Numerical Analysis – Computational Session

Simplified Newton Method for Implicit Runge–Kutta Schemes

Marco Sutti

January 7, 2026

The main reference for this computational session is the Numerical Analysis class of December 2, 2025, and the book by E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II – Stiff and Differential-Algebraic Problems* [1]. You can also have a look at the Wikipedia page on Radau methods.

1 Objective

This session aims to study the implementation of an implicit Runge–Kutta method for stiff systems of ordinary differential equations. In particular, you should:

- implement an implicit Runge–Kutta method (Radau IIA, 3 stages);
- solve the Runge–Kutta equations using a *simplified Newton method* (as seen in class);
- approximate the Jacobian numerically using finite differences (compare it with the analytical formulas);
- implement step-size reduction by a γ factor when Newton's method fails.

This exercise follows the discussion in [1, Sec. IV.8].

2 Test Problem: The Van der Pol Equation

We consider the Van der Pol equation (a stiff system of ODEs)

$$y_1' = y_2 \qquad y_1(0) = 2 \tag{2.1}$$

$$y_2' = \left((1 - y_1^2)y_2 - y_1\right)/\varepsilon \qquad y_2(0) = -0.66 \tag{2.2}$$

with $\varepsilon = 10^{-2}$ on the interval $[0, 5]$. You need a function that computes the right-hand side of this differential equation, and another function that computes its Jacobian.

3 Radau IIA Method (3 Stages)

The 3-stage Radau IIA method is a fifth-order method defined by the Butcher tableau

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

You can copy-paste the following coefficients for your code:

```
sqrt6 = sqrt(6);

c = [(4 - sqrt6)/10;
     (4 + sqrt6)/10;
      1];

A = [  (88 - 7*sqrt6)/360,    (296 - 169*sqrt6)/1800, (-2 + 3*sqrt6)/225;
       (296 + 169*sqrt6)/1800, (88 + 7*sqrt6)/360,    (-2 - 3*sqrt6)/225;
       (16 - sqrt6)/36,        (16 + sqrt6)/36,        1/9 ];

b = A(end,:);
```

The Runge–Kutta equations for this 3-stage ($s = 3$) method are

$$Y_i = y_0 + h \sum_{j=1}^{3} a_{ij}\, f(x_0 + c_i h, Y_j), \quad i = 1, 2, 3. \tag{3.1}$$

To reduce the effect of round-off errors, we work with the smaller quantities

$$z_i = Y_i - y_0.$$

Then (3.1) becomes

$$z_i = h \sum_{j=1}^{3} a_{ij}\, f(x_0 + c_j h, y_0 + z_j), \quad i = 1, 2, 3.$$

As done in the class of December 2, define the stacked vector

$$Z = (z_1, z_2, z_3)^\top \in \mathbb{R}^{3d}.$$

Then

$$Z = (hA \otimes I_d) \underbrace{\begin{pmatrix} f(x_0 + c_1 h, y_0 + z_1) \\ f(x_0 + c_2 h, y_0 + z_2) \\ f(x_0 + c_3 h, y_0 + z_3) \end{pmatrix}}_{=:F(Z)}.$$

The nonlinear system can be written as

$$G(Z) = Z - (hA \otimes I_d)\, F(Z) = 0.$$

4 Simplified Newton Method

At each time step:

1. Approximate the Jacobian
$$\frac{\partial f}{\partial y}(x_0 + c_i h, y_0 + z_i) \approx \frac{\partial f}{\partial y}(x_0, y_0) =: J.$$

2. Compute J either numerically (using finite differences) or analytically.

3. Use the simplified Newton iteration
$$\Delta Z^{(k+1)} = -(I_{3d} - hA \otimes J)^{-1}\, G(Z^{(k)}),$$
$$Z^{(k+1)} = \Delta Z^{(k)} + \Delta Z^{(k+1)}.$$
Each iteration requires s evaluations of f and the solution of an sd-dimensional linear system. Note that the matrix $(I_{3d} - hA \otimes J)$ is the same for all iterations of the simplified Newton method. Its LU-decomposition is done only once and is usually very costly.

