A Linear Wave Equation - Part II

The goal of the second part of this investigation was to compare the use of the Forward-in-Time-Backward-in-Space (FTBS) scheme to other schemes such as Leap-Frog (LF), Runge-Kutta 3 (RK3), and Adams-Bashforth 3 (AB3) when determining the solution of a one-dimensional linear wave equation on a fixed domain. The plots below show the results of all four schemes using variable sizes of time-steps and various wavelengths, thereby delineating the effects of changing the Courant's number (C). Also shown are the resultant numerical errors and amplitudes of the solutions as functions of time and number of timesteps. The attached file contains the C++ code used to generate output files named "[METHOD]_full.csv", "[METHOD]_summary.csv", and "exact_solution.csv", with the following four methods: FTBS, LF, RK3, and AB3.

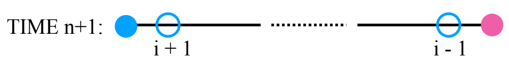
The FTBS equation used to update the solution of the wave is as follows:

$$u_i^{n+1} = u_i^n - C(u_i^n - u_{i-1}^n) \tag{1}$$

Here, C = c Δt/Δx. The interior nodes of the solution were solved for recursively using equation (1). The right boundary was solved for using the numerical solution for the node to the left of it. Due to the periodic nature of the boundary condition, the left boundary was equated to the solved value for the right boundary. Thus, a full advancement of the solution in time was achieved.

The Leap-Frog equation incorporated the previous time step to calculate the next one like so:

$$u_i^{n+1} = u_i^{n-1} - C\left(u_{i+1}^n - u_{i-1}^n\right) \tag{2}$$



The above was used for all the interior nodes. To calculate the right boundary, the i + 1 position was determined by that which was calculated for the second position in the vector of solutions for time n + 1, as depicted in the diagram. The left boundary was equated to the right boundary to satisfy the periodicity condition.

RK3, a multistep method, involved the generation of two intermediate approximates which were incorporated in the calculation for the next time step. The 3-stage process is outlined below:

$${}^{I}u_i = u_i^n - \frac{C}{6}\left(u_{i+1}^n - u_{i-1}^n\right) \tag{3}$$

$${}^{II}u_i = u_i^n - \frac{C}{4}\left({}^{I}u_{i+1} - {}^{I}u_{i-1}\right) \tag{4}$$

$$u_i^{n+1} = u_i^n - \frac{C}{2}\left({}^{II}u_{i+1} - {}^{II}u_{i-1}\right) \tag{5}$$

The purpose of this series of steps was to combine a predictor that is explicit to an inherently implicit scheme. We have the Euler-Forward predictor followed by a trapezoidal corrector. Overall, RK methods splits the time-step into smaller segments and combine several predictor-correctors. Once again, these series of calculations was followed for all the interior nodes. At each stage, the right boundary was calculated in the same manner as depicted in the diagram above. The left-most node was equated to the right boundary.

Finally, the AB3 iterations were calculated using the following:

$$u_i^{n+1} = u_i^n - \frac{C}{24}\left[23\left(u_{i+1}^n - u_{i-1}^n\right) - 16\left(u_{i+1}^{n-1} - u_{i-1}^{n-1}\right) + 5\left(u_{i+1}^{n-2} - u_{i-1}^{n-2}\right)\right] \tag{6}$$

The focus of this report will be to compare and contrast RK3 and Leap-Frog with FTBS. There will be minimal analysis of AB3 at this stage[1].

Experiment #1 serves as the base case for code testing with optimal C. Here, Δt is 2 seconds and the wavelength, λ_x, is 50 m. In this experiment, the exact solution and the calculated solution are the same for FTBS and Leap-Frog.

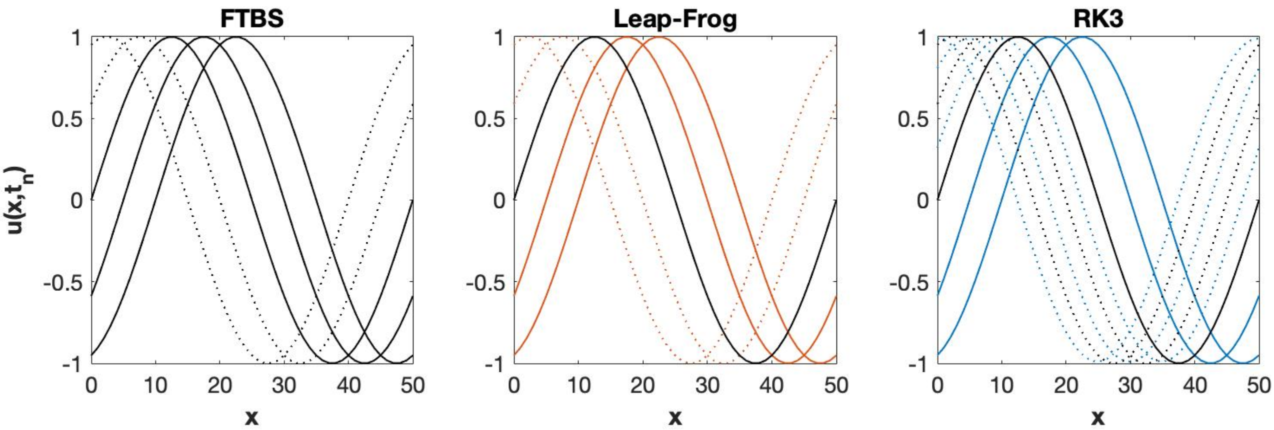


Figure 1A shows how the wave propagates with time, maintaining an amplitude of 1 with each time-step. Using the cyclic boundary condition, it can be deduced that the time taken for the wave to return to its initial condition is λ_x / c = 50 / 0.5 = 100 seconds. This holds for FTBS and Leap-Frog; however, the RK3 iterations show that a phase change occurs. This has to do with the phase component (θ) of the changing function, λ, that relates the solution at t to a solution at t + Δt.

$$\lambda \equiv e^{\theta i} \tag{7}$$

In RK3, we are changing Δt at every time-step. In effect, this means that C is being changed as well. By decreasing Δt in successive steps, there is a resultant decrease in θ (error) which slows down wave.

