

HOMEWORK 1 - Numerical Relativity 2023 - 2024

In the first sections of this homework, I will apply different numerical methods to solve the advection equation, and compute the evolution of an initial function on a grid with extent $x \in [0, 10]$ for a time $t \in [0, 20]$. In particular, the initial function is firstly chosen as a Gaussian, then substituted with a step function. By applying different methods using different initial conditions, I want to show the different properties of the methods, highlighting their strengths and problems. Specifically, I am going to focus on the production of oscillations and deviations from the expected results (also by checking with the L_2 norm).

In the last section, I will solve Burgers' equation and compute the evolution of an initial function on a grid with extent $x \in [0, 10]$ for a time $t \in [0, 20]$. In particular, the initial function will be a Gaussian, and my analysis will focus on underlining the differences in the results obtained using both the flux-conservative and the non flux conservative versions of the upwind scheme.

ADVECTION EQUATION - GAUSSIAN FUNCTION

Firstly, I will concentrate on the evolution of an initial Gaussian function given by:

$$u(x, t = 0) = \exp[-(x - x_0)^2] \quad (1)$$

with $x_0 = 5$. To evolve the system following the advection equation, I considered four different methods: FTCS, Lax-Friedrichs, Leapfrog and Lax-Wendroff. For each scheme, I varied some parameters to highlight the dependence of the quality of the result on them. In particular, I modified the number of points J in which I define my function and the Courant Factor c_f - consequently modifying the parameters $\Delta x = \frac{10}{(J-1)}$ and $\Delta t = c_f \frac{\Delta x}{a}$ -.

1 FTCS

Starting with the FTCS method - which is first order in time and second order in space -, the evolution of the initial function is given by equation 2:

$$u_{j+1}^{n+1} = u_j^n - \frac{a\Delta t}{2\Delta x} (u_{j+1}^n - u_{j-1}^n) \quad (2)$$

Firstly, I chose the following parameters: $J = 101$ and $c_f = 0.5$, obtaining the following results:

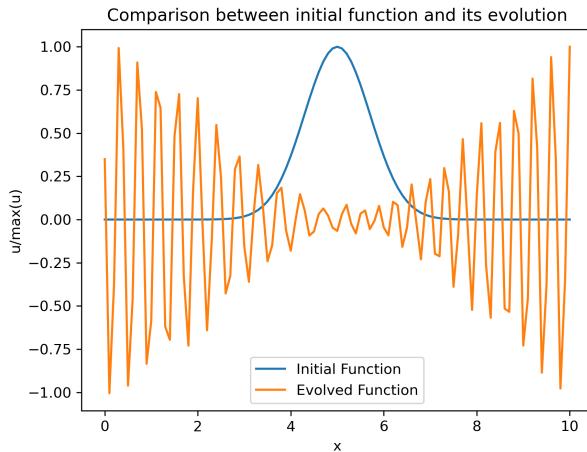


Figure 1: Comparison between the initial function and its evolution, both normalized for their maximum value.

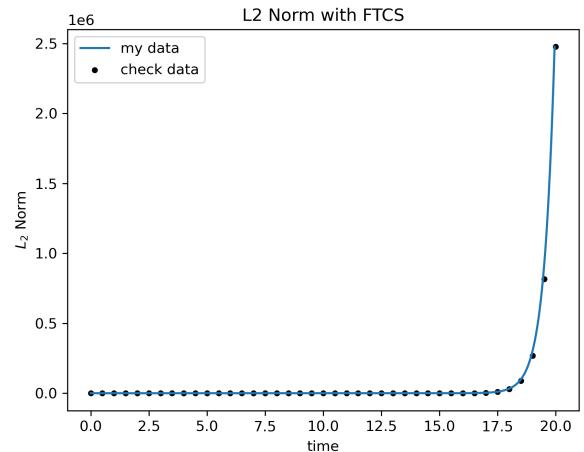


Figure 2: L_2 norm of the function computed at each step. Note that the norm diverges, reaching values of the order of $1e6$.

As we can see, the method fails to reproduce the analytical solution of the advection equation, introducing oscillations that reach orders of magnitude of $1e6$. This also happens if the accuracy in the sampling of the x is increased, where the oscillations reach orders of magnitude of $1e152$.

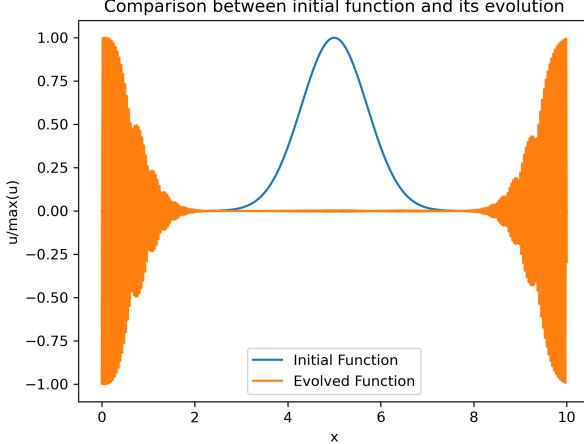


Figure 3: Comparison between the initial function and its evolution, both normalized for their maximum value.

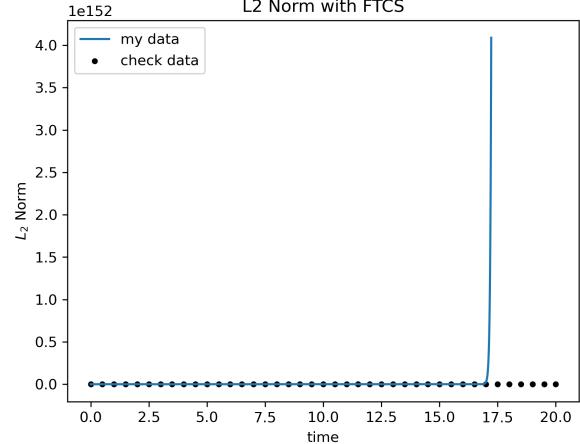


Figure 4: L_2 norm of the function computed at each step. Note that the norm diverges, reaching values of the order of $1e152$.

This can be explained by the fact that the method is unconditionally unstable since it does not satisfy the von Neumann stability analysis, a method used to determine the stability of numerical schemes for solving partial differential equations (PDEs). In fact, substituting in the trial solution $u_j^n = \xi^n e^{ikx_j}$ ¹ in equation 2, the following condition is obtained:

$$|\xi|^2 = 1 + \left[a \frac{\Delta t}{\Delta x} \sin(k\Delta x) \right]^2 > 1 \quad \forall k \quad (3)$$

which tells us that the amplitude of the numerical error is growing uncontrollably, thus making the method unstable.

