Linear algebra and scientific programming.

Let us consider a matrix $\underline{\mathcal{C}}$ define as :

$$\underline{\mathcal{C}} = \begin{bmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}$$

The aim here is to solve analytically and numericaly the ordinary differential equation (ODE)

$$\frac{d\boldsymbol{X}}{dt} = \underline{\mathcal{C}}\boldsymbol{X} \quad \text{with} \quad \boldsymbol{X}(t=0) = \boldsymbol{X}^0$$

At the time $T = 3$ the solution of this equation is $\boldsymbol{X}(3) = \exp(3\underline{\mathcal{C}})\,\boldsymbol{X}^0$. For $\boldsymbol{X}^0 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$

– Compute $\underline{\mathcal{C}}^2$, $\underline{\mathcal{C}}^3$ and $\underline{\mathcal{C}}^4$.

$$\underline{\mathcal{C}}^2 = \begin{pmatrix} 1 & 1 & -2 \\ -2 & 2 & 1 \\ 1 & -2 & 1 \end{pmatrix}, \quad \underline{\mathcal{C}}^3 = \begin{pmatrix} 0. & -3. & 3. \\ 3. & 0. & -3. \\ -3. & 3. & 0. \end{pmatrix} \quad \underline{\mathcal{C}}^4 = \begin{pmatrix} -3. & 6. & -3. \\ -3. & -3. & 6. \\ 6. & -3. & -3. \end{pmatrix}$$

– Compute $\tilde{\boldsymbol{X}}^{(k)}(3) = \sum_{m=0}^{k} \frac{3^m}{m!}\underline{\mathcal{C}}^m \boldsymbol{X}^0$ for k = 1, 2, 3 and 4.

$$\tilde{\boldsymbol{X}}^{(1)}(3) = \begin{pmatrix} 0. \\ -2. \\ 3. \end{pmatrix}, \quad \tilde{\boldsymbol{X}}^{(2)}(3) = \begin{pmatrix} 4.5 \\ 2.5 \\ -6. \end{pmatrix}, \quad \tilde{\boldsymbol{X}}^{(3)}(3) = \begin{pmatrix} -9. \\ 2.5 \\ 7.5 \end{pmatrix}, \quad \tilde{\boldsymbol{X}}^{(4)}(3) = \begin{pmatrix} 11.25 \\ -7.625 \\ -2.625 \end{pmatrix},$$

– Compute $\boldsymbol{X}(3) = \exp(3\underline{\mathcal{C}})\,\boldsymbol{X}^0$ and errors $e^{(k)}(3) = \boldsymbol{X}(3) - \tilde{\boldsymbol{X}}^{(k)}(3)$ and is norm $\|e^{(k)}(3)\|_2$.

$$\underline{\mathcal{C}} = \begin{pmatrix} 1 & 1 & 1. \\ 1 & \left(\frac{1-\imath\sqrt{3}}{2}\right)^2 & \left(\frac{1+\imath\sqrt{3}}{2}\right)^2 \\ 1 & -\frac{1-\imath\sqrt{3}}{2} & -\frac{1+\imath\sqrt{3}}{2} \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{-3+\imath\sqrt{3}}{2} & 0 \\ 0 & 0 & \frac{-3-\imath\sqrt{3}}{2} \end{pmatrix} \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{-1+\imath\sqrt{3}}{6} & \frac{-1-\imath\sqrt{3}}{6} \\ \frac{1}{3} & \frac{-1-\imath\sqrt{3}}{6} & \frac{-1+\imath\sqrt{3}}{6} \end{pmatrix}, \quad \boldsymbol{X}(3) = \begin{pmatrix} 0.3331 \\ 0.3269 \\ 0.3398 \end{pmatrix}$$

Let us now approximate the solution at the time $T = 3$ as follows :

$$\tilde{\boldsymbol{X}}(3)|_{\delta t} = \tilde{\boldsymbol{X}}^K \quad where \quad \delta t = \frac{3}{K}, \quad \tilde{\boldsymbol{X}}^{n+1} = \tilde{\boldsymbol{X}}^n + \delta t\underline{\mathcal{C}}\tilde{\boldsymbol{X}}^n, \quad 0 \le n \le K-1$$

– Compute $\tilde{\boldsymbol{X}}^K$ for $K = 1, 2, 3, 4, \ldots$.. (take care of the case $K = 3$)

$$\tilde{\boldsymbol{X}}^{K=1} = \begin{pmatrix} 0 \\ -2. \\ 3. \end{pmatrix}, \quad \tilde{\boldsymbol{X}}^{K=2} = \begin{pmatrix} 2.25 \\ 0.25 \\ -1.5 \end{pmatrix}, \quad \tilde{\boldsymbol{X}}^{K=3} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad \tilde{\boldsymbol{X}}^{K=6} = \begin{pmatrix} 0.328 \\ 0.344 \\ 0.328 \end{pmatrix},$$

$$\tilde{\boldsymbol{X}}^{K=10} = \begin{pmatrix} 0.338 \\ 0.322 \\ 0.33 \end{pmatrix},$$

- What is the limit $\lim_{K \longrightarrow \infty} \tilde{\boldsymbol{X}}^K$, compare it to $\boldsymbol{X}(3)$.