[1] This is mostly due to lack of time — I wasn't able to fully go through the stability analyses for AB3 and connect the results to them. All that is known is that AB3 is unstable when C = 1, which made it problematic to generate relevant figures as methods such as FTBS are increasingly dampened when C < 1.

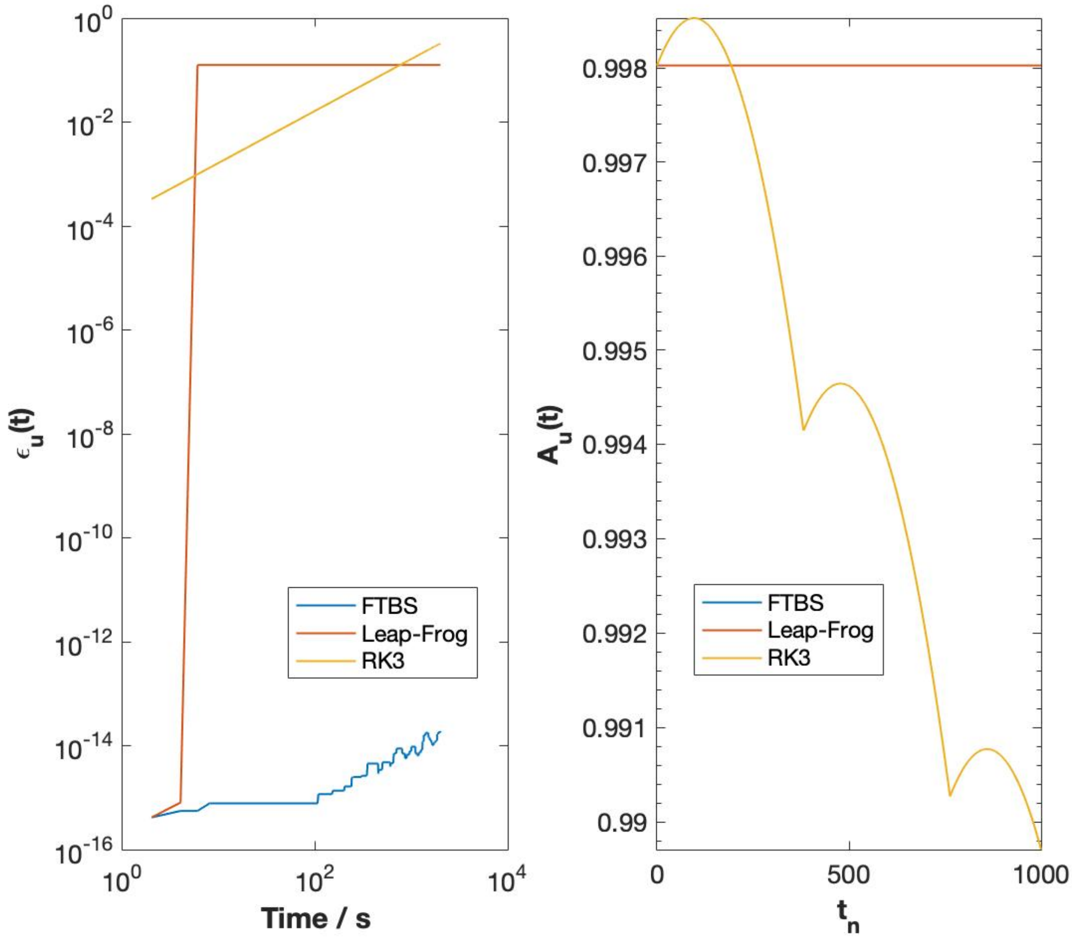


Figure 1B tells us that the error associated with the FTBS solutions in this experiment are minimal and are the result of small round-off errors accumulating in the computation for each time-step. In the case of Leap-Frog, the error is around 0.1 in magnitude but stays at this level throughout the experiment. As the errors for Leap-Frog and FTBS have such a small magnitude, we can see that the overall amplitude of the wave remains unchanged during the 2000 second period of observation. RK3, on the other hand, has an exponentially increasing error due to the increasing degree of phase change experienced by the propagating wave. Although not apparent in Figure 1A, the plot of amplitude for RK3 shows a slight decrease with distinct jumps. In contrast, FTBS and Leap-Frog simulations maintain their amplitude for the duration of the experiment. This deviation for RK3 can be attributed to the trapezoidal corrector component which attempts to make an implicit scheme explicit by establishing a first guess.

$$\frac{u(x,t+\Delta t)}{u(x,t)} = \lambda = e^{\omega_I \Delta t} - e^{i\omega_R \Delta t} = |\lambda| e^{i\omega_R \Delta t} \quad (8)$$

Through stability analyses, it can be concluded that phases errors in particular occur when ω_R for the discrete solution is different from that of the exact solution. As we saw previously, amplitude errors are evident in discrete solutions with $|\lambda| \neq 1$.

Experiment #2 allows us to investigate the effect of changing C by reducing the size of Δt. C is directly proportional to Δt so a decrease in Δt will result in a corresponding decrease in the value of C. We can compare simulations for Δt = 1.0, 2.5 seconds to see how and to what extent the numerical solutions deviate from the analytical solution.

As determined through consistency analyses previously, leading order error term is equal to 0 when 1 - C = 0 (i.e when C = 1). In experiment #1, we saw that when Δx = 1m, Δt = 2 seconds, and c = 0.5 m/s, we have C = 1, in which case there is no numerical diffusion. Experiment #2 effectively shows us what happens when Δt = 1 seconds and C < 1 (i.e, when there is artificial diffusion in the numerical solution). Alternatively, it also shows us the result when Δt = 2.5 seconds and C > 1.

