MSAS – Assignment #1: Simulation

Pietro Bosco, 218626

1 Implicit equations

Exercise 1

Let $\mathbf{f}$ be a two-dimensional vector-valued function $\mathbf{f}(\mathbf{x}) = (x_2^2 - x_1 - 2,\ -x_1^2 + x_2 + 10)^\top$, where $\mathbf{x} = (x_1, x_2)^\top$. Find the zero(s) of $\mathbf{f}$ by using Newton's method with $\partial\mathbf{f}/\partial\mathbf{x}$ 1) computed analytically, and 2) estimated through finite differences. Which version is more accurate?

(3 points)

Firstly the two components of the function, $f(1) = x_2^2 - x_1 - 2$ and $f(2) = -x_1^2 + x_2 + 10$, are plotted in the 2-D plane in order to get a better visualization of the problem and spot the zeros of $\mathbf{f}$:

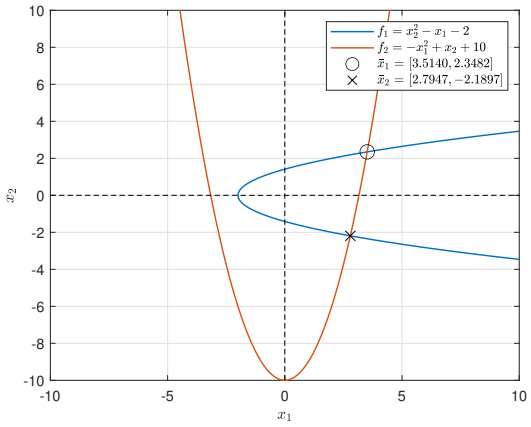

Figure 1: Zeros of $\mathbf{f}$

The values of zeros are computed through the implementation of two Newton's methods, one computed analytically, and the other through finite differences. With the support of the plot it is possible to compute the zeros with the two methods starting from an educated guess.
The implementation of the analytical Newton's method, the Jacobian matrix of $\mathbf{f}$ is calculated and then fed into the function:

$$J = \begin{bmatrix} -1 & 2x_2 \\ -2x_1 & 1 \end{bmatrix} \tag{1}$$

For the implementation of the second method, the Jacobian matrix is estimated in the function by means of the forward differences model with a perturbation $\epsilon = 0.01$:

$$f'(x) = \frac{f(x+\epsilon) - f(x)}{\epsilon} \tag{2}$$

Finally, the zeros are computed and the two Newton's methods compared. Tab. 1 shows that the two methods bring to the same result. However, as displayed in Tab. 2, they do not share the same level of accuracy, which can be computed by evaluating the norm of the function in the zero. It is clear by looking at the table that the analytical Netwton's method is the most accurate.

Method	First guess	Second guess	First result $\bar{x}_1$	Second result $\bar{x}_2$
Analytical	[2;1]	[2;-1]	[3.5140; 2.3482]	[2.7947;-2.1897]
Finite differences	[2;1]	[2;-1]	[3.5140; 2.3482]	[2.7947;-2.1897]

Table 1: Results

Method	First result $\bar{x}_1$	Second result $\bar{x}_2$
Analytical	2.1648e-09	8.2996e-07
Finite differences	2.0440e-13	2.5344e-08

Table 2: Accuracy of the methods

2 Numerical solution of ODE

Exercise 2

The Initial Value Problem $\dot{x} = x - 2t^2 + 2$, $x(0) = 1$, has analytic solution $x(t) = 2t^2 + 4t - e^t + 2$. 1) Implement a general-purpose, fixed-step Heun's method (RK2); 2) Solve the IVP in $t \in [0, 2]$ for $h_1 = 0.5$, $h_2 = 0.2$, $h_3 = 0.05$, $h_4 = 0.01$ and compare the numerical vs the analytical solution; 3) Repeat points 1)–2) with RK4; 4) Trade off between CPU time & integration error. (4 points)

A function that exploit a second order Runge-Kutta method, also known as Heun method, is implemented; the solution of the IVP obtained with such function is then compared to the evolution of the solution obtained through an analytical method for four different values of the time step's size h. The results are displayed in Fig.2, as well as the integration error in Fig.3:

