Chapter 8

Numerical Solution of Ordinary Differential Equation

Ordinary Differential Equation (ODE):

An Ordinary Differential Equation (ODE) in numerical methods is an equation that relates a function with its derivatives — meaning it shows how a quantity changes with respect to another (usually time).

It has the form:

$$\frac{dy}{dx} = F(x_i, y_i)$$

Where,

x = Independent variable

y = Dependent variable

$f(x, y)$ = some given function that defines the rate of change of y

Numerical Solution of Ordinary Differential Equations (ODEs)

Numerical methods for solving ODEs are used when a differential equation cannot be solved analytically (by exact formulas). These techniques — such as Euler's method, Runge–Kutta methods, and Predictor–Corrector methods — provide approximate solutions at discrete points.

- Numerical ODE methods help us understand and predict how quantities change over time when analytical solutions are not available.

Why We Use Them:

- Many real-world problems involve complex systems where finding an exact mathematical solution is impossible or too difficult.
- Numerical methods allow us to simulate and analyze such systems step by step using a computer.

Real-Life Applications:

- **Engineering:** Predicting motion of vehicles, vibrations, or heat transfer.
- **Biology:** Modeling population growth or spread of diseases.
- **Physics:** Simulating planetary motion or electric circuits.
- **Medicine:** Modeling drug concentration in the bloodstream over time.

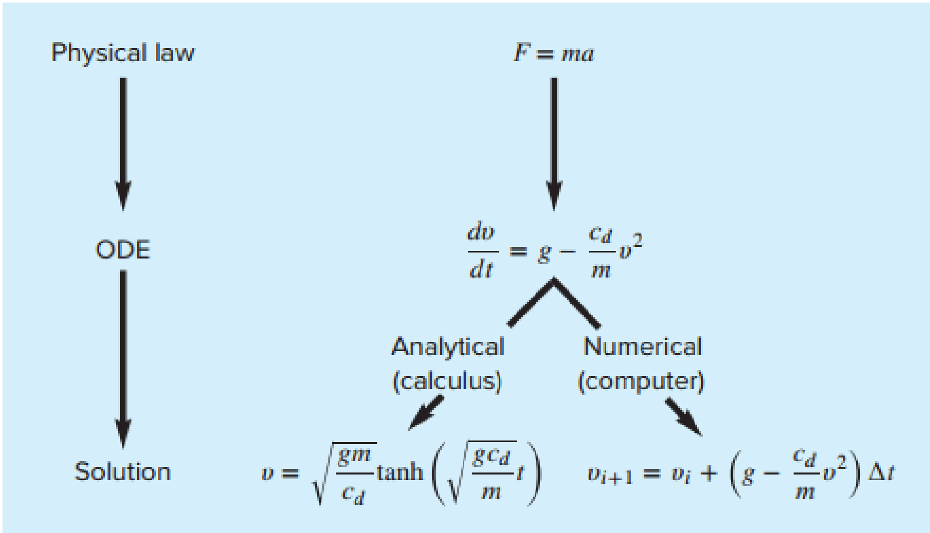


FIGURE PT6.1
The sequence of events in the development and solution of ODEs for engineering and science. The example shown is for the velocity of the free-falling bungee jumper.

The equation above is called 1st order and ordinary because:

- Only first derivative is involved.
- There is only one independent variable t.

Unlike ordinary equations, which have a number (real or complex) as its solution. The solution to an ODE is a function that satisfies the equation. Finding an analytical solution to an ODE involves integral calculus. An example is presented below (assume that y is a function of x).

$$y' = e^{(x-y)}$$
$$\frac{dy}{dx} = e^{(x-y)}$$
$$\frac{dy}{dx} = \frac{e^x}{e^y}$$

This particular ODE is in separable form

$$dxe^x = dye^y$$
$$\int e^x dx = \int e^y dy$$
$$e^x + C_1 = e^y + C_2$$
$$e^y = e^x + C$$
$$y = \ln(e^x + C)$$

If we are given an *initial value condition* $y(0) = y_0$. We can determine the constant C

$$y_0 = \ln(1 + C)$$

$$e^{y_0} = 1 + C$$
$$C = 1 - e^{y_0}$$

As evident from the discussions above, the analytical solutions to ODEs are generally difficult to find. In numerical solution to ODEs, we try to estimate the solution function at discrete points thus getting some discrete values rather than a full function.

