ETH Lectures 401-0663-00L Numerical Methods for Computer Science
401-2673-00L Numerical Methods for CSE

Homework Problems and Projects

Prof. R. Hiptmair, SAM, ETH Zurich

Autumn Term 2021, Version of October 2, 2022
(C) Seminar für Angewandte Mathematik, ETH Zürich

Link to the current version of this homework collection

0.1 General Information

0.1.1 Weekly Homework Assignments

All problems will be published in this single ".pdf" file. Every week, we publish a list of problem numbers on the lecture Moodle page. These are the current assignments for that week. Details are given on the lecture Moodle page.

0.1.2 Importance of Homework

Homework assignments are **not** mandatory. However, it is **very important** that you constantly exercise with the material you learn. "Solving" the homework assignments one week before the main exam by looking at the solutions will likely result in failure.

We provide hints and solutions from the beginning. It is your responsibility to look at the solutions only if you are stuck with a difficult problem and you tried to find a solution for a sufficient amount of time.

Most homework problems come with an estimate for the time required for its solution. These times are what we expect a bright and well-prepared student to require for that problem during an exam.

0.1.3 Corrections and Grading of Assignments

Homework assignments will not be graded.

However, you may submit your handwritten solutions by handing them to your tutor directly or uploading them via the Moodle upload interface, see the lecture Moodle page for instructions. The assistants will

Problem 12-4: Semi-implicit Runge-Kutta SSM

General implicit Runge-Kutta methods as introduced in [Lecture → Section 12.3.3] entail solving systems of non-linear equations for the increments, see [Lecture → Rem. 12.3.3.7]. Semi-implicit Runge-Kutta single step methods, also known as Rosenbrock-Wanner (ROW) methods [Lecture → Eq. (12.4.0.6)] just require the solution of linear systems of equations. This problem deals with a concrete ROW method, its stability and aspects of implementation.

Requires the contents of [Lecture → Section 12.1]

We consider the following autonomous ODE:

$$\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}) \tag{12.4.1}$$

and discretize it with a *semi-implicit* Runge-Kutta SSM, a so-called **Rosenbrock method**,

$$\begin{aligned} \mathbf{W}\mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_0)\,, \\ \mathbf{W}\mathbf{k}_2 &= \mathbf{f}(\mathbf{y}_0 + \frac{1}{2}h\mathbf{k}_1) - ah\mathbf{J}\mathbf{k}_1\,, \\ \mathbf{y}_1 &= \mathbf{y}_0 + h\mathbf{k}_2\,, \end{aligned} \tag{12.4.2}$$

where

$$\mathbf{J} = D\mathbf{f}(\mathbf{y}_0)\,, \quad \mathbf{W} = \mathbf{I} - ah\mathbf{J}\,, \quad a = \frac{1}{2+\sqrt{2}}\,.$$

(12-4.a) ⚀ (15 min.) Compute the stability function S of the Rosenbrock method (12.4.2), that is, compute the (rational) function $S(z)$, such that

$$y_1 = S(z)y_0, \quad z := h\lambda,$$

when we apply the method to perform one step of size h, starting from y_0, of the linear scalar model ODE $\dot{y} = \lambda y, \lambda \in \mathbb{C}$.

SOLUTION of (12-4.a):

For a scalar ODE, the Jacobian is just the derivative w.r.t. y, whence $J = Df(y) = f'(y) = (\lambda y)' = \lambda$. The quantity $\mathbf{W}$ is, therefore, a scalar quantity as well:

$$W = 1 - ahJ = 1 - ah\lambda$$

If we plug everything together and assume h is small enough:

$$\begin{aligned} y_1 &= y_0 + h\frac{\mathbf{f}(\mathbf{y}_0 + \frac{1}{2}h\mathbf{k}_1) - ah\mathbf{J}\mathbf{k}_1}{W} \\ &= y_0 + h\frac{\mathbf{f}(\mathbf{y}_0 + \frac{1}{2}h\frac{\mathbf{f}(\mathbf{y}_0)}{W}) - ah\mathbf{J}\frac{\mathbf{f}(\mathbf{y}_0)}{W})}{W} \\ &= y_0 + h\frac{\lambda(\mathbf{y}_0 + \frac{1}{2}h\frac{\lambda\mathbf{y}_0}{1-ah\lambda}) - \frac{ah\lambda^2 y_0}{1-ah\lambda}}{1-ah\lambda} \\ &= \frac{(1-ah\lambda)^2 + h\lambda(1-ah\lambda) + \frac{1}{2}h^2\lambda^2 - ah^2\lambda^2}{(1-ah\lambda)^2}y_0 \\ &= \frac{(1+a^2z^2-2az) + z(1-az) + \frac{1}{2}z^2 - az^2}{(1-az)^2}y_0 \\ &= \frac{1+(1-2a)z + (\frac{1}{2}-2a+a^2)z^2}{(1-az)^2}y_0 \end{aligned}$$

