

Mat 3440 Simple Runge–Kutta methods

Håkon Hoel

Spring 2026

Forward Euler

$$\bar{X}_{n+1} = \bar{X}_n + F(t_n, \bar{X}_n)h$$

Matlab:

```
function [t,X] = forward_euler(F,tspan,X0,N)
    % solves ode  $X' = F(t,X)$ 
    % over  $tspan = [a,b]$  with  $X(a) = X0$ .
    %  $h = (b-a)/N = \text{diff}(tspan)/N$ 

    h = diff(tspan)/N;
    t = (0:N)*h + tspan(1);
    X = zeros(length(X0),N+1);
    X(:,1) = X0;
    for n = 1:N
        X(:, n+1) = X(:,n) + F(t(n),X(:,n))*h;
    end
end
```

end

Implicit Euler

$$\bar{X}_{n+1} = \bar{X}_n + F(t_{n+1}, \bar{X}_{n+1})h$$

Nonlinear equation to solve in X :

$$G(X) = X - \bar{X}_n - F(t_{n+1}, X)h$$

Matlab:

```
function [t, X] = implicit_euler(F, tspan, X0, N)
    h = diff(tspan)/N; t = (0:N)*h + tspan(1);
    X = zeros(length(X0),N+1); X(:,1) = X0;
    options = optimoptions('fsolve', 'Display', 'off');

    for n = 1:N
        X_curr = X(:,n);
        G = @(X_next) X_next - X_curr - h * F(t(n+1), X_next);
        X(:,n+1) = fsolve(G, X_curr, options);
    end
end
```

Crank–Nicolson

$$\bar{X}_{n+1} = \bar{X}_n + \frac{F(t_n, \bar{X}_n) + F(t_{n+1}, \bar{X}_{n+1})}{2} h$$

Nonlinear equation to solve in X :

$$G(X) = X - \bar{X}_n - \frac{F(t_n, \bar{X}_n) + F(t_{n+1}, X)}{2} h$$

Matlab:

```
function [t, X] = crank_nicolson(F, tspan, X0, N)
    h = diff(tspan)/N; t = (0:N)*h + tspan(1);
    X = zeros(length(X0),N+1); X(:,1) = X0;
    options = optimoptions('fsolve', 'Display', 'off');

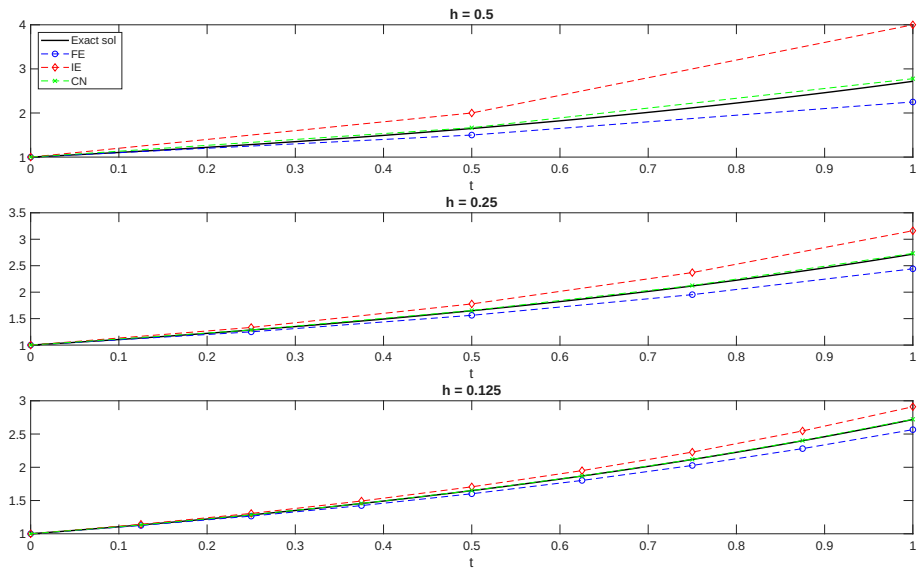
    for n = 1:N
        X_curr = X(:,n);
        G = @(X_next) X_next - X_curr - ...
            (F(t(n),X_curr) + F(t(n+1), X_next))*h/2;
        X(:,n+1) = fsolve(G, X_curr, options);
    end
end
```

Numerical example

$$\begin{aligned}x'(t) &= x(t) & t \in [0, 1] \\ x(0) &= 1\end{aligned}$$

Main Matlab lines:

```
tspan = [0,1];  
F = @(t,X) X; X0 = 1;  
N = 2.^(1:6);  
for i=1:length(N)  
    [t,X_FE] = forward_euler(F,tspan,X0,N(i));  
    [t,X_IE] = implicit_euler(F,tspan,X0,N(i));  
    [t,X_CN] = crank_nicolson(F,tspan, X0, N(i));  
  
    % plot and compute errors  
end
```