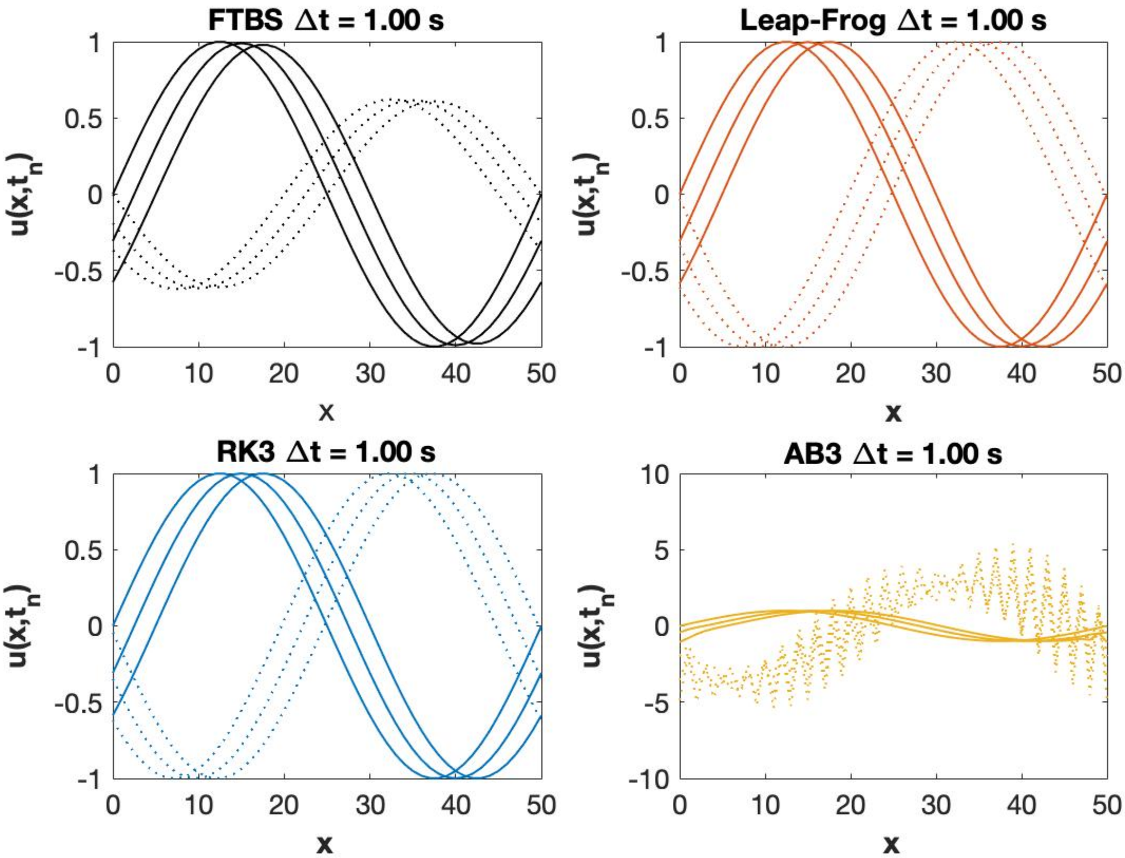


Figure 2A.1 captures the trends observed when the simulation is run for Δt = 1 second. The solid lines show the numerical result during the first few time-steps observed while the dotted lines represent the numerical solution obtained at arbitrary points in the future. The waves for FTBS are dampened. We saw that the smaller the Δt, the more pronounced the dampening effect as FTBS has a leading order error $\mathcal{O}(\Delta t^2, \Delta x^2)$.

This is because the smaller the Δt, the smaller the Courant's number, C, and the greater the artificial diffusion that is manifested in the calculations. In the case of Leap-Frog and RK3, the amplitude is conserved and there appears to be no phase error as well. This is because the leading order error in RK3 is $\mathcal{O}(\Delta t^3, \Delta x^2)$ and that in Leap-Frog is $\mathcal{O}(\Delta x^2)$, both of which are not diffusivity related errors.

