

Exam Numerical Mathematics 1, June 18th 2018, University of Groningen

Use of a simple calculator is allowed. All answers need to be justified.

There are three exercises, detailed on two pages in total. Every exercise will be graded from 1 to 10. To each sub-question in every exercise will be assigned a certain number of points (indicated in a box at the beginning of each subquestion) such that all subquestions sum up 9 points. The final grade will be the arithmetic mean of the grades of Exercises 1, 2 and 3.

Exercise 1

Questions (a)-(d) deal with the approximation of the function $y(x) = \cos(\pi x)$ for $x \in \mathcal{I} = [-1, 2]$.

- (a) 2.5 Give the general form of the interpolation polynomial expressed in the Lagrange characteristic polynomials. Then, compute the interpolation polynomial using the following interpolation points $x_0, x_1, x_2 \in \mathcal{I}$: $x_0 = 0, x_1 = 1, x_2 = 2$. Is it possible to find a different interpolation polynomial from the same set of points $\{x_0, y(x_0)\}, \{x_1, y(x_1)\}, \{x_2, y(x_2)\}$?

The Lagrange characteristic polynomials are given by

$$\phi_k(x) = \prod_{\substack{j=0 \\ j \neq k}}^n \frac{x - x_j}{x_k - x_j} \quad (\mathbf{0.5 \text{ pts}})$$

and the Lagrange form of the interpolant by

$$\Pi_n(x) = \sum_{k=0}^n y(x_k) \phi_k(x) \quad (\mathbf{0.5 \text{ pts}})$$

Replacing the values for $x_i, y(x_i)$

$$\begin{aligned} \Pi_3(x) &= (1) \frac{x-1}{0-1} \frac{x-2}{0-2} + (-1) \frac{x-0}{1-0} \frac{x-2}{1-2} + (1) \frac{x-0}{2-0} \frac{x-1}{2-1} \quad (\mathbf{0.5 \text{ pts}}) \\ &= \frac{1}{2}(x-1)(x-2) + x(x-2) \frac{1}{2}x(x-1) = x^2(0.5 + 1 + 0.5) + x(-0.5 - 1 - 2 - 0.5) + 1 \\ &= 2x^2 - 4x + 1 \quad (\mathbf{0.5 \text{ pts}}) \end{aligned}$$

This is the only possible interpolation polynomial of order 2 from this set of points (**0.25 pts**), since uniqueness can be proved (**0.25 pts**). Other possible answer is: If we do not fix the order of the polynomial, an infinite number of polynomials can be generated (**0.25 pts**) by, for instance, elevating $\phi_i(x)$ to some power (**0.25 pts**).

- (b) 0.5 Give the general formula of the n -th order Taylor expansion around α for some function $f(x)$. Use it for approximating $f(x) = y(x) = \cos(\pi x)$ around x_0 from the following data: $y(x_0), y'(x_0), y''(x_0)$.

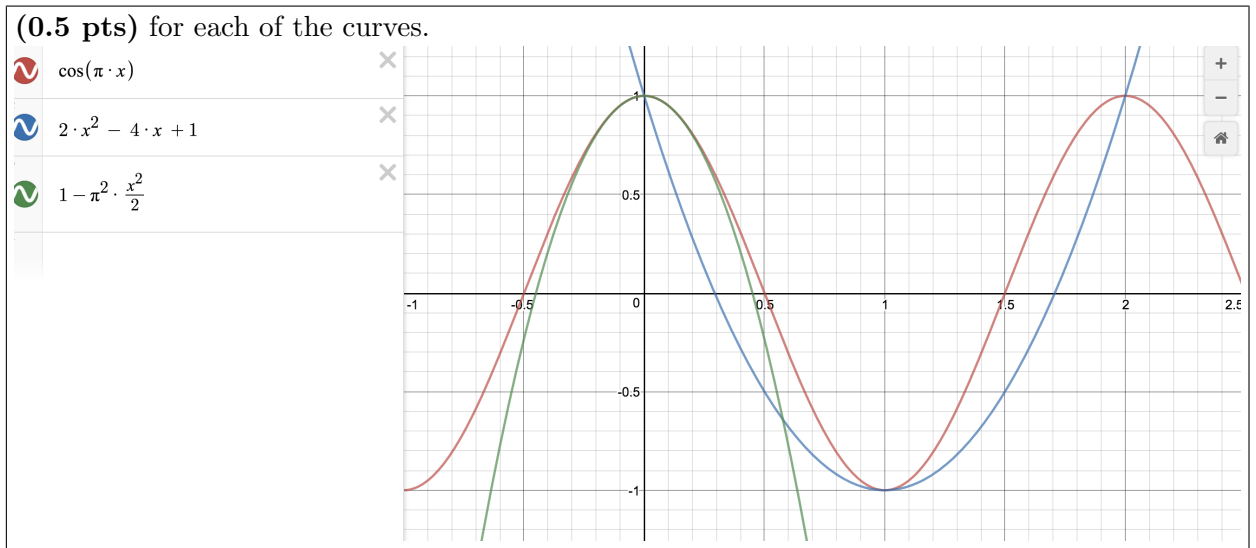
The general formula for the Taylor polynomial reads