2 LAX-FRIEDRICHES

The Lax-Friedrichs method applied to the advection equation gives us the following equation to compute the next step in time:

$$u_j^{n+1} = \frac{1}{2} (u_{j+1}^n + u_{j-1}^n) - \frac{a\Delta t}{2\Delta x} (u_{j+1}^n - u_{j-1}^n) \quad (4)$$

obtaining these results for $J = 101$ and $c_f = 0.5$:

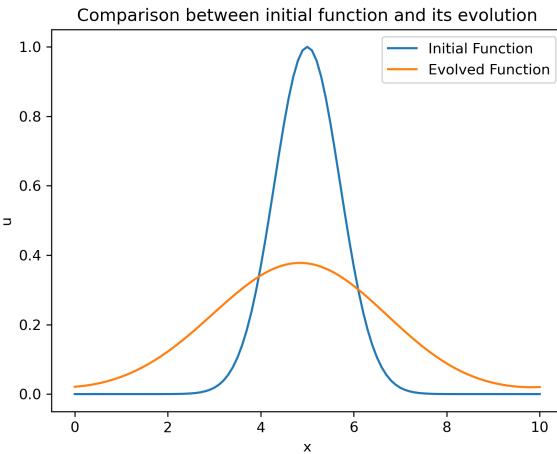


Figure 5: Comparison between the initial function and its evolution, both normalized for their maximum value.

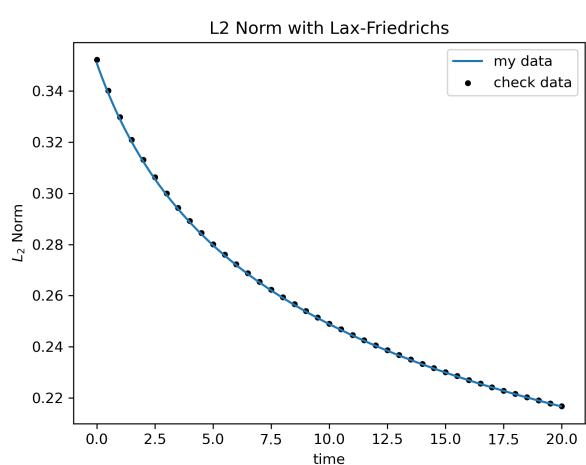


Figure 6: L_2 norm of the function computed at each step.

We can already notice that even with this method the L_2 norm of the function does not remain constant: it actually decreases during the evolution of the function. This situation improves but is still present if sampling the x domain with a larger amount of points - e.g. setting $J = 1001$ -, as it can be seen in the following figures:

¹Here using the Fourier transform of the function $u(x, t)$.

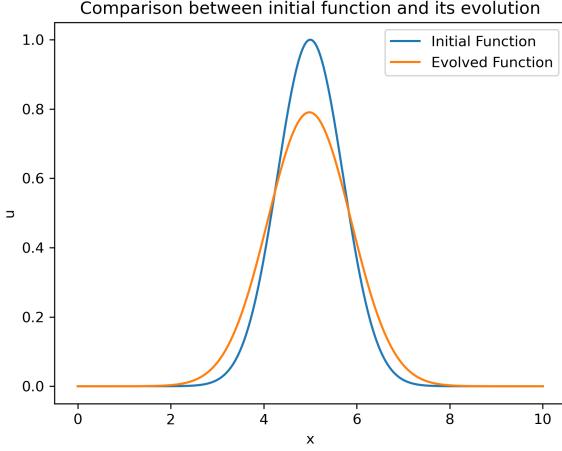


Figure 7: Comparison between the initial function and its evolution, both normalized for their maximum value.

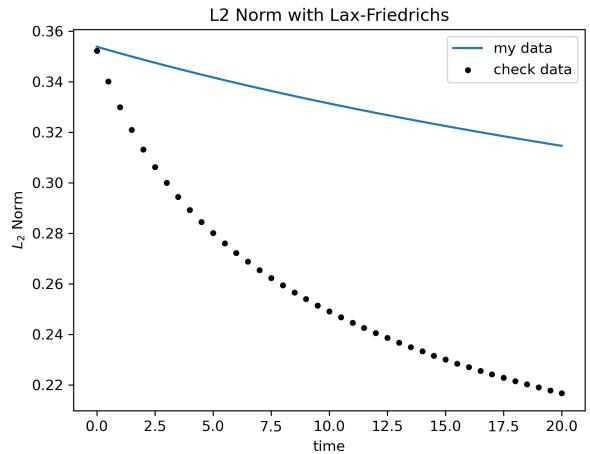


Figure 8: L_2 norm of the function computed at each step.

In the case of the Lax Friedrichs method, there are no oscillations that grow uncontrollably. In fact, if we consider the von Neumann stability analysis, we can notice that the method is conditionally stable. More specifically, substituting the trial solution $u_j^n = \xi^n e^{ikx_j}$ into equation 4, we obtain the following expression for $|\xi|^2$:

$$|\xi|^2 = 1 - \sin(k\Delta x)^2 \left[1 - \left(a \frac{\Delta t}{\Delta x} \right)^2 \right] \quad (5)$$

which results in the Courant-Friedrichs-Lowy (CFL) condition for the method to be stable:

$$|a| \frac{\Delta t}{\Delta x} \leq 1 \quad (6)$$

Since in the code $\Delta t = c_f \frac{\Delta x}{|a|}$, this condition translates in the following: $c_f \leq 1$. We can see that increasing the Courant factor to values higher than 1 makes the method unstable and therefore allows the amplitude to grow uncontrollably in the following images, where I set $c_f = 1.1$ and $J = 101$:

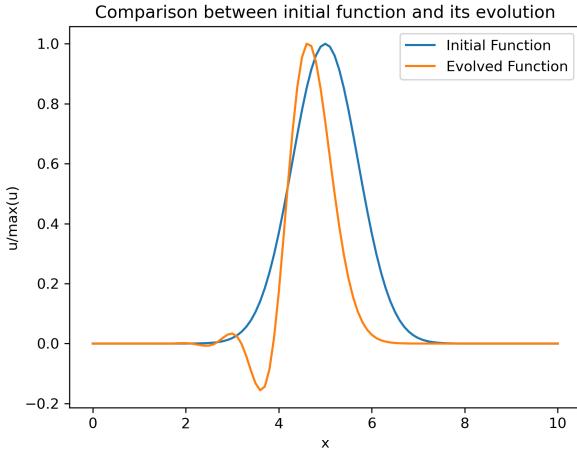


Figure 9: Comparison between the initial function and its evolution, both normalized for their maximum value.

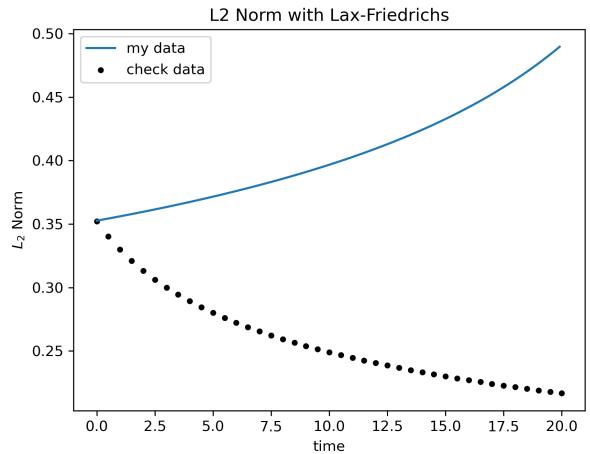


Figure 10: L_2 norm of the function computed at each step.

