Numerical Solutions to ODEs
Part I

CMPT 419/983
16/09/2019
Numerical Solutions of ODEs

• In general, $\dot{x} = f(x, u)$ does not have a closed-form solution
  • Instead, we usually compute numerical approximations to simulate system behaviour
  • Done through discretization: $t^k = kh, \ u^k := u(t^k)$
    • $h$ represents size of time step
  • Goal: compute $y^k \approx x(t^k)$

• Key considerations
  • Consistency: Does the approximation satisfy the ODE as $h \to 0$?
  • Accuracy: How fast does the solution converge?
  • Stability: Do approximation error remain bounded over time?
  • Convergence: Does the approximate solution converge to the true solution as $h \to 0$?
Euler Methods

- ODE: $\dot{x} = f(x,u), \ x(0) = x_0$
  - Discretization: $t^k = kh, \ u^k := u(t^k)$
  - Want: Approximate solution: $y^k \approx x(kh)$

- Forward Euler
  - Most naïve method; explicit method
    \[
    \frac{y^{k+1} - y^k}{h} = f(y^k, u^k) \Rightarrow y^{k+1} = y^k + hf(y^k, u^k)
    \]

- Backward Euler
  - Most basic implicit method
    \[
    \frac{y^{k+1} - y^k}{h} = f(y^{k+1}, u^k) \Rightarrow \text{solve for } y^{k+1} \text{ implicitly}
    \]
Visualizing Euler Methods

- Main consideration: what slope to use?
  - Forward Euler: slope at beginning
    \[ y^{k+1} = y^k + hf(y^k, u^k) \]
  - Backward Euler: slope at the end
    \[ y^{k+1} = y^k + hf(y^{k+1}, u^k) \]
Example

• \( \dot{x} = ax, \ x(0) = x_0 \)
  - Analytic solution: \( x(t) = x_0 e^{at} \)

• Forward Euler
  - \( y^{k+1} = y^k + hf(y^k, u^k) \)
  - \( y^{k+1} = y^k + ha y^k \)
  - \( y^{k+1} = (1 + ha)y^k \)
Example

%% Problem setup
x0 = 1;
a = -1;
h = 0.1;
T = 5;
tau = 0:h:T;

%% Exact solution
x_exact = @(t) exp(a*t);

figure
plot(tau, x_exact(tau), 'b.-')
Example

%% Forward Euler
f = @(x) a*x;
y_approx = -ones(size(tau));
y_approx(1) = x0;

% Initialize vector
for i = 2:length(tau)
    y_approx(i) = y_approx(i-1)*(1+h*a);
end

hold on
plot(tau, y_approx, 'r.-')
title(sprintf('a = %.1f', a))
legend('Exact', 'Approx.')
Example

• \( \dot{x} = ax, \ x(0) = x_0 \)
  • Analytic solution: \( x(t) = x_0 e^{at} \)

• Backward Euler
  • \( y^{k+1} = y^k + hf(y^{k+1}) \)
  • \( y^{k+1} = y^k + hax^{k+1} \)
  • \( y^{k+1} - hay^{k+1} = y^k \)
  • \( (1 - ha)y^{k+1} = y^k \)
  • \( y^{k+1} = \frac{y^k}{1 - ha} \)
Numerical Consistency: Forward Euler

• **Consistency**: ODE is satisfied as $h \to 0$
  
  • Forward Euler: $y^{k+1} = y^k + hf(y^k, u^k)$
  
  $\frac{y^{k+1} - y^k}{h} = f(y^k, u^k)$

• **Local truncation error**: Consistency requires $\frac{\|e^k\|}{h} \to 0$ as $h \to 0$
  
  • $\|e^k\|$: Error induced during one step, assuming perfect previous information
  
  • Forward Euler approximate solution:
  $$y^{k+1} = x(t^k) + hf(x(t^k), u^k)$$

• True solution:
  $$x(t^{k+1}) = x(t^k + h) = x(t^k) + h \frac{dx}{dt}(t^k) + \frac{h^2}{2} \frac{d^2x}{dt^2}(t^k) + O(h^3)$$
  
  $$= x(t^k) + hf(x(t^k), u^k) + \frac{h^2}{2} \frac{d^2x}{dt^2}(t^k) + O(h^3)$$
Numerical Consistency: Forward Euler

- Local truncation error: 
  \[ e^k = x(t^{k+1}) - y^{k+1} \]
  \[ = x(t^k) + hf(x(t^k), u^k) + \frac{h^2}{2} \frac{d^2x}{dx^2}(t^k) + O(h^3) - \left(x(t^k) + hf(x(t^k), u^k)\right) \]
  \[ = \frac{h^2}{2} \frac{d^2x}{dx^2}(t^k) + O(h^3) \]
  \[ = O(h^2) \]

