# Ordinary Differential Equations
# Initial Value Problems

*The question of whether computers can think is just like*
*the question of whether submarines can swim*

*Edsger W. Dijkstra*

# Topics to Be Discussed

- **This unit requires the knowledge of some very basic ordinary equations.**
- **The following topics will be presented:**
  - **Euler's method**
  - **Heun's method**
  - **Predictor-Corrector methods**
  - **Various Runge-Kutta methods with an emphasis on the second order methods**

# What Is an Initial Value Problem?

- **A first order differential equation has a form like this:**

$$y' = f(x, y)$$

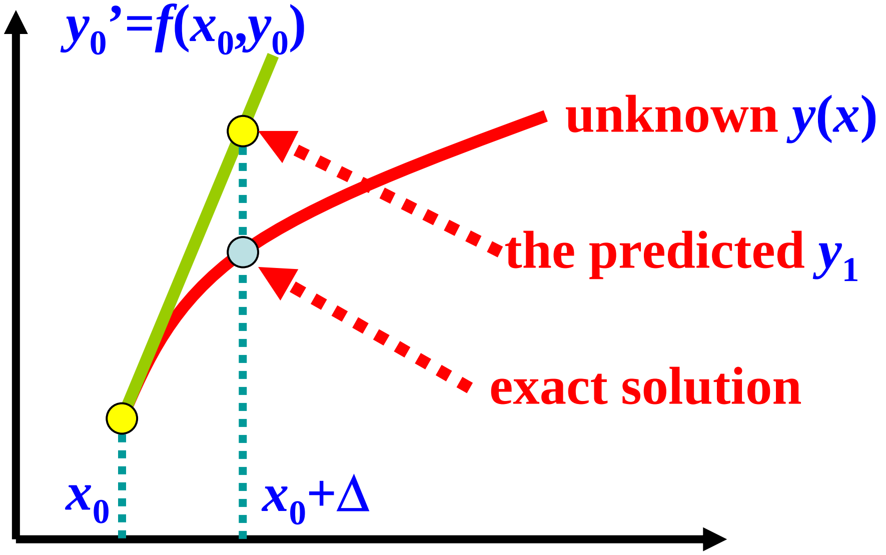
- **Here, $y$ is a function of $x$, and $y'$ is the derivative of $y$. Function $f(x,y)$ gives the relation of $x$ and $y$.**
- **However, solution to the above ODE is not unique, and an initial condition is required:**

$$y' = f(x, y) \quad \textbf{and} \quad y(x_0) = y_0$$

- **The problem is:** find $y$ with $y(x_0) = y_0$ step-by-step in an interval $[x_0, x_n]$ .

# Euler's Method: 1/6

- **The easiest method is perhaps Euler's.**
- **Since $y' = f(x,y)$ and $y_0 = y(x_0)$, we know the tangent slope of $y$ at $(x_0,y_0)$, *i.e.*, $y_0' = f(x_0,y_0)$.**
- **If $x_1 = x_0 + \Delta$ and $\Delta$ is small enough, we may** predict **$y(x+\Delta)$ using the tangent!**



# Euler's Method: 2/6

- How do we "predict" $y(x+\Delta)$?
- Using the forward difference method for differentiation, $y'(x_0)$ can be approximated as

$$y'(x_0) \approx \frac{y(x_0+\Delta) - y(x_0)}{\Delta}$$

- Replacing $y'$ yields:

$$\frac{y(x_0+\Delta) - y(x_0)}{\Delta} \approx f(x_0, y(x_0))$$

- Therefore, the prediction is:

$$y(x_0+\Delta) \approx y(x_0) + \Delta \times f(x_0, y(x_0))$$

# Euler’s Method: 3/6

- **Recall the following formula:**

$$y(x_0+\Delta)\approx y(x_0)+\Delta\times f(x_0,y(x_0))$$

- **We may start with an initial value $(x_0,y_0)$. Compute $x_1=x_0+\Delta$ and $y_1=y_0+\Delta\times f(x_0,y_0)$.**
- **Then, compute $(x_2,y_2)$ from $(x_1,y_1)$, etc.**
- **The algorithm below traces out the locus of $y$.**

```
! Initial values
! [a,b]
! $x_0$ = a
! $y_0$ = y(a) = y($x_0$)
! n: # of subdivisions
```

```
$\Delta$ = (b-a)/n
x = a
y = y(a)
DO i = 1, n
   y = y + $\Delta\times$f(x,y)
   x = x + $\Delta$
END
```