Initial Value Problems (IVP):

An Initial Value Problem (IVP) is a type of ordinary differential equation (ODE) where the value of the unknown function is given at a specific point. It tells us how a quantity changes (the ODE) and where to start (the initial condition).

In numerical methods, the initial value gives us the starting point to compute approximate values of y step by step using methods like:

- Euler's Method
- Runge-Kutta Method

General Form:

$$\frac{dy}{dx} = F(x_i, y_i)\,, \qquad y(x_0) = y_0$$

Here,

$\frac{dy}{dx}$ = derivative of y with respect to x

$f(x_i, y_i)$ = given function describing the rate of change

$y(x_0)$ = initial condition (the starting value)

Example:

Suppose we have:

$$\frac{dy}{dx} = x + y$$

And, an initial condition $y(0) = \ 1$ (That is, when $x = 0,\ y = 1$)

We need to find the function $y(x)$ that satisfies both:

1. The differential equation $\frac{dy}{dx} = x + y$
2. The initial condition y (0) = 1

Euler Method:

The Euler method is the simplest numerical method used to solve ordinary differential equations (ODEs), especially initial value problems (IVPs). It helps you approximate the solution step by step when you can't find an exact one. The essence of the Euler's method is the following

$$y_{i+1} = y_i + \phi h$$

$$new\ value = old\ value + slope\ \times step\ size$$

The first derivative provides a direct estimate of the slope at x_i (Fig. 22.1):

$$\phi = \frac{dy}{dx} = F(x_i, y_i)$$

Where $F(x_i, y_i)$ is the differential equation evaluated at x_i, y_i. Thus, the method can be summarized as

$$y_{i+1} = y_i + F(x_i, y_i)h$$

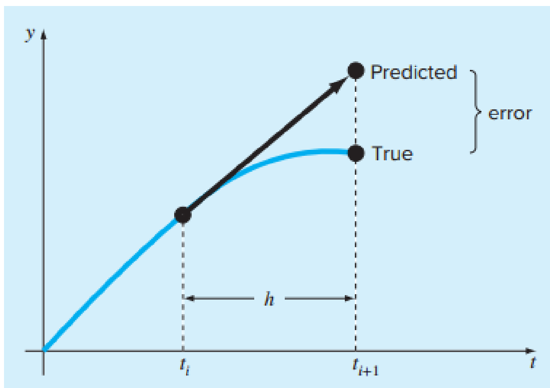


FIGURE 22.1
Euler's method.

Error Analysis and Stability of the Euler's Method

By using the Taylor's series expansion of y

$$y_{i+1} = y_i + y_i' h + \frac{y_i''}{2!} h^2 + \cdots + \frac{y_i^n}{n!} h^n + O(h^{n+1})$$

Where $O(h^{n+1})$ specifies that the truncation error is proportional to the step size h, raised to the $(n+1)$th power. Since we can see that by truncating the Taylor's series as following

$$y_{i+1} = y_i + F(x_i, y_i)h$$

The error is

$$E = O(h^2)$$

The stability of a solution method is another important consideration that must be considered when solving ODEs. A numerical solution is said to be unstable if errors grow exponentially for a problem for which there is a bounded solution.

The stability of a particular application can depend on three factors: the differential equation, the numerical method, and the step size. Insight into the step size required for stability can be examined by studying a very simple ODE:

$$\frac{dy}{dx} = -ay$$

If $y(0) = y_0$, calculus can be used to solve the equation as

$$y = y_0 e^{-ax}$$

Thus, the solution starts at y_0 and asymptotically approaches zero. Now suppose that we use Euler's method to solve the same problem numerically:

$$y_{i+1} = y_i + F(y_i, x_i)h$$

Substituting $\frac{dy}{dx} = -ay$ gives

$$y_{i+1} = y_i(1 - ah)$$

Which shows that if $|1 - ah| > 1$ then $y_i \to \infty$ as $i \to \infty$. Note that there are certain ODEs where errors always grow regardless of the method. Such ODEs are called ill-conditioned.

