MAP551 - PC 3

Paul Calot

October 17, 2020

0.1 Introduction

0.2 Numerical integration in the framework of application of Peano's theorem.

0.2.1 Study of the ODE

0.2.1.1

The function f has the following properties :

- It is continuous on $\mathbb{R}^+ \times \mathbb{R}$;
- It has a derivative on $\mathbb{R}^+ \times \mathbb{R} \setminus \{(0,0)\}$. Indeed $u \to \sqrt{|u|}$ is not differentiable at $u = 0$.

From 1), we deduce that Peano's theorem applies and thus there is a solution to (1). From 2), we can say that Cauchy-Lipschitz theorem does not apply.

0.2.1.2

We have first:

$$f(\Delta t, 0) = 4\Delta t cos(\frac{\pi log(\Delta t)}{log 2}) = 2^{-i+2}(-1)^i \tag{1}$$

We also have :

$$\forall |y| > t^2, f(t,y) = 4 sign(y)\sqrt{|y|} \tag{2}$$

0.2.1.3

Let us suppose that $u(0) \neq 0$.
First case : $u(0) > 0$.
In this case, $f(t, u(0)) = 4\sqrt{u(0)} > 0$ because $t^2 \leq u(0)$.
Consequently, we have :

$$\forall t \in]0, u(\tilde{t})[, f(t,u) > 0 \tag{3}$$

With $\tilde{t}$, the time such that $\tilde{t}^2 = u$.

The function u is consequently increasing on this interval and remains thus stictly positive. We can write for $t \neq \tilde{t}$:

$$\begin{aligned} d_t u = 4\sqrt{u} \Leftrightarrow \frac{dt_u}{\sqrt{u}} = 4 \\ \Leftrightarrow \sqrt{u} = 2t + \sqrt{u(0)} \end{aligned}$$

Then :

$$\begin{aligned} \tilde{t}^2 = u(\tilde{t}) \Leftrightarrow \tilde{t}^2 = (2\tilde{t} + \sqrt{u(0)})^2 \\ \Leftrightarrow -\tilde{t} = \sqrt{u(0)} \end{aligned}$$

since everything is positive.

This condition is never met. Consequently, we proved that for an initial condition $u(0) > 0$, $f(\mathbb{R}^+ \times \mathbb{R}^{\times +}) \subset \mathbb{R}^+ \times \mathbb{R}^{\times +}$. Consequently, the Cauchy-Lipchitz theorem applies and there is a unique solution to Cauchy's problem on $\mathbb{R}^{\times +}$.

The same applies for $u(0) < 0$: there is a unique solution to Cauchy's problem on $\mathbb{R}^{\times -}$.

Two examples of such solutions are given in figure 1.

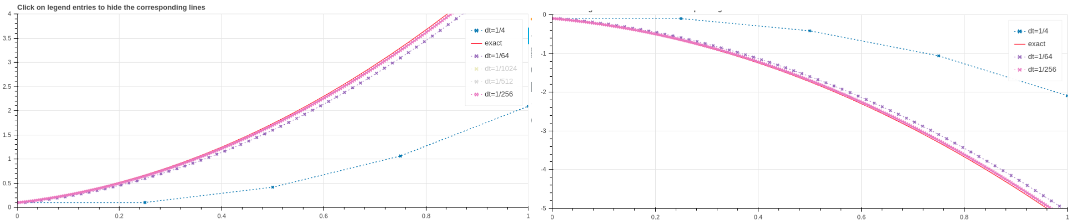

Figure 1: Analytic solutions (and some numerical ones) for $u_0 = 0.1$ (right) and $u_0 = -0.1$ (left).

0.2.2 Numerical integration

0.2.2.1

Recall for forward Euler that $y_{n+1} = y_n + \Delta t f(t, y_n)$ which yields with $\Delta t = 2^i$:

$$y_{n+1} = y_n + 2^i t f(n2^i, y_n)$$

We then have :

- Initial condition : $y_0 = y(0) = 0$
- First step : $y_1 = 0 + 2^i t f(0, 0) = 0$
- Second step : $y_2 = y_1 + \Delta t f(\Delta t, 0) = (-1)^i 4^{-i+1}$ (according to 0.2.1.2)
- Third step : $y_3 = y_2 + \Delta t f(2\Delta t, y_2)$ which yields : $y_3 = 3 \times (-1)^i 4^{-i+1}$.
- ...

By induction, we can prove that $\forall n \in \mathbb{N}, y_n = \alpha_n (-1)^i 4^{-i+1}$ with $\alpha_{n+1} = \alpha_n + 2\sqrt{\alpha_n}$. It's relatively straight forward under the hypothesis that $\forall n \in \mathbb{N}, n > 1, \alpha_n \geq \frac{n^2}{4}$ (which allows to remove the second term in the right hand side of the definition of $f, t > 0$).

Let's now make sure that : $\forall n \in \mathbb{N}, n > 1, \alpha_n \geq \frac{n^2}{4}$.

Once again by induction :

1. For $n = 2$, $\alpha_2 = 1 \geq \frac{1}{2}$

2. Let us suppose that for $k \in \mathbb{N}, k > 1$, $\alpha_k \geq \frac{k^2}{4}$ (inductive hypothesis). Then :

$$\begin{aligned}
\alpha_{k+1} &= \alpha_k + 2\sqrt{\alpha_k} \quad \text{(by recurrence relation)} \\
&\geq \frac{k^2}{4} + 2\sqrt{\frac{k^2}{4}} \quad \text{(by inductive hypothesis)} \\
&= \frac{k^2}{4} + k \\
&= \frac{k^2 + 4k}{4} \\
&\geq \frac{k^2 + 2k + 1}{4} \quad \text{(k>1)} \\
&= \frac{(k+1)^2}{4}
\end{aligned}$$

which proves the case for $k + 1$.

The associated numerical solutions are given in figure 2.

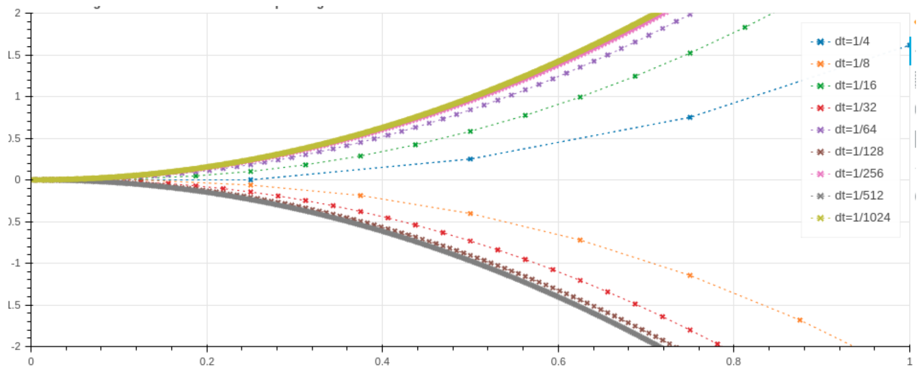

Figure 2: Numerical solutions for various i parities.