For T=100, 1000, 10000, ...

- Compute, numerically with a Fortran 90 language and scilab : $\tilde{\boldsymbol{X}}^{(k)}(T)$ for differents k and compare it to the exact solution.
- Compute, numerically with a Fortran 90 language and scilab : $\tilde{\boldsymbol{X}}^K$ for differents K (with $\delta t = \frac{T}{K}$) and compare it to the exact solution.
- plot the evolution of $\|\tilde{\boldsymbol{X}}^n\|_2$ as a function of n,
  - for $T = 100$ and $K = 100$.
  - for $T = 100$ and $K = 1000$.
  - for $T = 100$ and $K = 10000$.
  - for $T = 10000$ and $K = 10000$.

```
PROGRAM Simple
  IMPLICIT NONE
  INTEGER :: Nx, ix
  REAL    :: Lx, Dx, Pi
  REAL, DIMENSION(:), POINTER :: Coor, Var, VarNew


  ! Opening and rewind the file "DataFile.data" and
  ! associated it to the unit number 10
  ! Units 5 (keeboard) and 6 (sceen) are reserved
  ! -------------------------------------------------
  OPEN(UNIT=10, FILE="DataFile.data")

  ! By this declaration data in "DataFile.data"  are assumed
  ! to be in formatted form (readable).

  ! read the first line of the file and go to the next line
  ! The values readed are of the type of Nx and Lx (INTEGER and REAL)
  ! -------------------------------------------------------
  READ(10,*) Nx, Lx

  ! Close the Unit 1O and the associated file
  ! -----------------------------------------
  CLOSE(10)

  ! set the value of Dx to the result of the  operation
  !     at the right of equality symbol.
  ! -----------------------------------------------------
  Dx = Lx/(Nx-1)
  Pi = 4.0*ATAN(1.0)

  ! ALLOCATE the vectors Coor and Var to the range 1 to  Nx
  ! =====================================================-======
  ALLOCATE( Coor(1:Nx), Var(1:Nx), VarNew(1:Nx) )


  ! The memory to store the components of these variables
  ! is now up to date : we can make operations on it.
  ! =====================================================


  ! Loop to Define coordinates of points
  DO ix = 1, Nx
     Coor(ix) = (ix-1)*Dx
  END DO
```

```
  ! Loop to Define an initial value  Var
  DO ix = 1, Nx
     Var(ix) = SIN( 2.0*PI*Coor(ix) )
  END DO

  ! Open a file to an other unit 11
  ! (but it an be 10 because this unit is now free after close(10))
  ! To save the initial solution  in a formatted form.
  ! -----------------------------------------------------------
  OPEN(UNIT=11, FILE="InitVar.gnu")

  DO ix = 1, Nx
     WRITE(11, *) Coor(ix), Var(ix)
  END DO

  Write(6,*) " This is the actual end of the program"


END PROGRAM Simple
```

Compiling and executing a fortran file program. It is assume that the previous program is in a file name “Simple1.f90”

```
bash-3.2$ echo " 100  1" >DataFile.data
bash-3.2$ ifort -C  Simple1.f90  -o Run
bash-3.2$ ./Run
  This is the actual end of the program
bash-3.2$
```

The program has been run and now the file ”InitVar.gnu” contain the last initialiZation. We can view this formatted file with the software “gnuplot.

```
bash-3.2$ gnuplot
G N U P L O T
        Version 4.2 patchlevel 3
        last modified Mar 2008
        System: Darwin 9.6.0

        Copyright (C) 1986 - 1993, 1998, 2004, 2007, 2008
        Thomas Williams, Colin Kelley and many others

        Type `help` to access the on-line reference manual.
        The gnuplot FAQ is available from http://www.gnuplot.info/faq/

        Send bug reports and suggestions to <http://sourceforge.net/projects/gnu
```

```
Terminal type set to 'aqua'
gnuplot> plot "InitVar.gnu"
gnuplot> quit
bash-3.2$
```