§1.11 Stiff ODEs

Def. There are many ways of defining ~~stiff differential~~ equations.

The most important common feature of these definitions is that when such equations are being solved with standard numerical methods, the stepsize Δt is forced to be extremely small in order to maintain stability – far smaller than would appear to be necessary based on a consideration of the truncation error.

Rmk. Practical choice of Δt

When choosing Δt, we need to consider the two conditions:

(1) $\Delta t \leq \Delta t_{acc}$, where Δt_{acc} is the time step based on accuracy consideration.

→ Δt_{acc} depends on

(i) numerical methods that determines $\sum_{i=1}^{n}$,
(ii) solution smoothness,
(iii) required accuracy.

(2) $\Delta t \leq \Delta t_{stab}$, where Δt_{stab} is the time step based on stability consideration.

→ Δt_{stab} depends on the eigenvalues of $\frac{\partial f}{\partial u}$, and $\Delta t \leq \Delta t_{stab}$ will provide absolute stability.

Rmk. Typically, we would like to choose Δt based on accuracy considerations, i.e.,

$$\boxed{\Delta t \leq \Delta t_{acc} < \Delta t_{stab}}, \quad \text{--- (1)}$$

Rmk. If stability considerations force us to use a much smaller Δt than E_4^h indicates should be needed, i.e.,

$$\boxed{\Delta t \leq \Delta t_{stab} \ll \Delta t_{acc}}, \quad \text{--- (2)}$$

then this particular method is probably not optimal for this given IVP.

Rmk (2) is the case with stiff ODEs (and PDEs). The main reason that the stiffness occurs in solving ODEs and PDEs is that, in the given system (e.g., IVPs, BVPs, PDEs, etc) there exist two or more widely varying time scales.

→ In such system, a time scale is the time that it takes for

(i) a transient, or
(ii) a short-lived, or
(iii) a part of soln to die away exponentially from its initial value.

$\Delta t_{acc} \sim \frac{\Delta x}{|\lambda|}$

$\Delta t_{diff} \sim \frac{\Delta x^2}{k}$

→ The numerical soln of a stiff equ. (or a system) is very difficult because one needs to use a very small time step Δt in order to

obtain a stable, accurate soln of the short-time scale part of the soln.

→ Once this part of the soln is numerically achived, we are tempted to increase the time step Δt because the remaining part of the soln is very slowly changing and should yield a stable, accurate soln with relatively large Δt's.

→ Unfortunately, although the transient part of the soln has long ago become insignificant, the differential eqn has not changed, and large Δt's will bring on the instability on the transient part of the soln.

→ Very small time steps are the only soln to resolve this issue, which will be very costly in terms of computer time.

→ One natural way to avoid the small restricted Δt's is to resort to using "implicit methods", since they allow large time steps with large abolute stability regions.

Ex1. For systems of ODEs, stiffness is sometimes defined in terms of the "stiffness ratio" of the system, defined by

$$\lambda_{ratio} = \frac{\max |\lambda_p|}{\min |\lambda_p|}, \text{ where}$$

λ_p, $p = 1, \cdots, N$, are the eigenvalues of the Jacobian matrix $\left(\frac{\partial \underset{\sim}{f}}{\partial \underset{\sim}{u}}\right)_{N \times N}$, $\underset{\sim}{u} \in \mathbb{R}^N$, $\underset{\sim}{f} : \mathbb{R}^N \to \mathbb{R}^N$.

→ If the ratio is large, ∃ a large range of time scales in the problem.

→ It should be noted that a system of ODEs with a large stiffness ratio is not necessarily stiff.

(ex) Let λ_{max} be the eigenvalue with the largest modulus in the system, and assume $\lambda_{max} \in \mathbb{C}$. If λ_{max} lies very close to the imaginary axis (i.e, $Re(\lambda_{max}) \approx 0$) then it leads to highly oscillatory soln behavior with slow damping.