$$T_n(x) = f(\alpha) + f'(\alpha)(x - \alpha) + \frac{f''(\alpha)}{2}(x - \alpha)^2 + \dots + \frac{f^{(n)}(\alpha)}{n!}(x - \alpha)^n \quad (\mathbf{0.25 \text{ pts}})$$

The approximating polynomial using the given information then has the form

$$T_2(x) = y(x_0) + y'(x_0)(x - x_0) + \frac{y''(x_0)}{2}(x - x_0)^2 = 1 + 0 \cdot x - \frac{\pi^2 \cos(\pi 0)}{2}x^2 = 1 - \frac{\pi^2}{2}x^2 \quad (\mathbf{0.25 \text{ pts}})$$

- (c) 1.5 Draw $y(x) = \cos(\pi x)$ and the approximating Taylor and Lagrange polynomials, for $x \in \mathcal{I}$. Mark clearly the intersection points of each of the functions with the x - and y -axis, the (x, y) -positions of their minima and maxima, and the values at the extremes of \mathcal{I} .



- (d) **3.0** For each of the points $x = \{-0.25, 0, 1, 1.5\}$, which polynomial (Lagrange or Taylor) yields the most accurate approximation of $y(x)$? You can use either the previous plot or compute the errors. Additionally, comment on the accuracy comparison based on the conditions imposed to build both polynomials.

We will based the answer on the previous plot, but of course specific calculations of the function vales also are possible.

For $x = -0.25$, clearly the Taylor approximation is more accurate (**0.25 pts**). This is since the Lagrange interpolant is designed to approximate the function between the points $x = 0, 1.0, 2.0$, while the Taylor does it around (i.e. at both sides) of the development point, and -0.25 is very close to it (**0.5 pts**).

For $x = 0$, both polynomials are equally accurate (**0.25 pts**) (and matching indeed) since they are both designed to match the target function at that point (**0.5 pts**).

For $x = 1$, the Lagrange interpolant is more accurate (zero error indeed) (**0.25 pts**) since it is designed for matching the function at that point. In contrast, the Taylor approximation is very inaccurate since the asked point is far from the development point (**0.5 pts**).

For $x = 1.5$, the Lagrange interpolant is again more accurate (zero error indeed) (**0.25 pts**) since it is designed for approximating the function between $x = 1.0$ and $x = 2$. In contrast, the Taylor approximation is very inaccurate since the asked point is far from the development point (**0.5 pts**).

Question (e) deal with the numerical integration and it is not related to the previous questions.

- (e) **1.5** We will now define a new composite method to approximate the integral of $f(t)$ over an interval $t \in [a, b]$. The method consists in – after subdividing $[a, b]$ into an equidistant grid $[a, b] = [t_0, t_1] \cup \dots \cup [t_{N-1}, t_N]$ – approximating the integral on each interval $[t_i, t_{i+1}]$ using the following rule: a trapezoidal formula on $[t_i, (t_i + t_{i+1})/2]$ plus a trapezoidal formula on $[(t_i + t_{i+1})/2, t_{i+1}]$.

Give the composite numerical integration formula of the new method.

Derive an error formula for the new composite method.

Which method is more precise, this new one or the standard composite trapezoidal rule? Justify this according to the error formulae of both methods.