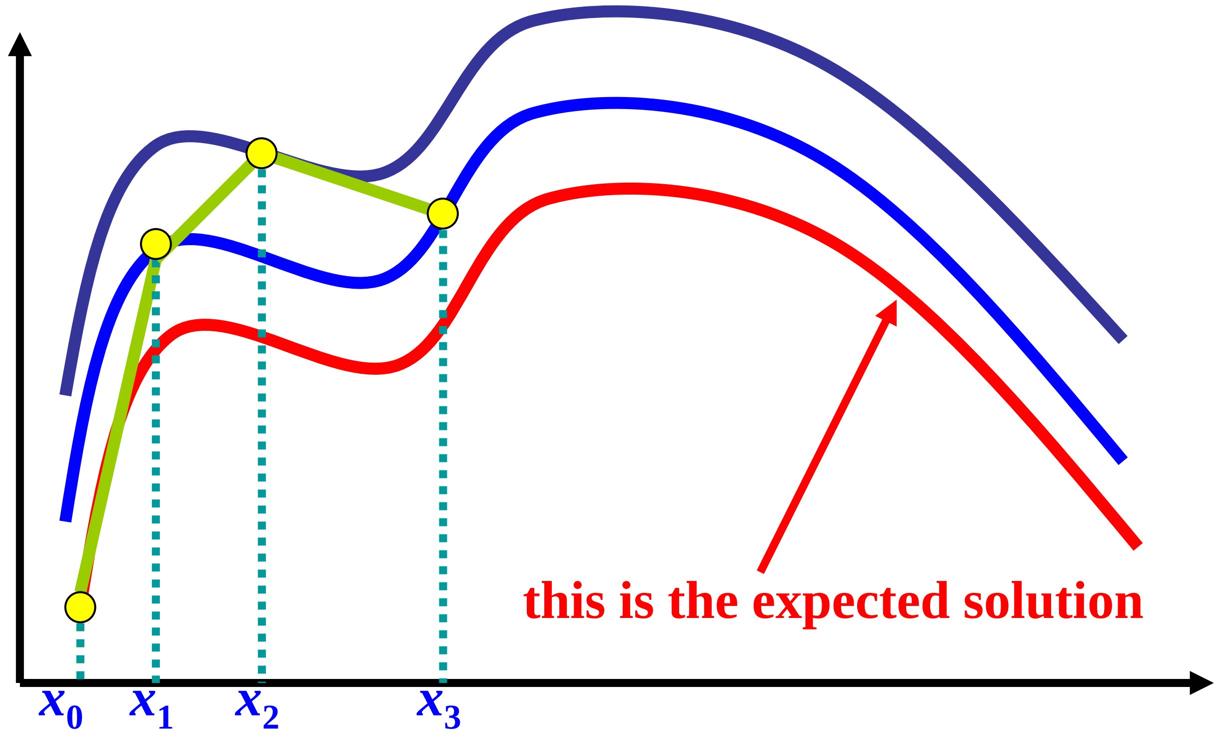
# Euler’s Method: 4/6

- **Consider $y' = x + y$ and $y(0)=2$ on $[0,1]$.**
- **Initial value is $x_0=0$ and $y_0=y(x_0)=2$.**
- **If $n = 4$, $\Delta=(1-0)/4=0.25$**
- **Predicted (next) $y = y + \Delta(x+y)$.**

| | $x$ | $y$ | $y'=f(x,y)$ | **Pred.** $y$ |
|---|---|---|---|---|
| **Init** | **0** | **2.0** | **2.0** | **$2.5=2+0.25\times(0+2)$** |
| **1** | **0.25** | **2.5** | **2.75** | **$3.1875=2.5+0.25\times2.75$** |
| **2** | **0.5** | **3.1875** | **3.6875** | **$4.109375=3.1875+0.25\times3.6875$** |
| **3** | **0.75** | **4.109375** | **4.859375** | **$5.3245187=4.109375+0.25\times4.859375$** |
| **4** | **1** | **5.3245187** | | |

# Euler’s Method: 5/6

- If Δ is sufficiently small, Euler’s method usually works fine.
- However, Euler’s method can produce large under- or over- predicted *y*.



this is the expected solution

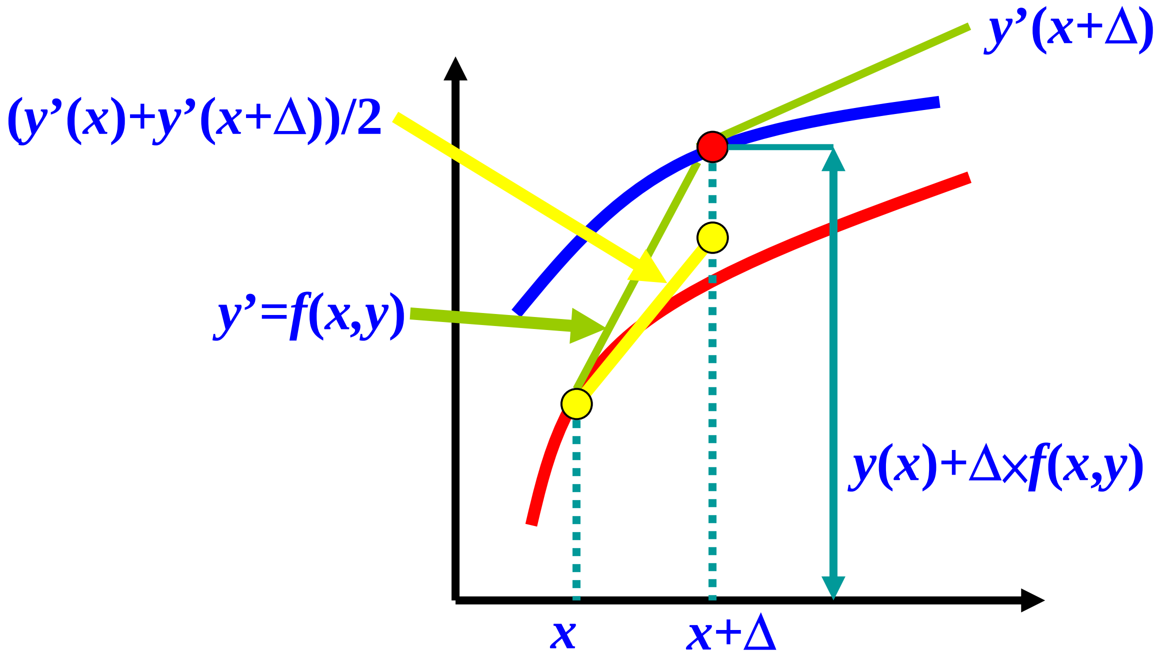
# Euler's Method: 6/6

- **Consider $y' = 1-y$ on [0,1] with $y(0)=0$.**
- **The solution is $y=1-e^{-x}$.**
- **The absolute error is increasing.**