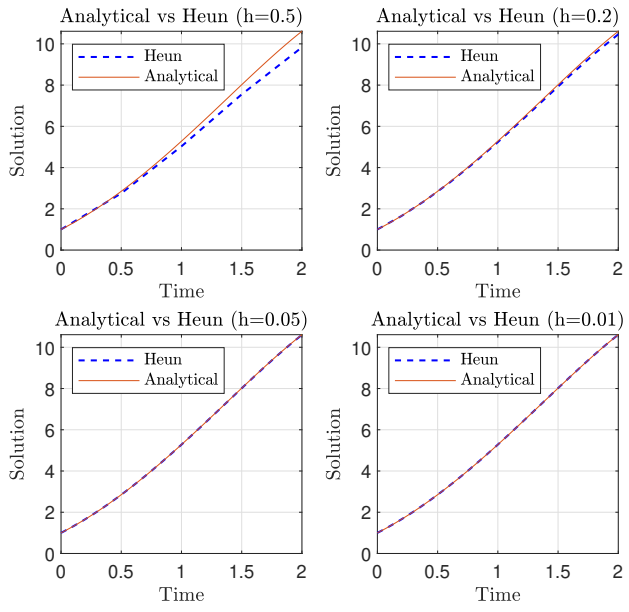

Figure 2: Heun method vs analytical comparison

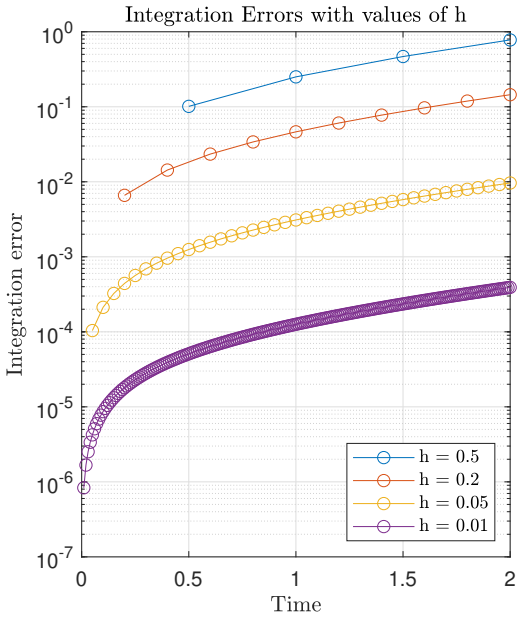

Figure 3: Integration errors

As the step size h decreases, the method becomes more accurate. However, in all the four cases it is clear that the integration error rises as the method goes on in time. Successively, the same process is carried out using a fourth order Runge-Kutta method instead; the IVP is solved once again both numerically and analitically, with the same step sizes as before. The results and the integration error are shown in Fig.4 and Fig.5 respectively:

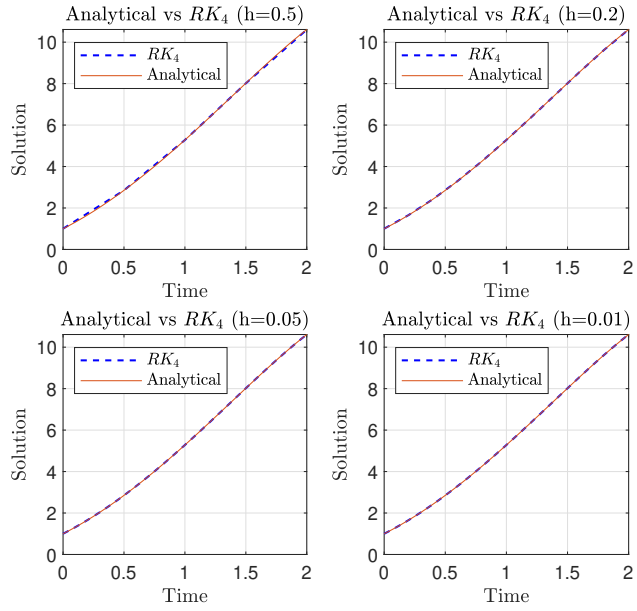

Figure 4: $RK4$ method vs analytical comparison

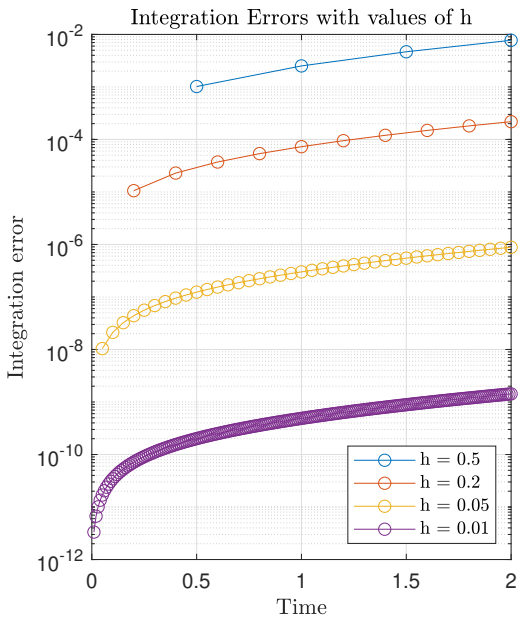

Figure 5: Integration errors

The integration method of fourth order is generally more accurate than the second order's one, even if the error still grows as the integration keeps going in time as before.
Lastly, the computational time taken by the numerical integration methods is computed. In order to get reliable results, a *for* cycle is implemented such that the function runs for 500.000 times and then the mean value is picked as CPU time; this is done for both *Heun* and $RK4$, for each value of h:

Method	**h = 0.5**	**h = 0.2**	**h = 0.05**	**h = 0.01**
Heun	$\sim 10^{-7}$	$\sim 10^{-7}$	$\sim 10^{-6}$	$\sim 10^{-6}$
$RK4$	$\sim 10^{-7}$	$\sim 10^{-7}$	$\sim 10^{-6}$	$\sim 10^{-5}$

In conclusion, it can be affirmed that unfortunately that the choice of a small step size in order to get an high accuracy comes with the cost of an higher computational time; this aspect shall be taken into account, if a small integration error is required.

Exercise 3

Let $\dot{\mathbf{x}} = A(\alpha)\mathbf{x}$ be a two-dimensional system with $A(\alpha) = [0, 1; -1, 2\cos\alpha]$. Notice that $A(\alpha)$ has a pair of complex conjugate eigenvalues on the unit circle; α denotes the angle from the $\mathrm{Re}\{\lambda\}$-axis. 1) Write the operator $F_{\mathrm{RK2}}(h, \alpha)$ that maps $\mathbf{x}_k$ into $\mathbf{x}_{k+1}$, namely $\mathbf{x}_{k+1} = F_{\mathrm{RK2}}(h, \alpha)\,\mathbf{x}_k$. 2)With $\alpha = \pi$, solve the problem "Find $h \geq 0$ s.t.max $(|\mathrm{eig}(F(h, \alpha))|) = 1$". 3) Repeat point 2) for $\alpha \in [0, \pi]$ and draw the solutions in the $(h\lambda)$-plane. 4) Repeat points 1)–3) with RK4.

(5 points)

The operator $\mathbf{F_{RK2}}$(h,α) that maps x_k into x_{k+1} has the following expression:

$$\mathbf{F_{RK2}}(h, \alpha) = \mathbf{I} + h\mathbf{A}(\alpha) + \frac{h^2}{2}\mathbf{A}^2(\alpha) \tag{3}$$

Now that the operator is defined, it is possible to solve the problem "Find h s.t. max(|eig($F(h,\alpha)$)|)=1" for $\alpha = \pi$, which results in $h = 2.0000$.

The same process is applied with Runge-Kutta of fourth order. The operator $\mathbf{F_{RK4}}$(h,α) has the following expression:

$$\mathbf{F_{RK4}}(h,\alpha) = \mathbf{I} + h\mathbf{A}(\alpha) + \frac{h^2}{2}\mathbf{A}^2(\alpha) + \frac{h^3}{6}\mathbf{A}^3(\alpha) + \frac{h^4}{24}\mathbf{A}^4(\alpha) \tag{4}$$

The problem "Find h s.t. max(|eig($F(h,\alpha)$)|)=1" for $\alpha = \pi$, with $\mathbf{F_{RK4}}$(h,α), yields $h = 2.7860$. Thanks to the two operators, and by letting α vary from 0 to π, it is possible to plot the stability domains of the methods $RK2$ and $RK4$ in the $\{h\lambda\}$-plane, as displayed in Fig.6 :

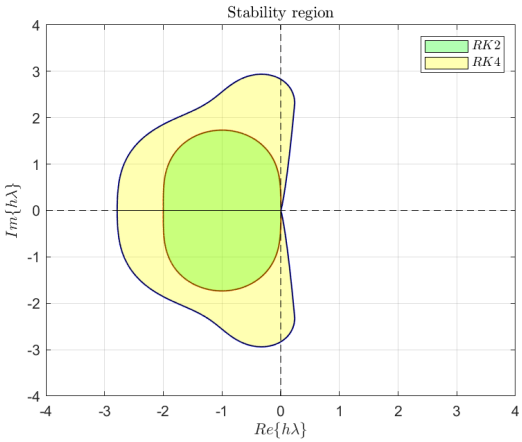

Figure 6: Stability domains of $RK2$ and $RK4$ in the $\{h\lambda\}$-plane

Exercise 4

Consider the IVP $\dot{\mathbf{x}} = A(\alpha)\mathbf{x}$, $\mathbf{x}(0) = [1,1]^T$, to be integrated in $t \in [0,1]$. 1) Take $\alpha \in [0,\pi]$ and solve the problem "Find $h \geq 0$ s.t. $\|\mathbf{x}_{\text{an}}(1) - \mathbf{x}_{\text{RK1}}(1)\|_\infty = \text{tol}$", where $\mathbf{x}_{\text{an}}(1)$ and $\mathbf{x}_{\text{RK1}}(1)$ are the analytical and the numerical solution (with RK1) at the final time, respectively, and $\text{tol} = \{10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$. 2) Plot the four locus of solutions in the $(h\lambda)$-plane; plot also the function evaluations vs tol for $\alpha = \pi$. 3) Repeat points 1)–2) for RK2 and RK4.

