Ordinary Differential Equations

- Types of Ordinary differential equations.
- Euler's method.

Differential equations

- Most fundamental and basic equations in physics as well as frequently occurring problems appear as differential equations.

Differential equations

- Most fundamental and basic equations in physics as well as frequently occurring problems appear as differential equations.
- Perhaps the most common use of computers in physics is for the solution of differential equations!

- Most fundamental and basic equations in physics as well as frequently occurring problems appear as differential equations.
- Perhaps the most common use of computers in physics is for the solution of differential equations!
- Examples :

$$m\frac{d^2x(t)}{dt^2} = -\,kx(t)$$
$$i\hbar\frac{\partial\psi(x,t)}{\partial t} = -\,\frac{1}{2m}\frac{\partial^2\psi(x,t)}{\partial x^2} + V(x)\psi(x,t)$$

Classification of differential equations

- Ordinary vs Partial differential equations.

Classification of differential equations

- Ordinary vs Partial differential equations.
- Linear vs non-linear differential equations.

Classification of differential equations

- Ordinary vs Partial differential equations.
- Linear vs non-linear differential equations.
- Homogeneous vs non-homogeneous differential equations.

Classification of differential equations

- Ordinary vs Partial differential equations.
- Linear vs non-linear differential equations.
- Homogeneous vs non-homogeneous differential equations.
- General solution vs particular solution of the differential equation.

Classification of differential equations

- Ordinary vs Partial differential equations.
- Linear vs non-linear differential equations.
- Homogeneous vs non-homogeneous differential equations.
- General solution vs particular solution of the differential equation.
- Initial value problem vs boundary value problem vs eigenvalue problem.

Simple Harmonic Oscillator

$$m\frac{d^2x(t)}{dt^2} = -kx(t)$$
$$x(t=0) = x_0$$
$$\left.\frac{dx}{dt}\right|_{t=0} = v_0$$

is initial value problem for the second order ordinary linear homogeneous differential equation

Ordinary vs Partial differential equations

- The ordinary differential equations (ODE) – have functions of one only independent variable.
  Example: stationary Schrödinger equation

Ordinary vs Partial differential equations

- The ordinary differential equations (ODE) – have functions of one only independent variable.
  Example: stationary Schrödinger equation
- The partial differential equations (PDE) – have functions of several independent variables.
  Example: time dependent Schrödinger equation for $\psi(\mathbf{r}, t)$

Linear vs Non-linear differential equation

- A linear differential equation – all of the derivatives appear in linear form and none of the coefficient depends on the dependent variable

$$a_0 x(t) + a_1 \frac{dx}{dt} + a_2 \frac{d^2x}{dt^2} + \ldots = c$$

Example:

$$m \frac{d^2 x(t)}{dt^2} = -kx(t)$$

Non-linear differential equation

- A nonlinear differential equation – if the coefficients depend on the dependent variable, OR the derivatives appear in a nonlinear form:
  Examples:

$$\frac{dx}{dt}\frac{d^2x}{dt^2} - x(t) = 0$$
$$t^2\frac{d^2x}{dt^2} - x^2(t) = 0$$

Order of the ODE

- The order n of an ordinary differential equation is the order of the highest derivative appearing in the differential equation

$$
\begin{aligned}
t^2 \frac{d^2 x(t)}{dt^2} - x(t) = 0 \quad & \text{second order} \\
t \frac{d^3 x(t)}{dt^3} - \frac{dx(t)}{dt} = 0 \quad & \text{third order}
\end{aligned}
$$

Example:

$$\frac{dx(t)}{dt} - x(t) = 0$$

General or partial solution

Example:

$$\frac{dx(t)}{dt} - x(t) = 0$$

- General Solution:

$$x(t) = Ce^t$$

General or partial solution

Example:

$$\frac{dx(t)}{dt} - x(t) = 0$$

- General Solution:

$$x(t) = Ce^t$$

- Partial solutions:

$$x(t) = 2.0e^t$$
$$x(t) = 4.8e^t$$

- A homogeneous equation: the each term contains either the function or its derivative, but no other functions of independent variables:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0$$

Homogeneous and non-homogeneous ODE

- A homogeneous equation: the each term contains either the function or its derivative, but no other functions of independent variables:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0$$

- A non-homogeneous equation: contains additional term (source terms, forcing functions) which do not involve the dependent variable:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = F_0\cos(\omega t)$$