Applying the trapezoidal rule for each subinterval we obtain in the interval $[t_i, t_{i+1}]$

$$\frac{1}{2} \frac{h}{2} f(t_i) + \frac{1}{2} \frac{h}{2} f\left(\frac{t_i + t_{i+1}}{2}\right) + \frac{1}{2} \frac{h}{2} f\left(\frac{t_i + t_{i+1}}{2}\right) + \frac{1}{2} \frac{h}{2} f(t_{i+1}) = \frac{h}{4} f(t_i) + \frac{h}{2} f\left(\frac{t_i + t_{i+1}}{2}\right) + \frac{h}{4} f(t_{i+1}) \quad (\mathbf{0.25 pts})$$

with $h = t_{i+1} - t_i$ since the grid is defined as equidistant. Then, summing over all the intervals we obtain the composite form:

$$\int_a^b f(t) \approx \sum_{i=0}^{N-1} \left(\frac{h}{4} f(t_i) + \frac{h}{2} f\left(\frac{t_i + t_{i+1}}{2}\right) + \frac{h}{4} f(t_{i+1}) \right) \quad (\mathbf{0.25 pts})$$

The error of the trapezoidal rule is $|(b-a)h^2 f''(\xi)/12|$ (**0.25 pts**), with $\xi \in [a, b]$. The new method is indeed equivalent to half the size of the grid elements (**0.25 pts**), so the same estimates applies but with $h/2$ (**0.25 pts**). Therefore we expect the new method to be more accurate. (**0.25 pts**)

Exercise 2

Consider the linear system $A_\varepsilon x = b$, where A_ε is given by:

$$A_\varepsilon = \begin{bmatrix} \varepsilon & -1 \\ -1 & \varepsilon \end{bmatrix}$$

with $\varepsilon \in \mathbb{R}$.

- (a) 1.0 Compute the eigenvalues of A_ε . Determine the range of values for ε , such that A_ε is positive definite.

The eigenvalues of A_ε are given by:

$$\det \begin{bmatrix} \varepsilon - \lambda & -1 \\ -1 & \varepsilon - \lambda \end{bmatrix} = (\varepsilon - \lambda)^2 - 1 = \lambda^2 - 2\varepsilon\lambda + (\varepsilon^2 - 1) = 0 \quad (\mathbf{0.25 pts})$$

leads to

$$\lambda_{\pm} = \frac{2\varepsilon \pm \sqrt{4\varepsilon^2 - 4(\varepsilon^2 - 1)}}{2} = \varepsilon \pm 1 \quad (\mathbf{0.25 pts})$$

$\lambda_+ > 0$ if $\varepsilon > -1$ and $\lambda_- > 0$ if $\varepsilon > 1$ (**0.25 pts**). Therefore, the matrix is positive definite if $\varepsilon > 1$. (**0.25 pts**)

Other ways to check positive definiteness are of course valid. But the eigenvalues need the eigenvalues later so computing them here is the shortest way.

- (b) 1.0 For ε in the previously computed range, calculate the 2-norm condition number $K_2(A_\varepsilon)$ of A_ε . Then, determine a such that

$$\lim_{\varepsilon \rightarrow a} K_2(A_\varepsilon) = \infty$$

Since the matrix is symmetric positive definite

$$K_2(A_\varepsilon) = \frac{\lambda_{max}}{\lambda_{min}} = \frac{\varepsilon + 1}{\varepsilon - 1} \quad (\mathbf{0.5 pts})$$

It also follows that $a = 1$ (**0.5 pts**). *There are ways to compute the condition number, the most general one being $K_2(A_\varepsilon) = \|A_\varepsilon\|_2 \|A_\varepsilon^{-1}\|_2$, what is fine if they get to the same result.*

- (c) 1.5 Compute the LU-factorization of A_ε .

We define the LU factorization such that $A_\varepsilon = L_\varepsilon U_\varepsilon$

$$A_\varepsilon = \begin{bmatrix} \varepsilon & -1 \\ -1 & \varepsilon \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \ell & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{21} \\ 0 & u_{22} \end{bmatrix} \quad (\mathbf{0.5 \ pts})$$

hence by solving a linear system for $\ell, u_{11}, u_{21}, u_{22}$ we obtain

$$L_\varepsilon = \begin{bmatrix} 1 & 0 \\ -1/\varepsilon & 1 \end{bmatrix} \quad (\mathbf{0.5 \ pts}), \quad U_\varepsilon = \begin{bmatrix} \varepsilon & -1 \\ 0 & \varepsilon - 1/\varepsilon \end{bmatrix} \quad (\mathbf{0.5 \ pts})$$

- (d) **[2.0]** We want to solve the linear system $A_\varepsilon x = b$ via the LU-factorization of A_ε , with ε in the range such that A_ε is positive definite. That is, we will first solve $L_\varepsilon y = b$ and then $U_\varepsilon x = y$.

Compute the Frobenius-norm condition numbers of L_ε and U_ε .