0.3 Conservative System and Euler Integration methods

0.3.1

We use system (6) :

$$\begin{aligned}
\frac{\partial E}{\partial t} &= (y_3\frac{\partial y_3}{\partial t} + y_4\frac{\partial y_4}{\partial t}) + (\frac{1-\mu}{{r_1}^2}\frac{\partial r_1}{\partial t} + \frac{\mu}{{r_2}^2}\frac{\partial r_2}{\partial t}) - (y_1\frac{\partial y_1}{\partial t} + y_2\frac{\partial y_2}{\partial t}) \\
&= y_3\Big(y_1 + 2y_4 - \frac{(1-\mu)(y_1+\mu)}{{r_1}^3} - \mu\frac{y_1 - 1 + \mu}{{r_2}^3}\Big) + y_4\Big(y_2 - 2y_3 - \frac{(1-\mu)y_2}{{r_1}^3} - \mu\frac{y_2}{{r_2}^3}\Big) \\
&+ \frac{1-\mu}{{r_1}^2}\frac{y_3(\mu+y_1) + y_2y_4}{\sqrt{(y_1+\mu)^2 + {y_2}^2}} + \frac{\mu}{{r_2}^2}\frac{y_3(\mu - 1 + y_1) + y_2y_4}{\sqrt{(y_1 - 1 + \mu)^2 + {y_2}^2}} - y_1y_3 - y_2y_4 \\
&= -\frac{y_3}{r_3}(1-\mu)(y_1+\mu) - \frac{y_3}{{r_2}^3}\mu(y_1 - 1 + \mu) - \frac{y_4y_2}{{r_1}^3}(1-\mu) \\
&- \mu\frac{y_2y_4}{{r_2}^3} + \frac{(1-\mu)}{{r_1}^3}(y_3(y_1+\mu) + y_2y_4) + \frac{\mu}{{r_2}^3}(y_3(y_1 - 1 + \mu) + y_2y_4) \\
&= 0
\end{aligned}$$

This proves that $E(y)$ is an invariant of (6).

In addition : $E(y) = \frac{1}{2}({y_3}^2 + {y_4}^2) - (\frac{1-\mu}{r_1} + \frac{\mu}{r_2}) - \frac{1}{2}({y_1}^2 + {y_2}^2) = T + V + C$ with :

1. T : kinetic energy ;
2. V : potential energy ;
3. C : This one is harder to interpret. It diminishes as ${y_1}^2 + {y_2}^2 = |Y|^2$, where Y is the position in the complex plane of the satellite, increases. It's not a kinetic term (no speed is involved). So the further away from the center of mass of the system Earth + Moon the satellite is, the less energy it has. Consequently, it's a coupling term which acts proportionnaly to the squared distance between the satellite and the origin.

The figure 3 gives the RK45 Scipy solution to this reduced three body problem.

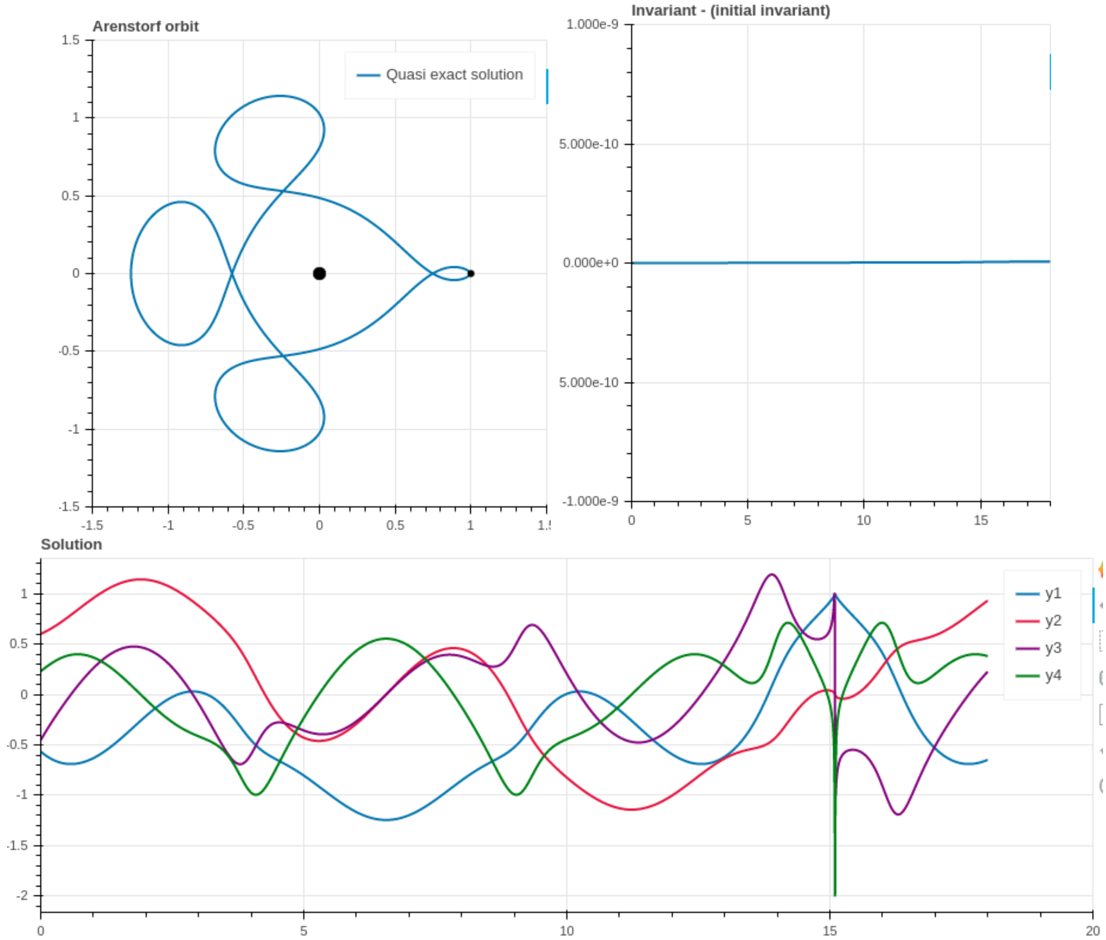

Figure 3: Quasi-exact solution to the reduced three body problem using Scipy solver and for one period only.

The evolution of y_i, $i \in \{1, 2, 3, 4\}$, shows that the solution is periodic of period roughly equals to 18 secs. The invariant seems almost conserved (a slight increase is visible at the end and this will be analysed in the following questions) despite choosen scheme which is not a symplectic one. Note that fixing $E(y)$ as an invariant is not enough to completely get the solution (two more equations would be required). The evolution of y_i, $i \in \{1, 2, 3, 4\}$, also shows a stiffness around 15 secs.

0.3.2

Using forward euler scheme : The numerical solutions for $n_t = 1800, 180000$ are given figure 4.