|  | $x$ | $y$ | **exact $y$** | **\|error\|** | **% of exact** |
|---|---|---|---|---|---|
| **1** | **1/6** | **0.1666667** | **0.1535182** | **0.01314841** | **8.56** |
| **2** | **1/3** | **0.3055556** | **0.2834687** | **0.02208686** | **7.79** |
| **3** | **1/2** | **0.4212963** | **0.3934693** | **0.02782699** | **7.07** |
| **4** | **2/3** | **0.5177469** | **0.4865829** | **0.03116405** | **6.40** |
| **5** | **5/6** | **0.5981224** | **0.5654018** | **0.03272063** | **5.79** |
| **6** | **1** | **0.6651020** | **0.6321206** | **0.03298146** | **5.22** |

# Heun's Method: 1/4

- **Heun's method tries to reduce the over- or under- shot of Euler's method.**
- **Since $y(x+\Delta)$ may be an overshot, one may use the average of $y'(x)$ and $y'(x+\Delta)$ as a corrector.**



# Heun’s Method: 2/4

- **Heun’s method consists of two steps:**
  - Step 1: **Use $x$ and $\Delta$ to compute $y^P(x+\Delta)$ with Euler’s method. This is a *predictor*:**

$$y^P(x+\Delta) = y(x) + \Delta f(x, y)$$

  - Step 2: **Use the $(y'(x) + y'(x+\Delta))/2$ as the new derivative to correct over- or under-shot This is a *corrector*:**

$$y(x+\Delta) = y(x) + \Delta \frac{f(x, y) + f(x+\Delta, y^P(x+\Delta))}{2}$$

# Heun's Method: 3/4

- **Heun's method is just a little more complex than Euler's method.**

```
! initialization
! [a,b]: interval
! $x_0$ = a
! $y_0$ = y(a) = y($x_0$)
! n : intervals
```

```
$\Delta$ = (b-a)/n
x = a
y = y(a)
DO i = 1, n
   $y^{pred}$ = y + $\Delta$×f(x,y)
   y = y + $\Delta$×(f(x,y) + f(x+$\Delta$,$y^{pred}$))/2
   x = x + $\Delta$
END DO
```

# Heun's Method: 4/4

- Consider $y' = 1-y$ on [0,1] with $y(0)=0$, again.
- The solution is $y=1-e^{-x}$.
- The absolute error is still increasing, but significantly smaller.

|  | $x$ | $y$ | exact $y$ | \|error\| | % of exact |
|---|---|---|---|---|---|
| 1 | 1/6 | 0.15277778 | 0.15351826 | 0.00074048 | 0.48 |
| 2 | 1/3 | 0.28221452 | 0.28346872 | 0.00125420 | 0.44 |
| 3 | 1/2 | 0.39187619 | 0.39346933 | 0.00159314 | 0.40 |
| 4 | 2/3 | 0.48478401 | 0.48658288 | 0.00179887 | 0.37 |
| 5 | 5/6 | 0.56349754 | 0.56540179 | 0.00190425 | 0.34 |
| 6 | 1 | 0.63018543 | 0.63212055 | 0.00193512 | 0.31 |

# Predictor-Corrector Methods: 1/5

- **Heun’s method is a special and the simplest form of the predictor-corrector methods.**
- **If we look at the corrector step and remove the superscript *P* from the predictor, we have**

$$y(x+\Delta) = y(x) + \Delta\frac{f(x,y) + f(x+\Delta, y(x+\Delta))}{2}$$

- ***y*(*x*+Δ) is the unknown value that we want to compute, and it appears in *both* sides of the equation.**

# Predictor-Corrector Methods: 2/5

- **To make this observation clearer, replacing $y(x+\Delta)$ with a $z$ yields:**

$$z = y(x) + \Delta \frac{f(x,y) + f(x+\Delta, z)}{2}$$

- **Since $z$ is an unknown, the above is an equation of variable $z$, and the desired $z$ is a root!**
- **If such a $z$ can be found, say $z^*$, the equation is perfectly balanced as follows:**

$$z^* = y(x) + \Delta \frac{f(x,y) + f(x+\Delta, z^*)}{2}$$

- **Therefore, this predictor-corrector method reduces to root finding.**

# Predictor-Corrector Methods: 3/5

- **How to solve the following equation in *z*?**

$$z = y(x) + \Delta \frac{f(x,y) + f(x+\Delta, z)}{2}$$

- **Since it is not easy to find the derivative with respect to *z*, Newton's method is out.**
- **However, it is in a perfect form of fixed-point iteration (*i.e.*, $z=g(z)$).**
- **We may use $y^P$ as an initial value, followed by a number of iterations of fixed-point iteration to find a better corrector $y^C$.**
- **Note that fixed-point iteration method does not always converge.**

