Chapter 5

ODE solvers

We often encounter ordinary differential equations (ODE) in physics. For example, Newton's equation is a second-order DE in time. In this chapter we will detail the important numerical methods to solve ODEs.

Explicit schemes

In most of our discussion we focus on first-order DEs since DEs of any any order can be reduced to a set of 1st order DEs. For example, $\ddot{x} = F(x,t)$ is equivalent to a set of two first-order DEs:

$$\dot{x} = v; \quad \dot{v} = F(x,t).$$

In the following discussion we will describe how to solve 1st-order DEs.

ODE Solver

$$\frac{dx}{dt} = \dot{x} = f(x,t) \qquad (1)$$

whose solution is

$$x^{(n+1)} - x^{(n)} = \int_{t_n}^{t_{n+1}} f(x,t)dt$$

where $x^{(n+1)}, x^{(n)}$ are shorthand for $x(t_{n+1})$ and $x(t_n)$ respectively. We obtain the solution by performing integration, which are

$$\int_{t^{(n)}}^{t^{(n+1)}} f(x,t)dt \approx hf(x^{(n)}, t^{(n)})$$ (2) called Forward difference

$$\approx hf(x^{(n+1)}, t^{(n+1)})$$ (3) Backward diference

$$\approx hf(x^{(n+1/2)}, t^{(n+1/2)})$$ (4) Midpoint rule.

$$\approx \frac{1}{2}h\left[f(x^{(n)}, t^{(n)}) + f(x^{(n+1)}, t^{(n+1)})\right]$$ (5) Trapezoid rule

When we use three points, we obtain

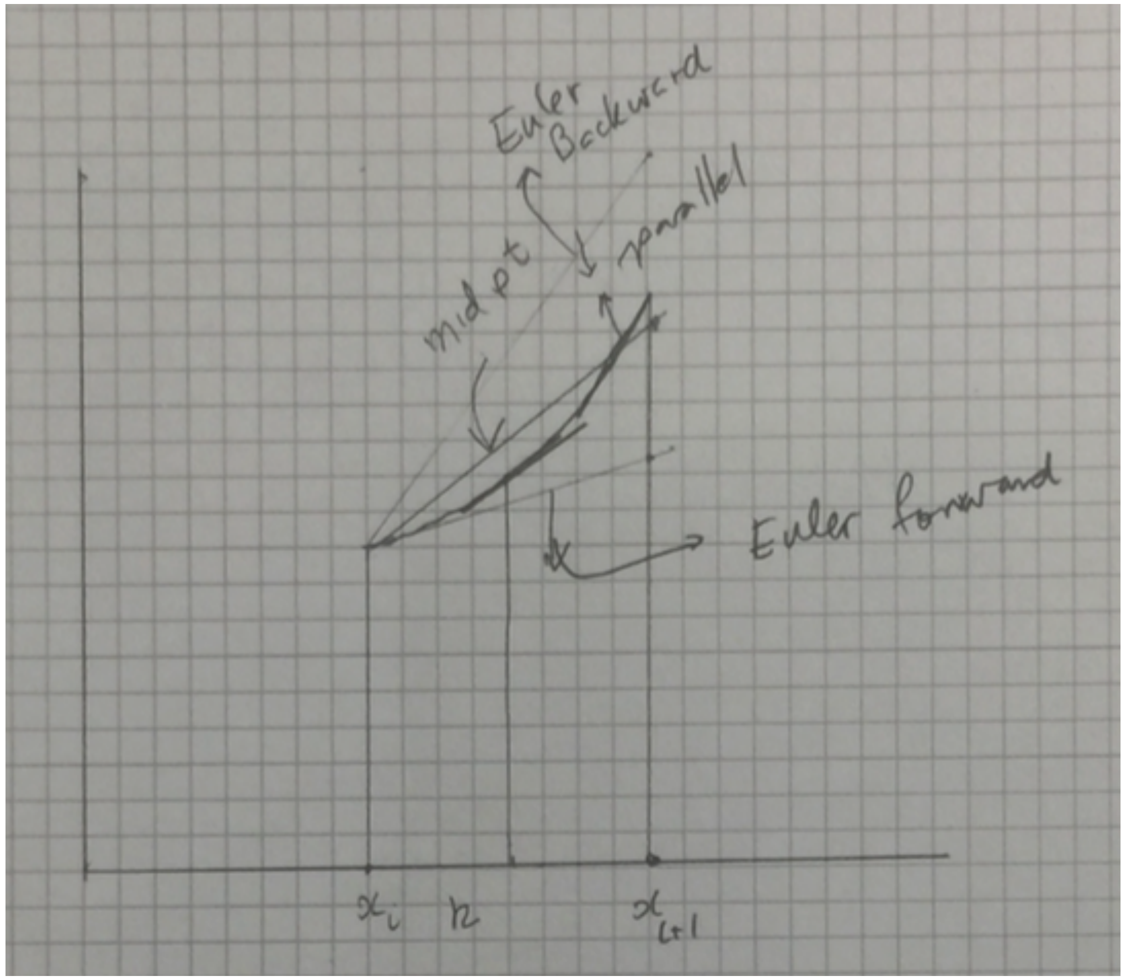
$$x^{(n+1)} - x^{(n-1)} \approx \frac{1}{3}h\left[f(x^{(n-1)}, t^{(n-1)}) + 4f(x^{(n)}, t^{(n)}) + f(x^{(n+1)}, t^{(n+1)})\right] \qquad (6)$$

