§1.6. Methods of Derivations

- There are several different ways of deriving a given discrete method.

- We consider several different ways to derive the forward Euler's method.

(1) Taylor series method

Taylor series expansion of $u(t^{n+1}) = u(t^n + \Delta t)$ around t^n:

$$u(t^{n+1}) = u(t^n) + \Delta t\, u'(t^n) + \frac{\Delta t^2}{2} u''(t^n) + H.O.T$$

: higher
... terms

Ignoring terms higher than $O(\Delta t)$, we obtain

$$u(t^{n+1}) \approx u(t^n) + \Delta t\, u'(t^n) = u(t^n) + \Delta t\, f(t^n, u^n)$$

$\Rightarrow$ The approximation is given by

$$v^{n+1} = v^n + \Delta t\, f(t^n, v^n).$$

<u>Rmk</u>. We see that if we keep the expansion up to $O(\Delta t^p)$, then the method becomes a p^{th}-order accurate method.

The limitation doing so for $p > 1$ is, although straightforward, is that we are only given

$$u'(t) = f(t, u(t)),$$

and we must compute the higher order derivatives by repeated differentiation of the function, that is,

$$u''(t) = f_u(t,u)\,u' + f_t(t,u)$$

$$= f_u(t,u)\,f(t,u) + f_t(t,u),$$

and so on until we get $u^{(p)}(t)$.

This becomes very messy and impractical.

(2) Finite difference approximation

If we replace the derivative $u'(t^n)$ by a first-order forward difference approximation;

$$u'(t^n) \approx \frac{v^{n+1} - v^n}{\Delta t},$$

then this gives the forward Euler's method.

(3) Numerical Quadrature

If we integrate $u'(t) = f(t, u(t))$ over $[t^n, t^{n+1}]$, we get

$$u(t^{n+1}) - u(t^n) = \int_{t^n}^{t^{n+1}} f(t; u(t))\, dt$$

Now using a left-hand rectangular quadrature rule, $\int_a^b g(t)\, dt \approx (b-a)\, g(a)$,

We get $\int_{t^n}^{t^{n+1}} f(t, u(t))dt \approx \Delta t\, f(t^n, u(t^n))$

Hence we get the approximation,

$$v^{n+1} - v^n = \Delta t\, f(t^n, v^n),$$

Which is again the forward Euler's method.

(4) Method of undetermined coefficients

At $t = t^n$, we know two things:

$$\begin{cases} u(t^n) , \\ u'(t^n) \doteq f(t^n, u^n) \end{cases}$$

The idea is to linearly combine them for the next prediction, $u(t^{n+1})$; that is to say,

$$u^{n+1} = \alpha u^n + \beta u'(t^n) \quad --- \quad ①$$

We now force ① in such a way that the first two monomials

$$1, t, t^2, t^3, \ldots$$

are exactly predicted using ① in order to determine α & β.

⇒ (i) For the first monomial, $u(t) = 1$,
① becomes:

$$1 = \alpha \cdot 1 + \beta \cdot 0 \quad \Rightarrow \quad \boxed{\alpha = 1}.$$

(ii) For the second monomial, $u(t) = t$,
① becomes

$$t^{n+1} = 1 \cdot t^n + \beta \cdot 1 \Rightarrow \beta = t^{n+1} - t^n = \Delta t$$

⇒ ① finally becomes $u^{n+1} = u^n + \Delta t \, u'(t^n)$, or
$$v^{n+1} = v^n + \Delta t \, f(t^n, v^n).$$

(5) Polynomial Interpolation fitting.

Given the two pairs of values:

$$\begin{cases} (t^n, u(t^n)), \\ (t^n, u'(t^n)), \end{cases}$$

we try to fit the two data points using a polynomial.

Since there are two data points, we can use a polynomial of degree 1 to fit them.

That is, consider a 1st degree poly in t:

$$p(t) = \alpha + \beta t.$$

We now determine α & β s.t.

$$\begin{cases} p(t^n) = u^n, \quad \text{and} \\ p'(t^n) = u'(t^n) = f(t^n, u^n) \end{cases}$$

$$\Rightarrow \quad \begin{cases} p(t^n) = \alpha + \beta t^n = u^n \\ p'(t^n) = \beta = f(t^n, u^n) \end{cases}$$

$$\Rightarrow \quad \begin{cases} \beta = f(t^n, u^n) \\ \alpha = u^n - t^n f(t^n, u^n). \end{cases}$$

$$\Rightarrow \quad p(t) = (u^n - t^n f(t^n, u^n)) + t f(t^n, u^n).$$

$\Rightarrow$ We now use this to predict at t^{n+1}:

$$p(t^{n+1}) = u^n - t^n f(t^n, u^n) + t^{n+1} f(t^n, u^n)$$

$$= u^n + (t^{n+1} - t^n) f(t^n, u^n)$$

$$= u^n + \Delta t\, f(t^n, u^n).$$

$\Rightarrow$ The approximation becomes

$$v^{n+1} = v^n + \Delta t\, f(t^n, v^n).$$

<u>Rmk</u> In general, this type of polynomial that interpolates

$$\begin{cases} \text{the function, and} \\ \text{their derivatives up to } p\text{th order,} \end{cases}$$

is called the <u>Hermite interpolation</u>