Moreover, even if the CFL condition is satisfied, we can notice that using the Lax-Friedrichs method the function decreases in amplitude during its evolution. This is due to the presence of a dissipative term $\frac{\partial^2 u}{\partial x^2}$. In fact, we can rewrite the Lax-Friedrich scheme as follows, showing that we are actually solving the advection equation with the addition of a dissipative term:

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = \frac{1}{2} \frac{|a|}{c_f} \Delta x \frac{\partial^2 u}{\partial x^2} + \mathcal{O}(\Delta x^2) \quad (7)$$

3 LEAPFROG

Another method that can be applied to the advection equation is the leapfrog scheme, which is a second order approximation method in both space and time and returns the following equations for the function evolution:

$$\begin{cases} u_j^{n+1} = \frac{1}{2} (u_{j+1}^n + u_{j-1}^n) - \frac{a\Delta t}{2\Delta x} (u_{j+1}^n - u_{j-1}^n) & \text{if } n = 0 \\ u_j^{n+1} = u_j^n - \frac{a\Delta t}{2\Delta x} (u_{j+1}^n - u_{j-1}^n) & \text{otherwise} \end{cases} \quad (8)$$

where in the case of $n = 0$ I used the Lax-Friedrichs method to overcome the necessity of the Leapfrog scheme to know the system's state at the stage $n = -1$ - which I do not have. Already in the case with $J = 101$ and $c_f = 0.5$, we can see an improvement on the evolution of the initial function with respect to the previous cases - as it can be seen in the following figures:

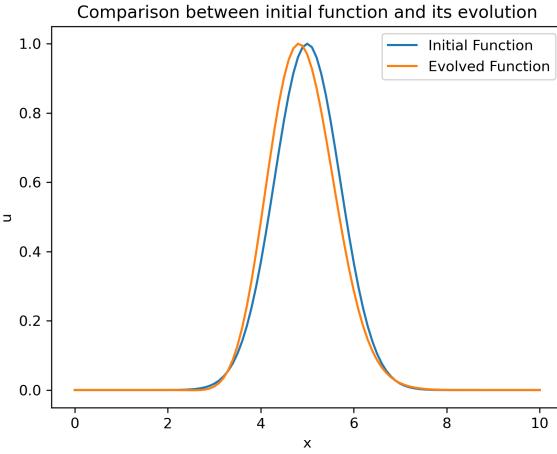


Figure 11: Comparison between the initial function and its evolution, both normalized for their maximum value.

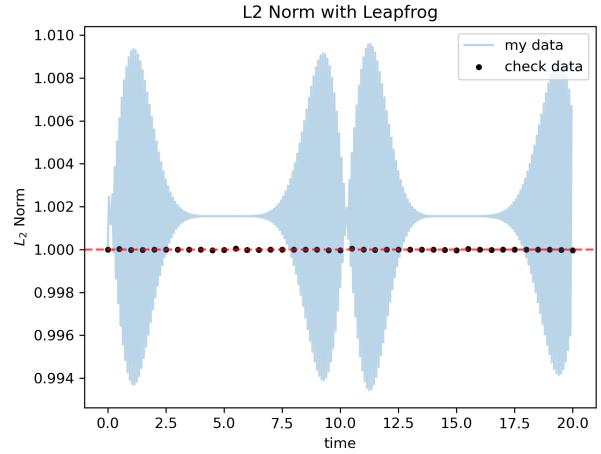


Figure 12: L_2 norm of the function computed at each step, normalized to its initial value.

The situation gets even closer to the analytical solution if I increment the sampling parameter. By setting $J = 1001$, the following results are obtained:

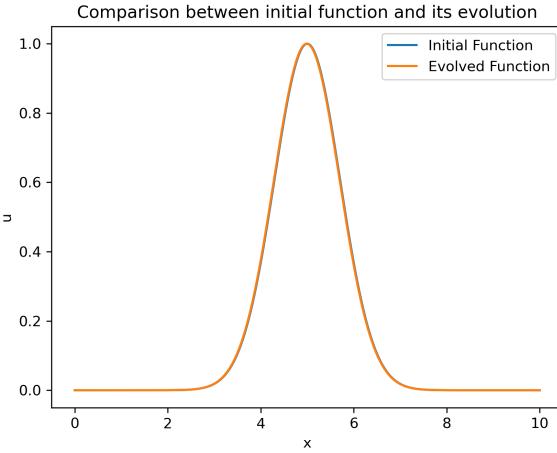


Figure 13: Comparison between the initial function and its evolution, both normalized for their maximum value.

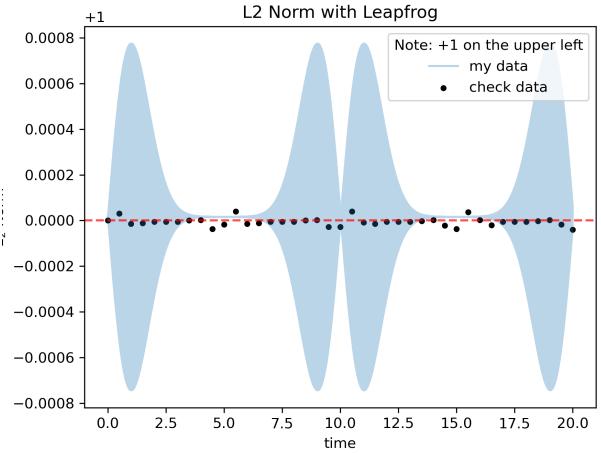


Figure 14: L_2 norm of the function computed at each step, normalized to its initial value.

As the previous method, if we consider the von Neumann stability analysis to check under which conditions the Leapfrog method is stable, we still recover the CFL condition $c_f \leq 1$.

4 LAX-WENDROFF

Another second order both in time and space method with which I can evolve an initial function following the advection equation is the Lax-Wendroff method, which returns the following relation for the evolution of the function:

$$u_j^{n+1} = u_j^n - \frac{a\Delta t}{2\Delta x} (u_{j+1}^{n+1} - u_{j-1}^{n+1}) + \frac{a^2\Delta t^2}{2\Delta x^2} (u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}) \quad (9)$$

As the previous method, the Lax-Wendroff is able to get close to the analytical solution. However, as it can be already seen from the following figures, the solution shows some oscillations.