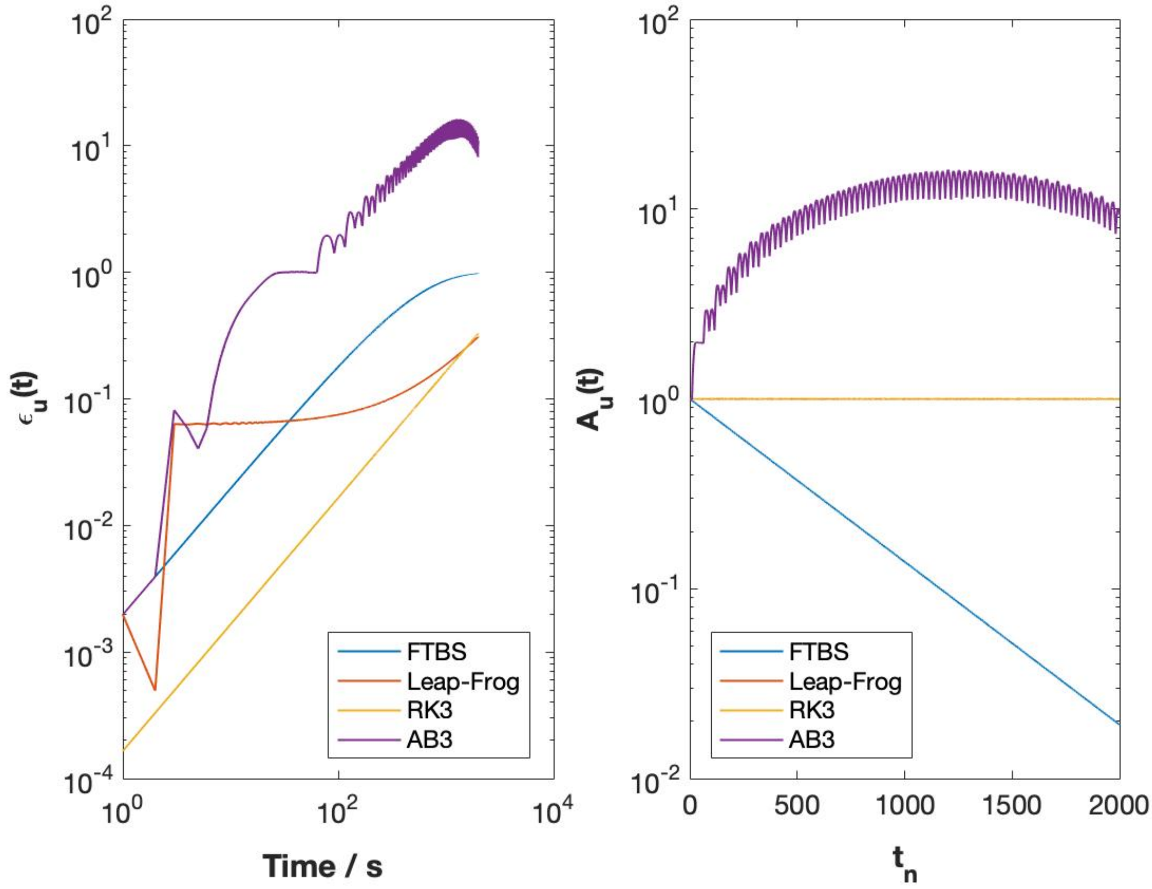


Figure 2A.2 shows how error and amplitude vary over time and time-steps respectively. Once again we see that amplitude is conserved by RK3 and Leap-Frog but is decreasing due to artificial diffusion in the case of FTBS. The associated errors increase exponentially as we progress through time for RK3 and FTBS; however, RK3 is consistently 10 orders of magnitude smaller in error as compared to FTBS — this is until FTBS reaches a maximum error of 1, at which point the wave has been completely suppressed and there are no other sources of error. The Leap-Frog simulation experiences a large jump in the error within the first two time steps. Following this, the rate of error accumulation is slower than for RK3 and FTBS. At the end of 2000 seconds, the error is approximately the same in magnitude as is for RK3.

When we increase the size of Δt, we also increase the value of C.

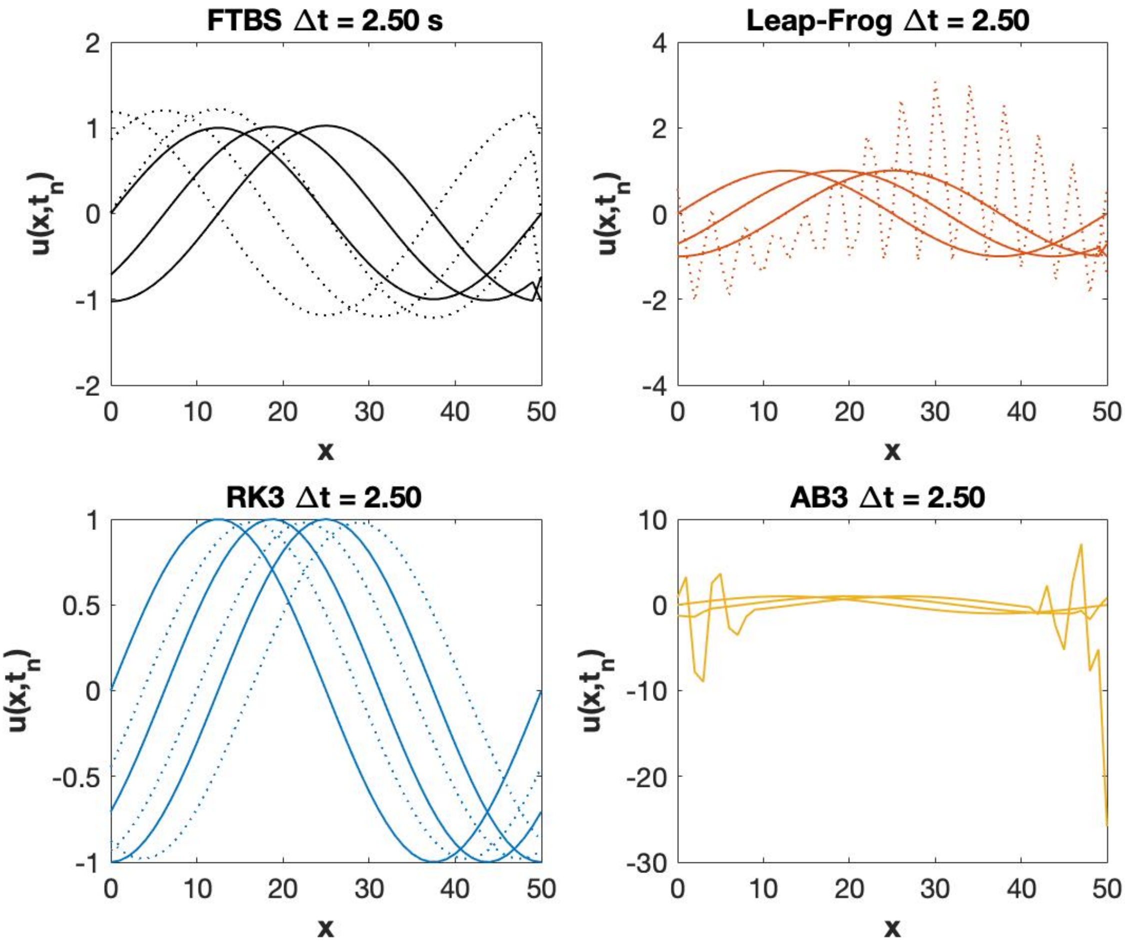


Figure 2B.1 illustrates the case when $\Delta t = 2.50$ seconds. When $C > 1$, we introduce "negative diffusion" which amplifies the amplitude in the FTBS simulation. The greater the value of C, or the greater the value of Δt, the faster this occurs. Leap-Frog also exhibits an amplification but one which oscillates in magnitude and decreases the frequency of the wave. On the other hand, the amplitude in the RK3 simulation remains unchanged and only shows a phase error. This is once attributed to the nature of the leading order error term which suggests less dependence on Δt compared to FTBS and Leap-Frog.