(4 points)

The operators of $RK1$,$RK2$ and $RK4$ are exploited to solve the problem "Find $h \geq 0$ s.t. $\|\mathbf{x}_{\text{an}}(1) - \mathbf{x}_{\text{RK1}}(1)\|_\infty = \text{tol}$", being $\mathbf{x}_{an} = e^{\mathbf{A}(\alpha)}\mathbf{x}_0$ the analytical solution. The four locus of solutions, each one for each value of tolerance, are shown in Fig.7, Fig.8, and Fig.9:

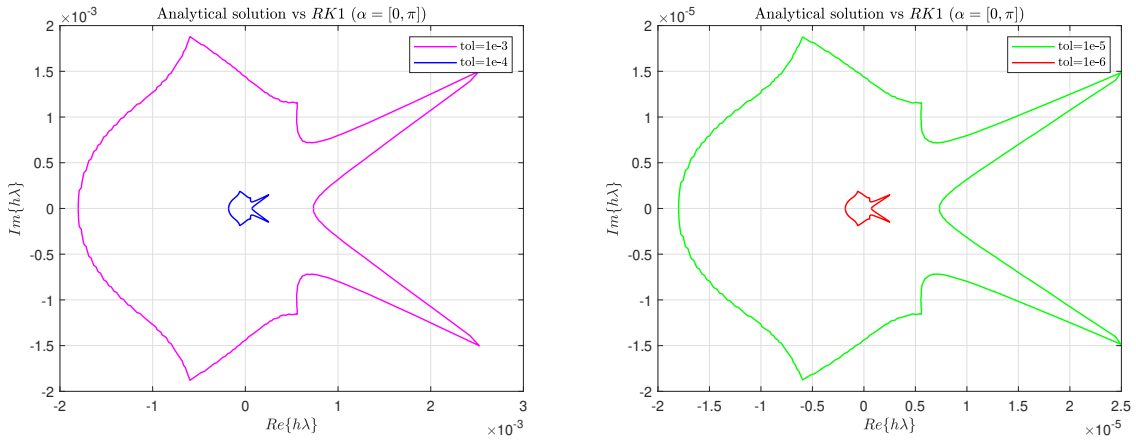

Figure 7: Locus of solutions of $RK1$

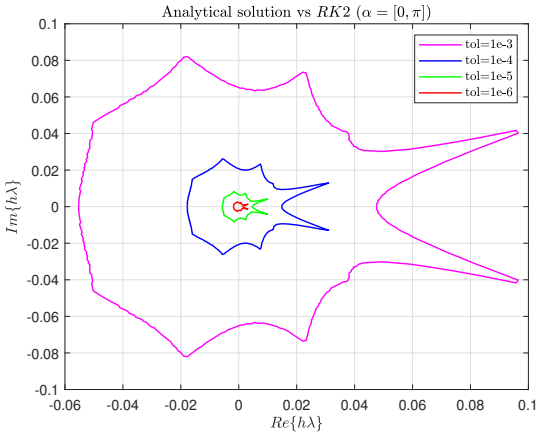

Figure 8: Locus of solutions of $RK2$

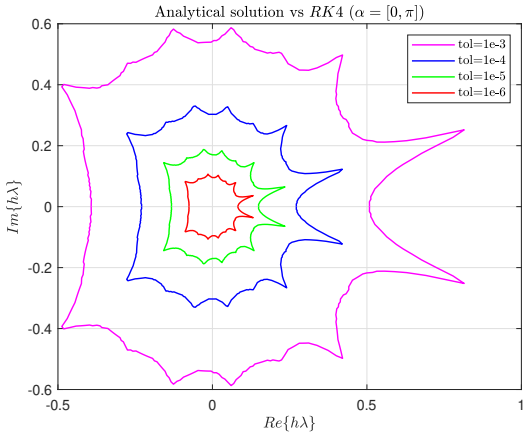

Figure 9: Locus of solutions of $RK4$

Notice that the locus of solutions for $RK1$ for $tol = 10^{-5}$ and $tol = 10^{-6}$ has been plotted in a different graph since it would have been much smaller than the other two locus.
In conclusion, having fixed $\alpha = \pi$, it is possible to point out how the number of function evaluations change for the three methods for different tolerances; more specifically, an higher tolerance brings a lower number of function evaluations:

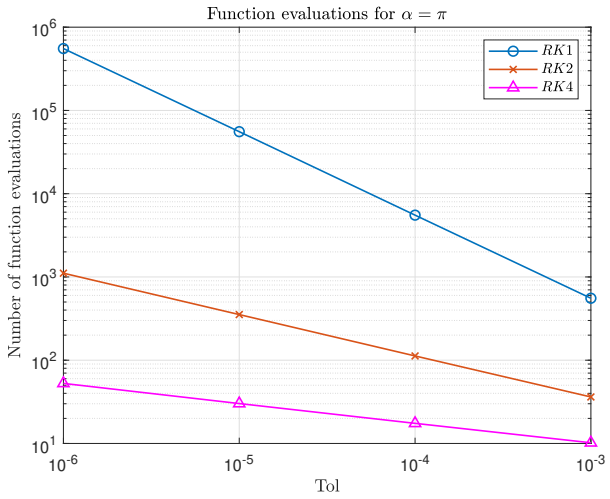

Figure 10: Function evaluations

The higher number of function evaluations of $RK1$ with respect to $RK2$ and $RK4$ is related to the fact that $RK1$ requires a smaller time step h than $RK2$ and $RK4$ to reach the same tolerance, therefore an higher number of function evaluations.

Exercise 5

Consider the backinterpolation method $\text{BI2}_{0.4}$. 1) Derive the expression of the linear operator $B_{\text{BI2}_{0.4}}(h, \alpha)$ such that $\mathbf{x}_{k+1} = B_{\text{BI2}_{0.4}}(h, \alpha)\mathbf{x}_k$. 2) Following the approach of point 3) in Exercise 3, draw the stability domain of $\text{BI2}_{0.4}$ in the $(h\lambda)$-plane. 3) Derive the domain of numerical stability of BI2_θ for the values of $\theta = [0.1, 0.3, 0.7, 0.9]$.

(5 points)

The procedure of derivation of the operator $B_{BI2_{0.4}}(h, \alpha)$ implies firstly the definition of the second order Runge-Kutta method:

$$\begin{cases} \mathbf{x}_{k+1}^P = \mathbf{x}_k + h\mathbf{f}(\mathbf{x}_k, t_k) \\ \mathbf{x}_{k+1}^C = \mathbf{x}_k + \frac{h}{2}[\mathbf{f}(\mathbf{x}_k, t_k) + \mathbf{f}(\mathbf{x}_{k+1}^P, t_{k+1})] \end{cases} \tag{5}$$

By introducing $\theta \in [0, 1]$ it is possible to split the time step into two fractions: θh and $(1-\theta)h$. By substituing $\mathbf{f}(\mathbf{x}_k, t_k) = \mathbf{A}(\alpha)\mathbf{x_k}$ in 5, and merging the first equation into the second,having taken into account a time step of θh, the following equation is obtained:

$$\mathbf{x}_{k+\theta h} = [\mathbf{I} + \theta h\mathbf{A} + \frac{h^2\theta^2}{2}\mathbf{A}^2]\mathbf{x}_k \tag{6}$$

The same procedure is repeated with a time step of $-(1-\theta)h$, meaning that the system is going from $\mathbf{x}_{k+1}$ to $\mathbf{x}_{k+\theta h}$:

$$\mathbf{x}_{k+\theta h} = [\mathbf{I} - (1-\theta)h\mathbf{A} + \frac{h^2(1-\theta)^2}{2}\mathbf{A}^2]\mathbf{x}_{k+1} \tag{7}$$

Now, 6 and 7 are equaled and $\mathbf{x}_{k+1}$ is made explicit:

$$\mathbf{x}_{k+1} = [(\mathbf{I} - (1-\theta)h\mathbf{A} + \frac{h^2(1-\theta)^2}{2}\mathbf{A}^2)^{-1}(\mathbf{I} + \theta h\mathbf{A} + \frac{h^2\theta^2}{2}\mathbf{A}^2)]\mathbf{x}_k \tag{8}$$

From 8 the $B_{BI2_{0.4}}(h, \alpha)$ operator is picked:

$$B_{BI2_{0.4}}(h, \alpha) = [(\mathbf{I} - (1-\theta)h\mathbf{A} + \frac{h^2(1-\theta)^2}{2}\mathbf{A}^2)^{-1}(\mathbf{I} + \theta h\mathbf{A} + \frac{h^2\theta^2}{2}\mathbf{A}^2)] \tag{9}$$

Now that $B_{BI2_{0.4}}(h, \alpha)$ is finally defined, it is possible to identify the stability region of the method for five different values of θ in the $\{h\lambda\}$-plane:

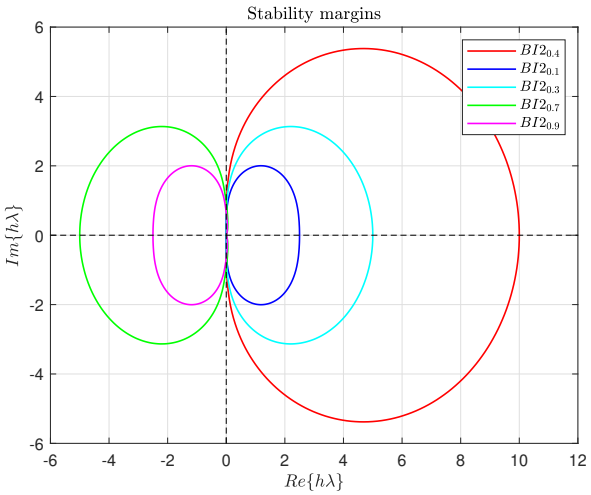

Figure 11: Stability margins of $BI2_\theta$

It is important to point out that the regions inside the margins associated to $\theta = [0.1, 0.3, 0.4]$ are unstable, while the regions inside the margins associated to $\theta = [0.7, 0.9]$ are stable.

Exercise 6

Consider the IVP $\dot{\mathbf{x}} = B\mathbf{x}$ with $B = [-180.5, 219.5; 179.5, -220.5]$ and $\mathbf{x}(0) = [1, 1]^T$ to be integrated in $t \in [0, 5]$. Notice that $\mathbf{x}(t) = e^{Bt}\mathbf{x}(0)$. 1) Solve the IVP using RK4 with $h = 0.1$; 2) Repeat point 1) using implicit extrapolation technique IEX4; 3) Compare the numerical results in points 1) and 2) against the analytic solution; 4) Compute the eigenvalues associated to the IVP and represent them on the $(h\lambda)$-plane both for RK4 and IEX4; 5) Discuss the results. (4 points)

The IVP can be solved numerically by means of the operators F_{RK4} and F_{IEX4} that map $\mathbf{x}_k$ in $\mathbf{x}_{k+1}$. They are compared with the analytical solution in Fig.12 and 13 :