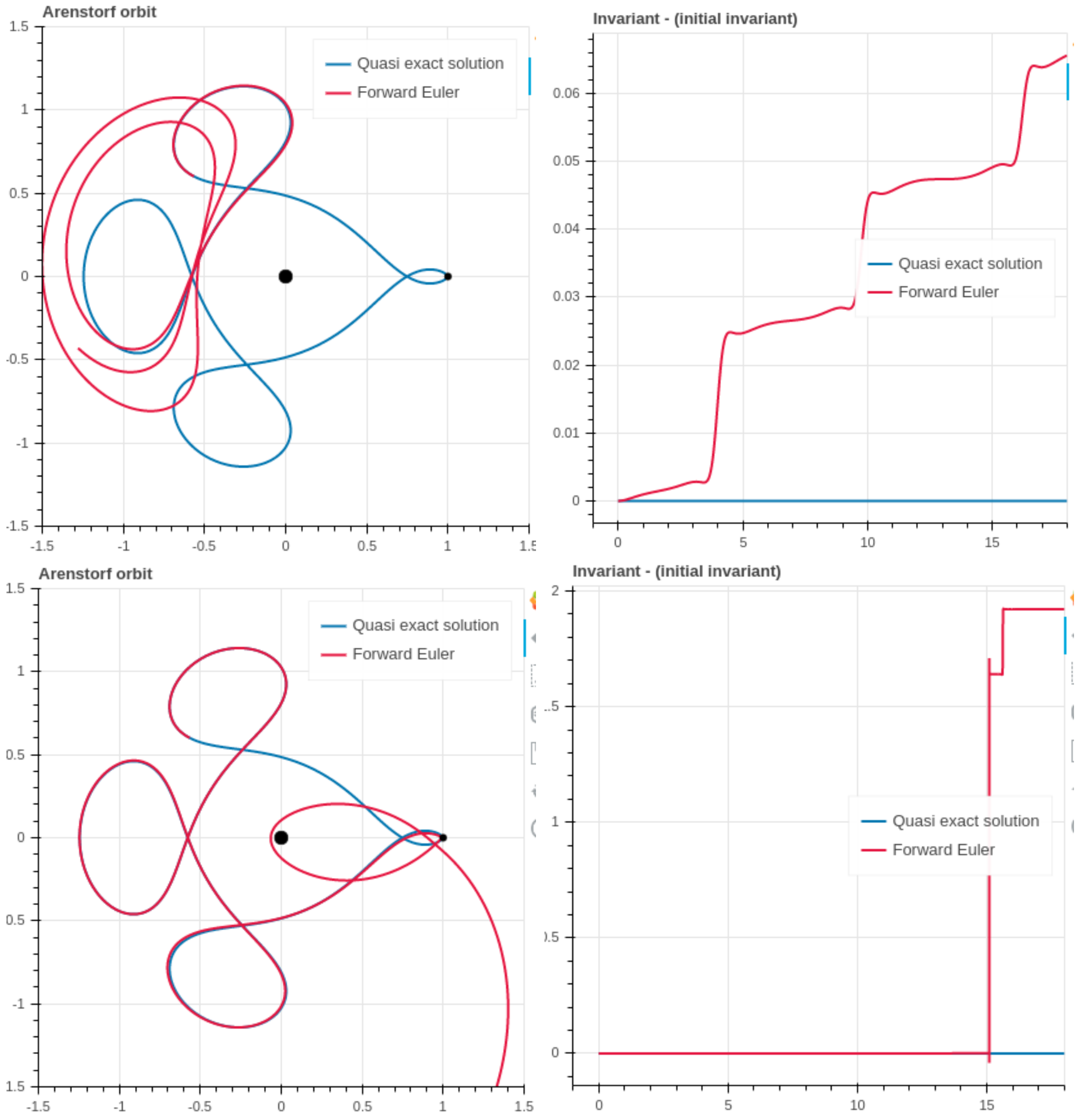

Figure 4: Solution to the reduced three body problem using forward euler scheme. Top is for $n_t = 1800$ points, bottom is for $n_t = 180000$.

1. For $n_t = 180000$: the scheme diverges after roughly 15 secs whereas it was able to nicely stick to the quasi exact solution up to this moment. This is due to the high stiffness identified in the previous question. Indeed, forward euler scheme is associated with a high sensibility to stiff systems which by definition admit at least one very low eigenvalue : $\lambda < 0, |\lambda| >> 1$. The lack of discretization steps around that 15 sec makes the scheme unstable. By increasing n_t, it is possible to get better results. For such a n_t value, there is no periodic phenomenon.

2. $n_t = 1800$: this time, the discretization step is not sufficient to allow the stability of the algorithm early-on.

Remarks :

1. Note that the invariant is increasing in the explicit euler scheme ;
2. Increasing the number of steps for the given period of plotting can improve the stability, however computing the numerical solution becomes higly time consuming for still poorly results.
3. Also, recall that the method converges but is of order one and consequently increasing 10 times the number of discretization steps will only provide a 10 times increase accuracy. Consequently, even if the the method does not diverge because it is in its stability domain, it still will after some periods.

Using backward euler sheme : The numerical solutions for $n_t = 1800, 180000$ are given figure 5.

The same behavior than with forward euler is displayed here.

Several remarks though :

1. This time, the invariant decreases instead of increasing.
2. Backward euler scheme, as all implicit method, handles better the stiffness of a system. The error on the invariant for $n_t = 180000$ is 200 less important than for Euler implicit scheme. Consequently, we have better results that previously for the same discretization parameters.
3. Even if the method is better, it takes much more time to compute a solution as inverting the system is required at each time step to compute the next point.