which is called the Simpson rule. Note that the solution $x(t_{n+1})$ here depends on $x(t_n)$ and $x(t_{n-1})$, hence it is called a multi-step method.

Also note that the Forward difference scheme is explicit, and rest all are implicit or semi-implicit.

In the following figure we illustrate Euler's forward and backward schemes, as well as central-difference scheme. Clearly, the central difference scheme is more accurate.

Two major issues in the solution of DEs are accuracy and stability. We will discuss accuracy first.



Accuracy

How much is the error in a given scheme. Firstly we consider the forward difference scheme, which is also called Euler's Scheme

$$x^{(n+1)} = x^{(n)} + hf(x^{(n)}, t^{(n)})$$

According to the Taylor's series. the solution of Eq. (1) is

$$x^{(n+1)} = x^{(n)} + h\dot{x} + \frac{h^2}{2}\ddot{x} + \ldots$$

$$= x^{(n)} + hf(x^{(n)}, t_n) + \frac{h^2}{2}\frac{df}{dt} + \ldots$$

with $\dfrac{df}{dt} = \dfrac{\partial f}{\partial t} + \dot{x}\dfrac{\partial f}{\partial x}$

Clearly Euler's method has

Error per step $= \dfrac{1}{2}h^2\ddot{x}(x^{(n)}, t^{(n)})$

Error in n steps if the errors are additive $= \dfrac{1}{2}nh\ddot{x}(x^{(n)}, t^{(n)})$,

which is possible if df/dt has the same sign at all the steps.

since $nh = t_{final}$, hence the error is proportional to *t*, which is not good for accuracy.

Example: Consider a DE $\dot{x} = \alpha x$ whose exact solution is $x(t) = x(0)e^{\alpha t}$.

In one step, the exact solution yields

$$x^{(n+1)} = x^{(n)}\exp(\alpha h)$$

However Euler's scheme yields

$$x^{(n+1)} = x^{(n)}(1 + \alpha h)$$

For small *h*,

$$\exp(\alpha h) = 1 + \alpha h + \frac{1}{2}(\alpha h)^2 + \ldots.$$

So the error is $\dfrac{1}{2}(\alpha h)^2 x^{(n)}$ consistent with the earlier formula.

The important consideration for the DEs is stablity. The notion of stability of DEs differs from usual definition of stability of dynamical systems.

