_{m+2n}^{k}-2U_{m+1n}^{k}+U_{mn}^{k}}{\sqrt{q}(\Delta x)^2} - U_{mn}^{k-1} + U_{mn-1}^{k-1} - U_{mn-1}^{k+1}\right]$$

$$U_{mn}^{k+1} = \frac{-1}{(\frac{2\Delta y}{\sqrt{q}\Delta t}+1)}\left[2\Delta y\frac{-2U_{mn}^{k}+U_{mn}^{k-1}}{\sqrt{q}\Delta t} - \Delta y\Delta t\frac{U_{mn}^{k}-2U_{m-1n}^{k}+U_{m-2n}^{k}}{\sqrt{q}(\Delta x)^2} - U_{mn}^{k-1} + U_{mn-1}^{k-1} - U_{mn-1}^{k+1}\right]$$

Grid Independence

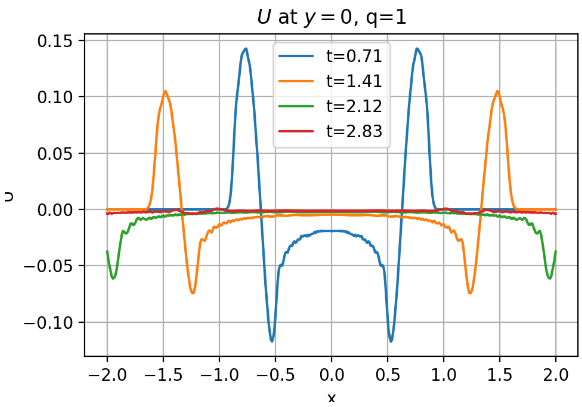
- A similar analysis was performed for the 2D wave equation for grid independence. The plot below shows the accuracy compared to a solution with $N=257$ for a fixed $L=513$. Note for stability for the case $q=1$, if $h_x=h_y$ we require $\Delta t/h_x \leq \frac{1}{\sqrt{2}}$. Ideally, higher values of N would be tested if more computational power was available.
- Setting a tolerance of 1e-3, the plot shows that the change from $N=129$ to $N=257$ changes the solution by roughly 1e-3, so a value of N around 300 will be sufficient for a maximum error of below 1e-3.

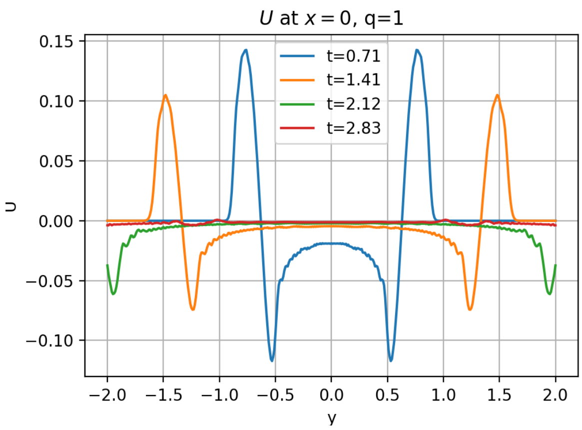

Grid Independance Test, delta=0.2, boundary=1, x=0, L=513, t=1.0
Error
t
N=33
N=65
N=129