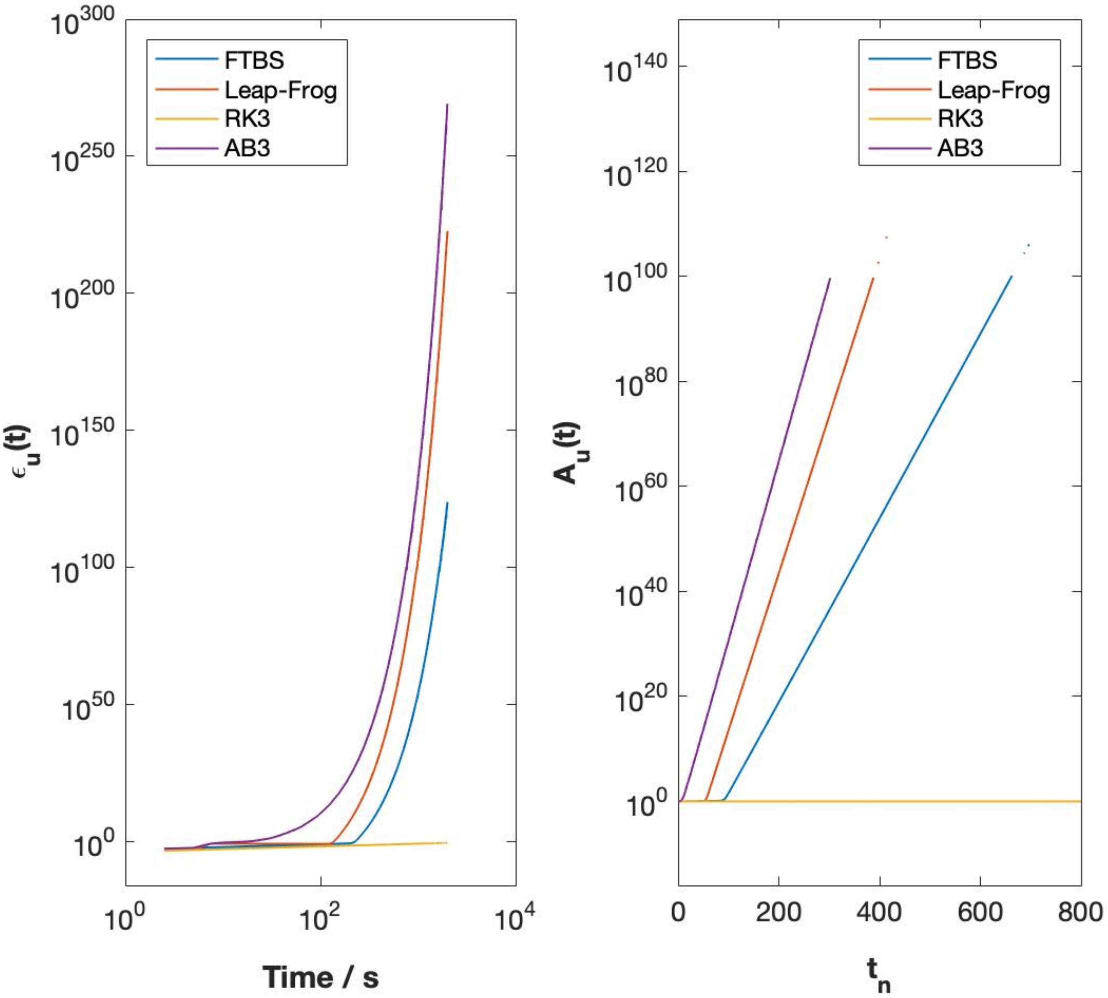


Figure 2B.2 allows us to very clearly trace the particular moment in time and the exact time-step for when the solution expands uncontrollably for FTBS and Leap-Frog. We see that Leap-Frog is the first to "blow up" out of proportion but FTBS is close behind. Once this threshold is crossed, both FTBS and Leap-Frog follow the same trajectory in increasing error; however, the amplitudes increase at different rates, albeit both exponential. This gives insight into the nature of the expansion for the Leap-Frog case which showed both a decrease in frequency of the propagating wave and an uneven rate of increase in amplitude.

Experiment #3 plays with changes in wavelength. By reducing the wavelength to 10 m, we can see the importance of grid size in relation to the distance travelled by the solution in one time-step. This is important as both amplitude and phase errors impact mostly short waves (i.e when λ_x / $\Delta x \leqq 4$). This is why finer grid spacing is a relevant concern.