→ $\lambda_{max} = a + bi$, $0 < |a| < \varepsilon$, $b \neq 0$

→ $u(t) = c_1 e^{at} \cos bt + c_2 e^{at} \sin bt$

In this case, Δt_{acc} also needs to be very small for accuracy reason, and Δt_{acc} may be the same magnitude as Δt_{stab} anyways,

$$\frac{\Delta t_{acc}}{\Delta t_{stab}} \sim O(1).$$

Hence the system is not really stiff.

Ex2. For a scalar ODE, $\underline{\lambda_{ratio} = 1}$ always, since there is only one eigenvalue.

However, there could exist more than one time scale due to nonhomogeneous terms $g(t)$.

Consider the IVP:

$$\begin{cases} u'(t) = f(t,u) + g(t) \\ \qquad\quad = \lambda(u - \cos t) - \sin t \\ u(0) = 1. \quad \cdots (3) \end{cases}$$

→ One particular soln satisfying the IC is

$$u(t) = \cos t. \quad \cdots (4)$$

→ Jacobian matrix is a scalar (indeed an eigenvalue),

$$\frac{\partial f}{\partial u} = \frac{\partial}{\partial u}(\lambda u) = \lambda. \quad (\because) \ \lambda_{ratio} = \frac{|\lambda|}{|\lambda|} = 1.$$

→ Now we modify our IC to

$$u(t^0) = u^0, \quad u^0 \neq 0, \ u^0 \neq 1. \quad \cdots (5)$$

we obtain a soln different from $u(t) = \cos t$, given as

$$u(t) = \underbrace{e^{\lambda(t-t_0)}(u^0 - \cos(t^0))}_{(*)} + \underbrace{\cos t}_{(**)}. \quad \cdots (6)$$

→ If $\lambda < 0$ (or $\mathrm{Re}(\lambda) < 0$ if $\lambda \in \mathbb{C}$), then $(*) \to 0$ and hence $\underline{(6) \to (4) \text{ with decay rate } \lambda.}$

→ In general, we can consider (5) as a perturbation of the original problem (3)

→ In this way, perturbing the given IVP with ③ gives rise to a yet new soln ⑥ which asymptotically approaches to ④ as $t \to \infty$ with different rate λ (for different λ).

→ This is an example illustrating that there are two different time scales of :

(i) a part of soln that vanishes away exponentially from its unperturbed soln (i.e, (*)), and

(ii) the unperturbed soln (i.e, ④, or (**))

Ex 3. Consider Ex 2 with $\lambda = -10^6$ using

(i) Forward Euler (FE)

(ii) Backward Euler (BE)

(iii) Trapezoidal method (Trap)

Case 1: IC is given as $u(0) = 1$.

→ There is no transient soln and $u(t) = \text{cst}$.

(i) FE: $0 \leq \Delta t \leq \frac{-2}{\lambda} \Rightarrow \underline{\Delta t \leq 2 \times 10^{-6}}$ (very small)

(ii) BE & Trap: one can use large Δt as they are A-stable.

→ FE is impractical.

Case 2: IC is given as $u(0) = 1.5$

→ There is an initial rapid transient toward $u(t) = \text{cst}$ on a time scale of about 10^{-6}.

→ Since FE is not practical to use anyway, we only consider BE & Trap.

→ For both BE & Trap, they are both absolutely stable and their solns are bounded, and convergent.