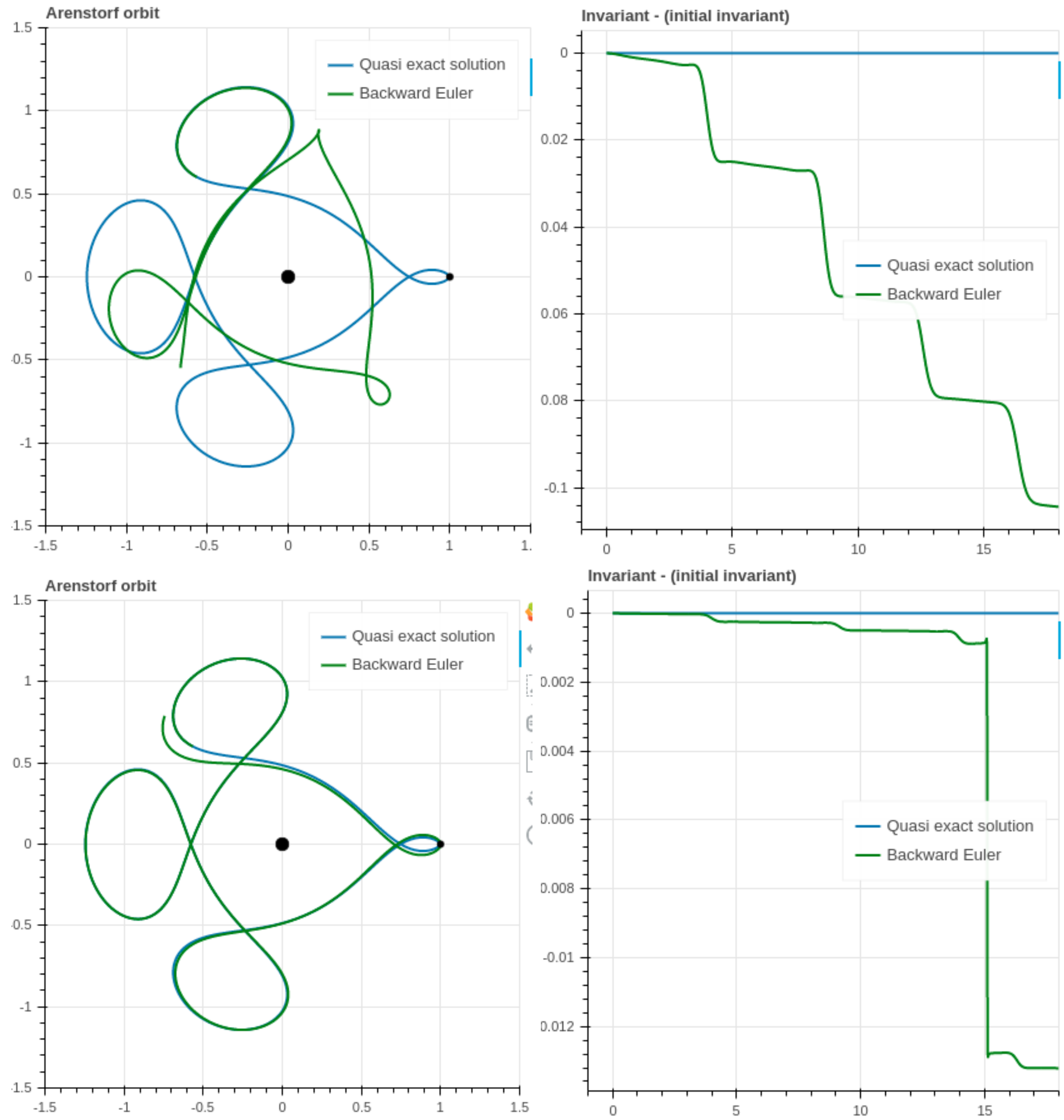

Figure 5: Solution to the reduced three body problem using backward euler scheme. Top is for $n_t = 1800$ points, bottom is for $n_t = 180000$.

Conclusion : Euler explicit scheme does not do well with this system which is stiff. Euler implicit is a little bit better but still very far from the quasi-exact solution.

0.3.3

The figure 6 shows that RK45 solver yields poor results after only 4 periods. Indeed, the accuracy on the invariant achieved by this scheme can not preserve the periodicity of the solution for more than 3 periods. On the figure, we see that a $\Delta E = E_{t^n} - E_0 \approx 1.5e - 11$ is enough to yield the loss of periodicity between the third period and the fourth one. The accuracy needed on that invariant is thus approximately $1.5e-11$ for as many periods as we want to plot. Since the accuracy loss on the invariant seems roughly the same per period, an accuracy of $\frac{1.5e-11}{N}$ per period is required to plot N periods.

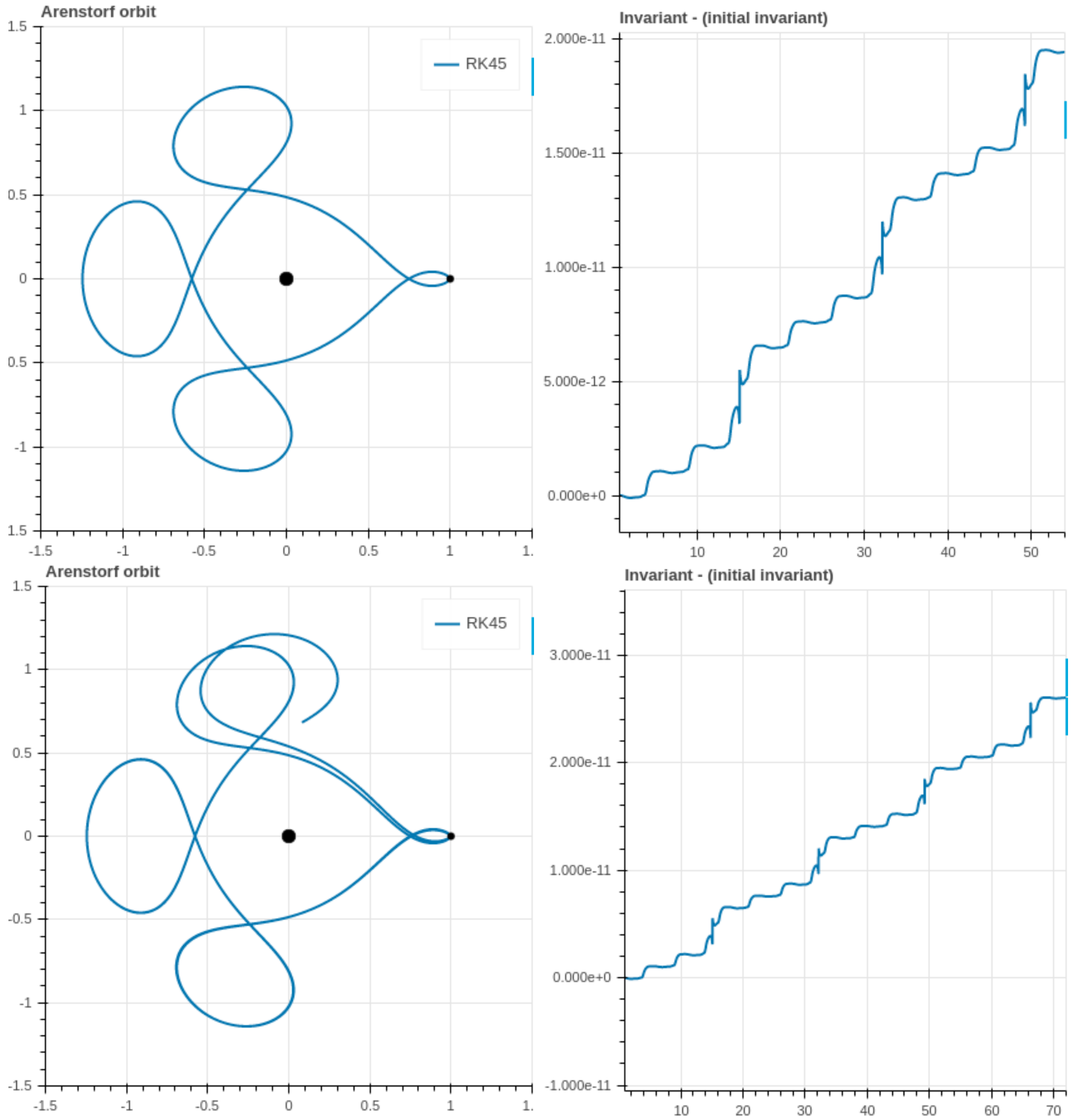

Figure 6: Quasi exact solution using RK45 solver for 3 periods (top pictures) and 4 periods (bottom pictures).