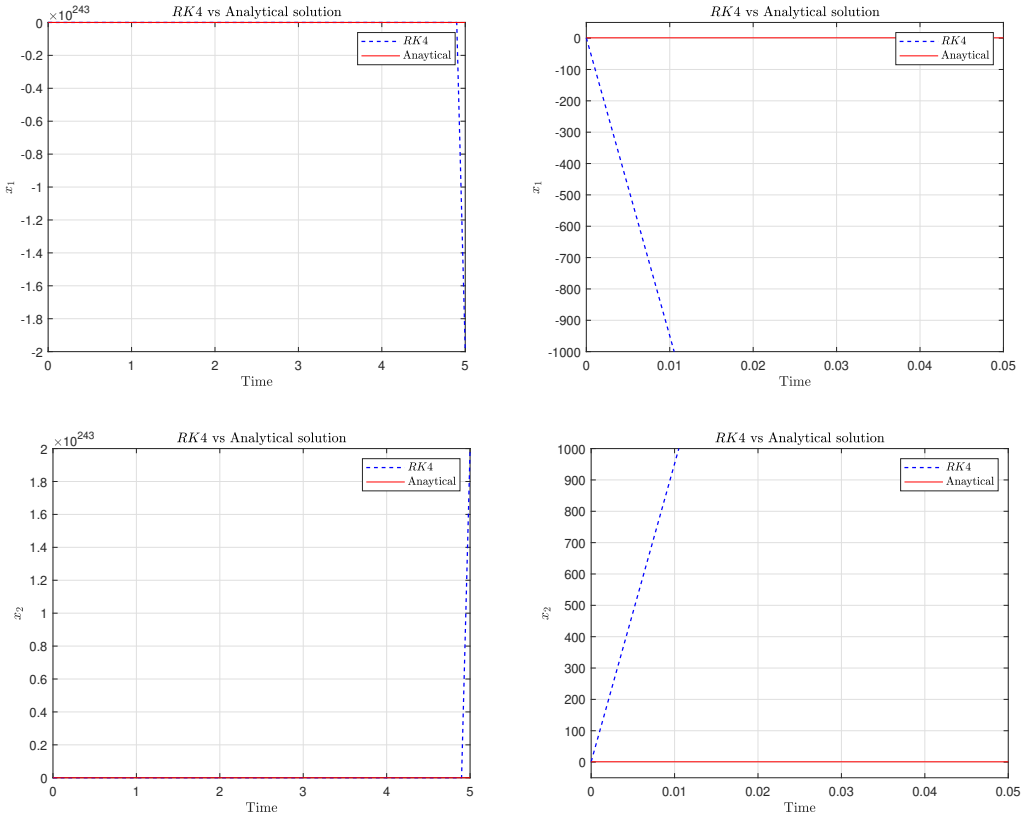

Figure 12: $RK4$ vs Analytical solution for x_1 and x_2

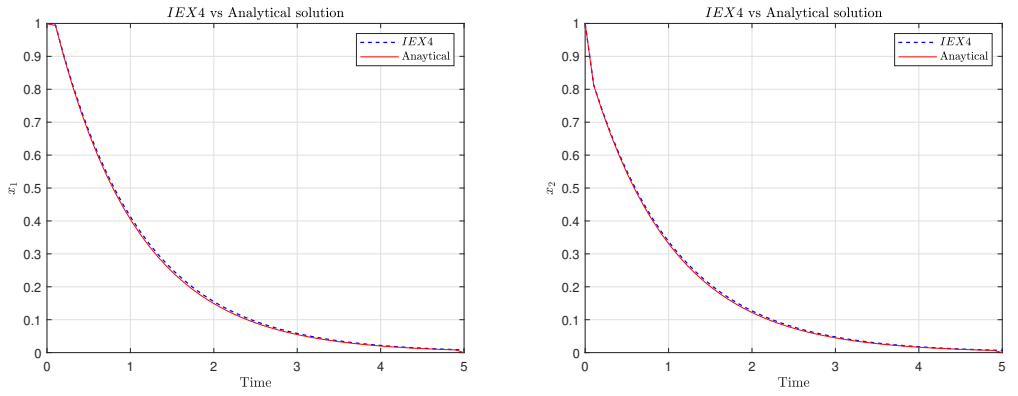

Figure 13: $IEX4$ vs Analytical solution for x_1 and x_2

As shown by the graphs, the numerical solution provided by $RK4$ diverges and does not keep track of the analytical one; on the other hand, the solution coming from $IEX4$ is able to

converge to the analytical solution. This behaviour can be explained by plotting the stability domains of the tho methods and the $h\lambda$ points related to the IVP in Fig.14:

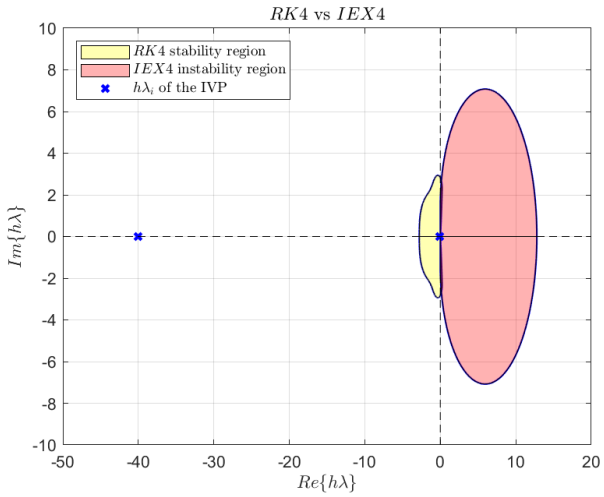

Figure 14: Stability regions of $RK4$ and $IEX4$

The two points are $h\lambda_1 = -0.1$ and $h\lambda_2 = -40$; the region of stability of $RK4$ does not cover a sufficiently wide area to include both the two points within its stability margin, while the instability area of $IEX4$ is located in the right semi-plane and does not cover the two points.

Exercise 7

Consider the two-dimensional IVP

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}-\frac{5}{2}\left[1+8\sin(t)\right]x_1\\ (1-x_1)x_2 + x_1\end{bmatrix}, \qquad \begin{bmatrix}x_1(t_0)\\ x_2(t_0)\end{bmatrix} = \begin{bmatrix}1\\ 1\end{bmatrix}$$

1) Solve the IVP using AB3 in $t \in [0, 3]$ for $h = 0.1$; 2) Repeat point 1) using AM3, ABM3, and BDF3; 3) Discuss the results.

(5 points)

The methods related to $AB3$,$AM3$,$ABM3$, and $BDF3$ are implemented as functions and then the IVP is solved. The solutions are exposed in Fig.15,16,17 and 18 :