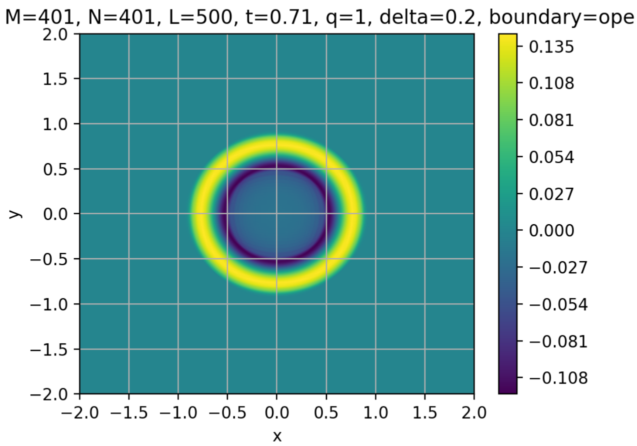
2D Wave Results

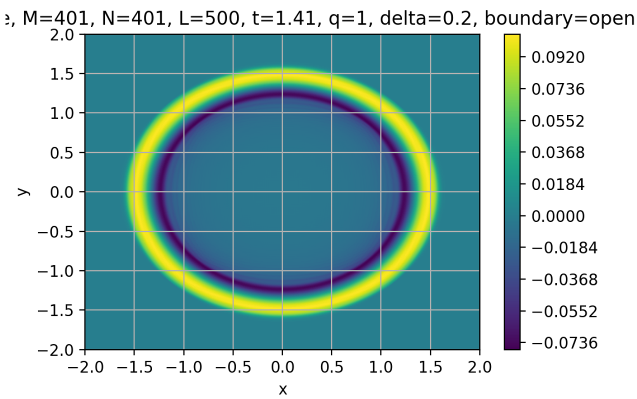
Again, please see the link for animations of the waves that accompany this report: https://leonwu4951.github.io/comp-pdes/

- The figures below show contour plots of the solutions for different times, and also the $(x, y = 0)$ and $(x = 0, y)$ data planes, for $q = 1$.





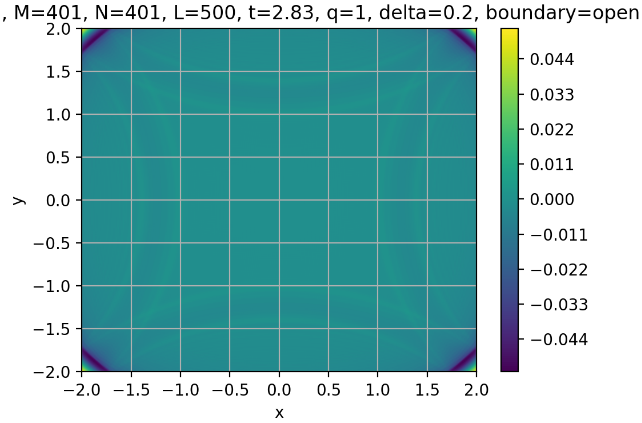


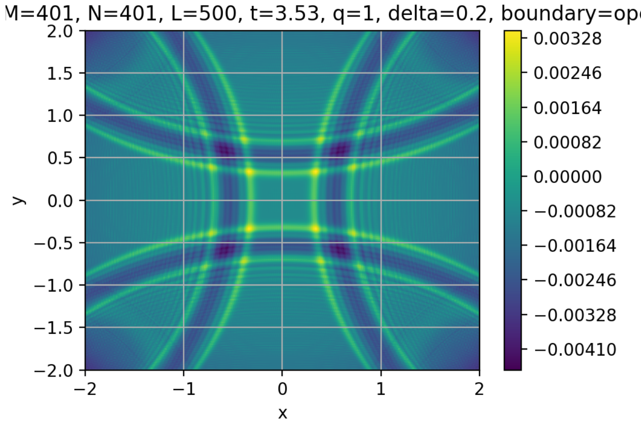


- The accuracy of the results for q not equal to 1 are lower. In order to overcome this, a few steps were taken:

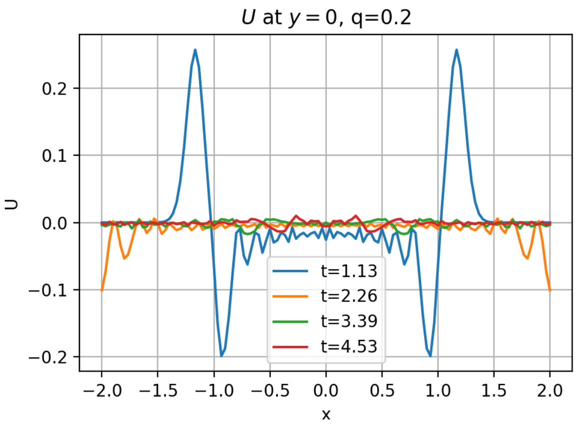
- The boundary conditions and interior formulas were modified to include q, so that the accuracy of the solutions was improved. (See the boundary conditions derived about).