0.4 Stability, order and accuracy for non-stiff and stiff equations

0.4.1 Stiffness ?

0.4.1.1

The given Cauchy problem verifies the hypotheses of Cauchy-Lipschitz theorem ($t, u \to k(cos(t) - u(t))$ is of class C^1 on $\mathbb{R}^+ \times \mathbb{R}$ and for all $k > 1$). Consequently, there exists an unique solution and we simply have to verify that this is the one given.

Note that u is indeed of class C^1. We have :

1. $d_t u(t) = \frac{k}{k^2+1}(-ksin(t) + cos(t)) - kc_0 e^{-kt}$
2. $k(cos(t) - u(t)) = \left[\frac{k}{k^2+1}(cos(t) - ksin(t))\right] - kc_0 e^{-kt}$

They are equals quantities. Consequently, we found the solution of Cauchy's problem.

0.4.1.2 $k >> 1$

In this case, we have roughly : $u(t) \approx cos(t) + c_0 e^{-kt}$ with $c_0 \approx (u_0 - 1)e^{kt_0}$. That yields to the identification of two regions :

- $\forall kt << 1, u(t) \approx c_0 e^{-kt} + cos(t)$ which decreases very abruptly with t increasing, hence the stiffness.
- $\forall kt >> 1, u(t) \approx cos(t)$.

These two regions are plotted on figure 7.

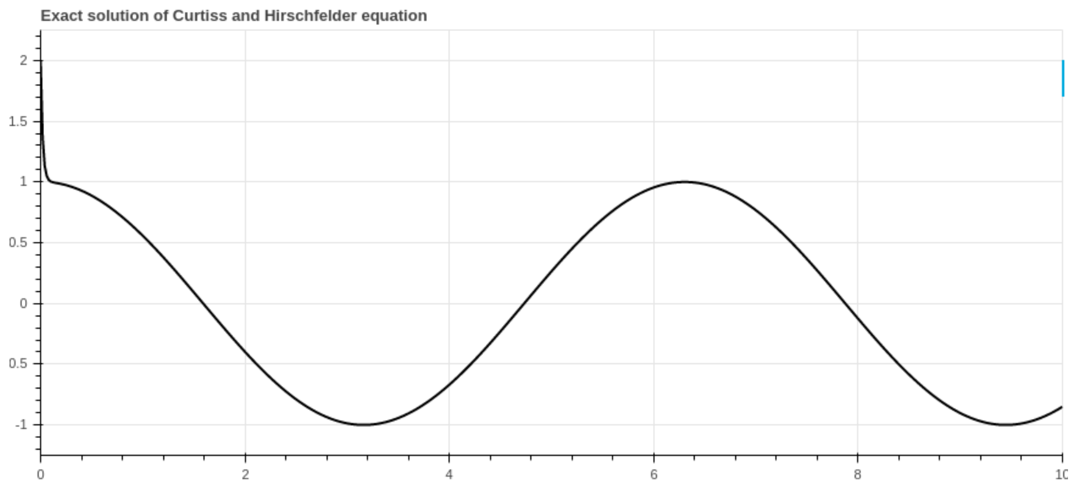

Figure 7: Plotting of the two visible regions. First one is for $t < 0.1$ roughly, and second therafter.

0.4.1.3 $u_0 = \frac{k^2}{k^2+1}$

This yields to $c_0 \approx 0$ which yields to only one region in this case : $u(t) \approx cos(t)$ (cf. figure 8), hence the lack of stiffness, even for large k (the solution depending only slightly of k).
The stiffness can then come from various origins :

1. Initial condition ;
2. Intrinsically from the system.

In our case, the system seems intrinsically stiff but the specific initial condition $u_0 = \frac{k^2}{k^2+1}$ prevents stiffness from appearing.

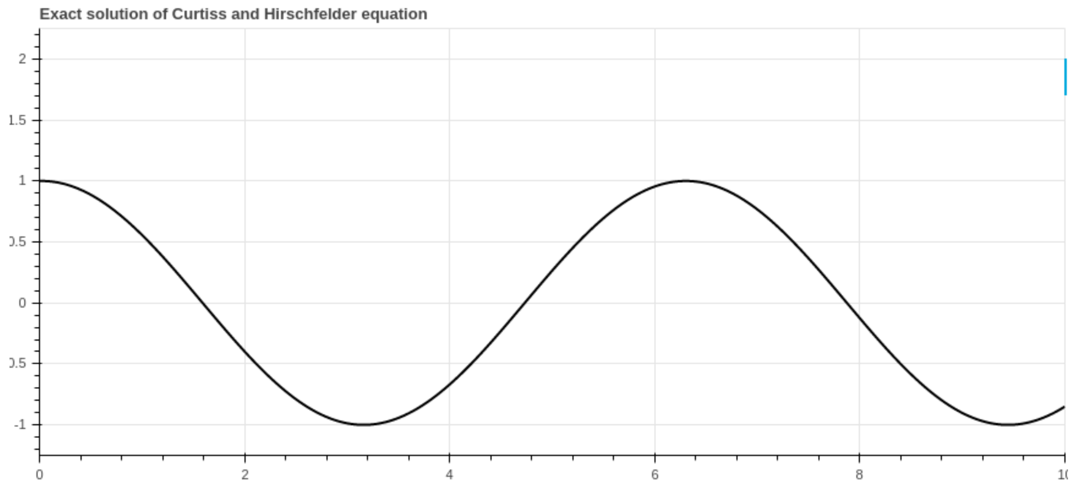

Figure 8: Plotting the numerical solution for $u_0 = \frac{k^2}{k^2+1}$

0.4.2 Explicit Euler

0.4.2.1

We take $k = 50$ and $u_0 = 2$.

We have the following regimes, with $T = 2$ sec (cf. 9) :

- $n_t < 50$: divergent oscillations ;
- $n_t \approx 51$: limits of convergence (oscillations still occur) ;
- $n_t > 52$: convergence without (much) oscillations. The more n_t increases, the less oscillations.

Keep in mind that the values given are only indicatives. It's roughly around those values that changes occur.