In most of our discussion on DEs, we consider $\dot{x} = \alpha x$ as a prime example. Note that Eq. (1) reduces to the above near the zero of f(x).

Proof: If the concerned zero of f(x) is $x=x_0$, then near $x=x_0$, Eq. (1) becomes

$$\dot{x}' = f'(x_0, t)x'$$

where x'= x-x. Hence studying $\dot{x} = \alpha x$ is useful for studying general equation o type Eq. (1).

Stability

We again consider $\dot{x} = \alpha x$. For $\alpha > 0$, the exact and numerical solution grow. So the chief concerns for such equations are related to the accuracy.

For $\alpha < 0$, the exact solution converges to zero. However if $|1 + \alpha h| > 1$, x_{n+1} oscillates and $|x_{n+1}|$ grows with n, contrary to the exact solution. Due to this observation, the system is said to be unstable.

A Method is *stable* if it produces a bounded solution when the solution of the DE is bounded; otherwise it is *unstable*.

Conditionally stable: If the system is stable for some set of parameters, and unstable for some other set of parameters.

Unconditionally stable: Stable for all parameter values

Unconditionally unstable: not stable for any parameter.

For complex α, the region of stability is $|1 + \alpha h| < 1$ as illustrated in the figure given below:

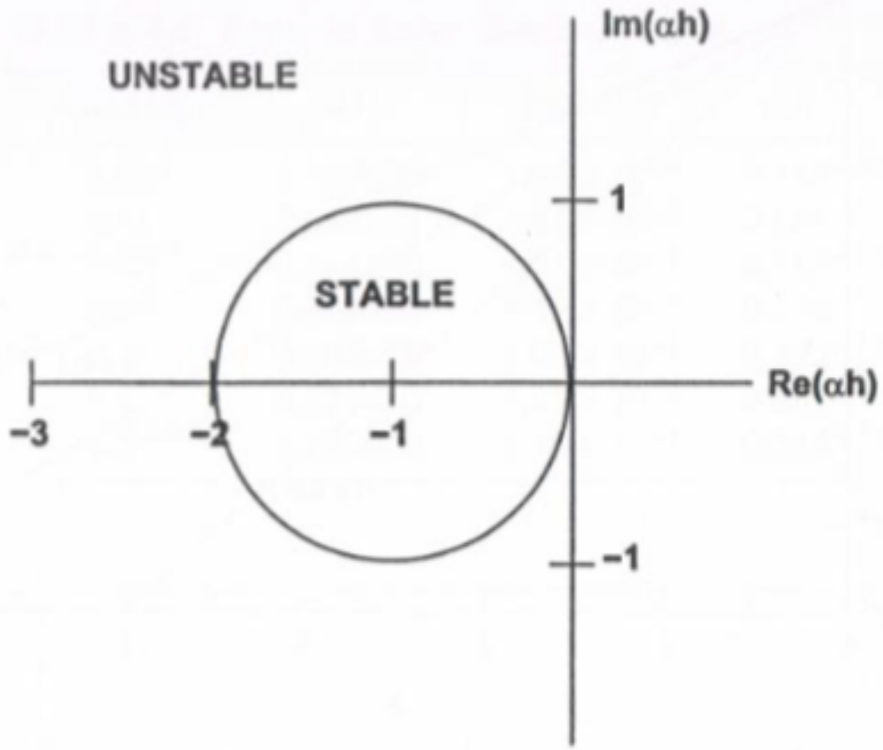


FIGURE 4.3. Stable and unstable regions for the Euler metho

Examples:

(1) $\dot{x} = -2x \quad x(t) = x(0)e^{-2t}$

Euler's scheme yields condition for stability as $|1 - 2h| < 1$, or $-1 < 1 - 2h < 1$, or *h*<1. Hence the system is stable for *h*<1 and unstable for h>1. Hence the system is said to be **conditionally stable**.