Approximation error and order

Recall that for small $h > 0$, it holds that

$$e_{FE}(h) := |\bar{X}_{FE}(T) - X(T)| \leq Ch$$

$$e_{IE}(h) := |\bar{X}_{IE}(T) - X(T)| \leq Ch$$

$$e_{CN}(h) := |\bar{X}_{CN}(T) - X(T)| \leq Ch^2$$

This is corroborated by the experiment, where we observe that

$$e_{FE}(h/2) \approx \frac{e_{FE}(h)}{2}, \quad e_{IE}(h/2) \approx \frac{e_{IE}(h)}{2}, \quad e_{CN}(h/2) \approx \frac{e_{CN}(h)}{4}.$$

h	$e_{FE}(h)$	$e_{IE}(h)$	$e_{CN}(h)$
5.0000×10^{-1}	4.6828×10^{-1}	1.2817×10^0	5.9496×10^{-2}
2.5000×10^{-1}	2.7688×10^{-1}	4.4221×10^{-1}	1.4330×10^{-2}
1.2500×10^{-1}	1.5250×10^{-1}	1.9200×10^{-1}	3.5501×10^{-3}
6.2500×10^{-2}	8.0353×10^{-2}	9.0122×10^{-2}	8.8552×10^{-4}
3.1250×10^{-2}	4.1292×10^{-2}	4.3727×10^{-2}	2.2126×10^{-4}
1.5625×10^{-2}	2.0937×10^{-2}	2.1545×10^{-2}	5.5306×10^{-5}

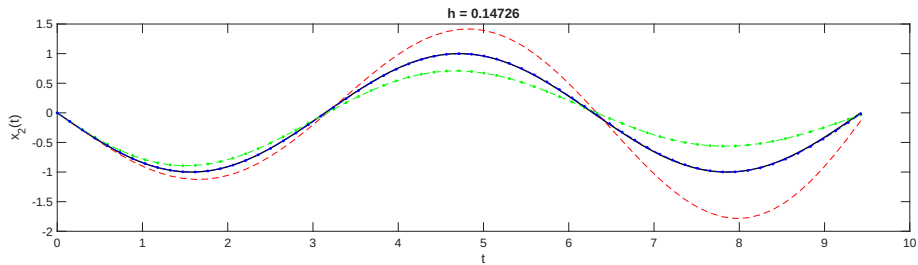
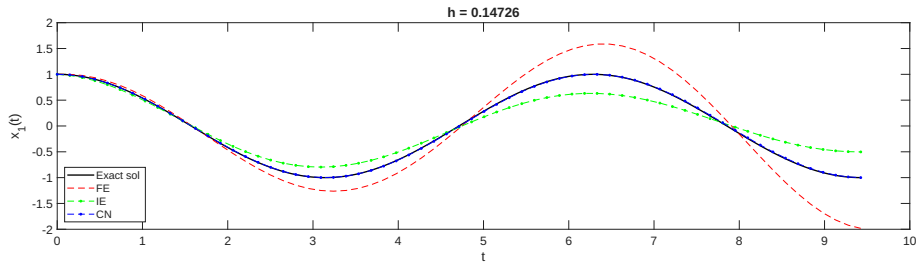
Numerical example – harmonic oscillator

$$X'(t) = AX \quad \text{with} \quad A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \text{for} \quad t \in [0, 3\pi]$$

$$x(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Main Matlab lines:

```
tspan = [0,3*pi];  
A = [0 1; -1 0];  
F = @(t,X) A*X; X0 = [1; 0];  
  
N = 2.^(5:10);  
for i=1:length(N)  
    [t,X_FE] = forward_euler(F,tspan,X0,N(i));  
    [t,X_IE] = implicit_euler(F,tspan,X0,N(i));  
    [t,X_CN] = crank_nicolson(F,tspan, X0, N(i));  
    % plot and compute errors  
end
```



Approximation error and order

Recall that for small $h > 0$, it holds that

$$e_{FE}(h) := |\bar{X}_{FE}(T) - X(T)| \leq Ch$$

$$e_{IE}(h) := |\bar{X}_{IE}(T) - X(T)| \leq Ch$$

$$e_{CN}(h) := |\bar{X}_{CN}(T) - X(T)| \leq Ch^2$$

Experiment again supports theoretical convergence orders

$$e_{FE}(h/2) \approx \frac{e_{FE}(h)}{2}, \quad e_{IE}(h/2) \approx \frac{e_{IE}(h)}{2}, \quad e_{CN}(h/2) \approx \frac{e_{CN}(h)}{4}.$$

h	$e_{FE}(h)$	$e_{IE}(h)$	$e_{CN}(h)$
2.9452e-01	2.8298e+00	7.4768e-01	6.7243e-02
1.4726e-01	9.9137e-01	4.9897e-01	1.6977e-02
7.3631e-02	4.1395e-01	2.9286e-01	4.2546e-03
3.6816e-02	1.8936e-01	1.5922e-01	1.0643e-03
1.8408e-02	9.0609e-02	8.3082e-02	2.6612e-04
9.2039e-03	4.4326e-02	4.2444e-02	6.6531e-05