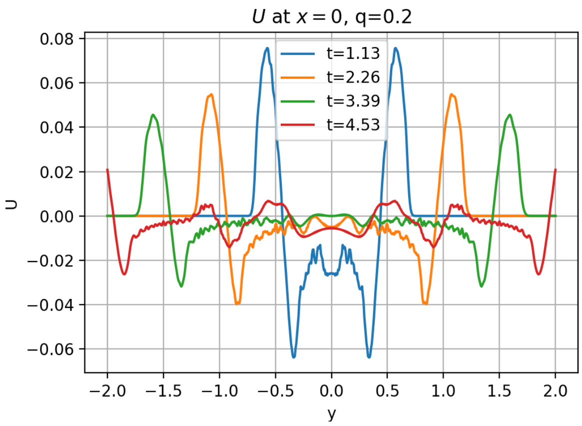
- For different values of q, the values of M and N (the number of gridpoints in x and y respectively), were adjusted in order to increase the accuracy of the solutions. In this case, M and N were set so that $M/N = q$. This is done in order to capture the faster moving solution in the x direction for low values of q, and vice versa.

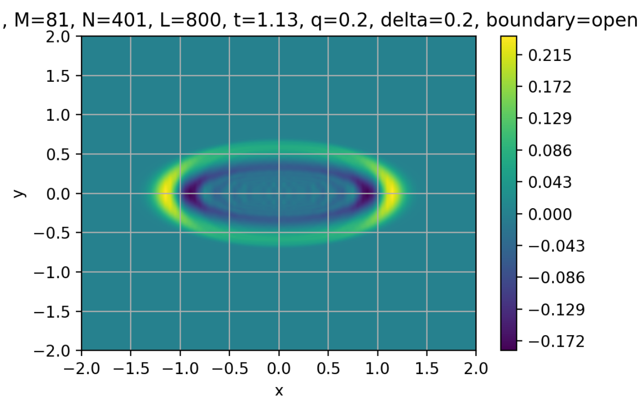


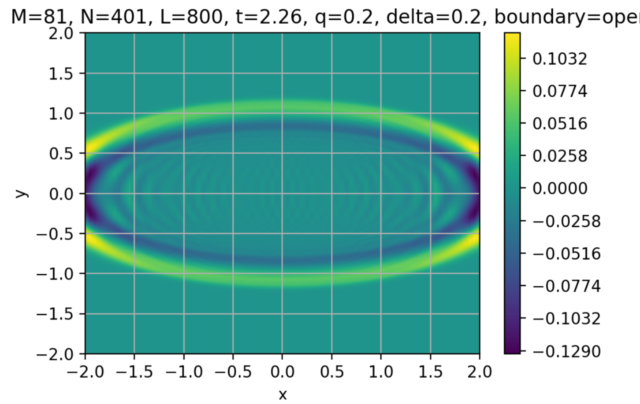


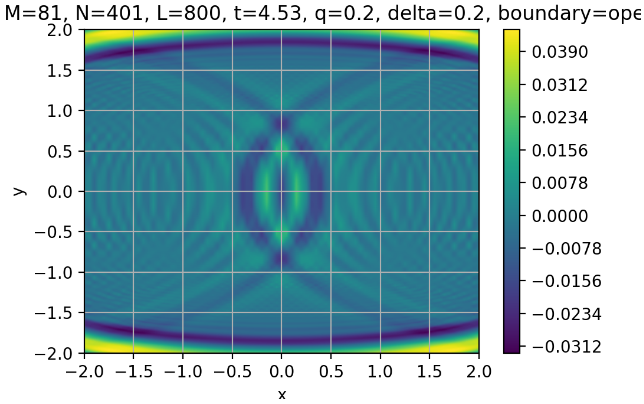
- Take note of the scales on the colorbar for the plots. The solutions for the 2d wave equation are not perfectly accurate, and the boundaries reflect some waves back, although these are very small because of the boundary conditions that were derived and implemented.
- The treatment of the schemes for q not equal to 1 now give similar accuracies to $q = 1$ as seen in the contour plots (looking at the magnitude of the solution) after the main wave passes the boundaries.