Example Problem: Solve for the velocity and position of the free-falling bungee jumper using Euler's method. Assuming that at $t = 0$, $x = 0$, and, $v = 0$. Predict up to $t = 10s$ with a step size of $h = 2s$. The gravitational acceleration is $g = 9.8m/s^2$, and the jumper has a mass of $m = 68.1\ kg$ with a drag coefficient of $c_d = 0.25\ kg/m$.

Solution: The ODEs can be used to compute the slopes at $t = 0$ as

$$\frac{dx}{dt} = 0$$
$$\frac{dv}{dt} = 9.81 - \frac{0.25}{68.1}(0)^2 = 9.81$$

Euler's method is then used to compute the values at $t_1 = 2s$

$$x = 0 + 0(2) = 0$$
$$v = 0 + 9.81(2) = 19.62$$

The process can be repeated to compute the results at $t_2 = 4s$ as

$$x = 0 + 19.62(2) = 39.24$$
$$v = 19.62 + \left(9.81 - \frac{0.25}{68.1}(19.62)^2\right)2 = 36.41368$$

Proceeding in this manner results in the following table

TABLE 22.3 Distance and velocity of a free-falling bungee jumper as computed numerically with Euler's method.

t	x_{true}	v_{true}	x_{Euler}	v_{Euler}	$\varepsilon_t(x)$	$\varepsilon_t(v)$
0	0	0	0	0		
2	19.1663	18.7292	0	19.6200	100.00%	4.76%
4	71.9304	33.1118	39.2400	36.4137	45.45%	9.97%
6	147.9462	42.0762	112.0674	46.2983	24.25%	10.03%
8	237.5104	46.9575	204.6640	50.1802	13.83%	6.86%
10	334.1782	49.4214	305.0244	51.3123	8.72%	3.83%

Although the foregoing example illustrates how Euler's method can be implemented for systems of ODEs, the results are not very accurate because of the large step size. In addition, the results for distance are a bit unsatisfying because x does not change until the second iteration. Using a much smaller step greatly mitigates these deficiencies.

Example: Suppose $\frac{dp}{dt} = t^2 + 0.2p$, $p(0)$= 10. Calculate the value at t = 2 sec using Euler method with step size h = 0.5.

Solution:
We are given the initial value problem (IVP):

$\frac{dp}{dt} = t^2 + 0.2p$,   p(0) =10

We have to estimate the value of $p(t)$ at $t = 2$ using the Euler method.

$$P_{n+1} = P_n + f(t_n, p_n)h$$

Where,

$f(t,p) = t^2 + 0.2p$
$h = 0.5$
Initial condition: $P_0 = 10, t_0 = 0$
We'll compute: $t = 0,\ 0.5,\ 1.0,\ 1.5,\ 2.0$

Step 1 : $t_0 = 0, P_0 = 10$
$f(0{,}10) = 0^2 + 0.2(10) = 2$
$P_1 = 10 + 0.5(2) = 10{+}1 = 11$

Step 2 : $t_1 = 0.5, P_1 = 11$
$f(0.5{,}11) = 0.5^2 + 0.2(11) = 2.45$
$P_2 = 11 + 0.5(2.45) = 12.225$

Step 3 : $t_2 = 1, P_2 = 12.225$

$f(1{,}12.225) = 1^2 + 0.2(12.225) = 3.445$

$P_3 = 12.225 + 0.5(3.445) = 13.9475$

Step 4 : $t_3 = 1.5, P_3 = 13.9475$

$f(1.5{,}13.9475) = 1.5^2 + 0.2(13.9475) = 5.0395$

$P_4 = 13.9475 + 0.5(5.0395) = 16.46725$

Final answer: At t= 2, $p(2) = 16.46725$

Runge kutta:

The Runge-Kutta (RK) method and Euler method are both techniques used in numerical analysis to solve ordinary differential equations (ODEs) of the form: The Runge–Kutta method (often written as RK method) is a family of iterative methods used to find numerical solutions of ordinary differential equations (ODEs) — especially when an exact solution is hard or impossible to get. It improves accuracy compared to simpler methods like Euler's method.

$$\frac{dy}{dx} = f(x,y) \qquad y(x_0) = y_0$$

Why we use Runge-Kutta method:

- ✓ **Accuracy:**
  - RK methods, especially the popular 4th-order Runge-Kutta (RK4), provide much higher accuracy than Euler's method for the same step size.
  - Euler's method is only first-order accurate, meaning the error per step is proportional to the step size h, while RK4 is fourth-order, so error per step is proportional to h^5 and total error is proportional to h^4.
- ✓ **Stability:**
  - RK methods are more stable than Euler's method, especially for stiff equations or problems where the solution changes rapidly.
- ✓ **No need for derivatives beyond first order:**
  - Unlike some other methods (like Taylor series methods), RK methods do not require higher-order derivatives of f(x,y), making them practical for complex problems.