# Predictor-Corrector Methods: 4/5

- **The following is the iterative computation for a better corrector with the fixed-point iteration.**
- **`MAX` is the maximum number of iterations, and $\varepsilon$ is a tolerance value.**
- There are more powerful predictor-corrector methods.

```
$y^{pred} = y + \Delta \times f(x,y)$
DO k = 1, MAX
   $y^{correct} = y + \Delta \times (f(x,y) + f(x+\Delta, y^{pred}))/2$ ! Fixed-point
   IF ($|(y^{correct} - y^{pred})/y^{correct}| < \varepsilon$) EXIT
   $y^{pred} = y^{correct}$
END DO
$y = y^{correct}$
```

# Predictor-Corrector Methods: 5/5

- **Consider $y' = 1-y$ on [0,1] with $y(0)=0$, again.**
- **The solution is $y=1-e^{-x}$.**
- **The absolute error is still increasing, but better than Heun's method.**

**$\varepsilon = 0.0005$**

| | $x$ | $y$ | exact $y$ | \|error\| | % of exact | Iters |
|---|---|---|---|---|---|---|
| 1 | 1/6 | 0.15384677 | 0.15351826 | 0.00032851 | 0.21 | 4 |
| 2 | 1/3 | 0.28401792 | 0.28346872 | 0.00054920 | 0.19 | 3 |
| 3 | 1/2 | 0.39416370 | 0.39346933 | 0.00069436 | 0.18 | 3 |
| 4 | 2/3 | 0.48736477 | 0.48658288 | 0.00078189 | 0.16 | 3 |
| 5 | 5/6 | 0.56622791 | 0.56540179 | 0.00082612 | 0.15 | 3 |
| 6 | 1 | 0.63295889 | 0.63212055 | 0.00083834 | 0.13 | 3 |

# Runge-Kutta Methods: 1/22

- **Runge-Kutta methods is** a family **of methods.**
- **In fact, Runge-Kutta methods are extensions to many popular methods, including Euler's and Heun's methods.**
- **Euler's method computes $y(x+\Delta)$ by adding $\Delta\times f(x,y)$ to the current $y$.**
- **Runge-Kutta methods try to express this correction as a *linear combination* of a number of "correction" steps.**

# Runge-Kutta Methods: 2/22

- **Runge-Kutta methods express $y(x+\Delta)$ in the following way:**

$$y(x+\Delta) = y(x) + \Delta \times \Omega(x, y, \Delta)$$

- **Function $\Omega(x,y,\Delta)$ is a function of the current point $(x,y)$ and the step-size $\Delta$, and is usually referred to as the *increment function*.**
- **The increment function is the linear combination of two sets of values, $c_i$'s and $k_i$'s ($1 \leq i \leq n$):**

$$\Omega(x, y, \Delta) = c_1k_1 + c_2k_2 + \cdots + c_nk_n = \sum_{i=1}^{n} c_ik_i$$

# Runge-Kutta Methods: 3/22

- **The $c_i$'s are values to be determined, and are usually $c_i \geq 0$ and $c_1+c_2+\ldots+c_n=1$.**
- **The $k_i$'s are defined by a recurrence relation similar to Heun's method.**
- **Here are the definitions of the $k_i$'s:**

$$\begin{aligned}
k_1 &= f(x, y) \\
k_2 &= f(x + p_2\Delta, y + (a_{2,1}k_1)\Delta) \\
k_3 &= f(x + p_3\Delta, y + (a_{3,1}k_1 + a_{3,2}k_2)\Delta) \\
&\vdots \\
k_n &= f(x + p_n\Delta, y + (a_{n,1}k_1 + a_{n,2}k_2 + \cdots + a_{n,n-1}k_{n-1})\Delta)
\end{aligned}$$

# Runge-Kutta Methods: 4/22

- **Let us look at $p_i$'s, $a_{i,j}$'s and $k_i$'s more closely:**

$$
\begin{aligned}
k_1 &= f(x, y) \\
k_2 &= f(x + p_2\Delta, y + (a_{2,1}k_1)\Delta) \\
k_3 &= f(x + p_3\Delta, y + (a_{3,1}k_1 + a_{3,2}k_2)\Delta) \\
&\vdots \\
k_n &= f(x + p_n\Delta, y + (a_{n,1}k_1 + a_{n,2}k_2 + \cdots + a_{n,n-1}k_{n-1})\Delta)
\end{aligned}
$$

- **Those $p_i$'s and $a_{i,j}$'s are predetermined values.**
- **$k_1$ is $f(x,y)$, $k_2$ uses $k_1$ in a way like a corrector.**
- **$k_3$ uses $k_1$ and $k_2$ for "correction" purpose, etc.**
- **A Runge-Kutta method uses up to $k_j$ is referred to as an *order* $j$ Runge-Kutta method.**

# Runge-Kutta Methods: 5/22

- **Consider a *first-order* Runge-Kutta method.**
- **It uses $k_1 = f(x,y)$ and the increment function is:**

$$\Omega(x,y,\Delta) = c_1 k_1$$

- **Thus, first order Runge-Kutta method is**

$$y(x+\Delta) = y(x) + \Delta \times c_1 k_1 = y(x) + \Delta \times c_1 f(x,y)$$

- **If we chose $c_1 = 1$, first-order Runge-Kutta method becomes Euler's method.**