Three major categories of ODE

- Initial-value problems – involve time-dependent equations with given initial conditions:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0 \qquad x(t=0) = x_0, \quad \left.\frac{dx}{dt}\right|_{t=0} = v_0$$

Three major categories of ODE

- Initial-value problems – involve time-dependent equations with given initial conditions:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0 \qquad x(t=0) = x_0, \quad \left.\frac{dx}{dt}\right|_{t=0} = v_0$$

- Boundary-value problems – involve differential equations with specified boundary conditions:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0 \qquad x(t=a) = x_a, \qquad x(t=b) = x_b$$

Three major categories of ODE

- Initial-value problems – involve time-dependent equations with given initial conditions:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0 \qquad x(t=0) = x_0, \quad \left.\frac{dx}{dt}\right|_{t=0} = v_0$$

- Boundary-value problems – involve differential equations with specified boundary conditions:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0 \qquad x(t=a) = x_a, \qquad x(t=b) = x_b$$

- Eigenvalue problems – Involve solutions with selected parameters in the equations

Three major categories of ODE

- Initial-value problems – involve time-dependent equations with given initial conditions:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0 \qquad x(t=0) = x_0, \quad \left.\frac{dx}{dt}\right|_{t=0} = v_0$$

- Boundary-value problems – involve differential equations with specified boundary conditions:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0 \qquad x(t=a) = x_a, \qquad x(t=b) = x_b$$

- Eigenvalue problems – Involve solutions with selected parameters in the equations

In reality, a problem may have more than just one of the categories active.

- Propagation problems – are initial value problems in open domains where the initial values are marched forward in time (or space) . The order may be one or greater. The number of initial values must be equal to the order of the differential equation.

- Propagation problems – are initial value problems in open domains where the initial values are marched forward in time (or space) . The order may be one or greater. The number of initial values must be equal to the order of the differential equation.
- Equilibrium problems – are boundary-value problems in closed domains where boundary values are specified at boundaries of the solution domain. The order of ODE must be at least two.

Three general classifications in physics

- Propagation problems – are initial value problems in open domains where the initial values are marched forward in time (or space) . The order may be one or greater. The number of initial values must be equal to the order of the differential equation.
- Equilibrium problems – are boundary-value problems in closed domains where boundary values are specified at boundaries of the solution domain. The order of ODE must be at least two.
- Eigenproblems – are a special type of problems where the solution exists only for special values of a parameter.

Converting n^{th} order to n linear equations

Any n^{th} order linear differential equation can be reduced to n coupled first order differential equations. Example:

$$m\frac{d^2x(t)}{dt^2} - kx(t) = 0$$

is the same as:

$$\frac{dx(t)}{dt} = v(t)$$
$$m\frac{dv(t)}{dt} = -kx(t)$$

Initial value problems

Initial values problems are solved by marching methods using finite difference methods.

Initial value problems

Initial values problems are solved by marching methods using finite difference methods.
The objective of a finite difference method for solving an ODE is to transform a calculus problem into an algebra problem by:

Initial value problems

Initial values problems are solved by marching methods using finite difference methods.
The objective of a finite difference method for solving an ODE is to transform a calculus problem into an algebra problem by:

- Discretizing the continuous physical domain into a discrete finite difference grid

Initial value problems

Initial values problems are solved by marching methods using finite difference methods.
The objective of a finite difference method for solving an ODE is to transform a calculus problem into an algebra problem by:

- Discretizing the continuous physical domain into a discrete finite difference grid
- Approximating the exact derivatives in the ODE by algebraic finite difference approximations (FDAs)

Initial value problems

Initial values problems are solved by marching methods using finite difference methods.
The objective of a finite difference method for solving an ODE is to transform a calculus problem into an algebra problem by:

- Discretizing the continuous physical domain into a discrete finite difference grid
- Approximating the exact derivatives in the ODE by algebraic finite difference approximations (FDAs)
- Substituting the FDA into ODE to obtain an algebraic finite difference equation (FDE).

Initial values problems are solved by marching methods using finite difference methods.
The objective of a finite difference method for solving an ODE is to transform a calculus problem into an algebra problem by:

- Discretizing the continuous physical domain into a discrete finite difference grid
- Approximating the exact derivatives in the ODE by algebraic finite difference approximations (FDAs)
- Substituting the FDA into ODE to obtain an algebraic finite difference equation (FDE).
- Solving the resulting algebraic FDE

Three groups of FDs for solving initial-value ODEs

- Single point methods advance the solution from one grid point to the next grid point using only the data at a single grid point. (most significant method – 4th order Runge-Kutta.