(2) $\dot{z} = iz \quad z(t) = z(0)e^{it}$. Hence for the exact solution shows oscillations, and $|z(t)|^2 = |z(0)|^2$. Note that the magnitude and phase of $\exp(-i\pi t)$ are 1 and $-\pi t$ respectively.

In one time step of size h, $z(t) = z(0)e^{ih}$, hence the amplitude and phase of e^{ih} are 1 and h respectively. However, for the Euler's scheme

$$z^{(1)} = x^{(0)}(1 + ih)$$

The amplitude and phase of (1+ih) are $\sqrt{1+h^2}$ and $\tan^{-1} h$ respectively that differs from those of the exact solution. Hence the system is said to **unconditionally unstable**.

Solve the above problems in terms of real and imaginary parts.

$$z = x + iy$$

$$\dot{x} = -y; \quad \dot{y} = x$$

or

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}$$

The difference equations are

$$x^{(n+1)} = x^{(n)} - hy^{(n)}$$

$$y^{(n+1)} = y^{(n)} + hx^{(n)}$$

$$\begin{pmatrix} x^{(n+1)} \\ y^{(n+1)} \end{pmatrix} = \begin{pmatrix} 1 & = h \\ h & 1 \end{pmatrix}\begin{pmatrix} x^{(n)} \\ y^{(n)} \end{pmatrix}$$

The above matrix has Tr = 2, and det = $1 + h^2$

Hence the eigenvalues of the matrix are $(1 \pm h)$. The solution grows along the eigen direction of 1+h. Hence, the system is unconditionally unstable, consistent with the arguments given above.

(3) DEs with variable coefficients: $\dot{x} = -2xt = f(x, t)$ whose solution is $x(t) = \exp(-t^2)$ that converges to $x(t) \to 0$ as $t \to \infty$. Euler's scheme yields $x^{(n+1)} = x^{(n)}(1 - 2ht^{(n)})$. Hence the solution becomes unstable when $|1 - 2ht^{(n)}| > 1$. That is the solution may be stable at early times, but it becomes unstable later.

Here, locally we should write $x^{(n+1)} = x^{(n)}(1 + h(\partial f/\partial x)_n)$.

(4) Nonlinear equation: $\dot{x} = -2x^2 t = f(x, t)$ has an exact solution $x(t) = x_0/[1 + \alpha x_0 t^2]$ that converges to $x(t) \to 0$ as $t \to \infty$. Euler's scheme yields

$$x^{(n+1)} = x^{(n)}(1 - 2ht^{(n)}[x^{(n)}]^2)$$

Though t_n increases with *n*, but $x^{(n)}$ decreases faster than $t^{(n)}$. Hence the $|1 - 2ht^{(n)}[x^{(n)}]^2| < 1$ for small *h* at all times. Therefore, the system is unconditional stable at all times.

Implicit Schemes

Backward or implicit Euler Method

$x^{(n+1)} = x^{(n)} + hf(x^{(n+1)}, t^{(n+1)})$

Here $f(x^{(n+1)}, t^{(n+1)})$ is to be computed $t = t^{(n+1)}$ with $x^{(n+1)}$ which is unknown. Hence the solution is in terms of itself. This is the reason it is called an implicit scheme.

This scheme is stable for the following reasons.

Let's take the earlier example: $\dot{x} = \alpha x$.

In the backward of implicit Euler's method

$$x^{(n+1)} = x^{(n)} + \alpha h x^{(n+1)}$$

$$x^{(n+1)} = \frac{1}{1 - \alpha h} x^{(n)}$$

Since $\frac{1}{|1 - \alpha h|} < 1$ for negative α, the system is unconditionally stable. In addition, $\frac{1}{|1 - i\pi h|} < 1$, hence the oscillatory solution of $\dot{z} = \alpha z$ is stable.

