CS 775: Advanced Computer Graphics

Lecture 8 : Stiff Systems and Implicit Solvers

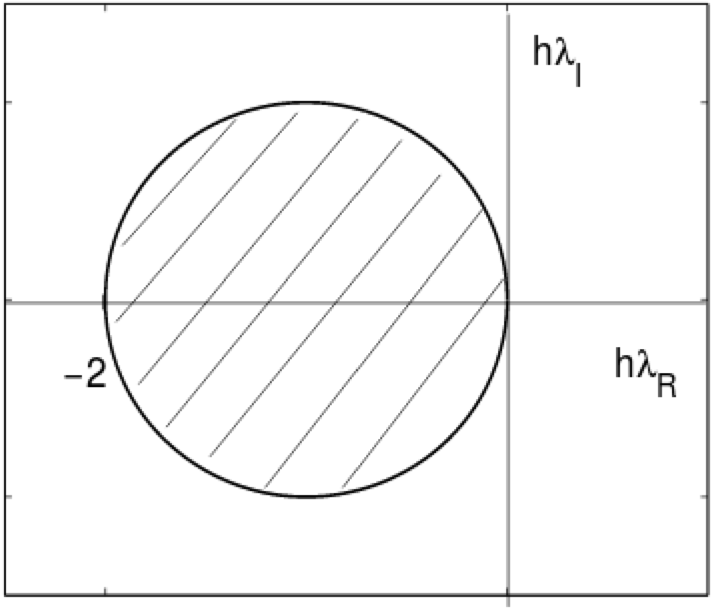
Explicit Euler - Stability

- Let the derivative function be linear $\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x}$
- Consider the x parallel to the largest eigenvector of A $\frac{d\mathbf{x}}{dt} = \lambda\mathbf{x}$
- Euler update $\dot{\mathbf{x}}_{n+1} = \mathbf{x}_n + h\lambda\mathbf{x}_n$
- Solution $\dot{\mathbf{x}}_n = (1+h\lambda)^n \mathbf{x}_0$
- Stable when

$$|1+h\lambda| \le 1 \Rightarrow -1 < 1+h\lambda < 1 \Rightarrow h < \frac{2}{|\lambda|}$$

- Real, negative eigenvalues
- Imaginary eigenvalues

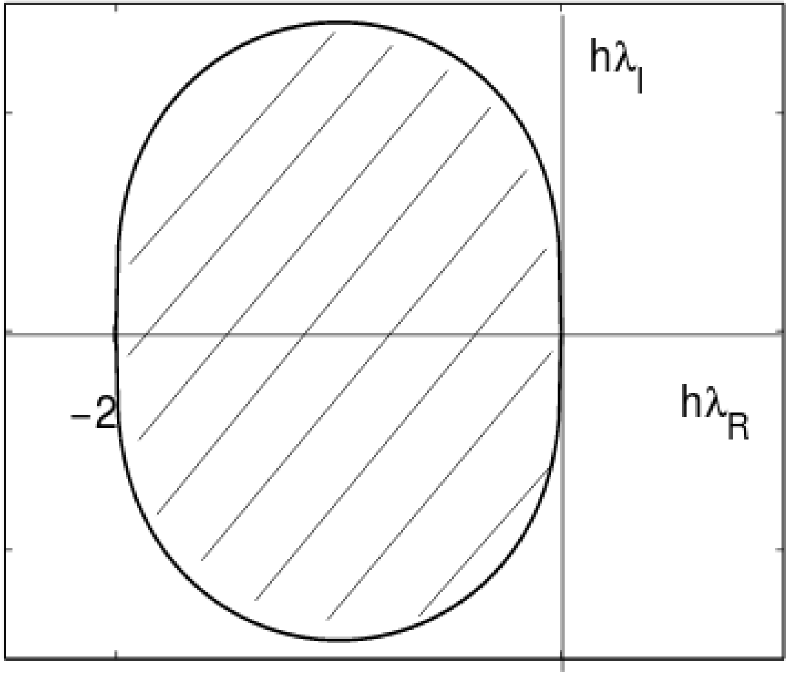
Stability region for simple Euler method