# Runge-Kutta Methods: 6/22

- **Heun's method calculates $y(x+\Delta)$ as follows:**

$$y(x+\Delta)=y(x)+\Delta\frac{f(x,y)+f(x+\Delta,y^P(x+\Delta))}{2}$$

- **$y^P(x+\Delta)$ is computed as follows:**

$$y^P(x+\Delta)=y(x)+\Delta f(x,y)$$

- **Therefore, Heun's method is a second-order Runge-Kutta method:**

$$\begin{aligned} y(x+\Delta) &= y(x)+\Delta\left[\frac{1}{2}f(x,y)+\frac{1}{2}f(x+\Delta,y^P(x+\Delta))\right] \\ &= y(x)+\Delta\left[\frac{1}{2}\overset{k_1}{\boxed{f(x,y)}}+\frac{1}{2}\overset{k_2}{\boxed{f(x+\Delta,y+\overset{k_1}{\boxed{f(x,y)}}\Delta)}}\right] \\ &= y(x)+\Delta\left[c_1k_1+c_2k_2\right] \end{aligned}$$

$p_2=1$ $a_{2,1}=1$

# Runge-Kutta Methods: 7/22

- **Let us examine general second-order Runge-Kutta methods.**
- **A second-order Runge-Kutta method is:**

$$y(x+\Delta) = y(x) + \Delta\left[c_1 f(x,y) + c_2 \overset{k_2}{\boxed{f(x+p_2\Delta, y+a_{2,1} f(x,y)\Delta)}}\right]$$

- **We wish to expand $k_2$. This will need the two-variable Taylor series as follows:** **$h^2$, $k^2$ and higher terms**

$$g(x+h, y+k) = g(x,y) + h\frac{\partial g}{\partial x} + k\frac{\partial g}{\partial y} + (....)$$

- **The following shows the result of expanding $k_2$:**

$$f(x+p_2\Delta, y+a_{2,1} f(x,y)\Delta)$$

**$\Delta^2$ and higher terms**

$$= f(x,y) + (p_2\Delta)\frac{\partial f}{\partial x} + (a_{2,1} f(x,y)\Delta)\frac{\partial f}{\partial y} + (...)$$

# Runge-Kutta Methods: 8/22

- **Plugging the expanded $k_2$ back into $y(x+\Delta)$ yields the following important result:**

$$
\begin{aligned}
y(x+\Delta) \quad = \quad & y(x) \\
+ \quad & \Delta(c_1+c_2)f(x,y) \\
+ \quad & \Delta^2\left[(c_2p_2)\frac{\partial f}{\partial x}+(c_2a_{2,1}f(x,y))\frac{\partial f}{\partial y}\right] \\
+ \quad & \boxed{(.....)}
\end{aligned}
$$

**$\Delta^3$ and higher terms**

# Runge-Kutta Methods: 9/22

- **Consider the function $y(x+\Delta)$ itself. It can be expanded using Taylor series as follows:**

**$\Delta^3$ and higher terms**

$$y(x+\Delta) = y(x) + \Delta y'(x) + \frac{\Delta^2}{2!} y''(x) + \boxed{(\ldots)}$$

- **Note that $y'(x) = f(x,y)$ and $y''(x)$ can be computed using the chain-rule:**

$$\begin{aligned} y''(x) &= \frac{d(y')}{dx} = f'(x,y) \\ &= \frac{\partial f(x,y)}{\partial x} + \frac{\partial f(x,y)}{\partial y}\boxed{\frac{dy}{dx}} \\ &= \frac{\partial f(x,y)}{\partial x} + \frac{\partial f(x,y)}{\partial y} f(x,y) \end{aligned}$$

**this is $y'(x)=f(x,y)$**

# Runge-Kutta Methods: 10/22

- **Plugging *y'(x)* and the computed *y''(x)* back into *y(x+Δ)*, we have the second important result:**