Let us analyze the accuracy of the above scheme:

$$x^{(n+1)} = \frac{1}{(1 - \alpha h)} x^{(n)}$$

$$= 1 + \alpha h + \alpha^2 h^2 + \ldots$$

Hence, for $\alpha < 0$, the error $= \frac{1}{2}(\alpha h)^2$ to the leading order.

For nonlinear equations like $\dot{x} = -tx^3$, the implicit Euler's scheme yields

$$x^{(n+1)} = x^{(n)} - ht^{(n+1)}(x^{(n+1)})^3$$

which is solved using iterative procedure, e.g. using Newton-Raphson method (to be discussed later).

How to test convergence

$h \to h/2$ see if error between the consecutive steps is decreasing.. If yes, the solution is converging.

Trapezoid Rule

In this scheme

$$x^{(n+1)} - x^{(n)} = \frac{h}{2}\left[f(x^{(n)}, t^{(n)}) + f(x^{(n+1)}, t^{(n+1)})\right]$$

which is a combination of explicit and implicit schemes, hence it is called semi-implicit method as well.

For the example, $\dot{x} = \alpha x$

$$x^{(n+1)} = x^{(n)} + \frac{h}{2}\alpha[x^{(n)} + x^{(n+1)}]$$

Therefore,

$$x^{(n+1)} = \left(\frac{1 + \alpha h/2}{1 - \alpha h/2}\right) x^{(n)}$$

$$\approx \left(1 + \alpha h + \frac{1}{2}(\alpha h)^2 + \frac{1}{4}(\alpha h)^3 + \dots\right) x^{(n)}$$

Hence this scheme is accurate up to O(h^2), and the error is O(h^3).

For the equation, $\dot{z} = iz$,

Euler's implicit scheme yields

$$z^{(n+1)} = \left(\frac{1 + ih/2}{1 - ih/2}\right) z^{(n)} = Az^{(n)}$$

for which $|A| = 1$, which is consistent with the exact solution. Note however that the phase is $2\tan^{-1}(\pi h/2)$ that differs from the exact value of h.

Section 3

Predictor-Corrector

Other Approaches

Consider $\dot{x} = f(x,t)$.

If the higher-order derivatives are easy to compute, then

$$\ddot{x} = \frac{\partial f}{\partial t} + \dot{x}\frac{\partial f}{\partial x}$$

Therefore

$$x^{(n+1)} = x^{(n)} + h\dot{x} + \frac{h^2}{2}\ddot{x} + HOT$$

$$= x^{(n)} + \frac{h}{2}\left[2\dot{x} + h\frac{\partial f}{\partial t} + nh\dot{x}\frac{\partial f}{\partial x}\right]$$

which will be accurate up to O(h^2).

Predictor-Corrector (PC)

Implicit Scheme based on Trapezoid rule is

$$x^{(n+1)} = x^{(n)} + \frac{h}{2}\left[f(x^{(n)}, t^{(n)}) + f(x^{(n+1)}, t^{(n+1)})\right]$$

Since x$^{(n+1)}$ is unknown, the computation of $f(x^{(n+1)}, t^{(n+1)})$ is difficult. An easier way is to

$$x^{(n+1)*} = x^{(n)} + hf(x^{(n)}, t^{(n)}) \quad \text{[Predictor step]}$$

$$x^{(n+1)} = x^{(n)} + \frac{h}{2}\left[f(x^{(n)}, t^{(n)}) + f(x^{(n+1)*}, t^{(n+1)})\right] \quad \text{[Corrector]}$$

The above scheme is accurate up to O(h^2).