→ However, Trap is not good enough and BE works much better,

→ To see why when $\Delta t = 0.1 \Rightarrow \lambda \Delta t = -10^5$,

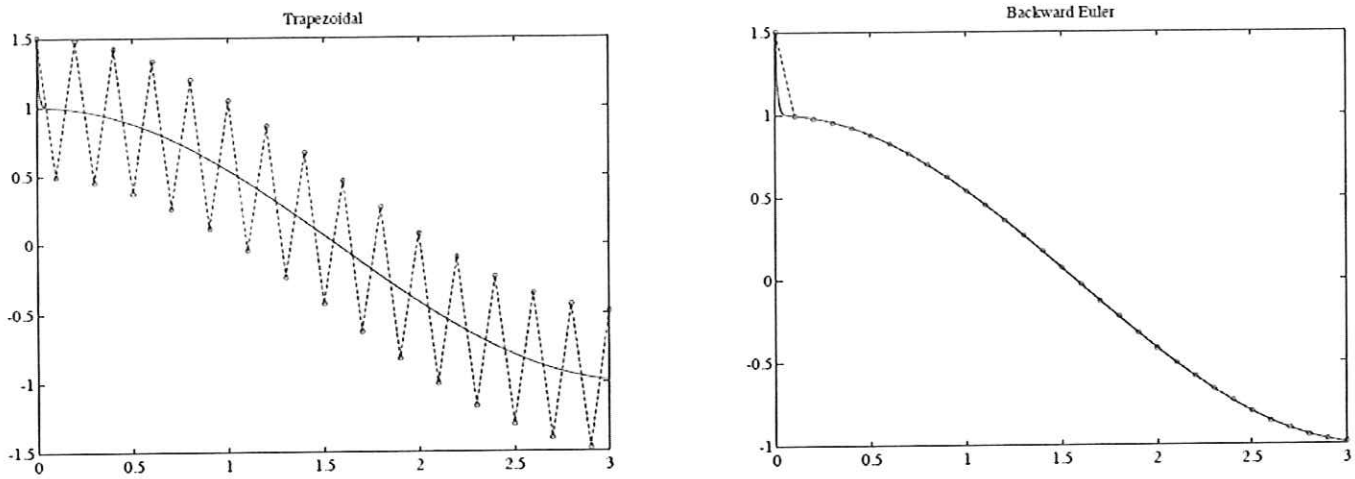
(i) Trap: the growth factor r is given by

$$r = \frac{1 - \frac{\lambda \Delta t}{2}}{1 + \frac{\lambda \Delta t}{2}} = -0.99996 \approx -1.0$$

⇒ The numerical soln is zig-zagging by crossing the $u(t) = \text{cost}$ curve depending on n = even & odd, with very slow rate of decay to $u(t) = \text{cost}$,

(ii) BE: $r = \frac{1}{1 + \lambda \Delta t} \approx -10^{-6}$

⇒ The soln converges to $u(t) = \text{cost}$ very rapidly.



(a) (b)

Figure 9.4: Comparison of (a) Trapezoidal method and (b) Backward Euler on a stiff problem with an initial transient (Case 2 of Example 9.3).

(Ex) **BDF methods** (Backward Difference Formula methods)

→ One class of very popular & effective methods for stiff problems.

→ the method has the form of

$$\boxed{\sum_{j=0}^{r} \alpha_j U^{n+j} = \Delta t\, \beta_r f(U^{n+r})}$$

popular choices are:

$\boxed{r=1}$ $U^{n+1} = U^n + \Delta t\, f(U^{n+1})$ (BE)

$\boxed{r=2}$ $3U^{n+2} - 4U^{n+1} + U^n = 2\Delta t\, f(U^{n+2})$

$\boxed{r=3}$ $11U^{n+3} - 18U^{n+2} + 9U^{n+1} - 2U^n = 6\Delta t\, f(U^{n+3})$

$\boxed{r=4}$ $25U^{n+4} - 48U^{n+3} + 36U^{n+2} - 16U^{n+1} + 3U^n = 12\Delta t\, f(U^{n+4})$

$\boxed{r=5}$ $137U^{n+5} - 300U^{n+4} + 300U^{n+3} - 200U^{n+2} + 75U^{n+1} - 12U^n = 60\Delta t\, f(U^{n+5})$

$\boxed{r=6}$ $147U^{n+6} - 360U^{n+5} + 450U^{n+4} - 400U^{n+3} + 225U^{n+2} - 72U^{n+1} + 10U^n = 60\Delta t\, f(U^{n+6})$

Rmk. BDF with $r>6$ are not zero-stable and hence they are not suitable for computation.