Since $\frac{1}{2} - 2a + a^2 = 0$, it follows:

$$S(z) = \frac{1 + (1-2a)z}{(1-az)^2}$$

▲

(12-4.b) ⚂ (15 min.) [depends on Sub-problem (12-4.a)]

Compute the first 4 terms of the Taylor expansion of $S(z)$ around $z = 0$. What is the maximal $q \in \mathbb{N}$ such that

$$|S(z) - \exp(z)| = O(|z|^q)$$

for $|z| \to 0$? Deduce the maximal possible order of the method Eq. (12.4.2).

HINT 1 for (12-4.b): The idea behind this sub-problem is elucidated in [Lecture → Rem. 12.1.0.21]. Apply [Lecture → Lemma 12.1.0.23]. ⌟

SOLUTION of (12-4.b):

We compute:

$$\begin{aligned} S(0) &= 1, \\ S'(0) &= 1, \\ S''(0) &= 4a - 2a^2, \\ S'''(0) &= 18a^2 - 12a^3 , \end{aligned}$$

therefore

$$S(z) = 1 + z + (2a - a^2)z^2 + (3a^2 - 2a^3)z^3 + O(z^4) .$$

We can also compute the expansion of $\exp$:

$$\exp(z) = 1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3 + O(z^4) .$$

Since $2a - a^2 = \frac{1}{2}$ but $3a^2 - 2a^3 = \frac{1}{2}(\sqrt{2} - 1) \neq \frac{1}{6}$, we have:

$$|S(z) - \exp(z)| = O(|z|^3) .$$

Using [Lecture → Lemma 12.1.0.23], we deduce that the maximal order q of the scheme is $q = 2$.

▲

(12-4.c) ⚂ (15 min.) Implement a C++ function:

```
template <class Function, class Jacobian, class StateType>
std::vector<StateType> solveRosenbrock(
   Function &&f, Jacobian &&df,
   const StateType &y0, unsigned int N, double T);
```

that applies the Rosenbrock method (12.4.2) for solving an initial-value problem for an autonomous ODE $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$.

It takes as input functor objects for $\mathbf{f}$ and $D\mathbf{f}$ (e.g., as lambda functions), an initial data (vector or scalar) $\texttt{y0} = \mathbf{y}(0)$, a number of steps N and a final time T. The function returns the sequence of states generated by the single step method up to $t = T$, using N equidistant steps of the Rosenbrock method.

SOLUTION of (12-4.c):

The implementation relies on EIGEN's direct solvers for dense linear systems.

C++11-code 12.4.3: Solution of Sub-problem (12-4.c)

```
template <class Func, class DFunc, class StateType>
std::vector<StateType> solveRosenbrock(const Func & f, const DFunc & df,
                                       const StateType & y0, unsigned int N,
                                          double T) {
    const double h = T/N;
    const double a = 1. / (std::sqrt(2) + 2.);

    // Will contain all time steps
    std::vector<StateType> res(N+1);
    // Push initial data
    res.at(0) = y0;

    // Some temporary
    StateType k1, k2;
    Eigen::Matrix2d J, W;

    // Main loop: *up to N (performs N steps)*
    for(unsigned int i = 1; i <= N; ++i) {
        StateType & yprev = res.at(i-1);

        // Jacobian computation
        J = df(yprev);
        W = Eigen::Matrix2d::Identity() - a*h*J;

        // Reuse factorization for each step
        auto W_lu = W.partialPivLu();

        // Increments
        k1 = W_lu.solve(f(yprev));
        k2 = W_lu.solve(f(yprev+0.5*h*k1) - a*h*J*k1);

        // Push new step
        res.at(i) = yprev + h*k2;
    }

    return res;
}
```