**$\Delta^3$ and higher terms**

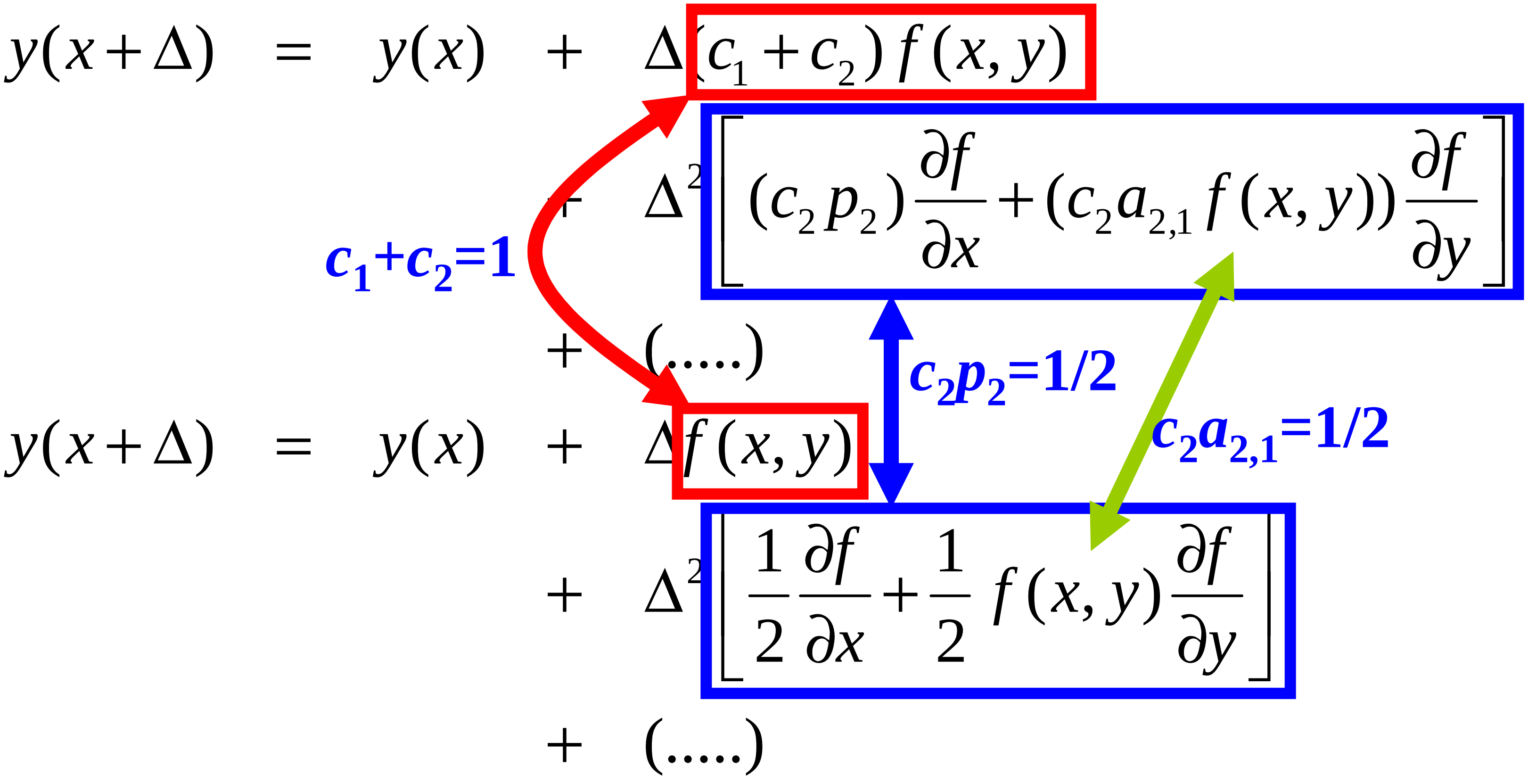
$$y(x+\Delta) = y(x)+\Delta f(x,y)+\frac{\Delta^2}{2}y''(x)+\boxed{(.....)}$$

$$= y(x)+\Delta f(x,y)+\Delta^2\left[\frac{1}{2}\frac{\partial f}{\partial x}+\frac{1}{2}f(x,y)\frac{\partial f}{\partial y}\right]+\boxed{(.....)}$$

**$\Delta^3$ and higher terms**

# Runge-Kutta Methods: 11/22

- **Now compare the two results of *y(x+Δ)* we obtained.** They should be equal!



$$
\begin{aligned}
y(x+\Delta) &= y(x) + \Delta\boxed{(c_1+c_2)f(x,y)} \\
&+ \Delta^2\boxed{\left[(c_2p_2)\frac{\partial f}{\partial x} + (c_2a_{2,1}f(x,y))\frac{\partial f}{\partial y}\right]} \\
&+ (\ldots..)
\end{aligned}
$$

$$
\begin{aligned}
y(x+\Delta) &= y(x) + \Delta\boxed{f(x,y)} \\
&+ \Delta^2\boxed{\left[\frac{1}{2}\frac{\partial f}{\partial x} + \frac{1}{2}f(x,y)\frac{\partial f}{\partial y}\right]} \\
&+ (\ldots..)
\end{aligned}
$$

$c_1+c_2=1$

$c_2p_2=1/2$

$c_2a_{2,1}=1/2$

# Runge-Kutta Methods: 12/22

- From comparing the coefficients of $\Delta$ and $\Delta^2$, we have the following:

$$\begin{aligned} c_1 + c_2 &= 1 \\ c_2 p_2 &= 1/2 \\ c_2 a_{2,1} &= 1/2 \end{aligned} \quad \textbf{or} \quad \begin{aligned} c_1 &= 1 - c_2 \\ p_2 &= 1/(2c_2) \\ a_{2,1} &= 1/(2c_2) \end{aligned}$$

- Since four unknowns (*i.e.*, $c_1$, $c_2$, $p_2$, $a_{2,1}$) cannot be solved from three equations, one must fix a variable to some value.
- Varying the value of $c_2$ yields a series of second-order Runge-Kutta method.

# Runge-Kutta Methods: 13/22

- **If $c_2 = \frac{1}{2}$, then $c_1 = \frac{1}{2}$, $p_2 = 1$ and $a_{2,1} = 1$.**
- **Second-order Runge-Kutta method becomes:**

$$\begin{aligned} y(x+\Delta) &= y(x)+\Delta\left[\frac{1}{2}f(x,y)+\frac{1}{2}f(x+\Delta,y+f(x,y)\Delta)\right] \\ &= y(x)+\frac{\Delta}{2}\left[f(x,y)+f(x+\Delta,y+f(x,y)\Delta)\right] \end{aligned}$$

- **This is Heun's method! We saw it earlier.**

$$\boxed{\begin{aligned} c_1 &= 1-c_2 \\ p_2 &= a_{2,1}=\frac{1}{2c_2} \end{aligned}}$$

# Runge-Kutta Methods: 14/22

- **If $c_2$=1, then $c_1$=0, $p_2 = a_{2,1} = ½$.**
- **Second order Runge-Kutta method becomes:**

$$\begin{aligned} y(x+\Delta) &= y(x)+\Delta k_2 \\ &= y(x)+\Delta f\left(x+\frac{1}{2}\Delta, y+\frac{1}{2}\Delta f(x,y)\right) \end{aligned}$$