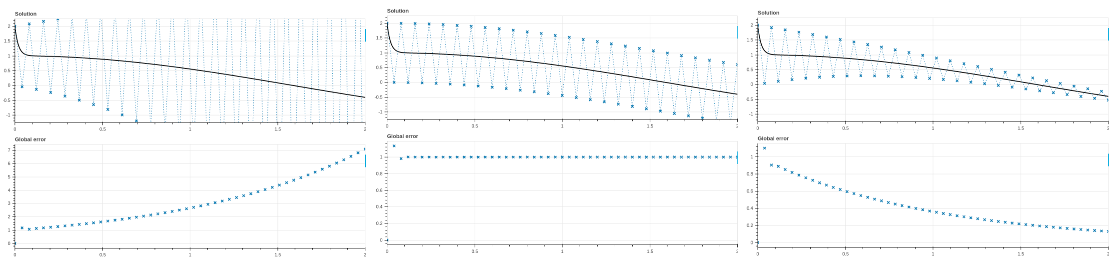

Figure 9: Plotting the three identified regimes for forward euler scheme, $n_t = 50, 51, 52$ (left to right),$k = 50$, $u_0 = 2$ and $T = 2$.

0.4.2.2

For a simpler system defined by

$$\begin{cases} Y' = -kY \\ Y_0 > 0 \end{cases} \tag{4}$$

the forward euler relation yields : $y_{n+1} = (1 - k\Delta t)y_n = z(\Delta t)y_n$.

By studying z, we find that the stability of the scheme requires : $|z(\Delta t)| < 1 \Leftrightarrow dt < \frac{2}{k} \Leftrightarrow n_t > \frac{kT}{2}$.

In our case, we have $T = 2$ sec, $k = 50$, which yields $n_t = 50$. So the scheme is A-stable for $n_t > 50$.

0.4.2.3

From the lessons, we know that the Explicit Euler scheme is consistent of order one. It consequently matches the expected order (cf. 10).

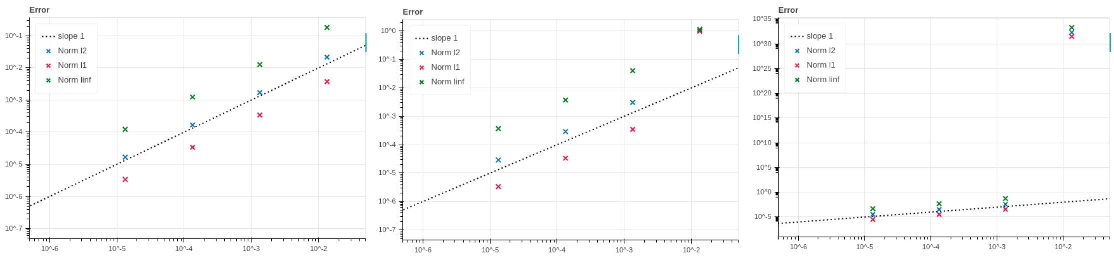

Figure 10: Log-log Plot of the error $\epsilon(\Delta t)$ for $k = 50, 150, 200$, $u_0 = 2$ and $T = 2$.

For the higher stiffness, a "bigger" error that is not consistent with the order one appears for $\Delta t \approx 10e-2$. This is because the scheme is not stable in this case (cf. previous questions). Also, the constant error increases with the stiffness of the system. This means that the initial error keeps getting bigger which was to be expected given the explicit nature of the scheme.

0.4.2.4

The figure 11 clearly shows the initial error increasing as k increases until it does not converge anymore for $k = 200$. As for the order diagram, for the higher stiffness, a "bigger" error that is not consistent with the order one appears for $\Delta t \approx 10e-2$. This is because the scheme is not stable in this case (cf. previous questions). When the method is A-stable however (which means when n_t is high enough), there seems to be a slightly bigger error but nothing meaningful. The notion of order is not absolute and should be considered with respect to the proven stability of the scheme.

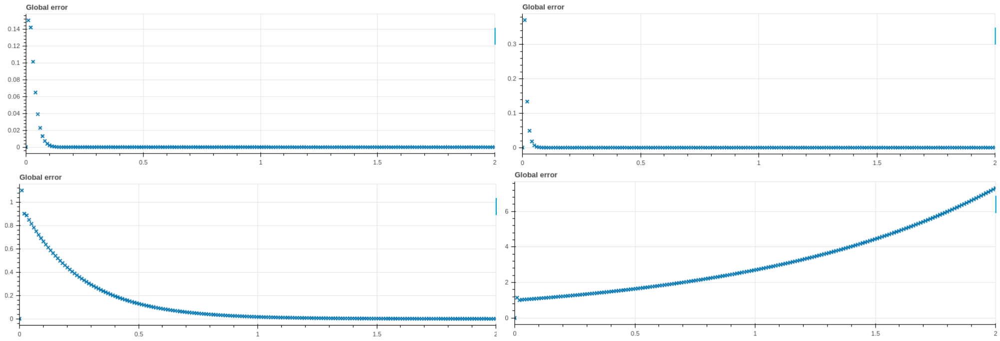

Figure 11: Plot of the global error for various stiffness : $k = 60, 100, 195, 200$ (from left to right, top to bottom), $n_t = 200$, $u_0 = 2$.

In figure 12, we can see that doubling the stiffness requires roughly doubling the discretization accuracy to keep a low error. In 12, we doubled both stiffness and discretization points. This is consistant with question 0.4.2.2.

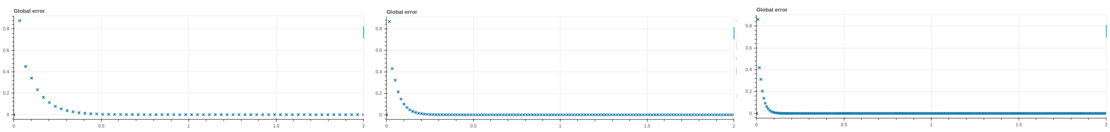

Figure 12: Plot of the global error for (stiffness, number discretization points) couples : $(50, 60)$, $(100, 120)$ and $(200, 240)$ (from left to right), $u_0 = 2$.

0.4.2.5 $u_0 = \frac{k^2}{k^2+1}$

In this case, it is harder to really observe the same three regimes as in the previous case ($u_0 = 2$) because the initial error, previously caused by the stiffness which is non-existant here, is much lower. However we roughly have the same behavior as we can see in figure 13.

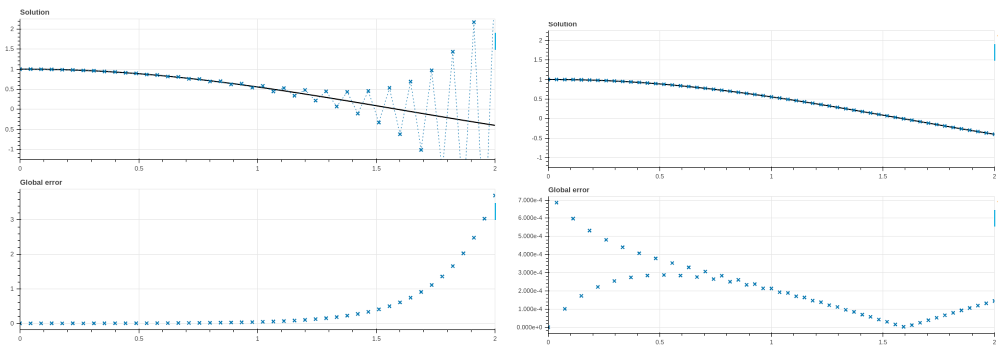

Figure 13: Plotting the identified regimes for forward euler scheme, $n_t = 51, 55$ (left to right),$k = 50$, $u_0 = \frac{k^2}{k^2+1}$ and $T = 2$. The system being not stiff for this initial condition, there is less abrupt changes and the scheme is able to follow the solution better. Knowing what happens between $n_t = 51$ and $n_t = 55$ is harder than previously as regime (divergent, convergent, oscillary) requires long-time view of the numerical integrations which is complicated to have here.