Proof using example $\dot{x} = \alpha x$. The predictor-corrector scheme yields

$$x^{(n+1)} = x^{(n)} + \frac{h}{2}[\alpha x^{(n)} + \alpha x^{(n+1)}]$$

$$= x^{(n)} + \frac{h}{2}[\alpha x^{(n)} + \alpha x^{(n)} + \alpha h x^{(n)}]$$

$$= \left[1 + \alpha h + \frac{\alpha h^2}{2}\right] x^{(n)} + HOT$$

Hence PC scheme is accurate up to 2nd order**.**

Runge-Kutta 2nd order (RK2)

This is another PC scheme in which

$$x^{(n+1/2)} = x^{(n)} + \frac{1}{2}hf(x^{(n)}, t^{(n)})$$

$$x^{(n+1)} = x^{(n)} + hf(x^{(n+1/2)}, t^{(n+1/2)})$$

This scheme has the same accuracy as the predictor-corrector method, which is easy to show using Taylor's expansion.

Runge-Kutta 4th order scheme (RK4)

[Euler half-step predictor]

$$x^{(n+1/2)*} = x^{(n)} + \frac{h}{2} f(x^{(n)}, t^{(n)})$$

[Backward Euler half-step corrector]

$$x^{(n+1/2)**} = x^{(n)} + \frac{h}{2} f(x^{(n+1/2)*}, t^{(n+1/2)})$$

[Mid point rule, full-step predictor]

$$x^{(n+1)***} = y^{(n)} + hf\left(x^{(n+1/2)**}, t^{(n+1/2)}\right)$$

[Simpson rule full-step corrector]

$$y_{n+1} = x^{(n)} + \frac{h}{6}\left[f(x^{(n)}, t^{(n)}) + 2f(x^{(n+1/2)*}, t^{(n+1/2)})\right]$$

$$+\frac{h}{6}\left[2f(x^{(n+1/2)**}, t^{(n+1/2)} + f(x^{(n+1)***}, t^{(n+1)})\right]$$

This scheme is 4th order accurate. Error = O(h^5).

Multistep method

These schemes are not covered in detail, but they are very useful.

Here $x^{(n+1)}$ not only depends on $x^{(n)}$, but also on the values at earlier times: $x^{(n-1)}$, $x^{(n-2)}$. This is the reason why it is called multistep method.

Example: Leap-frog method for which

$$x^{(n+1)} = x^{(n-1)} + 2hf(x^{(n)}, t^{(n)}).$$

For our usual example $\dot{x} = \alpha x$, Leap-frog yields

$x^{(n+1)} = x^{(n-1)} + 2\alpha h x^{(n)}$. (1)

Clearly the solution is of the form ρ^n, whose substitution in the above yields

$$\rho^{n+1} = \rho^{n-1} + 2\alpha h \rho^n$$

$$\Rightarrow \quad \rho^2 = 1 + 2\alpha h \rho$$

$$\Rightarrow \rho^2 - 2\alpha h \rho - 1 = 0$$

whose solutions are

$$\rho_{1,2} = \frac{2\alpha h \pm \sqrt{(4\alpha h)^2 + 4}}{2} = \alpha h \pm \sqrt{(\alpha h)^2 + 1}$$

Hence

$$\rho_1 \approx 1 + \alpha h + \frac{1}{2}(\alpha h)^2 + HOT$$

$$\rho_2 \approx -1 + \alpha h - \frac{1}{2}(\alpha h)^2$$

Clearly,

$$x^{(n)} = a_1 \rho_1^n + a_2 \rho_2^n$$

It is easy to see that ρ_2 is a spurious solution. If $\rho_2 = 0$, the $x^{(n+1)}/x^{(n)} = \rho_1$, which should be compared with its exact counterpart $\exp(h)$. Note that $\exp(h)$ and ρ_1 match to O(h^2), hence leap-frog scheme is second-order accurate. But this is achievable only if a_2 is set to zero. This is one of the challenges of this scheme.

Multistep method

Advantage: (a) For the same accuracy, 1/2 the calculation compared to single-step method. Compare with RK2.

(b) Leap-frog method is symmetric under time reversal. To go from $x^{(n+1)}$ to $x^{(n-1)}$, the scheme will be

$$x^{(n-1)} = x^{(n+1)} - 2\alpha h x^{(n)}$$

which is same as Eq. (1). This is not the case for Euler's or RK2 method.