- Consistency requires \( \frac{\|e^k\|}{h} \to 0 \) as \( h \to 0 \)
  \[ \frac{\|e^k\|}{h} = \left| \frac{h^2}{2} \frac{d^2x}{dx^2}(t^k) + O(h^3) \right| = \left| \frac{hd^2x}{2} \frac{d^2x}{dx^2}(t^k) + O(h^2) \right| \to 0 \]

- If \( \frac{\|e^k\|}{h} = O(h^p) \), then the numerical method is “order p”.
  - Forward Euler is an order 1 method, or first order method
Numerical Consistency

• More generally: $y^{k+1} = \sum_{n=k_1}^{k} \alpha_i y^i + h \sum_{n=k_2}^{k} \beta_i f(y^i, u^i)$

• Truncation error:
  
  $e^k := x(t^{k+1}) - \sum_{n=k_1}^{k} \alpha_n x(nh) - h \sum_{n=k_2}^{k} \beta_i f(x(nh), u^i)$

• Consistency requires $\frac{\|e^k\|}{h} \to 0$ as $h \to 0$

• If $\frac{\|e^k\|}{h} = O(h^p)$, then the numerical method is “order $p$”.
Numerical Stability: Forward Euler

• \( y^{k+1} = y^k + hf(y^k, u^k) \)
  • A map from \( y^k \) to \( y^{k+1} \)
  • Stability means \( y^k \) does not “blow up” when the true solution \( x(t^k) \) is bounded
  • Usually, stability requires that the time step \( h \) cannot be too large

• Example: \( \dot{x} = ax, a < 0 \)
  • \( y^{k+1} = (1 + ah)y^k \)
  • Stability requires \( |1 + ah| \leq 1 \iff -ah \leq 2 \)
  • For \( a = -10 \), we have \( |1 - 10h| \leq 1 \iff h \leq 0.2 \)
Numerical Stability: Backward Euler

- $y^{k+1} = y^k + hf(y^{k+1}, u^k)$
  - A map from $y^k$ to $y^{k+1}$
  - Stability means $y^k$ does not “blow up” when the true solution $x(t^k)$ is bounded
  - Usually, stability requires that the time step $h$ cannot be too large

- Example: $\dot{x} = ax, a < 0$
  - $y^{k+1} = \frac{y^k}{1-ah}$
  - Stability requires $\left|\frac{1}{1-ah}\right| \leq 1$
  - No restrictions on $h$, for any $a$!
Numerical Stability

• Example: $\dot{x} = ax$ with forward Euler
  • If $a = -10$, $h \leq 0.2$ is required for stability

• Example 2: $\dot{x} = ax$ with backward Euler
  • No restrictions on $h$, for any $a$
Numerical Stability

• More generally: \( y^{k+1} = \sum_{n=k_1}^{k} \alpha_i y^i + h \sum_{n=k_2}^{k} \beta_i f(y^i, u^i) \)
  
  • Desired property: the approximation \( y^k \) does not “blow up” when the true solution \( x(t^k) \) is bounded
  
  • Usually, this means time step \( h \) cannot be too large

• Specifically, one typically considers \( \dot{x} = ax, a < 0 \).
  
  • A stable numerical approximation to \( \dot{x} = ax, a < 0 \) has the property that \( y^k \to 0 \)
Numerical Convergence

• **Convergence:** \( \max_k \| x(t^k) - y^k \| \to 0 \) as \( h \to 0 \)
  
  • Maximum error goes to zero as time step goes to 0

• Dahlquist Equivalence Theorem
  
  • Consistency + stability \( \iff \) convergence

• Convergence rate
  
  • For order \( p \) methods: \( \max_k \| x(t^k) - y^k \| = O(h^p) \)
  
  • Forward and backward Euler: \( p = 1 \)
    
    • It takes \( \frac{t-t_0}{h} \), or \( O(\frac{1}{h}) \) steps, each incurring \( O(h^2) \) error
    
    • If we half \( h \), then the error also halves
Numerical Convergence

- Visualize convergence rate with Max error vs. $h$ plot

- Forward and backward Euler are both 1st order
  - Half the size of $h$ leads to half the error

- Usually, log-log plots are used to show a wide range of errors and $h$
  - Order $p$ method has a slope of $p$ (approximately).
Stiff equations

• ODEs with components that have very fast rates of change
  • Usually requires very small step sizes for stability

• Example: $\dot{x}_1 = ax_1$ with forward Euler
  • Stability requires $|1 + ha| \leq 1$
  • For $a = -100$, we have $|1 - 100h| \leq 1 \Leftrightarrow h \leq 0.02$

• Small step size is required even if there are other slower changing components like $\dot{x}_2 = x_1 - x_2$
  • Implicit methods (eg. backward Euler) are useful here

$$\begin{align*}
\dot{x}_1 &= -100x_1 \\
\dot{x}_2 &= x_1 - x_2 \\
\dot{x} &= \begin{bmatrix} -100 & 0 \\ 1 & -1 \end{bmatrix} x
\end{align*}$$