Midpoint Method - Stability

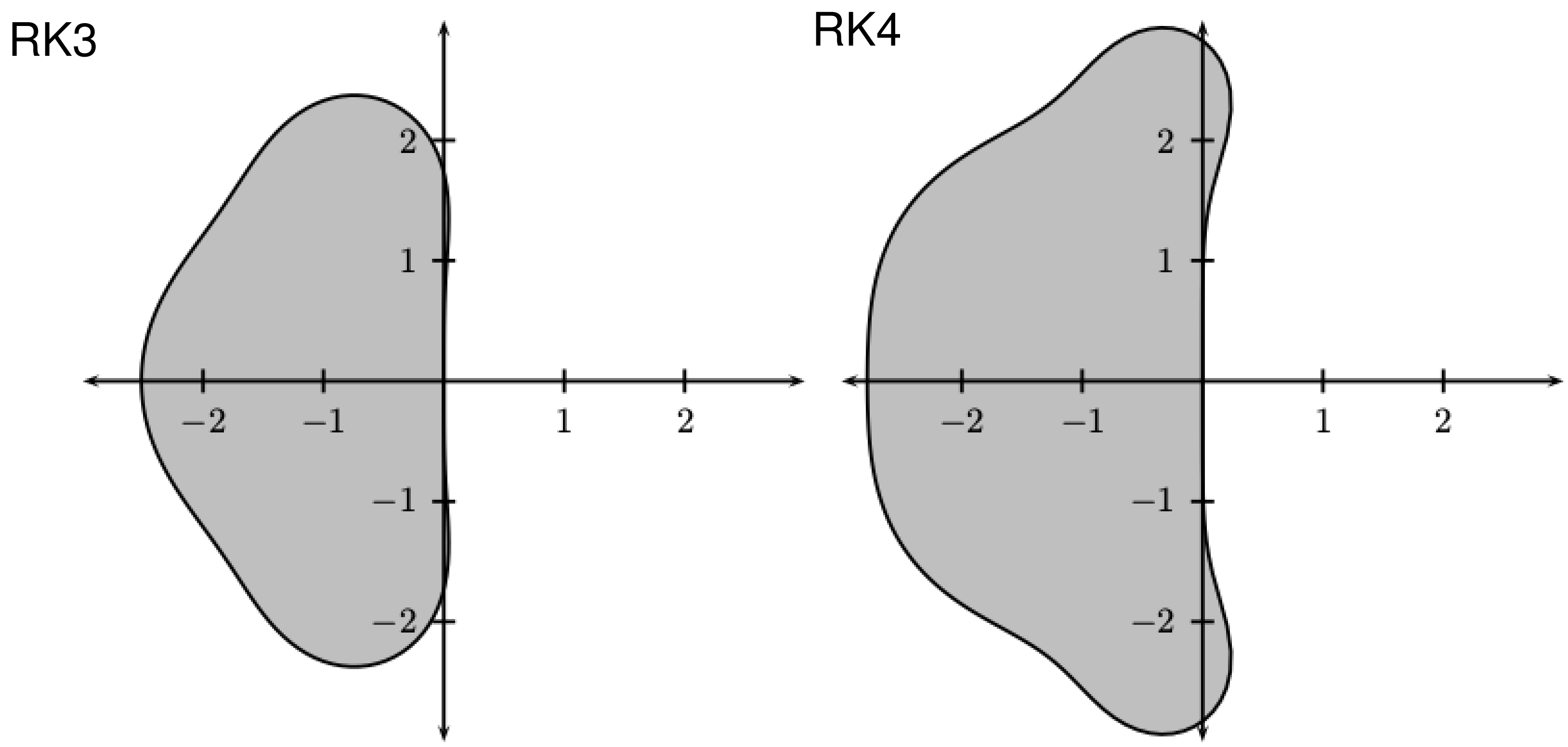
- Midpoint update
$$\boldsymbol{x}_{n+1}=\boldsymbol{x}_n+h\lambda\,\boldsymbol{x}_{n+\frac{1}{2}}$$
$$\boldsymbol{x}_{n+1}=\boldsymbol{x}_n+h\lambda\left(\boldsymbol{x}_n+\frac{1}{2}h\lambda\,\boldsymbol{x}_n\right)$$
$$\boldsymbol{x}_{n+1}=\left(1+h\lambda+\frac{1}{2}(h\lambda)^2\right)\boldsymbol{x}_n$$
- Let $h\lambda=a+ib$
- Then stable if

$$\left(1+a+\frac{1}{2}\left(a^2-b^2\right)\right)^2+\left(b+ab\right)^2\leq 1$$



Runge-Kutta Stability

- q-stage, p^{th}-order Runge-Kutta evaluates the derivative function q times in each iteration and its approximation of the next state is correct within $O(h^{p+1})$.
- RK4 is 4-stage, 4^{th}-order



Stiff Systems

- If we want to move a particle such that is always stays on the x-axis (particle on a wire), then we can model it as

$$\boldsymbol{X}(t)=\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} \qquad \dot{\boldsymbol{X}}(t)=\frac{d}{dt}\begin{bmatrix} x(t) \\ y(t) \end{bmatrix}=\begin{bmatrix} -x(t) \\ -ky(t) \end{bmatrix}$$