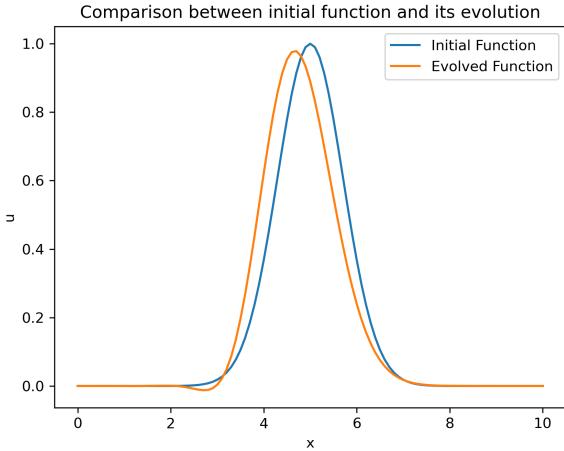


Figure 15: Comparison between the initial function and its evolution, both normalized for their maximum value.

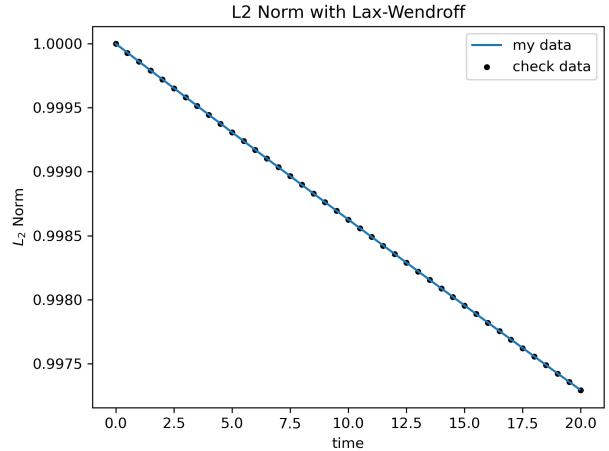


Figure 16: L_2 norm of the function computed at each step, normalized to its initial value.

The consistency with the analytical solution increases while increasing the resolution of the space grid. This can be seen in the following figures, obtained by setting $J = 1001$ and $c_f = 0.5$.

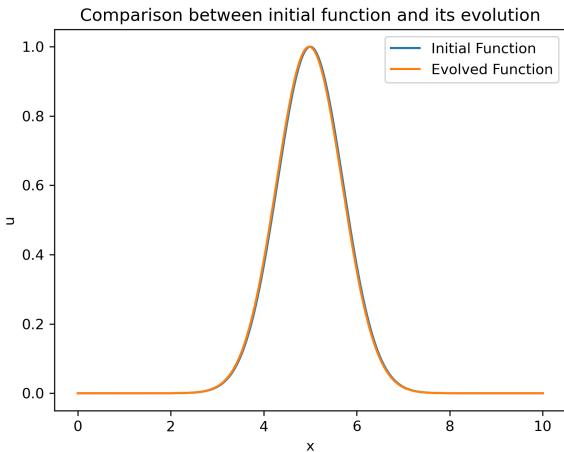


Figure 17: Comparison between the initial function and its evolution, both normalized for their maximum value.

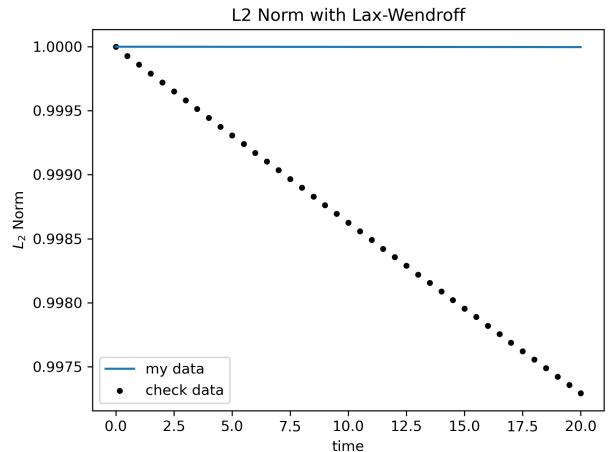


Figure 18: L_2 norm of the function computed at each step, normalized to its initial value.

The presence of oscillation can be understood by Taylor expanding equation 9, after which we obtain the following relation:

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = -\frac{1}{6} a \Delta x^2 \left(1 - a^2 \frac{\Delta t^2}{\Delta x^2} \right) \frac{\partial^3 u}{\partial x^3} + \frac{a^2 \Delta t \Delta x^2}{24} \left(1 - a^2 \frac{\Delta t^2}{\Delta x^2} \right) \frac{\partial^4 u}{\partial x^4} + \mathcal{O}(\Delta x^4) \quad (10)$$

where the term with $\frac{\partial^3 u}{\partial x^3}$ is a dispersive term while the $\frac{\partial^3 u}{\partial x^3}$ one is the dissipative term - subdominant with respect to the other one. The presence of a dispersive term implies that not all the points will move at the same velocity, therefore little oscillations in other directions will happen.

For the stability, we can still use the von Neumann stability analysis to obtain that also this method is conditionally stable, where the condition to satisfy is always the CFL condition $c_f \leq 1$. In fact, substituting in the trial solution $u_j^n = \xi^n e^{ikx_j}$ in equation 9, we obtain the following expression for $|\xi|^2$:

$$\xi^2 = 1 + c_f^2 (c_f^2 - 1) [1 - \cos(k \Delta x)]^2 \quad (11)$$

ADVECTION EQUATION - STEP FUNCTION

Now, I will concentrate on the evolution of an initial step function given by:

$$u(x, t = 0) = \begin{cases} 1 & \text{if } x \in [4, 6] \\ 0 & \text{elsewhere} \end{cases} \quad (12)$$

To evolve the system following the advection equation, I considered two different methods: Lax-Friedrichs and Lax-Wendroff. For each scheme, I varied some parameters to highlight the dependence of the quality of the result on them. In particular, I modified the number of points J in which I define my function and the Courient Factor c_f - consequently modifying the parameters $\Delta x = \frac{10}{(J-1)}$ and $\Delta t = c_f \frac{\Delta x}{a}$.

5 LAX-FRIEDRICH

The Lax-Friedrichs method applied to the advection equation gives us the following equation to compute the next step in time:

$$u_j^{n+1} = \frac{1}{2} (u_{j+1}^n + u_{j-1}^n) - \frac{a\Delta t}{2\Delta x} (u_{j+1}^n - u_{j-1}^n) \quad (13)$$

obtaining these results for $J = 101$ and $c_f = 0.5$:

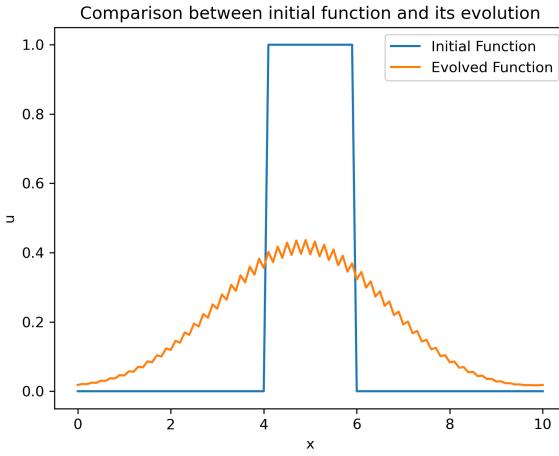


Figure 19: Comparison between the initial function and its evolution, both normalized for their maximum value.