Three groups of FDs for solving initial-value ODEs

- Single point methods advance the solution from one grid point to the next grid point using only the data at a single grid point. (most significant method – 4th order Runge-Kutta.
- Extrapolation methods evaluate the solution at a grid point for several values of grid size and extrapolate those results to get for a more accurate solution.

- Single point methods advance the solution from one grid point to the next grid point using only the data at a single grid point. (most significant method – 4th order Runge-Kutta.
- Extrapolation methods evaluate the solution at a grid point for several values of grid size and extrapolate those results to get for a more accurate solution.
- Multipoint methods advance the solution form one grid point to the next using the data at several known points (4th order Adams-Bashforth-Moulton method).

Using the Taylor series for x_{n+1} using the grid point n.

Using the Taylor series for x_{n+1} using the grid point n.

$$x_{n+1} = x_n + x'|_n \Delta t + \frac{1}{2} x''_n (\Delta t)^2 + \ldots + \frac{1}{m!} x^m|_n \Delta t^m + R^{m+1}$$

$$R^{m+1} = \frac{1}{(m+1)!} x^{m+1}(\tau) \Delta t^{m+1} \quad t \leq \tau \leq t + \Delta t$$

Finite difference approximations

Using the Taylor series for x_{n+1} using the grid point n.

$$x_{n+1} = x_n + x'|_n \Delta t + \frac{1}{2} x''_n (\Delta t)^2 + \ldots + \frac{1}{m!} x^m|_n \Delta t^m + R^{m+1}$$

$$R^{m+1} = \frac{1}{(m+1)!} x^{m+1}(\tau) \Delta t^{m+1} \quad t \le \tau \le t + \Delta t$$

Solving for $x'|_n$ yields:

$$x'|_n = \frac{x_{n+1} - x_n}{\Delta t} - \frac{1}{2} x''|_n \Delta t - \frac{1}{6} x'''|_n (\Delta t)^3$$

Finite difference approximations

Using the Taylor series for x_{n+1} using the grid point n.

- a first - order finite difference approximation:

$$x'|_n = \frac{x_{n+1} - x_n}{\Delta t} \quad \mathcal{O}(\Delta t)$$

Finite difference approximations

Using the Taylor series for x_{n+1} using the grid point n.

- a first - order finite difference approximation:

$$x'|_n = \frac{x_{n+1} - x_n}{\Delta t} \quad \mathcal{O}(\Delta t)$$

- a first - order backward-difference approximation:

$$x'|_{n+1} = \frac{x_{n+1} - x_n}{\Delta t} \quad \mathcal{O}(\Delta t)$$

Using the Taylor series for x_{n+1} using the grid point n.

- a first - order finite difference approximation:

$$x'|_n = \frac{x_{n+1} - x_n}{\Delta t} \quad \mathcal{O}(\Delta t)$$

- a first - order backward-difference approximation:

$$x'|_{n+1} = \frac{x_{n+1} - x_n}{\Delta t} \quad \mathcal{O}(\Delta t)$$

- A second-order centered difference approximation of x' at point $n + \frac{1}{2}$:

$$x'|_{n+\frac{1}{2}} = \frac{x_{n+1} - x_n}{\Delta t} \quad \mathcal{O}(\Delta t^2)$$

Finite difference equations

Consider the general first-order initial-value ODE:

$$x'(t) = f(x, t) \quad x(0) = x_0$$

Finite difference equations

Consider the general first-order initial-value ODE:

$$x'(t) = f(x,t) \quad x(0) = x_0$$

Use finite difference approximations:

$$x'|_n = \frac{x_{n+1} - x_n}{\Delta t} \qquad or \qquad x'|_{n+1} = \frac{x_{n+1} - x_n}{\Delta t}$$

Finite difference equations

Consider the general first-order initial-value ODE:

$$x'(t) = f(x,t) \quad x(0) = x_0$$

Use finite difference approximations:

$$x'|_n = \frac{x_{n+1} - x_n}{\Delta t} \qquad or \qquad x'|_{n+1} = \frac{x_{n+1} - x_n}{\Delta t}$$

Substitute into the ODE and solve for x_{n+1}:

$$x_{n+1} = x_n + f(x_n, t_n)\Delta t \qquad \text{Explicit finite difference}$$
$$x_{n+1} = x_n + f(x_{n+1}, t_{n+1})\Delta t \qquad \text{Implicit finite difference}$$

Errors – five types

- Errors in the initial data

Errors – five types

- Errors in the initial data
- Algebraic errors

Errors – five types

- Errors in the initial data
- Algebraic errors
- Truncation errors - cased by truncating the Taylor series approximation (decreases with decreasing of the step size).