- Use Euler's method to update

$$\boldsymbol{X}_{new}=\boldsymbol{X}_0+h\,\dot{\boldsymbol{X}}(t_0)=\begin{bmatrix} x(t) \\ y(t) \end{bmatrix}+h\begin{bmatrix} -x(t) \\ -ky(t) \end{bmatrix}$$

$$\boldsymbol{X}_{new}=\begin{bmatrix} (1-h)\,x_0 \\ (1-hk)\,y_0 \end{bmatrix}$$

- If |1-hk| > 1 then Euler's method will not converge.
- For very large stiffness constant k, step size h is very small

Implicit Euler

Explicit Euler $\quad \boldsymbol{X}_{new} = \boldsymbol{X}_0 + h f(\boldsymbol{X}_0)$

Implicit Euler $\quad \boldsymbol{X}_{new} = \boldsymbol{X}_0 + h f(\boldsymbol{X}_{new})$

Solving for $\boldsymbol{X}_{new}$ such that f, at time $t_0 + h$ points back at $\boldsymbol{X}_0$

$$f(\boldsymbol{X}_{new}) = f(\boldsymbol{X}_0) + \Delta\boldsymbol{X} f'(\boldsymbol{X}_0) \qquad \text{where} \quad \Delta\boldsymbol{X} = \boldsymbol{X}_{new} - \boldsymbol{X}_0$$

$$\Rightarrow \boldsymbol{X}_{new} = \boldsymbol{X}_0 + h f(\boldsymbol{X}_0) + h\Delta\boldsymbol{X} f'(\boldsymbol{X}_0)$$

and $\quad \Delta\boldsymbol{X} = \left(\frac{1}{h}\boldsymbol{I} - f'(\boldsymbol{X}_0)\right)^{-1} f(\boldsymbol{X}_0)$

$$f(\boldsymbol{X}, t) = \dot{\boldsymbol{X}}(t)$$

$$f(\boldsymbol{X}, t)' = \frac{\partial f}{\partial \boldsymbol{X}}$$

Implicit Euler

Solving our stiff system with implicit Euler

$$\Delta \boldsymbol{X} = \left(\frac{1}{h} \boldsymbol{I} - f'(\boldsymbol{X}_0) \right)^{-1} f(\boldsymbol{X}_0)$$

$$f(\boldsymbol{X}(t)) = \begin{bmatrix} -x(t) \\ -ky(t) \end{bmatrix}$$

$$f(\boldsymbol{X}(t))' = \begin{bmatrix} -1 & 0 \\ 0 & -k \end{bmatrix}$$

$$\Delta \boldsymbol{X} = \begin{bmatrix} \frac{1+h}{h} & 0 \\ 0 & \frac{1+kh}{h} \end{bmatrix}^{-1} \begin{bmatrix} -x_0 \\ -ky_0 \end{bmatrix} = -\begin{bmatrix} \frac{h}{h+1} x_0 \\ \frac{h}{1+kh} ky_0 \end{bmatrix}$$

Implicit Euler – Step Size

Largest step size the implicit solver can take for our problem

$$\lim_{h \to \infty} \Delta \boldsymbol{X} = \lim_{h \to \infty} -\begin{bmatrix} \dfrac{h}{h+1} x_0 \\ \dfrac{h}{1+kh} k y_0 \end{bmatrix}$$

$$= -\begin{bmatrix} x_0 \\ \dfrac{1}{k} k y_0 \end{bmatrix} = -\begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

$$\text{i.e., } \boldsymbol{X}_{new} = \boldsymbol{X}_0 + \left(-\boldsymbol{X}_0\right) = \boldsymbol{0}$$