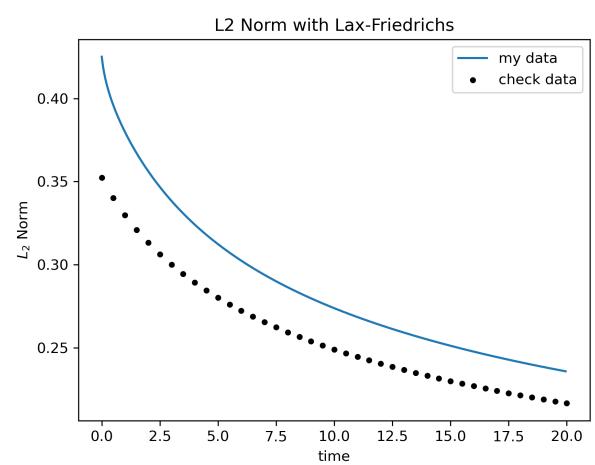


Figure 20: L_2 norm of the function computed at each step.

We can already notice that even in the step function scenario the L_2 norm of the function does not remain constant: it actually decreases during the evolution of the function. This situation improves but is still present if we sample the x domain with a larger amount of points - e.g. setting $J = 1001$ -, as it can be seen in the following figures:

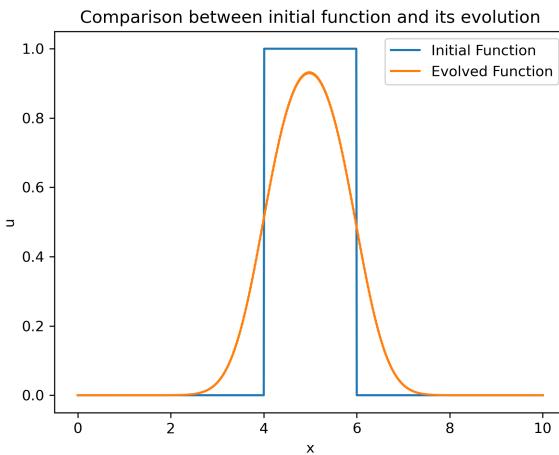


Figure 21: Comparison between the initial function and its evolution, both normalized for their maximum value.

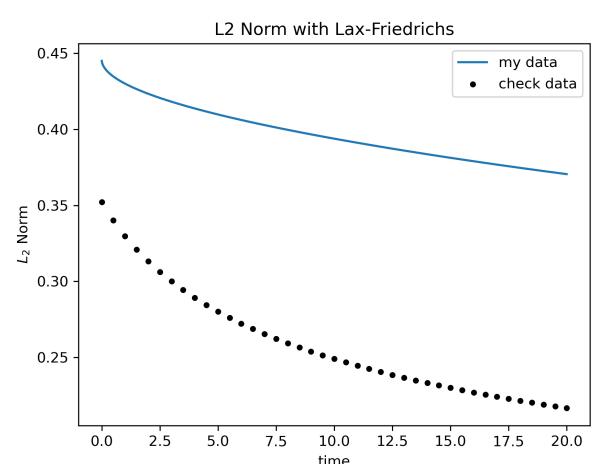


Figure 22: L_2 norm of the function computed at each step.

The dissipation present with this method is explained in section 2. However, in this scenario the impact of the dissipation is a lot more evident, with the step function being smoothed out into a shape similar to a Gaussian. Moreover, we can notice that there are no oscillations, which is a consequence of the monotonic nature of the Lax-Friedrichs scheme. In fact, according to Godunov's theorem, linear monotonic schemes suppress oscillations, at the cost of being restricted to first-order accuracy.

6 LAX-WENDROFF

We can also evolve the initial step function with the Lax-Wendroff method, which returns the following relation for the evolution of the function:

$$u_j^{n+1} = u_j^n - \frac{a\Delta t}{2\Delta x} (u_{j+1}^{n+1} - u_{j-1}^{n+1}) + \frac{a^2\Delta t^2}{2\Delta x^2} (u_{j+1}^{n+1} - 2u_j^n + u_{j-1}^{n+1}) \quad (14)$$

As it can be already seen from the following figures, the solution shows some oscillations, as we would expect from a 2nd order linear scheme due to the Godunov Theorem.

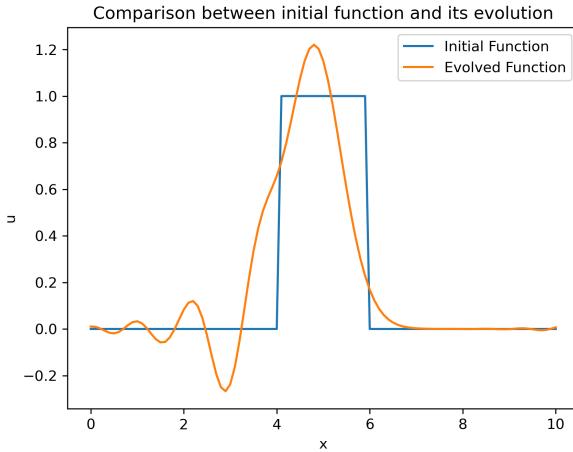


Figure 23: Comparison between the initial function and its evolution, both normalized for their maximum value.

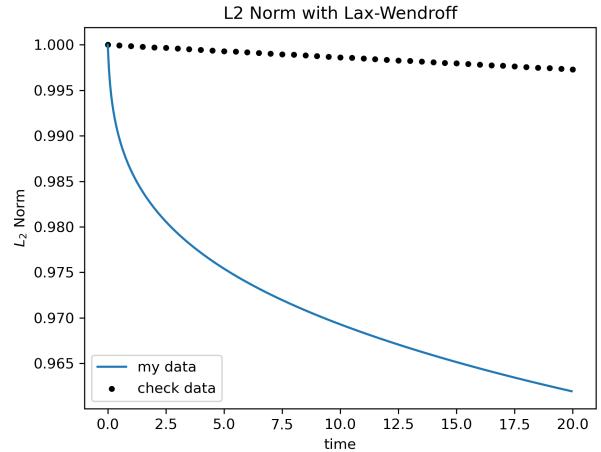


Figure 24: L_2 norm of the function computed at each step, normalized to its initial value.

The consistency with the analytical solution increases while increasing the resolution of the space grid. This can be seen in the following images, obtained by setting $J = 1001$ and $c_f = 0.5$.