Errors – five types

- Errors in the initial data
- Algebraic errors
- Truncation errors - cased by truncating the Taylor series approximation (decreases with decreasing of the step size).
- Round off errors - caused the finite word length (increases as the step size decreases: more steps and small difference between large numbers)

Errors – five types

- Errors in the initial data
- Algebraic errors
- Truncation errors - cased by truncating the Taylor series approximation (decreases with decreasing of the step size).
- Round off errors - caused the finite word length (increases as the step size decreases: more steps and small difference between large numbers)
- Inherited errors - the sum of all accumulated errors from all previous steps (means that the initial condition for the next is incorrect)

The explicit Euler method – first-order finite difference method for solving initial-value problem for ODE

The explicit Euler method for ODE

The explicit Euler method – first-order finite difference method for solving initial-value problem for ODE

Lets consider a general first-order ODE:

$$\frac{dx}{dt} = f(x,t) \quad \text{with} \quad x(t_0) = x_0$$

The explicit Euler method for ODE

The explicit Euler method – first-order finite difference method for solving initial-value problem for ODE
Lets consider a general first-order ODE:

$$\frac{dx}{dt} = f(x,t) \quad \text{with} \quad x(t_0) = x_0$$

Explicit finite difference (first order):

$$x'|_n = \frac{x_{n+1} - x_n}{\Delta t}$$

then

$$x_{n+1} = x_n + f(x_n, t_n)\Delta t$$

The explicit Euler method for ODE

- Explicit (since $f(t_n, x_n)$ does not depend on x_{n+1})

The explicit Euler method for ODE

- Explicit (since $f(t_n, x_n)$ does not depend on x_{n+1})
- Requires only one known point (singe point method)

- Explicit (since $f(t_n, x_n)$ does not depend on x_{n+1})
- Requires only one known point (singe point method)
- The local truncation error is $\mathcal{O}(\Delta t^2)$

The explicit Euler method for ODE

- Explicit (since $f(t_n, x_n)$ does not depend on x_{n+1})
- Requires only one known point (singe point method)
- The local truncation error is $\mathcal{O}(\Delta t^2)$
- The global error accumulated after n steps $\mathcal{O}(\Delta t)$

The explicit Euler method for ODE

- Explicit (since $f(t_n, x_n)$ does not depend on x_{n+1})
- Requires only one known point (singe point method)
- The local truncation error is $\mathcal{O}(\Delta t^2)$
- The global error accumulated after n steps $\mathcal{O}(\Delta t)$
- Problem: the method is conditionally stable for $\Delta t \leq \Delta t_{cr}$.

The implicit Euler method – first-order finite difference method for solving initial-value problem for ODE

The implicit Euler method for ODE

The implicit Euler method – first-order finite difference method for solving initial-value problem for ODE
Lets us chose $n+1$ as the base point for:

$$\frac{dx}{dt} = f(x,t) \quad \text{with} \quad x(t_0) = x_0$$

The implicit Euler method for ODE

The implicit Euler method – first-order finite difference method for solving initial-value problem for ODE
Lets us chose $n+1$ as the base point for:

$$\frac{dx}{dt} = f(x,t) \quad \text{with} \quad x(t_0) = x_0$$

Implicit finite difference (first order):

$$x'|_{n+1} = \frac{x_{n+1} - x_n}{\Delta t}$$

then

$$x_{n+1} = x_n + f(x_{n+1}, t_{n+1})\Delta t$$

The implicit Euler method for ODE

- Implicit (since $f(x_{n+1}, t_{n+1})$ does depend on x_{n+1})

The implicit Euler method for ODE

- Implicit (since $f(x_{n+1}, t_{n+1})$ does depend on x_{n+1})
- The implicit Euler is unconditionally stable.