Advantages of Runge-Kutta over Euler method:

Aspect	Euler Method	Runge-Kutta Method
Accuracy	Low (first-order)	High (commonly 4th-order, very accurate)
Stability	Less stable	More stable
Step size sensitivity	Requires very small step size for accuracy	Can use larger step size for similar accuracy
Error	Accumulates quickly	Accumulates slowly
Implementation	Very simple	Slightly more complex than Euler

Disadvantages of Runge-Kutta:

- **Computational cost:**
  - ✓ RK methods require multiple function evaluations per step (e.g., 4 evaluations for RK4), whereas Euler requires only 1.
  - ✓ This makes RK more computationally expensive, especially for very large systems.
- **Complexity:**
  - ✓ RK formulas are slightly more complicated to implement compared to the simple Euler formula.

General Problem form:

We have a first -order ODE:

$$\frac{dy}{dx} = f(x, y) \qquad y(x_0) = y_0$$

We want to find y at some next point $x_{n+1} = x_n + h$, where h is the step size.

- Instead of taking the slope only at the beginning (like Euler), Runge–Kutta methods compute several slopes within the interval, then combine them in a smart weighted average to get a more accurate prediction.

Different Order of RK Methods:

- **RK1: Euler's Method**
  - The equation is: $y_{i+1} = y_i + F(x_i, y_i)h$
  - Only one slope (beginning point)
  - Simple but not accurate (1st order)

- **RK2 : Second Order (Improved Euler/ Heun/ Midpoint)**

$$k_1 = f(x_n, y_n)$$

$$k_2 = f(x_n + h,\ y_n + hk_1)$$

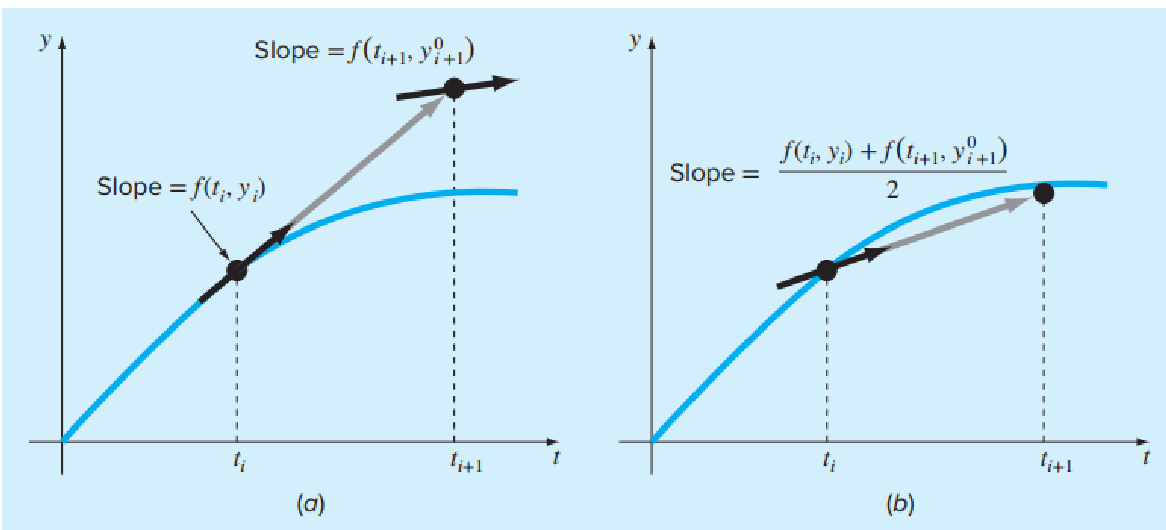
$$y_{i+1} = y_n + \frac{h}{2}(k_1 + k_2)$$

  - Uses average of two slopes.
  - More accurate

Improving the Euler's Method: Heun's Method

One method to improve the estimate of the slope involves the determination of two derivatives for the interval—one at the beginning and another at the end. The two derivatives are then averaged to obtain an improved estimate of the slope for the entire interval. This approach, called Heun's method, is depicted graphically in Fig. 22.4.