As for the log-log error, we now have (figure 14):

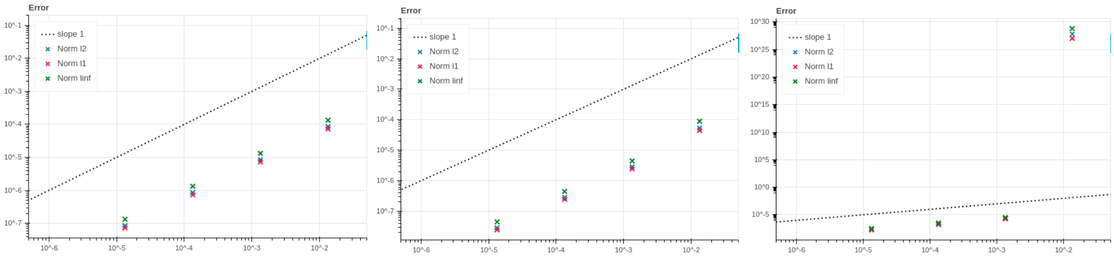

Figure 14: Log-log Plot of the error $\epsilon(\Delta t)$ for $k = 50, 150, 200$, $u_0 = 2$ and $T = 2$.

This is basically the same thing as before for the order and general evolution. We still have the explosion of the error for $k = 200$ and low n_t. However, one can see that the error is always lower (by at least a factor ten). This is due to the initial error that is much lower (almost non-existent) contrary to the previous case, as the stiffness of the system does not appear for this specific initial condition.

As for the global error, we now have (figure 15):

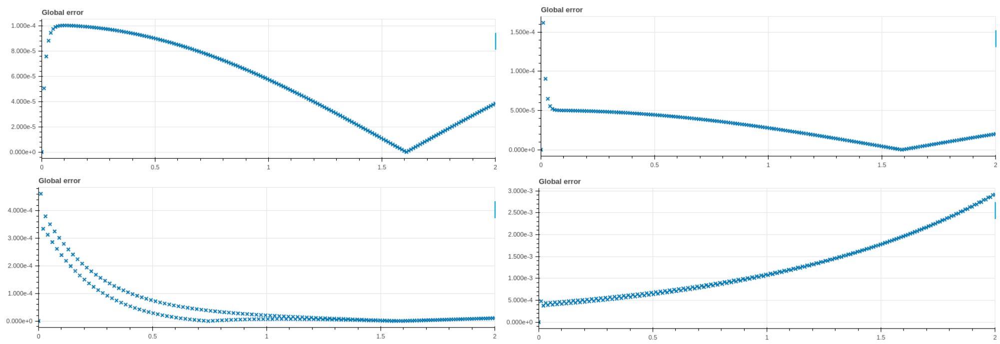

Figure 15: Plot of the global error for various stiffness : $k = 50, 100, 195, 200$ (from left to right, top to bottom), $n_t = 200$, $u_0 = \frac{k^2}{k^2+1}$.

This time, it is less obvious that the stiffness increases the error even if we still see it increasing slightly until explosion when it reaches the unstable domain.

0.4.2.6

Forward euler scheme does not allow for a great stability especially in the presence of stiffness, however it is relatively quick to compute. Also, its A-stability domain restricts the possiblity for (k, n_t) values.

0.4.3 Implicit Euler

0.4.3.1

Figure 16 shows that stiffness influences the numerical solution obtained with backward euler scheme much less than for forward euler scheme. Indeed, the error barely increases.

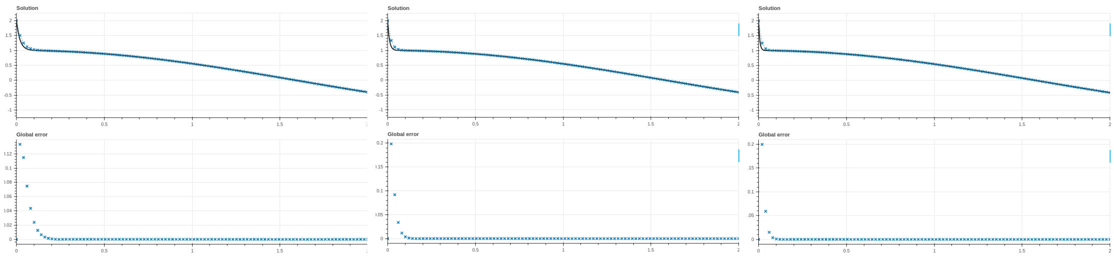

Figure 16: Stiffness influence for backward euler scheme with $n_t = 100$, $u_0 = 2$ and $k = 50, 100, 150$ (left to right).

Link with stability of the scheme :

For a simpler system defined by

$$\begin{cases} Y' = -kY \\ Y_0 > 0 \end{cases} \tag{5}$$

the backward euler relation yields : $y_{n+1} = y_n - k\Delta t y_{n+1} \Rightarrow y_{n+1} = z(\Delta t) y_n$ where $z(\Delta t) = \frac{1}{1+k\Delta t}$

The stability of the scheme requires that $|z(\Delta t)| < 1$ which is always true. So the scheme is A-stable. This is why we observe no unstability when computing the solutin for higher stiffness.

0.4.3.2

In the figure 17, we see that doubling n_t when doubling k allows to keep the same error. Note however that in the previous question we had seen very little changes too while keeping the exact same n_t and for various stiffness values.

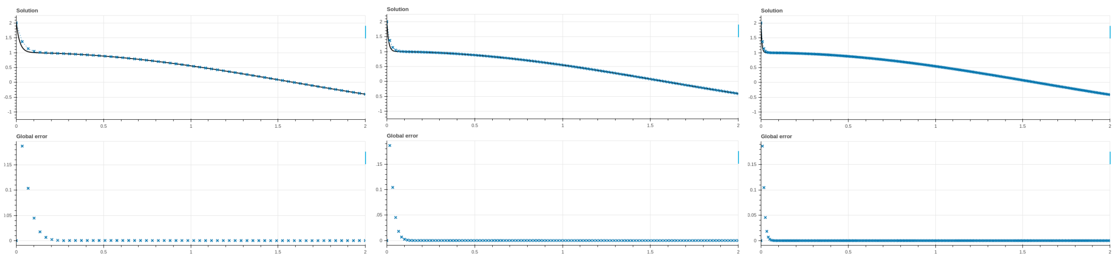

Figure 17: Numerical solutions for three (stiffness, discretization step) couple, from left to right : $(50, 60)$, $(100, 120)$, $(200, 60)$.