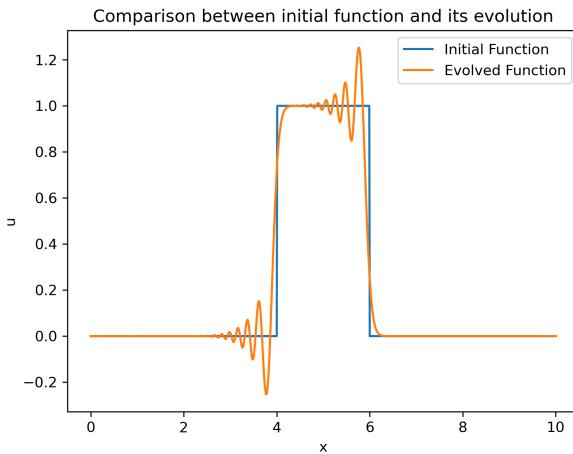


Figure 25: Comparison between the initial function and its evolution, both normalized for their maximum value.

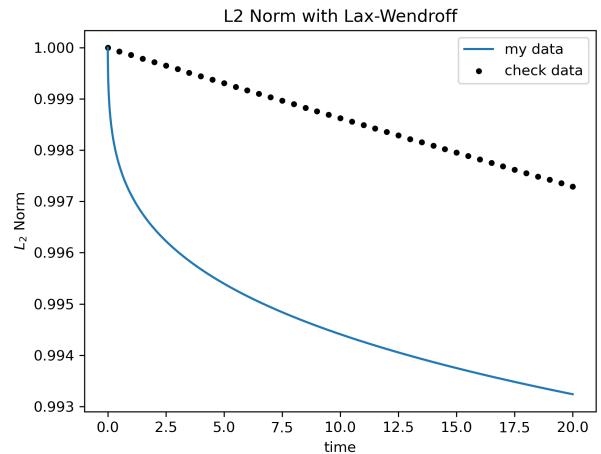


Figure 26: L_2 norm of the function computed at each step, normalized to its initial value.

Even in this case the impact of dissipative and dispersive terms are more relevant, with the effects being most dominant along the points at which the step function changes its value. In this case, we have that oscillations are present, due to the dispersion of the wave pack because of the different velocities.

Another interesting thing that can be seen is the case with $c_f = 1$. As a matter of fact, using this specific value for c_f leads to the step function remaining always the same. This happens because substituting $\Delta t = c_f \Delta x a$, where a is

the velocity with $c_f = 1$ in the formula of the Lax-Wendroff scheme, the function at the successive time step remains equal to the function at the precedent time step.

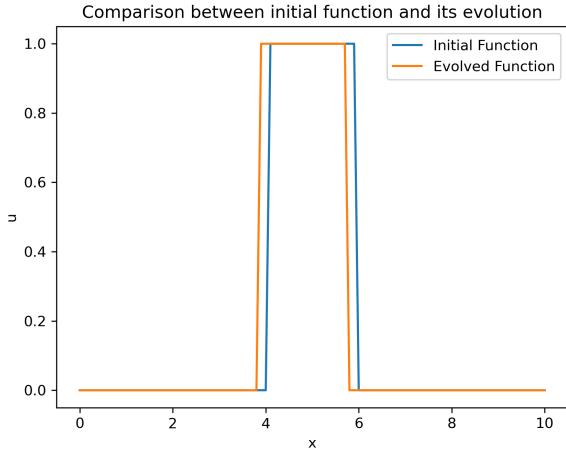


Figure 27: Comparison between the initial function and its evolution, both normalized for their maximum value.

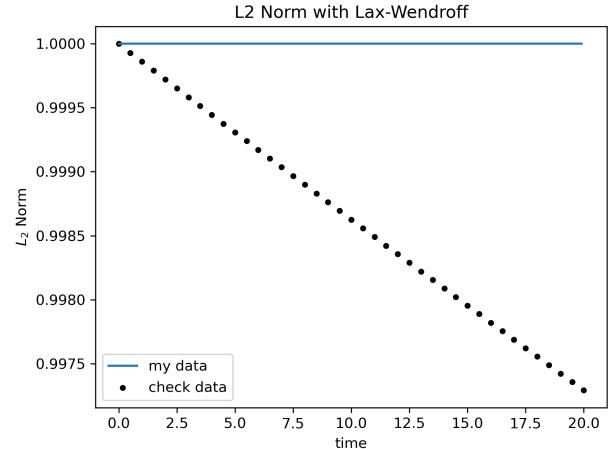


Figure 28: L_2 norm of the function computed at each step, normalized to its initial value.

BURGERS' EQUATION

To study the Burgers' equation, I will concentrate on the evolution of an initial Gaussian function given by:

$$u(x, t = 0) = 10 \cdot \exp [-(x - x_0)^2] \quad (15)$$

with $x_0 = 5$. In particular, we will use the upwind scheme both for the non flux-conservative version and the flux-conservative version, in order to underline the differences, their strengths and weaknesses.

7 NON FLUX-CONSERVATIVE

Firstly, we will consider the non flux-conservative version, where the partial differential equation is written as:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0 \quad (16)$$

For this scenario, the evolution of the initial function is given by the following expression:

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x} \cdot u_j^n \cdot (u_j^n - u_{j-1}^n) \quad (17)$$

At first, we consider $J = 101$ and $c_f = 0.5$. We can notice the formation of a shock wave, and that the L_2 norm is decreasing during the evolution.

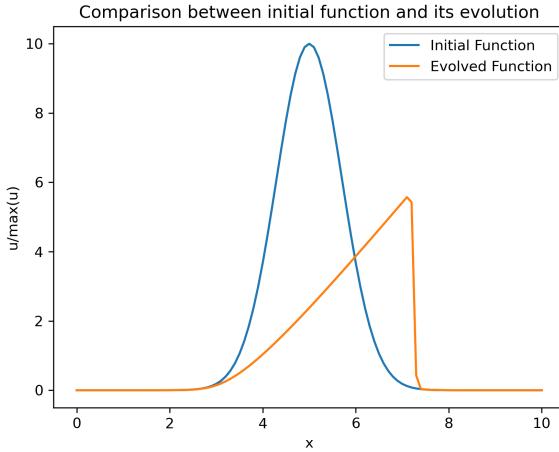


Figure 29: Comparison between the initial function and its evolution.

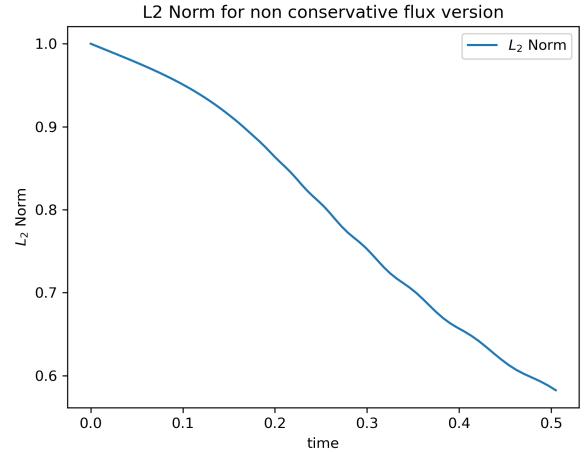


Figure 30: L_2 norm of the function computed at each step, normalized to the initial value.