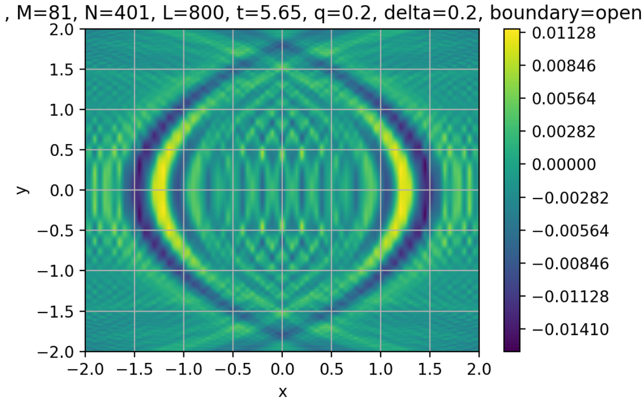


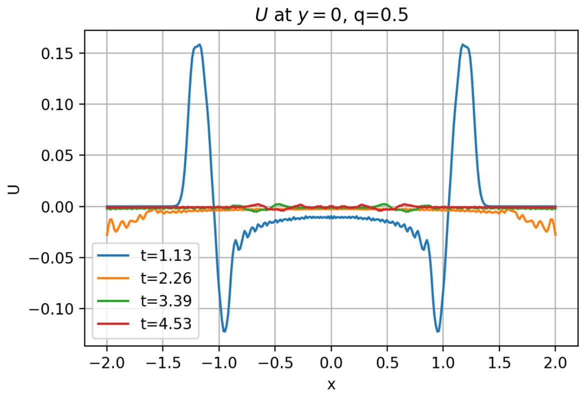


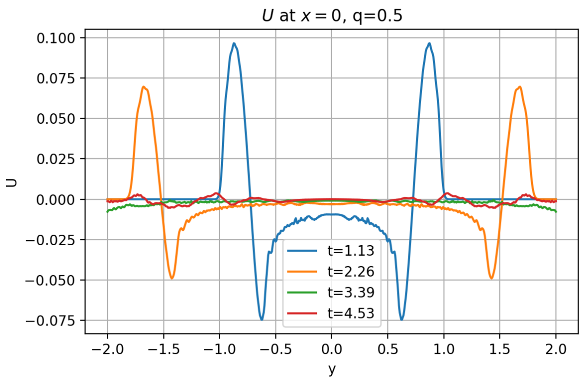


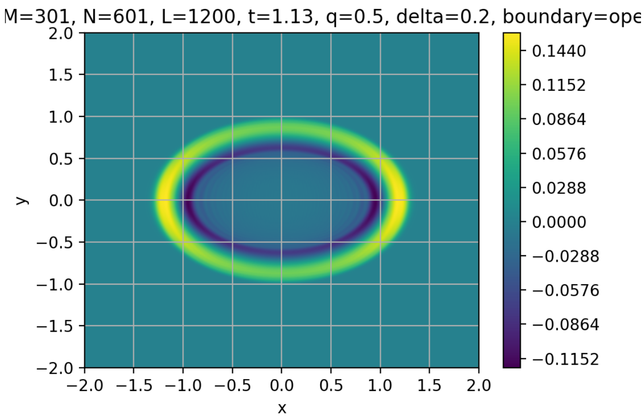


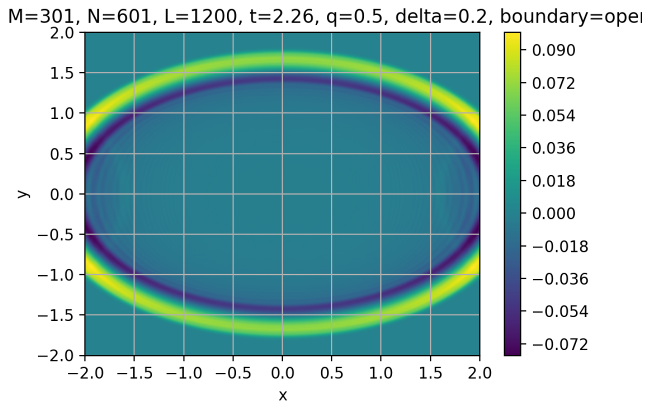

M=81, N=401, L=800, t=4.53, q=0.2, delta=0.2, boundary=ope
x
y

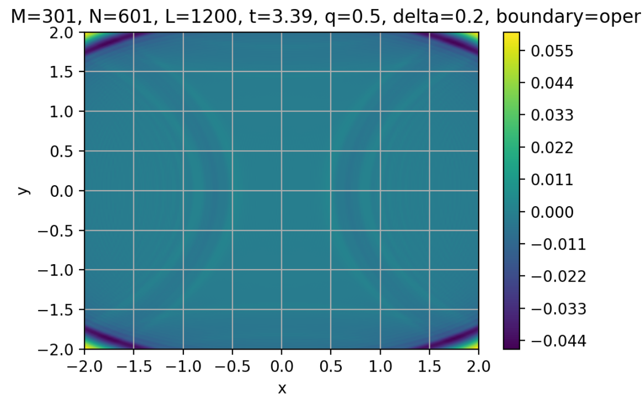

, M=81, N=401, L=800, t=5.65, q=0.2, delta=0.2, boundary=open
x
y

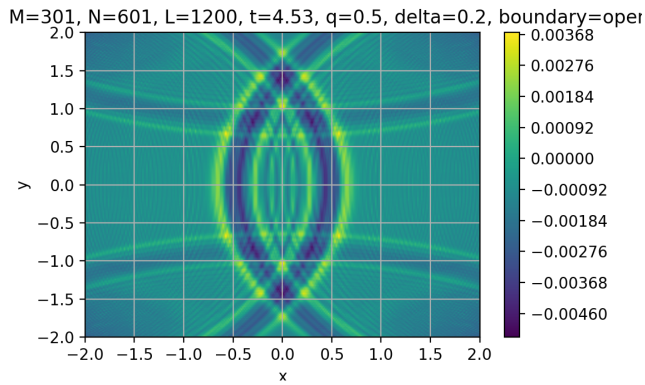

U at $y = 0$, q=0.5
t=1.13
t=2.26
t=3.39
t=4.53
x
U

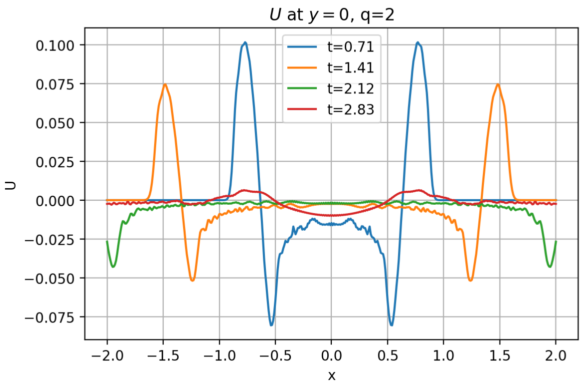

U at $x = 0$, q=0.5
t=1.13
t=2.26
t=3.39
t=4.53
y
U

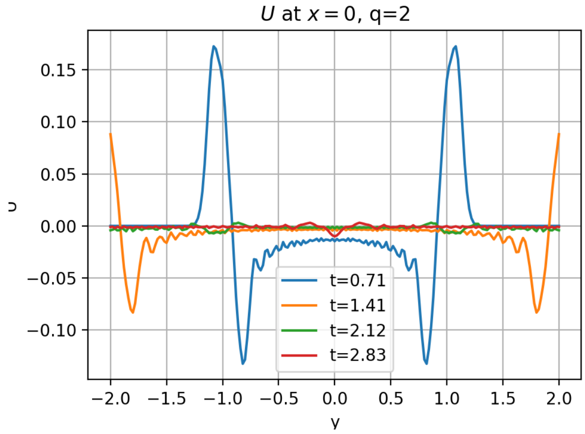