Since we chose the same couple (k, n_t) for the forward (cf. figure **??**) and backward euler schemes, we can compare them : the forward method seems roughly four times better for the initial error (≈ 0.9 *vs* ≈ 0.2). Consequently, we would need more discretization steps to roughly have the same accuracy. Figure 19 shows it even better when compared to figure 10.

Stability and accuracy are two different concepts. It's true however that the unstable scheme has no chance to be accurate (forward euler scheme with stiffness of 200 is an example). An extreme case of stable scheme with no accuracy would be to take this backward euler scheme with maximum stiffness and very few discretization points (18).

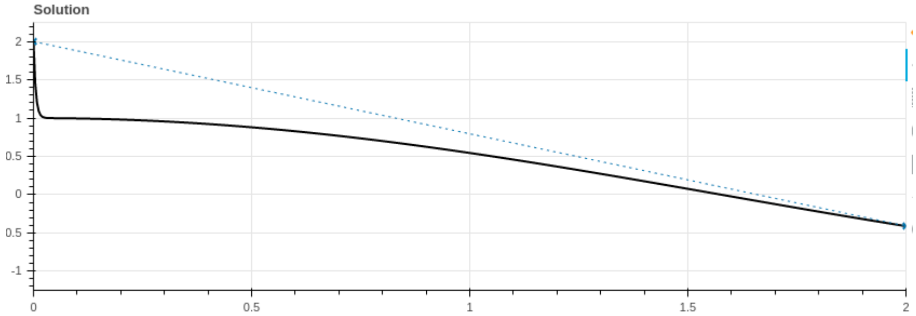

Figure 18: Extreme case : $k = 200$, $n_t = 2$. We can hardly say that it fits the analytical solution. That being said, one is expected to choose wisely the discretization parameters used for the numerical integration.

Conclusion : For a stiff system, explicit schemes are not the best, especially to achieve good accuracy. However, inplicit methods requires inverting the recurrence relation at each step which can be a very (very) long process. So unless it is necessary to have a very accurate result, explicit methods can do the job.

0.4.3.3

Figure 19 shows the exact same behavior of the logarithmic error depending on the time step. This is consistent with the A-stability of the backward scheme. This is also consistent with theory that indicates that the forward euler scheme is of order 1.

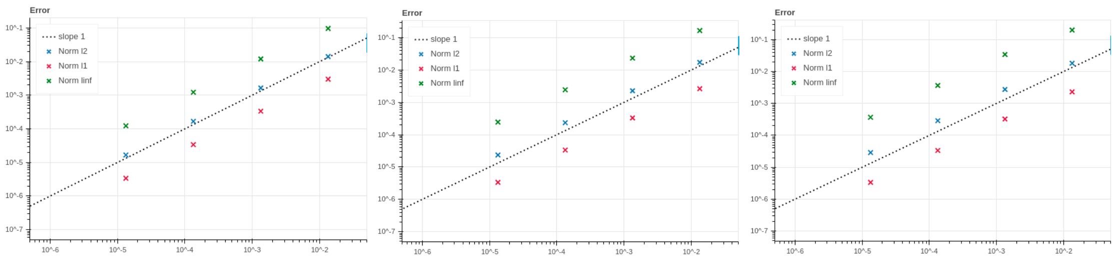

Figure 19: Log-log plot different stiffness value : 50, 100, 150.

The constant error does not seem to be modified a lot in the implicit case whereas it was visible in the explicit case (figure 10). Indeed, the constant error kept getting bigger with the stiffness in the previous case.

0.4.3.4 $u_0 = \frac{k^2}{k^2+1}$

Figure 20 shows the obtained numerical solutions for three different couples. Figure 21 shows the global error plot for various value of stiffness.

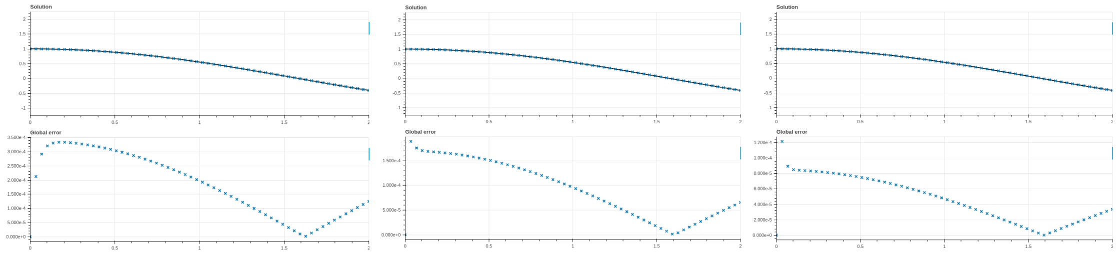

Figure 20: Numerical solutions for three (stiffness, discretization step) couple, from left to right : $(50, 60)$, $(100, 120)$, $(200, 240)$ for $u_0 = \frac{k^2}{k^2+1}$.

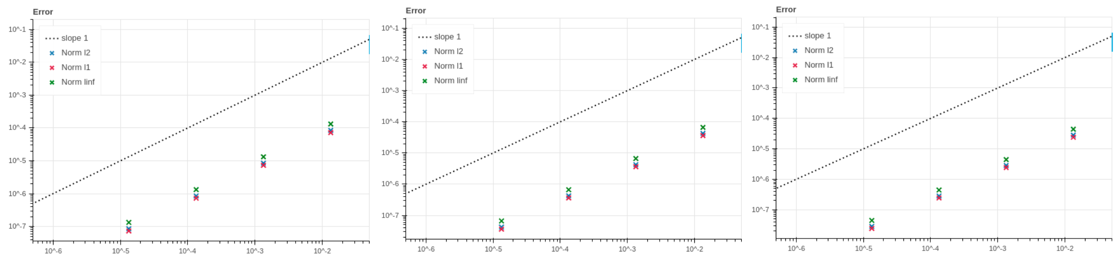

Figure 21: Log-log plot different stiffness value : 50, 100, 150.

This time we combined what was said for the explicit method, case $u_0 = \frac{k^2}{k^2+1}$, with the implicit method. As for the previous explicit case, the error is lower than for the case $u_0 = 2$. No obvious dependence is visible upon the initial error. In this case, it even makes more sense since we annihilated the stiffness of the system. Nothing much to add that has not already been said.

0.4.3.5

The implicit scheme allows for a very stable numerical solution even with high stiffness in the system. However, it can be time consuming as it's necessary to invert the recurrence relation at each time step. Consequently, this scheme should be avoided when an explicit (and quicker) scheme can be used and that the solution need not be super accurate. The explicit method can be as accurate as the implicit one given enough discretization points and still perform faster.