Disadvantage: Need to store more variables, e.g., leap-frog method requires storage of $x^{(n)}$ and $x^{(n-1)}$. Starting at t=0 is also requires a different method, e.g. Euler's method.

Other popular mutlistep methods are Adams-Bashforth and Adams-Moulton methods. The Adams-Bashforth method is written as

$$x^{(n+1)} = x^{(n)} + h\sum_{m=0}^{k} \beta_m f(x^{(n+1-m)}, t^{(n+1-m)})$$

with

	m=1	m=2	m=3	m=4	m=5
β_{1m}	1				
$2\beta_{2m}$	3	-1			
$12\beta_{3m}$	23	-16	5		
$24\beta_{4m}$	55	-59	37	-9	
$720\beta_{4m}$	1901	-2774	2616	-1274	251

As illustrations, the two lowest-order Adams-Bashforth schemes are

$$x^{(n+1)} = x^{(n)} + hf(x^{(n)}, t^{(n)})$$

$$x^{(n+1)} = x^{(n)} + \frac{h}{2}[3f(x^{(n)}, t^{(n)}) - f(x^{(n-1)}, t^{(n-1)})]$$

The Adams-Moulton method is written as

$$x^{(n+1)} = x^{(n)} + h\sum_{m=0}^{k} \beta_m f(x^{(n+1-m)}, t^{(n+1-m)})$$

with

	m=0	m=1	m=2	m=3	m=4
β_{1m}	1				
$2\beta_{2m}$	1	1			
$12\beta_{3m}$	5	8	-1		
$24\beta_{4m}$	9	19	-5	1	
$720\beta_{4m}$	251	646	-264	106	-19

As illustrations, the three lowest-order Adams-Moulton schemes are

$$x^{(n+1)} = x^{(n)} + hf(x^{(n+1)}, t^{(n+1)})$$

$$x^{(n+1)} = x^{(n)} + \frac{h}{2}[f(x^{(n+1)}, t^{(n+1)}) + f(x^{(n)}, t^{(n)})]$$

$$x^{(n+1)} = x^{(n)} + \frac{h}{12}[5f(x^{(n+1)}, t^{(n+1)}) + 8f(x^{(n)}, t^{(n)}) - 8f(x^{(n-1)}, t^{(n-1)})]$$

Section 5

System of equation, Stiff equations

Set of Equations

m equations

$$\dot{x}_i = f(t, x_0, x_1, \ldots, x_{m-1})$$

Numerical scheme:

Explicit scheme

$$x_i^{(n+1)} = x_i^{(n)} + hf(t, x_0^{(n)}, x_1^{(n)}, \ldots, x_{m-1}^{(n)})$$

Time step *n* equations

Implicit scheme:

$$x_i^{(n+1)} = x_i^{(n)} + hf(t, x_0^{(n+1)}, x_1^{(n+1)}, \ldots, x_{m-1}^{(n+1)})$$

solve iteratively.

For linear equations, matrix equation.

Application to Mechanics

Equation of motion of a particle in 1D

$$m\ddot{x} = f(x, \dot{x}, t)$$

Convert it to 2 first-order ODEs

$$m\dot{x} = p$$

$$\dot{p} = f(x, p/m, t)$$

Now apply the appropriate integration scheme.

Stiff equations

$$\dot{u} + au = v; \quad \dot{v} = -v$$

whose exact solutions are

$$v(t) = v(0)\exp(-t)$$

$$u(t) = c_1 \exp(-at) + \frac{1}{a}\exp(-t)$$

Two different time scales: 1/a, 1

If $a \gg 1$. Two very different time scales 1/a, 1.

Time stepping a problem

Euler explicit scheme:

$$\begin{pmatrix} u^{(n+1)} \\ v^{(n+1)} \end{pmatrix} = \begin{bmatrix} 1 - ah & h \\ 0 & 1 - h \end{bmatrix} \begin{bmatrix} u^{(n)} \\ v^{(n)} \end{bmatrix}$$

whose eigenvalues are $1 - ah; 1 - h$.