Then, we increasing the sampling in the space dimension, setting $J = 1001$, obtaining a better resolution. In fact, the position of the shock wave and the shock wave itself are better defined with respect to the previous case, while the L_2 norm decreases slightly slower.

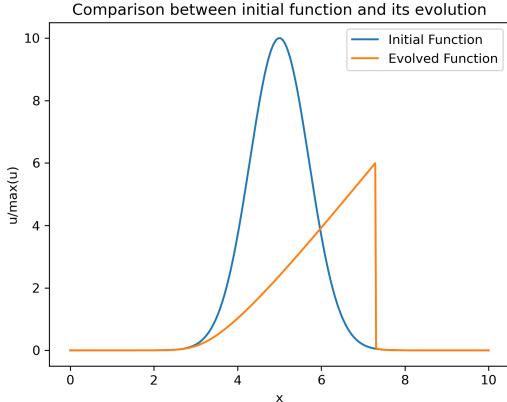


Figure 31: Comparison between the initial function and its evolution.

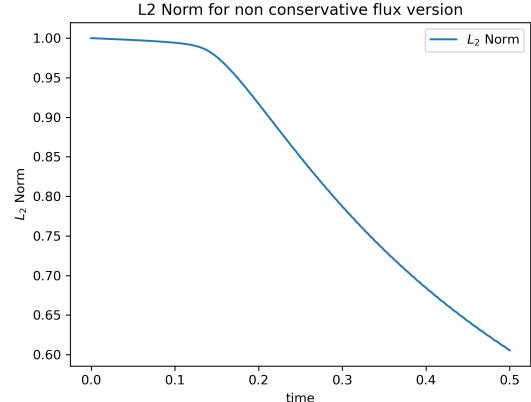


Figure 32: L_2 norm of the function computed at each step, normalized to its initial value.

8 FLUX-CONSERVATIVE

Then, we consider the flux-conservative version of the Burgers' equation, given by:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad \text{where } f(u) = \frac{1}{2}u^2 \quad (18)$$

The upwind scheme applied to the flux-conservative version returns the following equation for the evolution of the function:

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x} \left[\frac{1}{2}(u_j^n)^2 - \frac{1}{2}(u_{j-1}^n)^2 \right] \quad (19)$$

At first, I used $J = 101$ and $c_f = 0.5$. It is already noticeable that the result obtained in this case differs from the one described in the previous section - both in height and position of the shock wave. This can also be seen by the L_2 norm, which decreases less than the non conservative-flux version scenario.

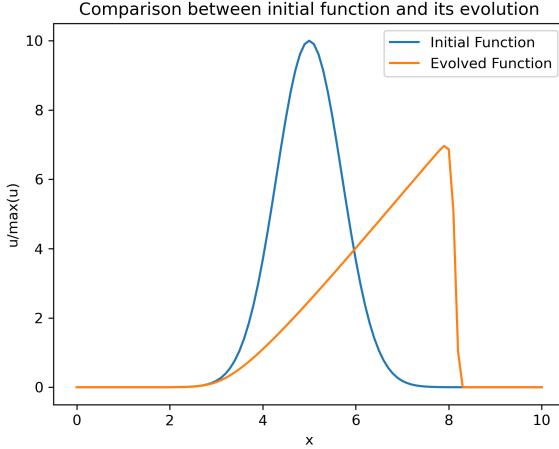


Figure 33: Comparison between the initial function and its evolution.

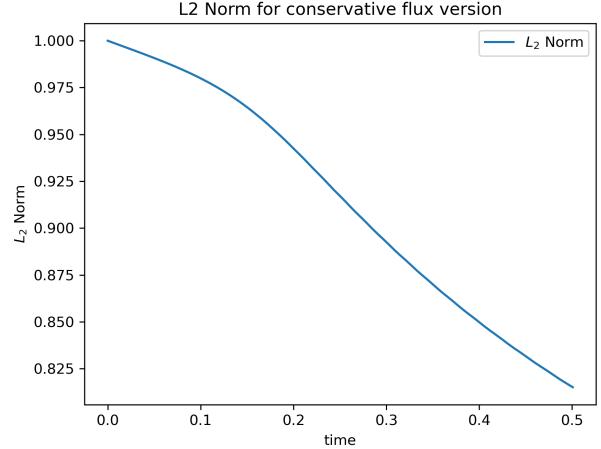


Figure 34: L_2 norm of the function computed at each step, normalized to its initial value.

As the previous scenario, the resolution increases while sampling the space component with more points. In fact, setting $J = 1001$ we obtain the following result:

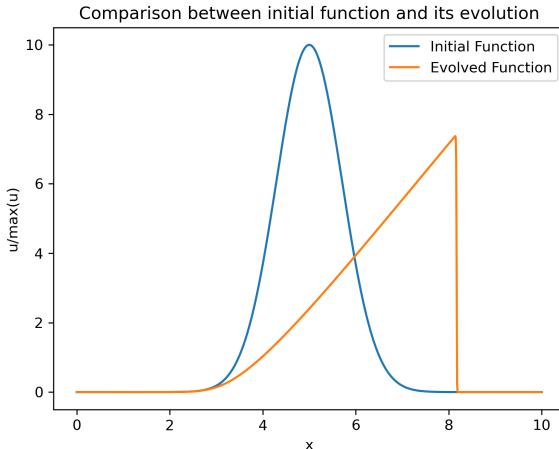


Figure 35: Comparison between the initial function and its evolution.

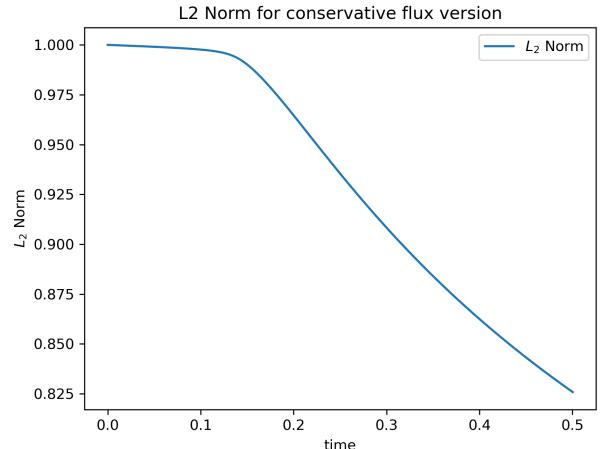


Figure 36: L_2 norm of the function computed at each step, normalized to its initial value.

9 COMPARISON FLUX CONSERVATIVE - NON FLUX CONSERVATIVE VERSIONS

Seen the previous sections, a comparison between the different version gives back interesting results. As we can see from the following figures, the generated shock wave evolves differently - and this happens for different timesteps and for both the sampling scenarios. In particular, we can notice a difference in both the height and the position of the resulting shock wave between the non flux-conservative and the flux-conservative versions.