▲

(12-4.d) ⚂ (15 min.) [depends on Sub-problem (12-4.c)]

Explore the order of the method (12.4.2) empirically by applying it to the IVP for the limit cycle [Lecture

→ Ex. 12.2.0.4]:

$$\mathbf{f}(\mathbf{y}) := \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{y} + \lambda(1 - \|\mathbf{y}\|^2)\mathbf{y} \,, \tag{12.4.4}$$

with $\lambda = 1$ and initial state $\mathbf{y}_0 = [1,1]^\top$ on $[0,10]$. Use uniform timesteps of size $h = 2^{-k}, k = 4,\ldots,10$ and compute a reference solution $\mathbf{y}_{\mathrm{ref}}$ with timestep size $h = 2^{-12}$. Monitor the maximal error on the temporal mesh

$$\max_j \|\mathbf{y}_j - \mathbf{y}_{\mathrm{ref}}(t_j)\|_2 \,,$$

and use it to estimate a rate of algebraic convergence by means of linear regression.

To do this write a C++ function

```
double cvgRosenbrock(void);
```

that tabulates the errors and returns the estimated rate of convergence.

SOLUTION of (12-4.d):

The Jacobian Df is:

$$D\mathbf{f}(\mathbf{y}) = \begin{bmatrix} \lambda(1 - \|\mathbf{y}\|^2) - 2\lambda y_0^2 & -1 - 2\lambda y_1 y_0 \\ 1 - 2\lambda y_1 y_0 & \lambda(1 - \|\mathbf{y}\|^2) - 2\lambda y_1^2 \end{bmatrix}$$

For the implementation, *cf.* `rosenbrock.cpp`.

▲

In complex analysis you might have hear about the maximum principle for holomorphic/analytic functions. The following is a special version of it.

Theorem 12.4.5. Maximum principle for holomorphic functions

Let

$$\mathbb{C}^- := \{z \in \mathbb{C} \mid Re(z) < 0\} \,.$$

Let $f : D \subset \mathbb{C} \to \mathbb{C}$ be non-constant, defined on $\overline{\mathbb{C}^-}$, and analytic in $\mathbb{C}^-$. Furthermore, assume that $w := \lim_{|z|\to\infty} f(z)$ exists and $w \in \mathbb{C}$, then:

$$\forall z \in \mathbb{C}^- : \quad |f(z)| < \sup_{\tau\in\mathbb{R}} |f(i\tau)| \,.$$

(12-4.e) ⚀ (20 min.) Appealing to Thm. 12.4.5 show that the method (12.4.2) is L-stable (*cf.* [Lecture → § 12.3.4.11]).

HINT 1 for (12-4.e): To investigate the A-stability, calculate the complex norm of $S(z)$ on the imaginary axis $\mathrm{Re}\, z = 0$ and apply the ⌟

SOLUTION of (12-4.e):

We start by proving that the method is A-stable [Lecture → § 12.3.4.4], meaning that $S(z) < 1, \forall z \in \mathbb{C}, \mathrm{Re}(z) < 0$. First of all, we can compute the complex norm of the stability function at $z = iy, y \in \mathbb{R}$ as:

$$|S(iy)|^2 = \frac{|1 + (1-2a)iy|^2}{|(1-aiy)^2|^2} = \frac{1 + (1-2a)^2y^2}{(1+a^2y^2)^2} = \frac{1 + (1-4a+4a^2)y^2}{(1+2a^2y^2+a^4y^4)} .$$

Notice that $1 - 4a + 4a^2 = 2a^2$, therefore:

$$|S(iy)|^2 = \frac{1 + 2a^2y^2}{1 + 2a^2y^2 + a^4y^4} < 1 .$$

The norm of the function S is bounded on the imaginary axis. Observe that $|S(z)| \to 0, |z| \to \infty$, which, follows from the fact that the degree of the denominator of S is bigger than the polynomial degree of the numerator. Notice that $1 - 2a = 1 - 2\frac{1}{\sqrt{(2)}+2} = \sqrt{2} - 2$.

The only pole of the function is at $z = 1/a$, therefore the function is holomorphic on the left complex plane.

Applying the theorem of the hint on concludes that the absolute value of the function is bounded by 1 on the left complex plane.

The L-stability follows immediately with A-stability and the fact that the absolute value of the function converges to zero as $|z| \to \infty$.

▲

End Problem 12-4 , 80 min.