IV.1 Examples of Stiff Equations

... Around 1960, things became completely different and everyone became aware that the world was full of stiff problems.

(G. Dahlquist in Aiken 1985)

Stiff equations are problems for which explicit methods don’t work. Curtiss & Hirschfelder (1952) explain stiffness on one-dimensional examples such as

\[ y' = -50(y - \cos x). \]  

(1.1)

Solution curves of Equation (1.1) are shown in Fig. 1.1. There is apparently a smooth solution in the vicinity of \( y \approx \cos x \) and all other solutions reach this one after a rapid "transient phase". Such transients are typical of stiff equations, but are neither sufficient nor necessary. For example, the solution with initial value \( y(0) = 1 \) (more precisely 2500/2501) has no transient. Fig. 1.2 shows Euler polygons for the initial value \( y(0) = 0 \) and step sizes \( h = 1.974/50 \) (38 steps) and \( h = 1.875/50 \) (40 steps). We observe that whenever the step size is a little too large (larger than 2/50), the numerical solution goes too far beyond the equilibrium and violent oscillations occur.

Looking for better methods for differential equations such as (1.1), Curtiss and Hirschfelder discovered the BDF method (see Sect. III.1): the approximation
$y \approx \cos x$ (i.e., $f(x, y) = 0$) is only a crude approximation to the smooth solution, since the derivative of $\cos x$ is not zero. It is much better, for a given solution value $y_n$, to search for a point $y_{n+1}$ where the slope of the vector field is directed towards $y_n$, hence

$$\frac{y_{n+1} - y_n}{h} = f(x_{n+1}, y_{n+1}). \quad (1.2)$$

This is the implicit Euler method. The dotted line in Fig. 1.1 consists of three implicit Euler steps and demonstrates impressively the good stability property of this method. Equation (1.1) is thus apparently “stiff” in the sense of Curtiss and Hirschfelder.

Extending the above idea “by taking higher order polynomials to fit $y$ at a large number of points” then leads to the BDF methods.

**Chemical Reaction Systems**

When the equations represent the behaviour of a system containing a number of fast and slow reactions, a forward integration of these equations becomes difficult. (H.H. Robertson 1966)

The following example of Robertson’s (1966) has become very popular in numerical studies (Willoughby 1974):

$$\begin{align*}
A &\quad 0.04 & B \\
B + B &\quad 3 \cdot 10^7 & C + B \\
B + C &\quad 10^4 & A + C
\end{align*} \quad (1.3)$$

which leads to the equations

$$\begin{align*}
A: &\quad y_1' = -0.04y_1 + 10^4y_2y_3 & y_1(0) = 1 \\
B: &\quad y_2' = 0.04y_1 - 10^4y_2y_3 - 3 \cdot 10^7y_2^2 & y_2(0) = 0 \quad (1.4) \\
C: &\quad y_3' = 3 \cdot 10^7y_2^2 & y_3(0) = 0.
\end{align*}$$

After a bad experience with explicit Euler just before, let’s try a higher order method and a more elaborate code for this example: DOPRI5 (cf. Volume 1). The numerical solutions obtained for $y_2$ with $Rtol = 10^{-2}$ (209 steps) as well as with $Rtol = 10^{-3}$ (205 steps) and $Atol = 10^{-6}$ · $Rtol$ are displayed in Fig. 1.3. Fig. 1.4 presents the step sizes used by the code and also the local error estimates. There, all rejected steps are crossed out.

We observe that the solution $y_2$ rapidly reaches a quasi-stationary position in the vicinity of $y_2 = 0$, which in the beginning ($y_1 = 1$, $y_3 = 0$) is at $0.04 \approx 3 \cdot 10^7y_2^2$, hence $y_2 \approx 3.65 \cdot 10^{-5}$, and then very slowly goes back to zero again. The numerical method, however, integrates this smooth solution by thousands of apparently unnecessary steps. Moreover, the chosen step sizes are more or less independent of the chosen tolerance. Hence, they seem to be governed by stability