Stability condition (assuming a>1):

$$ah < 2$$

Suppose $a = 100$, then $h < 2/100 = 0.02$.

For large a, h tends to be very small.

For initial condition (u,v) = (2,1).

$$v(t) = \exp(-t)$$

$$u(t) = (2 - \frac{1}{a})\exp(-at) + \frac{1}{a}\exp(-t)$$

Note that both u,v>0 for all t.

Semi implicit scheme like Crank Nicolson method can cure this problem.

$$u^{(n+1)} = u^{(n)} - ah\left(\frac{u^{(n)} + u^{(n+1)}}{2}\right) + hv^{(n)}$$

$$v^{(n+1)} = v^{(n)} - hv^{(n)}$$

Stable method since

$$u^{(n+1)} = \left(\frac{1 - ah/2}{1 + ah/2}\right) u^{(n)} + hv^{(n)}$$

$$v^{(n+1)} = (1 - h)v^{(n)}$$

The above scheme is stable for *a*>0.

Second method: Eliminate by a change of variable

$$\tilde{u} = \exp(at)u$$

The equations transform to

$$\dot{\tilde{u}} = v\exp(-at)$$

Not stiff any more.

Leap frog method, Verlet method for dynamics

$\dot{x} = v$

$\dot{v} = F(x)$

Time step

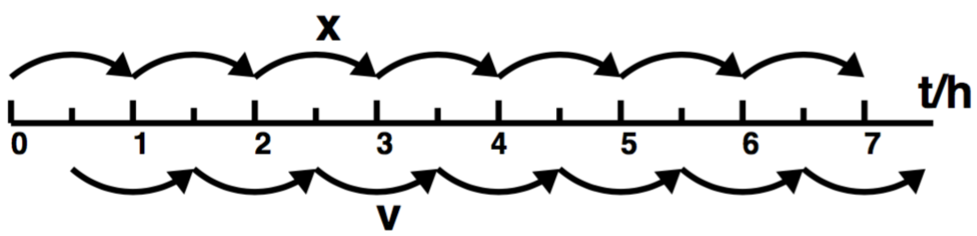
$x_{n+1} = x_n + hv_{n+1/2}$

$v_{n+3/2} = v_{n+1/2} + hF(x_{n+1})$

Note

$x_n = x_{n-1} + hv_{n-1/2}$ (1)

$v_{n+1/2} = v_{n-1/2} + hF(x_n)$ (2)



Init condition: (x_0, v_0)

Euler first step: $v_{1/2} = v_0 + \frac{h}{2}F(x_0)$

This scheme is time reversal symmetric

Under time reversal

$x_{n-1} = x_n - hv_{n-1/2}$

$v_{n-1/2} = v_{n+1/2} - hF(x_n)$

That are same as Eqs. (1,2).

Exercise

1. Solve simple oscillator problem ($\ddot{x}$=-x) using Euler and RK2 scheme for h=0.1,0.01,0.001. Analyze the error, and verify the error law. Go over a time period and see if you get energy conservation to a reasonable accuracy.
2. Solve the oscillator equation using Euler Backward scheme.
3. Solve $\ddot{x} = -x + x^3 + \sin(2\,t)$ using RK2 method for different initial conditions.
4. Solve Newon's equation for a pendulum with a massless string of length l and bob of mass m. Plot the angle and angular velocity as a function of time. Also make the phase space plot. It is better to use non-dimensional equation.
5. Solve Lorenz equation numerically:

$$\dot{x} = P(y - x)$$

$$\dot{y} = x(r - z) - y$$

$$\dot{z} = xy - \beta z$$

Make 3D plots of (x,y,z) for three sets of parameters: (1) P=10, r=0.5, β=1; (2) Pr=10, r=2, β=1; (3) Pr=10, r=28, β=1.