M=301, N=601, L=1200, t=1.13, q=0.5, delta=0.2, boundary=ope
x
y

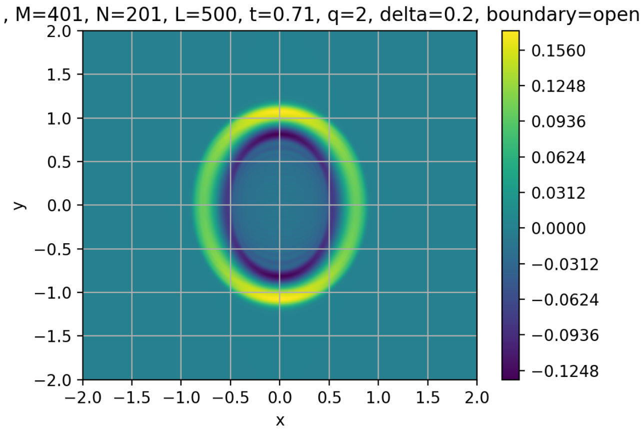

M=301, N=601, L=1200, t=2.26, q=0.5, delta=0.2, boundary=ope
x
y

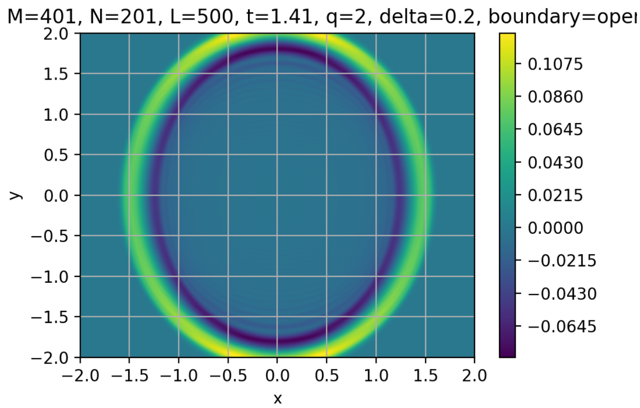

M=301, N=601, L=1200, t=3.39, q=0.5, delta=0.2, boundary=oper
x
y

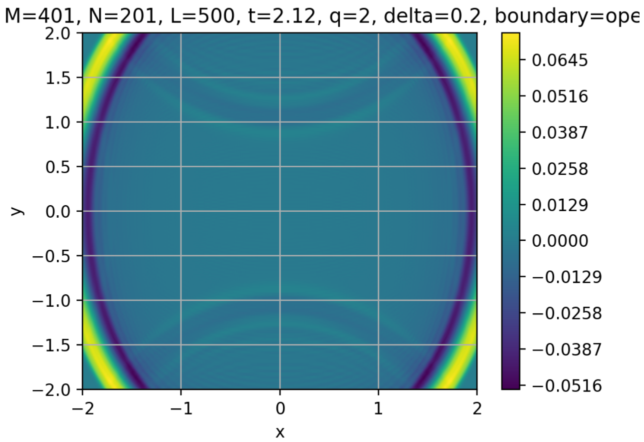

M=301, N=601, L=1200, t=4.53, q=0.5, delta=0.2, boundary=ope
x
y

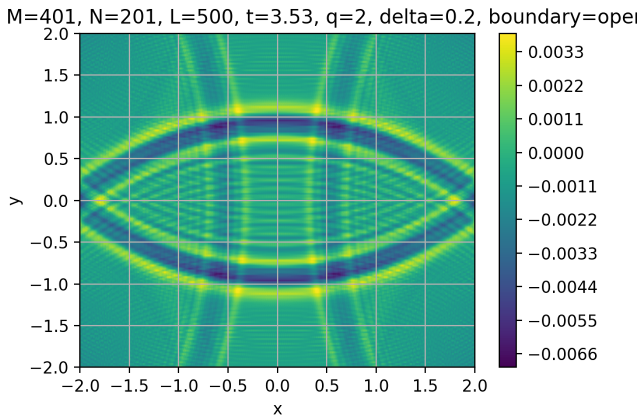

U at $y = 0$, q=2
t=0.71
t=1.41
t=2.12
t=2.83
x
U


U at $x = 0$, q=2
t=0.71
t=1.41
t=2.12
t=2.83
y
U


, M=401, N=201, L=500, t=0.71, q=2, delta=0.2, boundary=open
x
y


M=401, N=201, L=500, t=1.41, q=2, delta=0.2, boundary=ope
x
y


M=401, N=201, L=500, t=2.12, q=2, delta=0.2, boundary=ope
x
y


M=401, N=201, L=500, t=3.53, q=2, delta=0.2, boundary=ope
x
y

Perfectly Matched Layers

As an extension to this project, perfectly matched layers can be used to further dampen reflections at the boundaries. The idea is to define a function that is (nonzero for some outer region of the domain, and zero for the interior of the domain. This function can then be used to dissipate waves leaving the boundary so that the waves do not reflect back.