Numerical methods for ODEs

- Forward Euler method (explicit Euler):
  \[ y_{k+1} = y_k + hf(t_k, y_k) \]

- Backward Euler method (implicit Euler):
  \[ y_{k+1} = y_k + hf(t_{k+1}, y_{k+1}) \]

- Trapezoidal method:
  \[ y_{k+1} = y_k + \frac{h}{2} (f(t_k, y_k) + f(t_{k+1}, y_{k+1})) \]
Numerical methods for ODEs

- Heun’s method:

\[
\begin{align*}
K_1 &= f(t_k, y_k) \\
K_2 &= f(t_k + h, y_k + hK_1) \\
y_{k+1} &= y_k + \frac{h}{2}(K_1 + K_2)
\end{align*}
\]

- Classical Runge-Kutta:

\[
\begin{align*}
K_1 &= f(t_k, y_k) \\
K_2 &= f(t_k + \frac{h}{2}, y_k + \frac{h}{2}K_1) \\
K_3 &= f(t_k + \frac{h}{2}, y_k + \frac{h}{2}K_2) \\
K_4 &= f(t_k + h, y_k + hK_3) \\
y_{k+1} &= y_k + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4)
\end{align*}
\]
Local truncation error

- It is the error made in one step of the numerical method

- At time $t_{k+1}$, the local truncation error is

  $$y(t_{k+1}) - y_{k+1} \quad \text{assuming} \quad y_k = y(t_k)$$

- The local truncation error of forward Euler method is

  $$\frac{1}{2} h^2 y''(c) = O(h^2)$$
Global truncation error

- At time $t_{k+1}$, the global truncation error is $y(t_{k+1}) - y_{k+1}$, assuming the initial condition is known, $y_0 = y(0)$

- It reflects not only the error at that time step, but also the errors at all previous steps

- It is of primary interest. But only the local truncation error is easily estimated and controlled
Global truncation error

It is not the simple sum of all the local truncation errors in the previous time steps
Local VS Global

Good news:

Under certain conditions, it can be shown that the global truncation error is $O(h^q)$ when the local truncation error is $O(h^{q+1})$. 
## Local VS Global

<table>
<thead>
<tr>
<th>Numerical Methods</th>
<th>Local Truncation Error</th>
<th>Global Truncation Error</th>
<th>Order of Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>F Euler</td>
<td>$O(h^2)$</td>
<td>$O(h^1)$</td>
<td>1</td>
</tr>
<tr>
<td>B Euler</td>
<td>$O(h^2)$</td>
<td>$O(h^1)$</td>
<td>1</td>
</tr>
<tr>
<td>Trapezoidal</td>
<td>$O(h^3)$</td>
<td>$O(h^2)$</td>
<td>2</td>
</tr>
<tr>
<td>Heun</td>
<td>$O(h^3)$</td>
<td>$O(h^2)$</td>
<td>2</td>
</tr>
<tr>
<td>RK4</td>
<td>$O(h^5)$</td>
<td>$O(h^4)$</td>
<td>4</td>
</tr>
</tbody>
</table>
Different methods

- Euler method: 1
- Heun method: 2
- Classical Runge–Kutta method: 4

Order of accuracy

Discretization error dominates
Rounding error dominates
For a given tolerance, what step size should I use?

![Graph showing relative error vs step size for different methods: Euler method, Heun method, and Classical Runge-Kutta.](image-url)
# Lab

Comparison (tolerance = 0.0001)

<table>
<thead>
<tr>
<th></th>
<th>$h$</th>
<th>time</th>
<th>Steps</th>
<th>time per step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Euler</td>
<td>0.0002</td>
<td>0.0341</td>
<td>51368</td>
<td>0.00000007</td>
</tr>
<tr>
<td>Heun</td>
<td>0.0457</td>
<td>0.0003</td>
<td>219</td>
<td>0.00000014</td>
</tr>
<tr>
<td>RK4</td>
<td>0.3961</td>
<td>0.0001</td>
<td>26</td>
<td>0.00000034</td>
</tr>
</tbody>
</table>

A single step of RK4 takes longer time, but the total time is shorter.
Two definitions

Let $\tau$ denote the local truncation error:

**Def 1:** The numerical method is consistent with the ODE if $\tau / h \rightarrow 0$ when $h \rightarrow 0$.

A consistent method is a correct approximation of the ODE.

**Def 2:** The numerical method is convergent if the global error $\rightarrow 0$ when $h \rightarrow 0$.

Consistent + Stable $\rightarrow$ Convergent
Lab: Stability

- Stability problems - a limit on the choice of the step size

\[
\begin{align*}
    y' &= -50y \\
    y(0) &= 1
\end{align*}
\]
Using Forward Euler method to solve the ODE

\[
\begin{cases} 
  y' = -\lambda y \\
  y(0) = 1 
\end{cases}
\]

The stability condition is

\[ h < \frac{2}{\lambda} \]

Why?
Lab

- Backward Euler method, $h=0.05$
Lab

Using Backward Euler method to solve the ODE

\[
\begin{align*}
y' &= -\lambda y \\ y(0) &= 1
\end{align*}
\]

It is unconditionally stable.

Why?
Lab

Stiff problem  Explicit method

Implicit method