Universität Tübingen
Mathematisches Institut
Prof. Dr. Christian Lubich

Tübingen, May 11th, 2018

4th Exercise sheet – Numerics for instationary differential equations

Exercise 10: (Reformulation of the nonlinear equation systems in RKM)

In order to reduce the influence of rounding errors, one defines $Z_i = Y_i - y_0$. Show:

$$Z = Y - 1 \otimes y_0 = \tau(A \otimes I)F(Y),$$
$$y_1 = y_0 + (d^T \otimes I)Z,$$

where $d^T = b^T A^{-1}$, with the usual notations of the lecture. Formulate the simplified Newton method. Show that $d = e_s$ in the case of Radau methods and thus $y_1 = y_0 + Z_s$.

Exercise 11: (Stopping criteria for the Newton iteration, evaluation of the Jacobian)

(a) The simplified Newton method usually converges linearly: $\|\Delta Z^{(k+1)}\| \leq \theta \|\Delta Z^{(k)}\|$ with a θ, that hopefully satisfies $\theta < 1$. Show that in this case, the error after the $(k+1)$-th step satisfies

$$\|Z^{(k+1)} - Z\| \leq \frac{\theta}{1-\theta}\|\Delta Z^{(k)}\|\,.$$

Hint: telescope sum for $Z^{k+1} - Z^{k+1+j}$.

(b) One can estimate θ by $\theta_k = \|\Delta Z^{(k)}\|/\|\Delta Z^{(k-1)}\|$. Since the iteration error should not be greater than the local error, which should be $\approx$ `tol`, one stops the Newton iteration, if

$$\eta_k \|\Delta Z^{(k)}\| \leq \kappa\texttt{tol}, \qquad \eta_k = \frac{\theta_k}{1-\theta_k}.$$

This strategy can only be applied after at least two iterations. In order to make it possible to stop after the first iteration already, one uses $\eta_0 = \max\{\eta_{old}, eps\}$, where *eps* is the machine accuracy. A good choice for is $\kappa \in [0.01, 0.1]$ (resulting from numerical tests).

To improve efficiency, we limit the number of Newton iterations to $k_{\max} \in \{7, 8, 9, 10\}$. During these $k_{\max}$ steps, the computation is canceled and the step size τ is decreased (e. g. to $\tau/2$), if there exists a k with $\theta_k \geq 1$ (divergence), or if

$$\frac{\theta_k^{k_{\max}-k}}{1-\theta_k}\|\Delta Z^{(k)}\| > \kappa \cdot \texttt{tol}.$$

Think about why the left-hand side of this expression is a coarse estimate for the error after $k_{\max}$ iterations.

If convergence occurs after one step or if the last θ_k is very small, e. g. $\theta_k < 10^{-3}$, then you don't compute a new Jacobian in the next step but continue using the current one.

Programming exercise 2 :

Implement the Radau5-method (Radau IIA of order 5) with constant step size in MATLAB. The Radau IIA method is given by the Butcher tableau

$$
\begin{array}{c|ccc}
\frac{2}{5}-\frac{\sqrt{6}}{10} & \frac{11}{45}-\frac{7\sqrt{6}}{360} & \frac{37}{225}-\frac{169\sqrt{6}}{1800} & -\frac{2}{225}+\frac{\sqrt{6}}{75} \\
\frac{2}{5}+\frac{\sqrt{6}}{10} & \frac{37}{225}+\frac{169\sqrt{6}}{1800} & \frac{11}{45}+\frac{7\sqrt{6}}{360} & -\frac{2}{225}-\frac{\sqrt{6}}{75} \\
1 & \frac{4}{9}-\frac{\sqrt{6}}{36} & \frac{4}{9}+\frac{\sqrt{6}}{36} & \frac{1}{9} \\
\hline
 & \frac{4}{9}-\frac{\sqrt{6}}{36} & \frac{4}{9}+\frac{\sqrt{6}}{36} & \frac{1}{9}
\end{array}
$$

Realize the reformulation of the Newton method of exercise 10 and the stopping criteria of exercise 11.

The programm should display an error, if divergence occurs or if convergence after $k_{\max}$ iterations cannot be ensured.

Test your program by solving the *van der Pol equation*

$$
\begin{aligned}
y_1' &= y_2 \\
\varepsilon y_2' &= (1-y_1^2)y_2 - y_1
\end{aligned}
$$

with initial value $y_1(0) = 2,\ y_2(0) = -0.6$ for different values of von ε and τ. Integrate until $t = 2$ and plot for $\varepsilon = 5 \cdot 10^{-4}$ and $\tau = 10^{-4}$ the approximation to y_1.

Hint: You can proceed like this:

- Implement a function

  `vanderPol_rhs(epsilon,y)`

  and

  `jacobian(epsilon,y)`

  to compute the right-hand side f of the van der Pol-system $y' = f(\varepsilon, y)$ and to compute the corresponding Jacobian matrix, where $y = (y_1, y_2)^T$. Use these functions to compute the right-hand side of the equation system in each Newton iteration and to compute the matrix of the simplified Newton method in each Radau step according to exercise 10.

- A Radau step $t \longrightarrow t + \tau$ can be realized with a function

  `[y_new, eta_new, h] = radau_step(epsilon,J,A,h,t,TOL,y_old,eta_old)`

  where J is the Jacobian, A the coefficient matrix of the RKM, TOL the tolerance in the Newton method and η as in exercise 11.

Solutions are discussed on May 18th.
The programming exercise needs to be handed in by June 1st.
Contact person: Dominik Edelmann,
`edelmann@na.uni-tuebingen.de`, office hours Mo 14 - 16 and by arrangement per email.