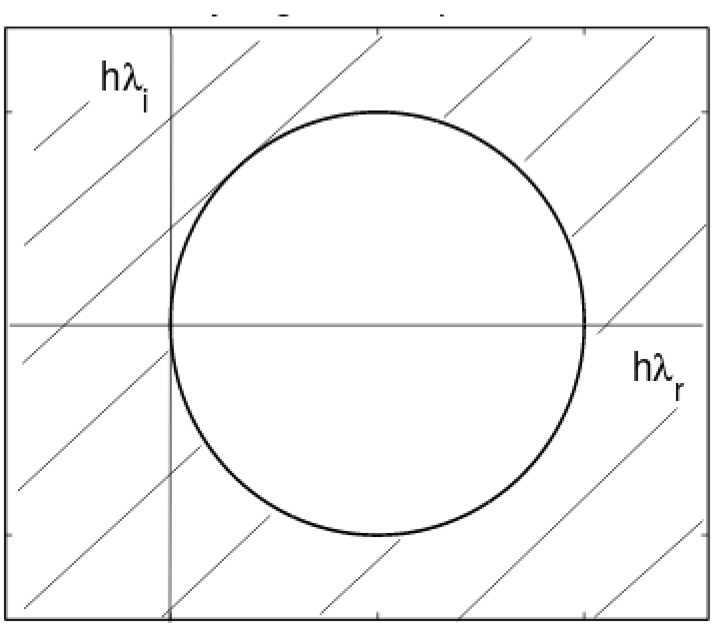
Implicit Euler - Stability

- Implicit update $$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + h\lambda\,\boldsymbol{x}_{n+1}$$

$$\Rightarrow \boldsymbol{x}_n = \boldsymbol{x}_0 \left(\frac{1}{1 - h\lambda} \right)^n$$

- Then the solver is stable if

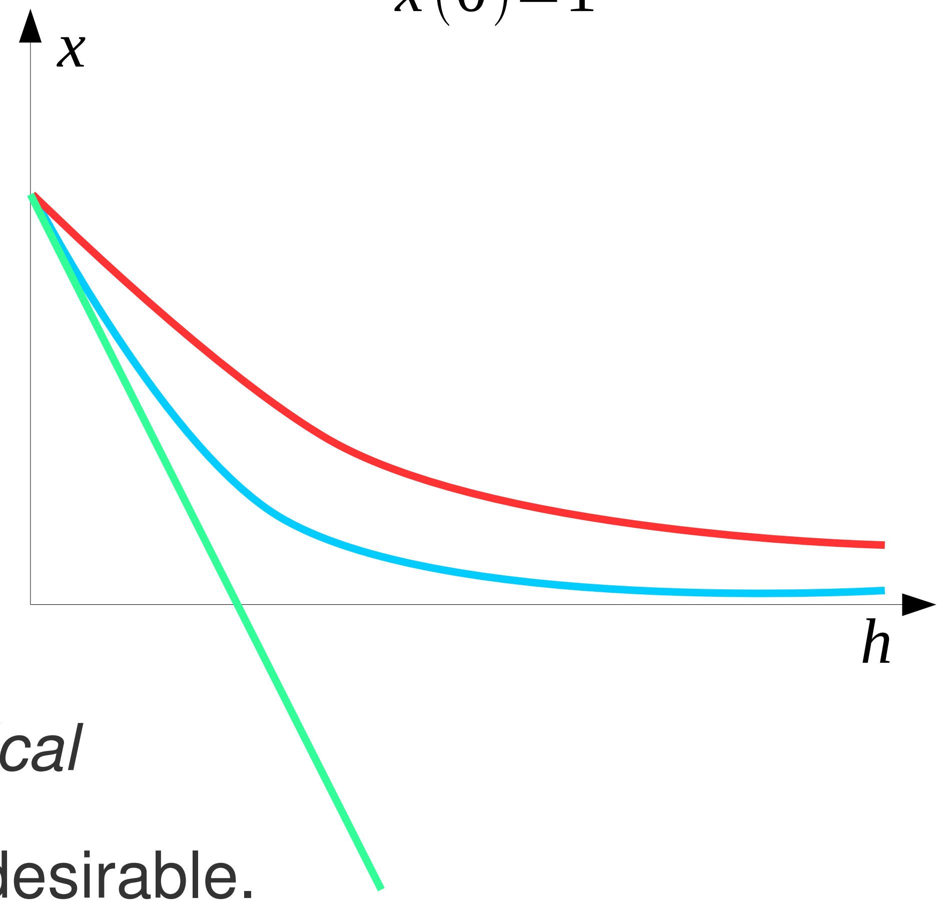
$$\frac{1}{|1 - h\lambda|} \leq 1 \quad \text{or} \quad |1 - h\lambda| > 1$$



Implicit Euler

$$\dot{x}(h) = -k\,x(h)$$
$$x(0) = 1$$

- Correct Soln $\quad x(h) = e^{-hk}$
- Explicit Euler $\quad x(h) = 1 - hk$
- Implicit Euler $\quad x(h) = \dfrac{1}{1+hk}$



- Implicit solver causes *numerical* damping. This is not always desirable.
- Stability over large step size does not imply accuracy.

Trapezoidal Rule

- Half explicit Euler, half implicit Euler

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + h\left(\frac{1}{2} f\left(\boldsymbol{x}_n\right) + \frac{1}{2} f\left(\boldsymbol{x}_{n+1}\right)\right)$$

- Where is this stable?