- **This is referred to as the *midpoint* method.**

$$\boxed{\begin{aligned} c_1 &= 1-c_2 \\ p_2 &= a_{2,1}=\frac{1}{2c_2} \end{aligned}}$$

# Runge-Kutta Methods: 15/22

- **If $c_2 = 2/3$, then $c_1 = 1/3$, $p_2 = a_{2,1} = ¾$.**
- **Second-order Runge-Kutta method becomes:**

$$y(x+\Delta) = y(x) + \Delta\left[\frac{1}{3}f(x,y) + \frac{2}{3}f\left(x+\frac{3}{4}\Delta, y+\frac{3}{4}\Delta f(x,y)\right)\right]$$

- **This is the *Ralston* method.**

$$\boxed{\begin{aligned} c_1 &= 1-c_2 \\ p_2 &= a_{2,1} = \frac{1}{2c_2} \end{aligned}}$$

# Runge-Kutta Methods: 16/22

- The following shows a possible second-order Runge-Kutta method program.
- The initialization part, however, is for Ralston method.

```
! Initialization
! [a,b] : input
! n : intervals

c2  = 2.0/3.0
c1  = 1 - c2
p2  = 1/(2*c2)
a21 = 1/(2*c2)
```

for Ralston method

```
Δ = (b-a)/n
x = a
DO i = 1, n
   k1 = f(x,y)
   k2 = f(x+p2*Δ, y+a21*k1*Δ)
   y  = y + Δ*(c1*k1 + c2*k2)
   x  = x + Δ
END
```

general second-order Runge-Kutta method

# Runge-Kutta Methods: 17/22

- **ODE $y' = 4e^{0.8x} - 0.5y$ with $y(0)=2$ has the following exact solution:**

$$y = \frac{4}{1.3}\left(e^{0.8x} - e^{-0.5x}\right) + 2e^{-0.5x}$$

- **The following uses $\Delta=1$ on [0,4]:**

**predictor-corrector**

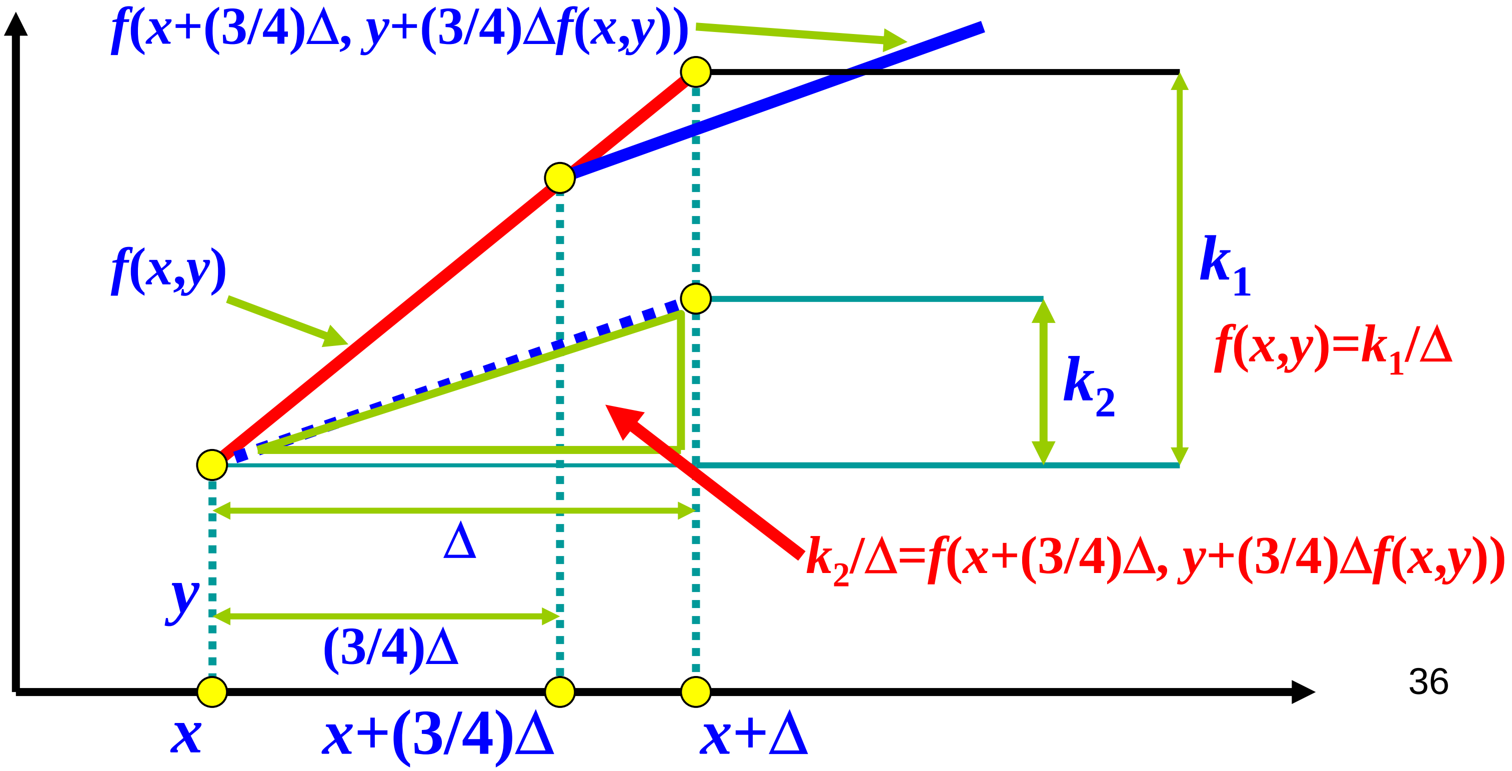
| x | Exact y | Euler | % | Heun | % | P-C | % | Ralston | % |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 2 | 2 | 0 | 2 | 0 | 2 | 0 | 2 | 0 |
| 1 | 6.19463 | 5.00000 | 19.3 | 6.70108 | 8.2 | 6.36053 | 2.7 | 6.44232 | 4.0 |
| 2 | 14.84392 | 11.40216 | 23.2 | 16.31978 | 9.9 | 15.30125 | 3.1 | 15.58216 | 5.0 |
| 3 | 33.67717 | 25.51321 | 24.2 | 37.19925 | 10.5 | 34.74091 | 3.2 | 35.45657 | 5.3 |
| 4 | 75.33897 | 56.84931 | 24.5 | 83.33777 | 10.6 | 77.72971 | 3.2 | 79.39618 | 5.4 |