4. Keep the Newton matrix fixed during the iteration.

5 Step-Size Reduction

Use a constant step size h. If the simplified Newton method does not converge within a prescribed number of iterations, then we reject the step, reduce the step size by a factor γ: $h \leftarrow \gamma h$ with $0 < \gamma < 1$, and finally retry the step. We stop if $h < h_{\min}$.

6 Your Tasks

1. Implement the Radau IIA method with simplified Newton iteration.
2. Implement finite-difference approximation of the Jacobian or compute it analytically.
3. Implement step-size reduction.
4. Plot the numerical solution y_1. You should get a plot like the one in Figure 1.
5. Briefly discuss Newton convergence and stiffness effects.

References

[1] Ernst Hairer and Gerhard Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, Heidelberg, 2nd edition, 1996.

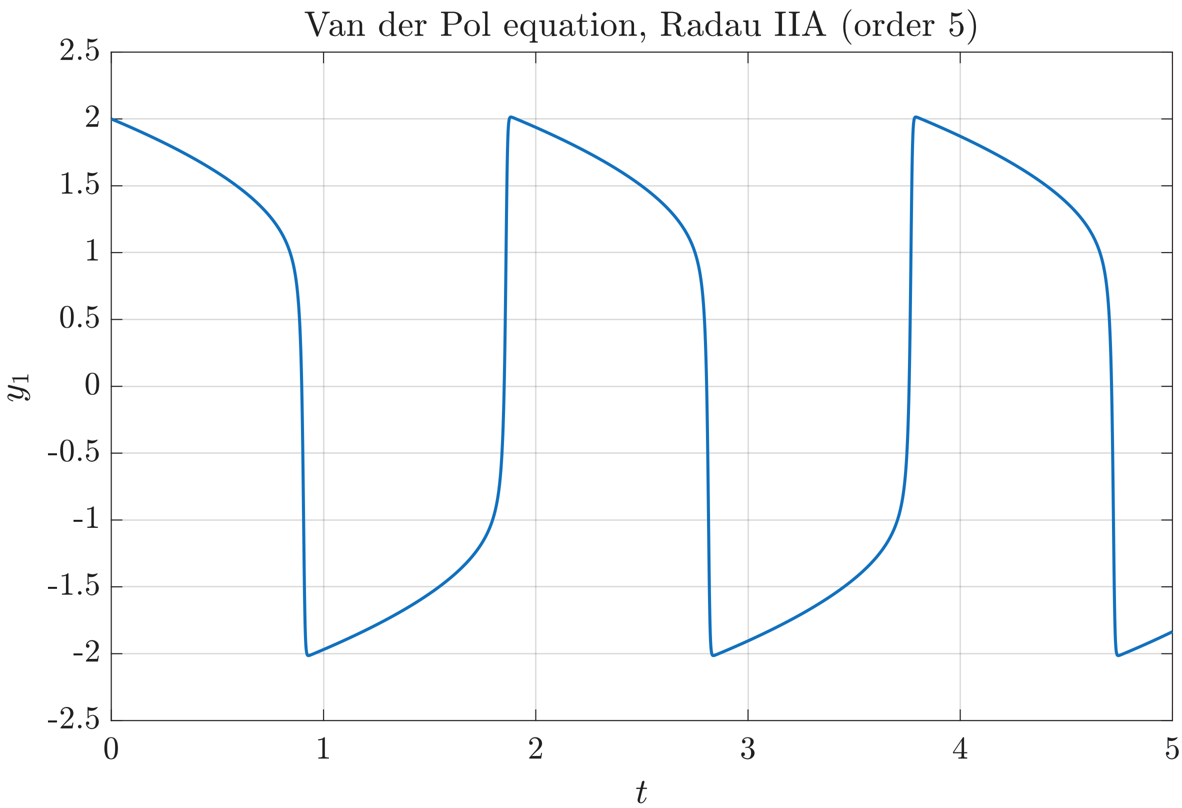

Figure 1: Numerical solution of the Van der Pol equation with the Radau IIA method.