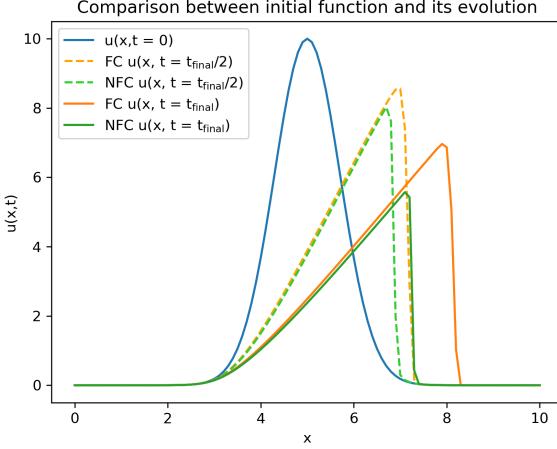


Figure 37: Comparison between the non flux-conservative and the flux-conservative version evolution at different times for $J = 101$.

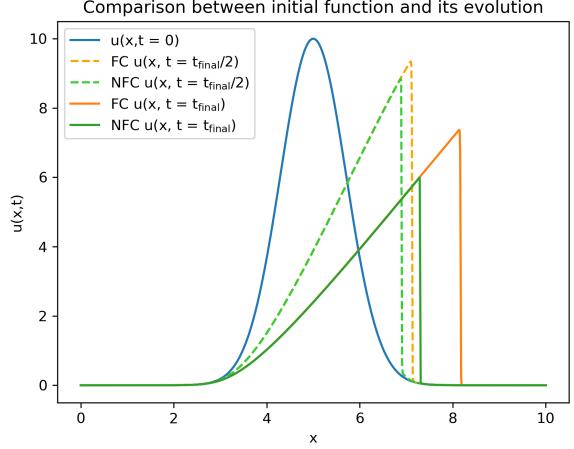


Figure 38: Comparison between the non flux-conservative and the flux-conservative version evolution at different times for $J = 1001$.

Regarding the position of the shock wave, we can notice that the non-flux conservative version evolution results in the shock wave being always behind with respect to the flux-conservative version one. Moreover, an increased sampling does not change the position of the shock wave, while it delays the moment at which the position of the shock wave starts to differ in the two evolutions.

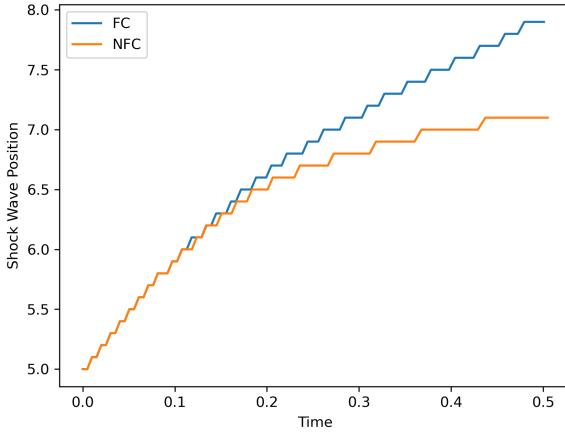


Figure 39: Comparison between the position of the shock wave for the non flux-conservative and the flux-conservative version evolution for $J = 101$.

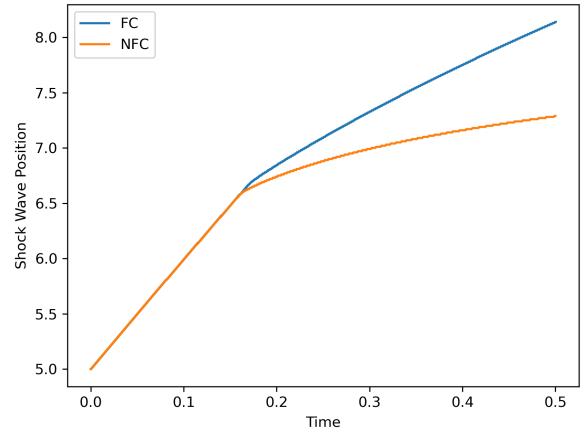


Figure 40: Comparison between the position of the shock wave for the non flux-conservative and the flux-conservative version evolution for $J = 1001$.

Regarding the height, we can see that the height of the shock wave decreases faster in the case of the non-flux conservative version. Moreover, increasing the sampling of the space components allows for a slower decrease of the maximum value of the function. Also, the separation between the non flux-conservative and the flux-conservative version is delayed to higher times.

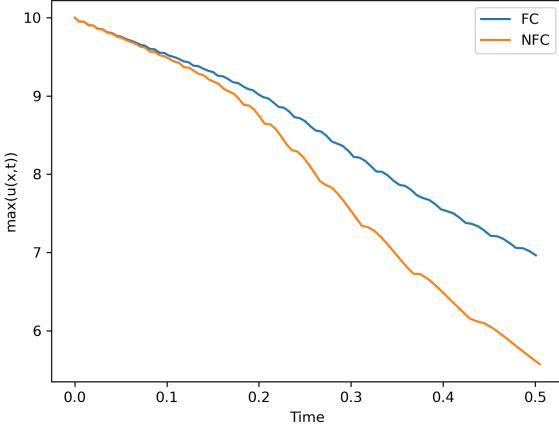


Figure 41: Comparison between the height of the shock wave for the non flux-conservative and the flux-conservative version evolution for $J = 101$.

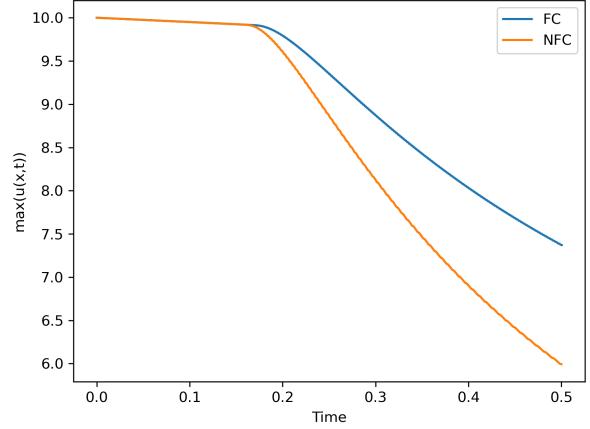


Figure 42: Comparison between the height of the shock wave for the non flux-conservative and the flux-conservative version evolution for $J = 1001$.

Therefore, using a non flux-conservative or a flux-conservative version deeply affects the results of the numerical code. In fact, these different behaviours are predicted by the Hou-Le Flock Theorem, for which non flux-conservative numerical methods do not converge to the correct solution if a shock wave is present. Instead, the convergence of the flux-conservative numerical method is guaranteed by the Lax-Wendroff Theorem, for which we have that a function $u(x, t)$ is a solution of the conservation law if the consistent and flux-conservative method converges to a function $u(x, t)$ for $\Delta x \rightarrow 0$.