FIGURE 22.4
Graphical depiction of Heun's method. (*a*) Predictor and (*b*) corrector.



Recall that in Euler's method, the slope at the beginning of an interval

$$y_i' = F(x_i, y_i)$$

is used to extrapolate linearly to y_{i+1}

$$y^0_{i+1} = y_i + F(x_i, y_i)h$$

For the standard Euler's method we would stop at this point. However, in Heun's method the y^0_{i+1} calculated is not the final answer, but an intermediate prediction. This is why we have distinguished it with a superscript 0. Equation above is called a *predictor equation*. It provides an estimate that allows the calculation of a slope at the end of the interval:

$$y'_{i+1} = F(x_{i+1}, y^0_{i+1})$$

Then the two slopes are averaged

$$\bar{y}' = \frac{F(x_i, y_i) + F(x_{i+1}, y^0_{i+1})}{2}$$

This average slope is then used to extrapolate linearly from y_i to y_{i+1} using Euler's method:

$$y_{i+1} = y_i + \bar{y}'h$$

which is called a *corrector equation*. As with any other iterative methods, the approximation error is

$$E_a = \left| \frac{y^j_{i+1} - y^{j-1}_{i+1}}{y^j_{i+1}} \right| 100\%$$

Insights into the error bound can also be gained by recognizing that this method is actually related to the trapezoidal rule. We shall demonstrate this for the case of polynomials where the ODE is solely a function of the independent variable x. In that case the predictor step is not required and the corrector is applied only once. For such cases, the technique is expressed concisely as,

$$y_{i+1} = y_i + \frac{F(x_i) + F(x_{i+1})}{2}h$$

The connection between the Heun's method and the trapezoidal method can be formally demonstrated by starting with the ordinary differential equation:

$$\frac{dy}{dx} = F(x)$$

The equation can be solved for y by integrating

$$\int_{y_i}^{y_{i+1}} dy = \int_{x_i}^{x_{i+1}} F(x)dx$$

Which yields

$$y_{i+1} - y_i = \int_{x_i}^{x_{i+1}} F(x)dx$$

Or,

$$y_{i+1} = y_i + \int_{x_i}^{x_{i+1}} F(x)dx$$

Now approximating the integral on the right side by using the trapezoidal rule (1^{st} order method) of integration

$$\int_{x_i}^{x_{i+1}} F(x)dx \cong \frac{F(x_i) + F(x_{i+1})}{2}h$$

Where $h = x_{i+1} - x_i$. Since we know that the error bound for the trapezoidal rule was

$$E = \mathcal{O}(h^3)$$

This same error bound also holds for the Heun's method, which is an improvement over the $\mathcal{O}(h^2)$ for the Euler's method.

Example: Suppose $\frac{dp}{dt} = t^2 + 0.2p$, $p(0)$= 10. Calculate the value at t = 2 sec using Heun's method with step size h = 0.5.

Solution:

$$P_{n+1} = P_n + \frac{h}{2}\left[f(t_n, P_n) + f(t_{n+1}, P_n + h \cdot f(t_n, P_n))\right]$$

Step 1 : $t_0 = 0, P_0 = 10$

$f_0 = 0^2 + (0.2 * 10) = 2$

$P^* = 10 + 0.5(2) = 10+1 = 11$ (Intermediate prediction)

$f^* = 0.5^2 + (0.2 * 11) = 2.45$

$P_1 = 10 + 0.5(\frac{2+2.45}{2}) = 11.1125$

Step 2 : $t_1 = 0.5, P_1 = 11.1125$

$f_1 = 0.5^2 + (0.2 * 11.1125) = 2.4725$

$P^* = 11.1125 + 0.5(2.4725) = 12.34875$ (Intermediate prediction)