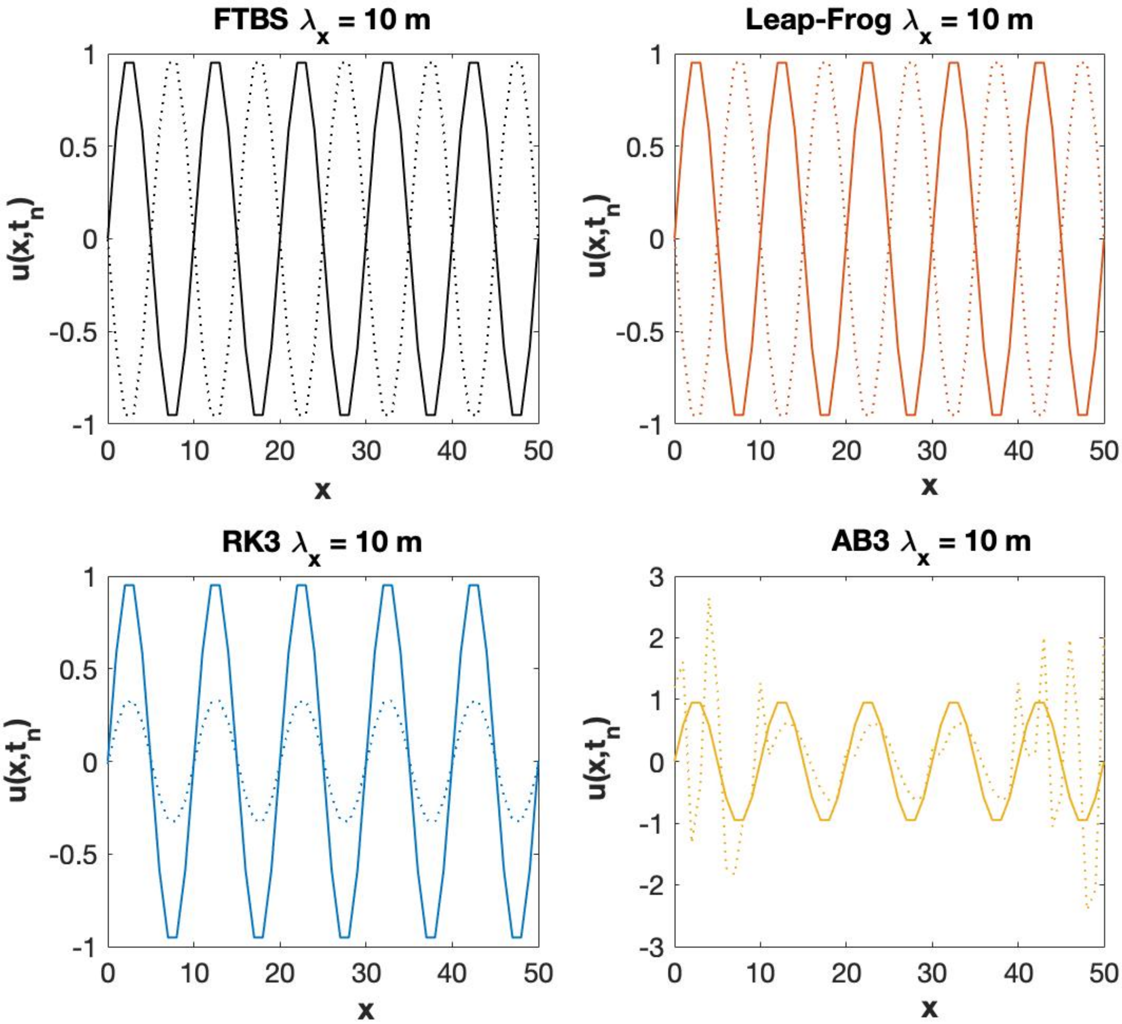


Figure 3A shows the results of all the scheme when λ_x = 10 m. Here, we see the opposite — RK3 is the most unstable as it experiences dampening. In contrast, Leap-Frog and FTBS retain both their amplitude and their phase speed throughout the 2000 seconds of observation. RK3 is a second order scheme centered in space. The stability function of RK3 is polynomial which means we generate two values of λ. Thus, it is difficult for the RK3 method to be stable in terms of amplitude as errors in amplitude stem from instances when $|\lambda| \neq 1$.