**6 iterations**

# Runge-Kutta Methods: 18/22

- **A geometric interpretation of the $k_i$'s.**
- **Take the Ralston method as an example.**

$$y(x+\Delta) = y(x) + \boxed{\Delta}\left[\frac{1}{3}\boxed{f(x,y)} + \frac{2}{3}\boxed{f\left(x+\frac{3}{4}\Delta,\ y+\frac{3}{4}\Delta f(x,y)\right)}\right]$$



# Runge-Kutta Methods: 19/22

- **Higher order Runge-Kutta methods are more accurate but more time consuming.**
- **They are derived based on the same principle used for second-order methods.**
- **The following is a commonly used third-order Runge-Kutta method:**

$$
\begin{aligned}
k_1 &= f(x, y) \\
k_2 &= f\left(x + \frac{1}{2}\Delta, y + \frac{1}{2}\Delta k_1\right) \\
k_3 &= f(x + \Delta, y - \Delta k_1 + 2\Delta k_2) \\
y(x+\Delta) &= y(x) + \frac{1}{6}\Delta\left(k_1 + 4k_2 + k_3\right)
\end{aligned}
$$

# Runge-Kutta Methods: 20/22

- **The following is the most commonly used fourth-order method, usually referred to as the *classical fourth-order Runge-Kutta method*:**

$$
\begin{aligned}
k_1 &= f(x, y) \\
k_2 &= f\left(x + \frac{1}{2}\Delta, y + \frac{1}{2}\Delta k_1\right) \\
k_3 &= f\left(x + \frac{1}{2}\Delta, y + \frac{1}{2}\Delta k_2\right) \\
k_4 &= f(x + \Delta, y + \Delta k_3) \\
y(x + \Delta) &= y(x) + \frac{1}{6}\Delta\left(k_1 + 2k_2 + 2k_3 + k_4\right)
\end{aligned}
$$

# Runge-Kutta Methods: 21/22

- **If more accurate results are required, use Butcher's sixth-order Runge-Kutta method:**

$$
\begin{aligned}
k_1 &= f(x, y) \\
k_2 &= f\left(x + \frac{1}{4}\Delta, y + \frac{1}{4}\Delta k_1\right) \\
k_3 &= f\left(x + \frac{1}{4}\Delta, y + \frac{1}{8}\Delta k_1 + \frac{1}{8}\Delta k_2\right) \\
k_4 &= f\left(x + \frac{1}{2}\Delta, y - \frac{1}{2}\Delta k_2 + \Delta k_3\right) \\
k_5 &= f\left(x + \frac{3}{4}\Delta, y + \frac{3}{16}\Delta k_1 + \frac{9}{16}\Delta k_4\right) \\
k_6 &= f\left(x + \Delta, y - \frac{1}{3}\Delta k_1 + \frac{2}{7}\Delta k_2 + \frac{12}{7}\Delta k_3 - \frac{12}{7}\Delta k_4 + \frac{8}{7}\Delta k_5\right) \\
y(x+\Delta) &= y(x) + \frac{1}{90}\Delta\left(7k_1 + 32k_3 + 12k_4 + 32k_5 + 7k_6\right)
\end{aligned}
$$

# Runge-Kutta Methods: 22/22

- **ODE $y' = 4e^{0.8x} - 0.5y$ with $y(0)=2$ has the following exact solution:**

$$y = \frac{4}{1.3}\left(e^{0.8x} - e^{-0.5x}\right) + 2e^{-0.5x}$$

- **The following uses $\Delta=1$ on [0,4]:**

| x | Exact y | Ralston | % | 3rd | % | 4th | % | 6th | % |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 2 | 2 | 0 | 2 | 0 | 2 | 0 | 2 | 0 |
| 1 | 6.19463 | 6.44232 | 4.0 | 6.17568 | 0.31 | 6.20104 | 0.10 | 6.19469 | 0.00 |
| 2 | 14.84392 | 15.58216 | 5.0 | 14.78616 | 0.39 | 14.86248 | 0.13 | 14.84410 | 0.00 |
| 3 | 33.67717 | 35.45657 | 5.3 | 33.53672 | 0.42 | 33.72135 | 0.13 | 33.67760 | 0.00 |
| 4 | 75.33897 | 79.39618 | 5.4 | 75.01767 | 0.43 | 75.43918 | 0.13 | 75.33994 | 0.00 |

# The End