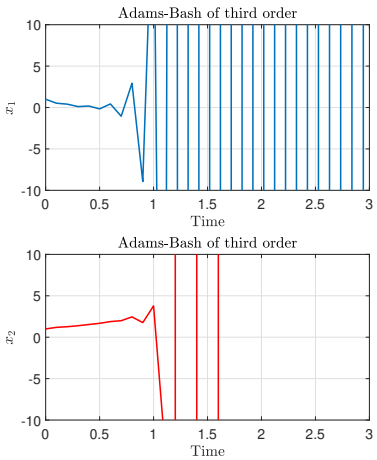

Figure 15: $AB3$

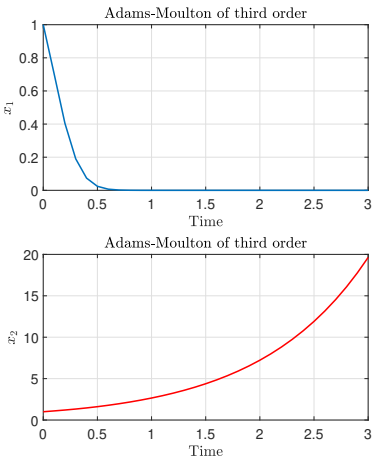

Figure 16: $AM3$

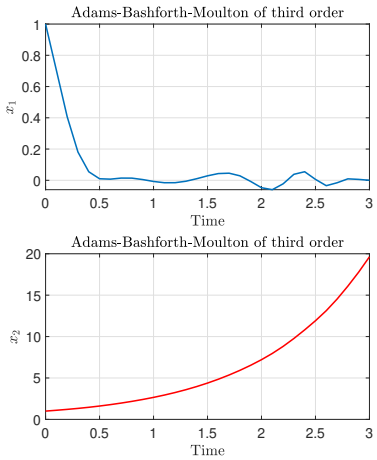

Figure 17: $ABM3$

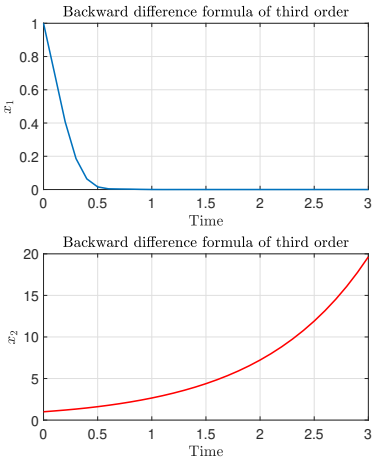

Figure 18: $BDF3$

It is plain that for every method the solution associated to x_2 diverges; regarding x_1, its behaviour changes among the methods. To explain this behaviour it is useful to analyze the evolution of the eigenvalues associated to the IVP. In order to do such a thing, firstly the analytic solution of the equation 10 :

$$\dot{x}_1 = -\frac{5}{2}[1 + 8sin(t)]x_1 \tag{10}$$

is computed, leading to 11:

$$x_1 = e^{-\frac{5}{2}(t-t_0)+20(cos(t)-cos(t_0)} \tag{11}$$

This expression is the merged in 12:

$$\dot{x}_2 = (1 - x_1)x_2 + x_1 \tag{12}$$

Now, by looking at the coefficients of x_1 and x_2 in 10 and 12, is possible to compute the evolution of the products $h\lambda_1$ and $h\lambda_2$ in Fig.19 :

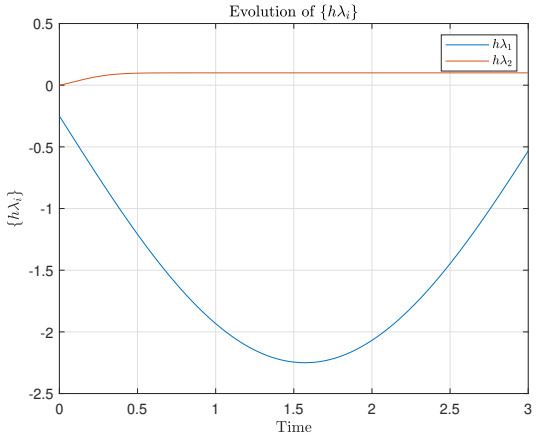

Figure 19: Evolution of $h\lambda_{1,2}$

Since the eigenvalues have no imaginary part, some considerations can be made just by considering the interceptions of the stability domain of each method with the real axes in the $\{h\lambda\}$-plane. The product $h\lambda_2$ assumes always a positive value between 0 and 0.5, a trait which can be found outside the stability regions of $AB3$,$AM3$ and $ABM3$ that are located in the left half of the plane, and inside the instability region of $BDF3$ which is in the right half: that explains the instability related to x_2. Regarding x_1:

- **AB3**: the stability domain of this method intercepts $Re\{h\lambda\}$ axes at 0 and ~ -0.55; $h\lambda_2$ drops below this value after short time;
- **AM3**: the stability region extends to cover a trait in the $Re\{h\lambda\}$ axes from 0 up to -6; $h\lambda_2$ never goes beneath the value of -2.5;
- **ABM3**: x_2 shows some oscillations around 0; that is because x_2 assumes for a short time window values below -1.7 which is where the stability domain of this method crosses $Re\{h\lambda\}$ together with 0 in the plane;
- **BDF3**: the $h\lambda$ values which lies on the $Re\{h\lambda\}$ axes in the left-half semiplane are entirely inside the stability region.