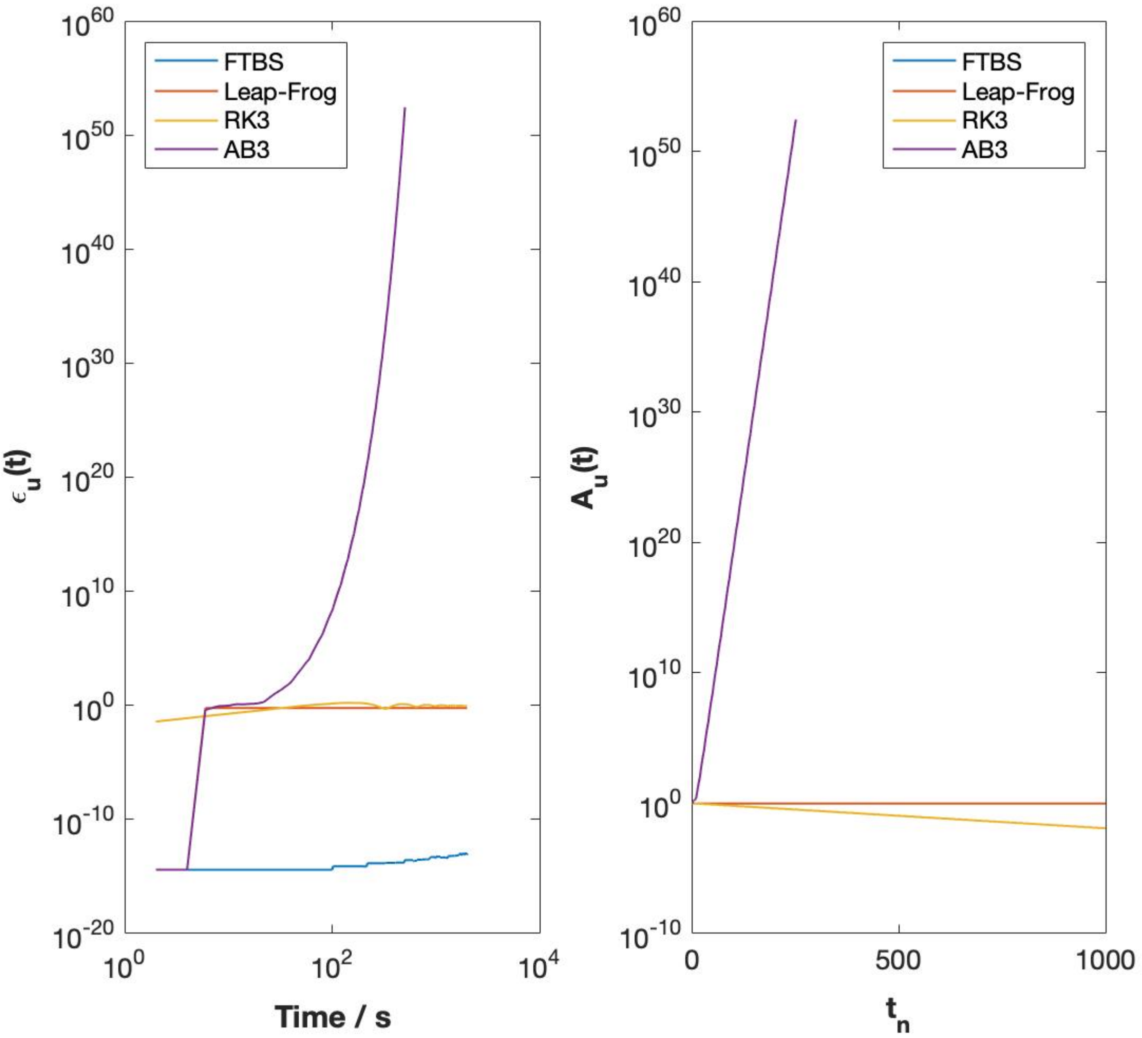


Figure 3B clearly shows the difference in the errors and amplitudes. Amplitude is conserved at 1 for Leap-Frog and FTBS but decreases gradually with he RK3 simulation. Errors in general consequently increase to 1 — until the RK3 wave has been fully dampened.

Finally, experiment #4 tested the case of the "square wave". Here, the initial conditions and analytical solution were derived from using a sign function which equates all values above 0 to +1 and all values below 0 to -1.

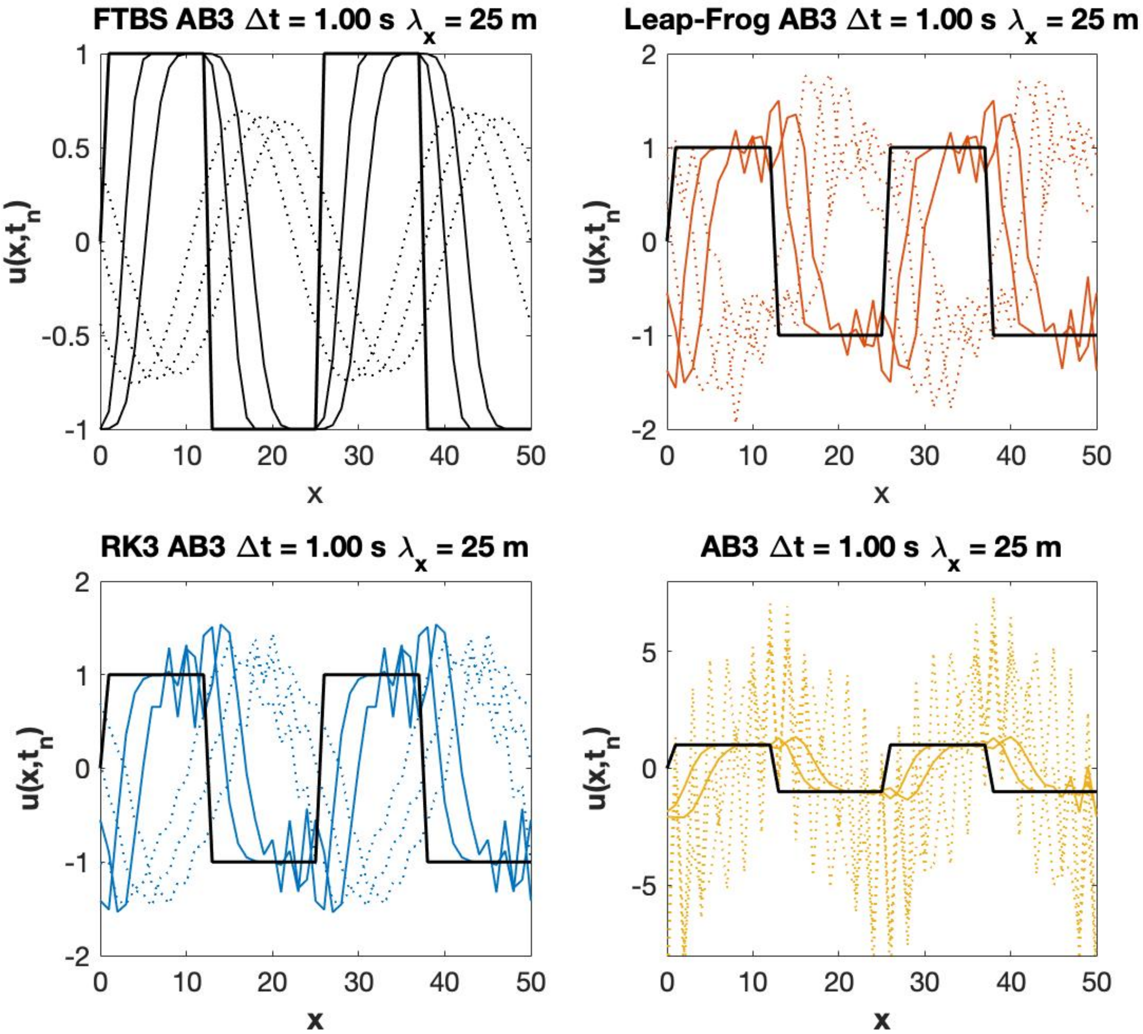


Figure 5A shows that the numerical solutions immediately begin to depart from the square wave initial condition. In FTBS, the 'edges' of the wave erode to add a sinusoidal curvature. The dotted lines, which represent the wave in an arbitrary state in the future, show that once the wave has achieved a sinusoidal shape with amplitude 1, the dampening process begins to act on the amplitude. For Leap-Frog and RK3, the 'edges' of the wave seem to propagate differently and accumulate additional crests which serve to increase the amplitude locally. However, both Leap-Frog and RK3 are not dampened, i.e while the amplitude increases a little on the 'edges' of the wave, the amplitude overall is conserved for the duration of the entire observation period.

