Problem Sheet 7

Problem 7.1 Implementation of an Extrapolated Single-Step Method

(7.1a) Implement a MATLAB function

```
y = extrapolate(Y,n,p),
```

that interpolates the sequence of pairs $\{n_i^{-1}, y_i\}_{i=1}^{k+1}$ with the polynomial

$$p(t) = \alpha_1 t^{p+k-1} + \alpha_2 t^{p+k-2} + \cdots + \alpha_k t^p + \alpha_{k+1}$$

and returns the extrapolated value $p(t = 0)$.

(7.1b) Implement a MATLAB function

```
y = ExplEul(f,y0,h,n),
```

that performs `n` steps of the explicit Euler method with step size `h`. The input `f` is an implementation of the right hand side and `y0` is the initial value.

(7.1c) In a MATLAB function

```
y = extraExplEul(f,y0,T,N,n),
```

implement the extrapolated explicit Euler method. The input is: `f`, the right hand side, `y0`, its initial value, `T`, the end point, `N`, the number of macro steps and the vector `n`, which represents the sequence of numbers of micro steps.

HINT: Use the function `extrapolate(Y,n,p)` from subproblem (7.1a).

(7.1d) Now, we consider the autonomous initial value problem

$$\dot{\mathbf{y}} = -\mathbf{y}, \quad \mathbf{y}(0) = 1.$$

Complete the template `extraExplEulKonv.m`, which performs a convergence study of the extrapolated explicit Euler method. Let $n = 1, ..., 6$ and $N = 2^2, 2^3, 2^4, 2^5$. What do you observe?

Problem 7.2 Lie-Trotter Splitting Method

(7.2a) Write a MATLAB function

```
function [t,y] = lietrotter(psi1, psi2, y0, tspan, N)
```

which implements the Lie-Trotter splitting method on the time interval `tspan`.

(7.2b) Write a MATLAB function

```
function [t,y] = strang(psi1,psi2,psi3,y0,tspan,N)
```

which implements the Strang splitting method on the time interval `tspan`.

(7.2c) Validate your implementations with the help of `cvg_main.m`.

Problem 7.3 Extrapolation of explicit Runge-Kutta methods as Runge-Kutta methods

Extrapolation methods based on explicit methods can be written as explicit Runge–Kutta methods. Here, we will choose an explicit 2-stage Runge-Kutta single step method with $c_1 = 0$ as our base method. Perform an extrapolation step with step size $h_1 = 2\,h_2$. Then, write the extrapolated method as a Runge–Kutta method (Butcher-tableau).

HINT: Use the Aitken-Neville algorithm [NUMODE, Eq. (2.4.5)],[NUMODE, Eq. (2.4.6)].

Published on 31 March 2015.
To be submitted by 14 April 2015.

References

[NUMODE] Lecture Slides for the course “Numerical Methods for Ordinary Differential Equations”, SVN revision # 63606.

Last modified on April 1, 2015