$f^* = 1^2 + (0.2 * 12.34875\ ) = 3.46975$

$P_2 = 11.1125 + 0.5(\frac{3.46975+2.4725}{2}) = 12.5981$

Step 3 : $t_2 = 1, P_2 = 12.5981$

$f_2 = 1^2 + (0.2 * 12.5981) = 3.5196$

$P^* = 12.5981 + 0.5(3.5196) = 14.3579$ (Intermediate prediction)

$f^* = 1.5^2 + (0.2 * 14.3579\ ) = 5.1216$

$P_3 = 12.5981 + 0.5(\frac{3.5196+5.1216}{2}) = 14.7584$

Step 4 : $t_3 = 1.5, P_3 = 14.7584$

$f_3 = 1.5^2 + (0.2 * 14.7584) = 5.2017$

$P^* = 14.7584 + 0.5(5.2017) = 17.3592$ (Intermediate prediction)

$f^* = 2^2 + (0.2 * 17.3592) = 7.4718$

$P_4 = 14.7584 + 0.5(\frac{5.2017+ 7.4718}{2}) = 17.9268$

Final answer: At t= 2, $p(2) = 17.9268$

RK4: Fourth Order (Most common)

The RK4 method improves the accuracy of estimating y_{n+1} (the value of y at the next step) by taking a weighted average of slopes at different points within the interval.

Unlike Euler's method (which uses only the slope at the beginning), RK4 uses four slopes.

Suppose we have an initial value problem (IVP):

$$\frac{dy}{dx} = f(x, y) \qquad\qquad y(x_0) = y_0$$

We want to find y at some point x numerically. Now compute the four slopes:

$$k_1 = f(x_n, y_n)$$

$$k_2 = f\left(x_n + \frac{h}{2},\ y_n + \frac{h}{2}k_1\right)$$

$$k_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2\right)$$

$$k_4 = f(x_n + h, y_n + hk_3)$$

$$y_{i+1} = y_n + \frac{h}{6}(k_1 + \ 2k_2 + \ 2k_3 + \ k_4)$$

- Uses weighted average of four slopes.
- Accuracy: 4th Order
- It takes four slopes (k1: beginning, k2, k3: midpoints, k4: end)

✓ Euler's method guesses the next position using only the current velocity.
✓ RK methods sample the velocity multiple times within the step and average them — so we get a smoother, more accurate path.

Explanation of the Slopes

- K1: slope at the beginning of the interval (like Euler's method)
- K2: slope at the midpoint, using K1
- K3: slope at the midpoint again, using K2
- K4: slope at the end of the interval, using K3

The combination $\frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$ is a weighted average giving a very accurate estimate.

Advantages:

- Much more accurate than Euler's method or even 2nd order methods.
- Local truncation error is O(h^5), so global error is O(h^4) — hence 4th order

Example:

We have a first -order ODE:

$$\frac{dy}{dx} = x + y \qquad y(0) = 1$$

With h = 0.1

- Compute k1, k2, k3, k4 using the formulas above.
- Calculate $y_1 = y_0 + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$

And repeat for the next steps.

Systems of ODEs

Many practical problems in engineering and science require the solution of a system of simultaneous ordinary differential equations rather than a single equation. Such systems may be represented generally as

$$\frac{dy_1}{dx} = F_1(x, y_1, y_2, \dots y_n)$$
$$\frac{dy_2}{dx} = F_2(x, y_1, y_2, \dots, y_n)$$
$$.$$
$$.$$
$$\frac{dy_n}{dx} = F_n(x, y_1, y_2, \dots, y_n)$$

The solution of such a system requires that n initial conditions be known at the starting value of x.

An example is the calculation of the bungee jumper's velocity and position that we set up at the beginning of this chapter. For the free-fall portion of the jump, this problem amounts to solving the following system of ODEs:

$$\frac{dx}{dt} = v$$
$$\frac{dv}{dt} = g - \frac{c_d}{m}v^2$$

If the stationary platform from which the jumper launches is defined as $t = 0$, the initial conditions would be $x(0) = v(0) = 0$.
All the methods discussed in this chapter for single equations can be extended to systems of ODEs.

Engineering applications can involve thousands of simultaneous equations. In each case, the procedure for solving a system of equations simply involves applying the one-step technique for every equation at each step before proceeding to the next step.