The implicit Euler method for ODE

- Implicit (since $f(x_{n+1}, t_{n+1})$ does depend on x_{n+1})
- The implicit Euler is unconditionally stable.
- However, if $f(x, t)$ is non-linear, then we need to use one of the methods for solving non-linear equations.

Stability of Euler method

Consider the 'linear test equation':

$$\frac{dx(t)}{dt} = \lambda x(t)$$

where $\lambda \in \mathbb{C}$ and $x(t=0) = x_0 \neq 0$.
The exact solution of this equation is:

$$x(t) = x_0 e^{\lambda t}$$

For $\Re(\lambda) < 0$, then the solution $x(t \to \infty) \to 0$.

Stability of explicit Euler equation

$$x_{n+1} = x_n + \lambda\,\delta t\,x_n$$
$$x_{n+1} = (1 + \lambda\,\delta t)^n x_0$$

So for the case when $\Re(\lambda) < 0$:

$$|1 + \lambda\,\delta t| < 1$$

Stability of explicit Euler equation

$$x_{n+1} = x_n + \lambda\,\delta t\,x_n$$
$$x_{n+1} = (1 + \lambda\,\delta t)^n x_0$$

So for the case when $\Re(\lambda) < 0$:

$$|1 + \lambda\,\delta t| < 1$$

If we restrict that $\lambda \in \mathbb{R}$:

$$-1 < 1 + \lambda\,\delta t < 1$$
$$-2 < \lambda\,\delta t < 0$$
$$0 < \delta t < t - \frac{2}{\lambda}$$

(as $\delta t > 0$ and $\lambda < 0$).

Stability of explicit Euler equation

$$x_{n+1} = x_n + \lambda\,\delta t\,x_n$$
$$x_{n+1} = (1 + \lambda\,\delta t)^n x_0$$

So for the case when $\Re(\lambda) < 0$:

$$|1 + \lambda\,\delta t| < 1$$

If we restrict that $\lambda \in \mathbb{R}$:

$$-1 < 1 + \lambda\,\delta t < 1$$
$$-2 < \lambda\,\delta t < 0$$
$$0 < \delta t < t - \frac{2}{\lambda}$$

(as $\delta t > 0$ and $\lambda < 0$).
The condition of stability is:

$$\delta t < -\frac{2}{\lambda}$$

Stability of implicit Euler equation

$$x_{n+1} = x_n + \lambda\,\delta t\,x_{n+1}$$
$$x_{n+1} = \frac{1}{1-\lambda\,\delta t} x_n$$
$$x_{n+1} = \left(\frac{1}{1-\lambda\,\delta t}\right)^n x_0$$

So for the case when $\Re(\lambda) < 0$:

$$|\frac{1}{1-\lambda\,\delta t}| < 1$$

Stability of implicit Euler equation

$$x_{n+1} = x_n + \lambda\,\delta t\, x_{n+1}$$
$$x_{n+1} = \frac{1}{1-\lambda\,\delta t} x_n$$
$$x_{n+1} = \left(\frac{1}{1-\lambda\,\delta t}\right)^n x_0$$

So for the case when $\Re(\lambda) < 0$:

$$|\frac{1}{1-\lambda\,\delta t}| < 1$$

If we restrict that $\lambda \in \mathbb{R}$:

$$|1-\lambda\,\delta t| > 1$$
$$1-\lambda\,\delta t > 1 \text{ or } 1-\lambda\,\delta t < -1$$
$$\lambda\,\delta t < 0 \text{ or } \lambda\,\delta t > 2$$

Stability of implicit Euler equation

$$x_{n+1} = x_n + \lambda\,\delta t\, x_{n+1}$$
$$x_{n+1} = \frac{1}{1-\lambda\,\delta t} x_n$$
$$x_{n+1} = \left(\frac{1}{1-\lambda\,\delta t}\right)^n x_0$$

So for the case when $\Re(\lambda) < 0$:

$$|\frac{1}{1-\lambda\,\delta t}| < 1$$

If we restrict that $\lambda \in \mathbb{R}$:

$$|1-\lambda\,\delta t| > 1$$
$$1-\lambda\,\delta t > 1 \text{ or } 1-\lambda\,\delta t < -1$$
$$\lambda\,\delta t < 0 \text{ or } \lambda\,\delta t > 2$$

As $\delta t > 0$ and $\lambda < 0$, the condition of stability is always satified. Thus, implicit Euler method is unconditionally stable.