How does the sensitivity to round-off errors – of the solution procedure using the LU-factorization – change when ε goes from a very large number to a ?

Since L and U are non-symmetric matrices, we should use the general definition of the F-norm condition number $K_F(A) = \|A\|_F \|A^{-1}\|_F$ (**0.25 pts**).

For L_ε , $\|L_\varepsilon\|_F = \sqrt{2 + 1/\varepsilon^2}$ (**0.25 pts**), and $\|L_\varepsilon^{-1}\|_F = \sqrt{2 + 1/\varepsilon^2}$ (**0.25 pts**). Therefore, $K_F(L_\varepsilon) = 2 + 1/\varepsilon^2$.

For U_ε , $\|U_\varepsilon\|_F = \sqrt{\varepsilon^2 + 1 + (\varepsilon - 1/\varepsilon)^2}$ (**0.25 pts**). Then,

$$U_\varepsilon^{-1} = \begin{bmatrix} \varepsilon - 1/\varepsilon & 1 \\ 0 & \varepsilon \end{bmatrix} \frac{1}{\varepsilon(\varepsilon - 1/\varepsilon)} = \begin{bmatrix} 1/\varepsilon & \frac{1}{\varepsilon(\varepsilon - 1/\varepsilon)} \\ 0 & \frac{1}{\varepsilon - 1/\varepsilon} \end{bmatrix}, \quad \|U_\varepsilon^{-1}\|_F = \sqrt{1/\varepsilon^2 + \frac{1}{\varepsilon^2(\varepsilon - 1/\varepsilon)^2} + \frac{1}{(\varepsilon - 1/\varepsilon)^2}}$$

and hence

$$K_F(U_\varepsilon) = \|U_\varepsilon\|_F \|U_\varepsilon^{-1}\|_F = \frac{(\varepsilon - 1/\varepsilon)^2 + 1 + \varepsilon^2}{|\varepsilon(\varepsilon - 1/\varepsilon)|} \quad (\mathbf{0.5 \ pts})$$

The larger the condition number, the more sensitive will be the results to perturbations (**0.25 pts**). Therefore, when $\varepsilon \rightarrow 1$ the solution of the system $L_\varepsilon y = b$ is less sensitive to round-off errors, compared with the system $U_\varepsilon x = y$, which becomes very sensitive since the condition number goes to infinity (**0.25 pts**).

Questions (e)-(f) deal with Richardson iterations and they are not related to the previous questions.

- (e) **[2.5]** We want to solve the following linear system for x_1, x_2 by using stationary Richardson iterations:

$$C \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, \quad C = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix} \quad (4)$$

Use an initial guess for the solution vector $[0, 0]^T$.

First write down the stationary Richardson iterations method for any system matrix and right-hand side. Then, by applying a few Richardson iterations, determine the value of the stationary relaxation parameter α such that the solution obtained with the Richardson method matches the exact solution of the linear system in the lowest possible number of iterations. Why can the optimal value α_{opt} not be applied for this matrix C ?

First, it is easy to verify that the exact solution is given by **(0.25 pts)**

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_2 \\ 2b_2 - b_1 \end{bmatrix}$$

Then, the stationary Richardson method reads **(0.50 pts)**

$$X^k = X^{k-1} + \alpha \left(\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} - CX^{k-1} \right)$$

with X^k the value after the k -th iteration. Given $X^0 = 0$, clearly

$$X^1 = \alpha \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad \text{(0.5 pts)}.$$

We may converge in one iteration if the exact solution would be proportional to the right-hand-side, and we could choose α as the proportionality factor. However, this is not the case in this exercise **(0.25 pts)**. Therefore, we proceed with a second iteration which results

$$X^2 = \alpha \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} + \alpha \left(\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} - \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix} \alpha \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \right) = \alpha \begin{bmatrix} 2b_1 \\ 2b_2 \end{bmatrix} - \alpha^2 \begin{bmatrix} 2b_1 - b_2 \\ b_1 \end{bmatrix} \quad \text{(0.5 pts)}$$

Clearly, if $\alpha = 1$ then the Richardson converges in two iterations. **(0.25 pts)**

The optimal value α_{opt} cannot be applied for this matrix C because it is not symmetric **(0.25 pts)**.

- (f) **1.0** Is it possible to directly apply a Jacobi preconditioner to the linear system (4)? If not, how can we rewrite the system such that we can use a Jacobi preconditioner?

No, because for Jacobi you need to take the inverse the diagonal which is singular **(0.5 pts)**. Since in our problem the diagonal has a zero, we need to apply pivoting to the original problem first **(0.5 pts)**.