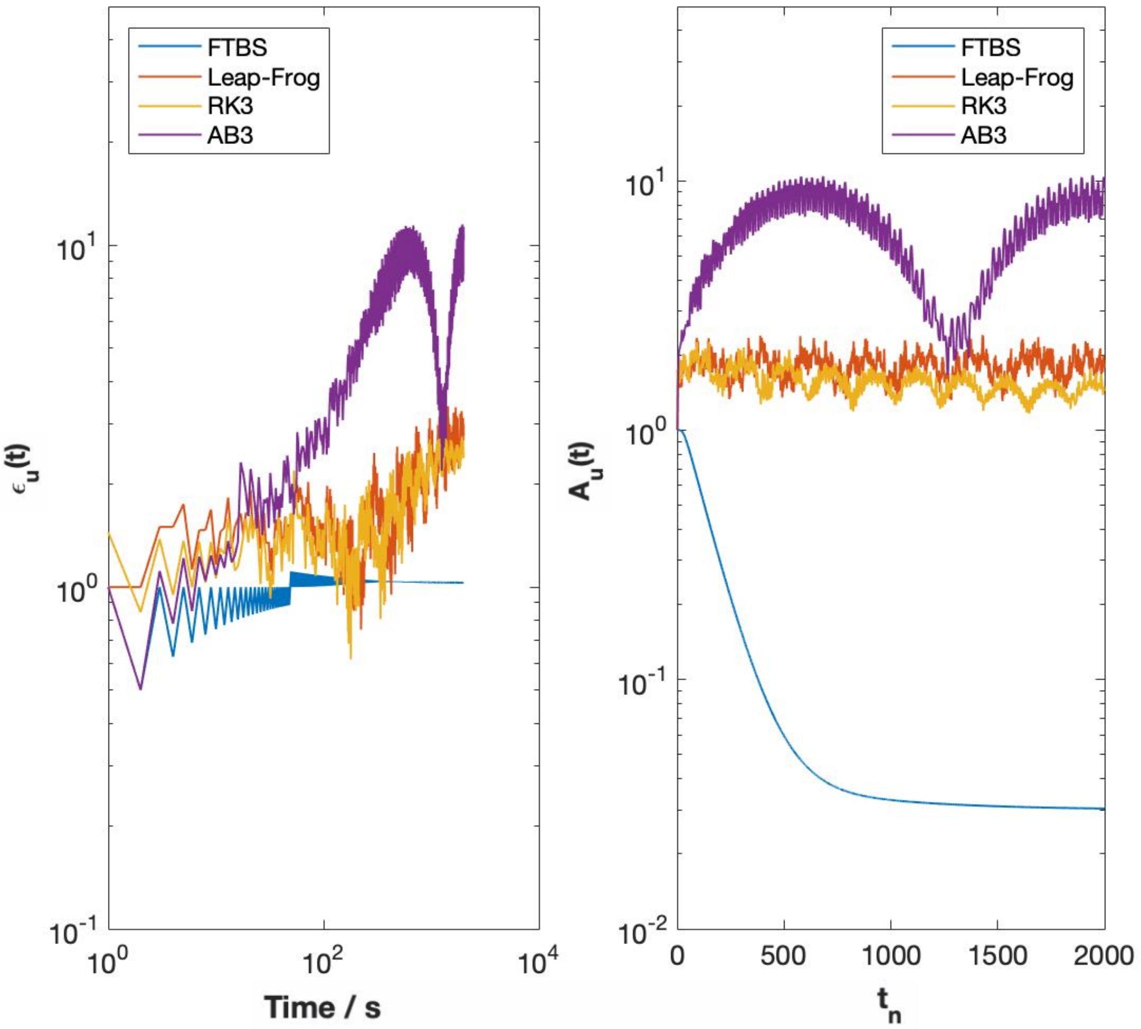


Figure 5B denotes the errors associated with the square wave simulation. The change in the numerical solution is two-fold for FTBS. First, there is a change in the shape of the curve from square to sinusoidal. Here, the amplitude is preserved. As such, the maximum error that is obtained is 1. Once the solution is perfectly sinusoidal, the secondary effect of dampening takes place. From this point on, the lowest error is 1. Errors thereafter fluctuate due to the exact solution being a square wave. We see that this second effect also corresponds to a rapid exponential decay in the amplitude of the numerical solution. For Leap-Frog and RK3, the amplitude oscillates within a certain range that is above 1 (around 1.5). This is because the propagation of the 'edge' of the square tends to increase until a certain peak and then start again at 0. This "catching up" of the edge with the curvature of the rest of the wave is the reason for the crests and the troughs in the graph of the error. The jagged nature of the plot mirrors the jagged-ness of the 'edge' when simulated using RK3 and Leap-Frog.

The table below encapsulates key statistics associated with each run of each experiment. C was calculated directly from its definition: $C = c\,\Delta t / \Delta x$. $\lambda_x / \Delta x$ denotes the ratio of the wavelength to grid size, which was important particularly in the discussion of experiment #3.

Experiment	**Trial**	**C**	**$\lambda_x / \Delta x$**
#1	$\Delta t = 2.00$ s	1	50
#2	$\Delta t = 1.00$ s	0.5	50
	$\Delta t = 2.50$ s	1.25	50
#3	$\Delta t = 2.00$ s	1	10
#4	$\Delta t = 1.00$ s	0.5	50