Exercise 3

Consider a system of ODEs modeling the electrical potential in a cell:

$$\begin{cases} u' = c_1 u(u - \alpha)(1 - u) - c_2 r, u(0) = u_0 \\ r' = c_2(u - r), r(0) = 0 \end{cases} \quad (5)$$

with $u(t)$ the electrical potential, and $r(t)$ the so called *restitution variable*. $c_1, c_2 > 0$ and $0 < \alpha < 1$ are model constants.

- (a) **1.0** Discretize (5) in time using the θ -method. Formulate system of equations as a vector root finding problem $G(u_{n+1}, r_{n+1}) = 0$, and give the explicit form for G . Here, $u_n \approx u(t_n), r_n \approx r(t_n)$ and assume $t_{n+1} = t_n + \tau, \tau > 0$.

Denoting **(0.5 pts)**

$$F(u, r) = \begin{bmatrix} c_1 u(u - \alpha)(1 - u) - c_2 r \\ c_2(u - r) \end{bmatrix}$$

The nonlinear system of equations results at each time step: **(0.5 pts)**

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = G(u_{n+1}, r_{n+1}) = - \begin{bmatrix} u_{n+1} \\ r_{n+1} \end{bmatrix} + \begin{bmatrix} u_n \\ r_n \end{bmatrix} + \tau(1 - \theta)F(u_n, r_n) + \tau\theta F(u_{n+1}, r_{n+1})$$

- (b) **1.0** Write down the Newton iteration for computing the $(k+1)$ -th iterand $u_{n+1}^{k+1}, r_{n+1}^{k+1}$ from the k -th iterand u_{n+1}^k, r_{n+1}^k . Do it first in terms of G . Then, give explicit expressions for the

vectors and matrix involved, but you do not need to invert any matrix.

The general form of the Newton iterations is **(0.25 pts)**:

$$\begin{bmatrix} u_{n+1} \\ r_{n+1} \end{bmatrix}^{k+1} = \begin{bmatrix} u_{n+1}^k \\ r_{n+1}^k \end{bmatrix} - [J(u_{n+1}^k, r_{n+1}^k)]^{-1} G(u_{n+1}^k, r_{n+1}^k)$$

with J the Jacobian of G .

First, we need to compute the Jacobian of the residual with respect to u_{n+1}, r_{n+1} **(0.5 pts)**:

$$J(u_{n+1}, r_{n+1}) = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} + \theta \tau \begin{bmatrix} c_1((2+2\alpha)u_{n+1} - 3(u_{n+1})^2 - \alpha) & -c_2 \\ c_2 & -c_2 \end{bmatrix}$$

The Newton iteration then has the form **(0.25 pts)**

$$\begin{bmatrix} u_{n+1} \\ r_{n+1} \end{bmatrix}^{k+1} = \begin{bmatrix} u_{n+1}^k \\ r_{n+1}^k \end{bmatrix} - [J(u_{n+1}^k, r_{n+1}^k)]^{-1} \left\{ - \begin{bmatrix} u_{n+1}^k \\ r_{n+1}^k \end{bmatrix} + \begin{bmatrix} u_n \\ r_n \end{bmatrix} + \tau(1-\theta)F(u_n, r_n) + \tau\theta F(u_{n+1}^k, r_{n+1}^k) \right\}$$

- (c) **1.5** Choose θ as in the forward Euler method. Then, compute two Newton iterations, namely u_{n+1}^1, r_{n+1}^1 and u_{n+1}^2, r_{n+1}^2 : first, using as initial guess $u_{n+1}^0 = u_0, r_{n+1}^0 = 0$, and second, using $u_{n+1}^0 = 0, r_{n+1}^0 = 0$.

The forward Euler method is equivalent to the θ -method with $\theta = 0$ **(0.25 pts)**. Hence, the Newton iteration simplifies to **(0.25 pts)**

$$\begin{bmatrix} u_{n+1} \\ r_{n+1} \end{bmatrix}^{k+1} = \begin{bmatrix} u_{n+1} \\ r_{n+1} \end{bmatrix}^k + \left\{ - \begin{bmatrix} u_{n+1}^k \\ r_{n+1}^k \end{bmatrix} + \begin{bmatrix} u_n \\ r_n \end{bmatrix} + \tau F(u_n, r_n) \right\} = \begin{bmatrix} u_n \\ r_n \end{bmatrix} + \tau F(u_n, r_n)$$

and therefore $u_{n+1}^1 = u_{n+1}^2, r_{n+1}^1 = r_{n+1}^2$ **(0.5 pts)**. The result is also independent of the initial guess **(0.5 pts)**

- (d) **2.0** Let J denote the Jacobian of the right-hand side of (5) with $c_1 = c_2$, evaluated at $(u, r) = (0, 0)$. Consider now the system of ODEs:

$$X'(t) = JX(t), \quad X(0) = X_0 \neq 0 \tag{6}$$

Prove that $X(t) \rightarrow 0$ when $t \rightarrow \infty$.

We already have computed the needed Jacobian when formulating the Newton method. Indeed, **(0.25 pts)**

$$\begin{bmatrix} c_1((2+2\alpha)u - 3u^2 - \alpha) & -c_2 \\ c_2 & -c_2 \end{bmatrix}$$

which evaluated in $(0, 0)$ leads to **(0.25 pts)**

$$J = \begin{bmatrix} -\alpha c_1 & -c_2 \\ c_2 & -c_2 \end{bmatrix} = c_1 \begin{bmatrix} -\alpha & -1 \\ 1 & -1 \end{bmatrix}$$

The eigenvalues λ of J/c_1 are given by the equation **(0.25 pts)**

$$\det \begin{bmatrix} -\alpha - \lambda & -1 \\ 1 & -1 - \lambda \end{bmatrix} = 0$$

or equivalently $(\alpha + \lambda)(1 + \lambda) + 1 = 0$, leading to **(0.5 pts)**

$$\lambda = \frac{-(\alpha + 1) \pm \sqrt{(\alpha + 1)^2 - 4(\alpha + 1)}}{2}$$

since $0 < \alpha < 1$, then $Re(\lambda) = -(\alpha + 1)/2 < 0$ **(0.5 pts)**. Since also $c_1 > 0$, $X(t)$ is $X(t) \rightarrow 0$ when $t \rightarrow \infty$ (asymptotically stable) **(0.25 pts)**.

- (e) **1.5** Using a backward Euler method, write a linear system of equations for $X(\tau)$ in terms of $X(0) = (u_0, 0)$ for a general value of τ . Check if $|u(\tau)| < |u(0)|$ for $\tau = 1/c_1$.

The backward Euler method for this problem reads **(0.5 pts)**

$$X(\tau) = X(0) + \tau J X(\tau)$$

so that the linear system reads

$$(\mathbb{I} - \tau J)X(\tau) = X(0)$$

specifically **(0.5 pts)**

$$\begin{bmatrix} 1 + \tau\alpha c_1 & \tau c_1 \\ -\tau c_1 & 1 + \tau c_1 \end{bmatrix} X(\tau) = \begin{bmatrix} u_0 \\ 0 \end{bmatrix}$$

If $\tau = 1/c_1$

$$\begin{bmatrix} 1 + \alpha & 1 \\ -1 & 2 \end{bmatrix} X(\tau) = \begin{bmatrix} u_0 \\ 0 \end{bmatrix}$$

hence $r_1 = u_1/2$, then $(1 + \alpha + 1/2)u_1 = u_0$. Therefore, since $\alpha > 0$, it holds that $|u(\tau)| < |u(0)|$ **(0.5 pts)**

- (f) **2.0** Discretize problem (6) using the forward Euler method. Derive conditions depending on τ , c_1 and the eigenvalues $\lambda_{\pm} = a \pm bi$ of J/c_1 such that $X(t_k) \rightarrow 0$ when $k \rightarrow \infty$. Assume $a < 0$.

The forward Euler method for this problem reads **(0.5 pts)**

$$X_{n+1} = B X_n, B = \mathbb{I} + \tau J$$

the eigenvalues of B are $1 + \tau c_1 \lambda_{\pm}$ **(0.5 pts)**, so that the stability condition reads **(0.5 pts)**

$$|1 + \tau c_1 \lambda_{\pm}| < 1$$

or equivalently

$$|1 + \tau c_1 \lambda_{\pm}|^2 = |1 + \tau c_1 a \pm \tau c_1 b i|^2 = (1 + \tau c_1 a)^2 + (\tau c_1 b)^2 = 1 + 2\tau c_1 a + (\tau c_1 a)^2 + (\tau c_1 b)^2 < 1$$

namely

$$2a + \tau c_1 (a^2 + b^2) < 0$$

so that the stability condition on τ results: **(0.5 pts)**

$$\tau < \frac{-2a}{c_1(a^2 + b^2)}$$