12.5. **Linear multistep methods.** The general linear $(p+1)$ step method has the form

$$y_{n+1} = \sum_{i=0}^{p} a_i y_{n-i} + h \sum_{i=-1}^{p} b_i f_{n-i},$$

where $f_{n-i} = f(x_{n-i}, y_{n-i})$ and the a_i and b_i are constants.

Remarks: Any of the a_is and b_is may be zero, but we assume either a_{n-p} or b_{n-p} is not zero (otherwise the method would not be a $p+1$ step method).

If $b_{-1} = 0$, then y_{n+1} is expressed as a linear combination of (computationally) known past values $y_n, \ldots, y_{n-p}$ and thus is easily computed. Such formulas are called *explicit* or forward integration formulas. If $b_{-1} \neq 0$, then the formula is an implicit equation for y_{n+1}, since y_{n+1} also appears on the right hand side. Such formulas are called *implicit* and must be solved by an iterative procedure.

The methods are called linear because values of f_{n-i} enter linearly. We do not assume that f_{n-i} is a linear function of y_{n-i}.

12.5.1. *Derivation.* One way of deriving such formulas is through numerical integration. Since $y' = f(x, y(x))$, we have

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} f(x, y(x))\, dx.$$

If we approximate the right hand side using the trapezoidal rule, i.e., replace f by a linear interpolating polynomial and integrate, we get

$$\begin{aligned} y(x_{n+1}) - y(x_n) &= (h/2)[f(x_{n+1}, y(x_{n+1})) + f(x_n, y(x_n))] - (h^3/12) f''(\xi, y(\xi)) \\ &= (h/2)[f(x_{n+1}, y(x_{n+1})) + f(x_n, y(x_n))] - (h^3/12) y'''(\xi). \end{aligned}$$

Omitting the error term, the resulting approximation scheme is:

$$y_{n+1} = y_n + (h/2)[f_{n+1} + f_n].$$

More generally, we can obtain linear multistep methods by replacing $f(x, y(x))$ by its interpolating polynomial using the points $x_n, x_{n-1}, \ldots, x_{n-p}$ and then integrating between x_{n-j} and x_{n+1}.

Recall that the Newton form of the interpolating polynomial interpolating $g(x)$ at the points $x_n, x_{n-1}, \ldots, x_{n-p}$ is given by

$$\begin{aligned} P(x) = \sum_{k=0}^{p} g[x_n, \ldots, x_{n-k}] \prod_{i=0}^{k-1} (x - x_{n-i}) &= g(x_n) + g[x_n, x_{n-1}](x - x_n) \\ + g[x_n, x_{n-1}, x_{n-2}](x - x_n)(x - x_{n-1}) &+ \ldots + g[x_n, \ldots, x_{n-p}](x - x_n) \cdots (x - x_{n-p+1}). \end{aligned}$$

In the above, we set $\prod_{j=0}^{-1} = 1$. Setting $g(x) = f(x, y(x))$, we get

$$\begin{aligned} y(x_{n+1}) - y(x_{n-j}) &= \int_{x_{n-j}}^{x_{n+1}} y'(x)\,dx = \int_{x_{n-j}}^{x_{n+1}} f(x, y(x))\,dx \\ &\approx \sum_{k=0}^{p} g[x_n, \ldots, x_{n-k}] \int_{x_{n-j}}^{x_{n+1}} \prod_{i=0}^{k-1} (x - x_{n-i})\,dx. \end{aligned}$$

We now examine some specific cases. For simplicity, we consider the case of equally spaced points $x_{i+1} - x_i = h$.

$$\begin{aligned} p = 0: &\qquad y_{n+1} = y_{n-j} + (j+1)hf_n \\ j = 0: &\qquad y_{n+1} = y_n + hf_n \quad \text{(Euler's method)} \\ j = 1: &\qquad y_{n+1} = y_{n-1} + 2hf_n \quad \text{(Midpoint rule)} \end{aligned}$$

When $p = 1$, we get

$$\begin{aligned} y_{n+1} &= y_{n-j} + h(j+1)f_n + f[x_n, x_{n-1}] \int_{x_{n-j}}^{x_{n+1}} (x - x_n)\,dx \\ &= y_{n-j} + h(j+1)f_n + (1/2)f[x_{n-1}, x_n](h^2 - j^2h^2) \\ &= y_{n-j} + hf_n[1 + j + (1/2)(1 - j^2)] + hf_{n-1}(1/2)(j^2 - 1). \end{aligned}$$

Hence, we obtain the methods:

$$\begin{aligned} j = 0: &\qquad y_{n+1} = y_n + (3/2)hf_n - (1/2)hf_{n-1}, \\ j = 1: &\qquad y_{n+1} = y_{n-1} + 2hf_n, \\ j = 2: &\qquad y_{n+1} = y_{n-2} + (3/2)hf_n + (3/2)hf_{n-1}. \end{aligned}$$

All these methods are explicit methods. To get implicit methods, we use the interpolating polynomial based on the points $x_{n+1}, x_n, x_{n-1}, \ldots, x_{n-p+1}$ and repeat this procedure.

Multistep methods can also be derived by using Taylor series expansions. In fact, we have already seen how Euler's method can be derived in this way. To get the midpoint rule, we use the expansions

$$\begin{aligned} y(x_n + h) &= y(x_n) + hy'(x_n) + (h^2/2)y''(x_n) + (h^3/6)y'''(\xi_n^+), \\ y(x_n - h) &= y(x_n) - hy'(x_n) + (h^2/2)y''(x_n) - (h^3/6)y'''(\xi_n^-). \end{aligned}$$

Subtracting these equations, we obtain:

$$y(x_n + h) - y(x_n - h) = 2hy'(x_n) + (h^3/6)y'''(\xi_n^+) + (h^3/6)y'''(\xi_n^-).$$

Neglecting higher order terms, we get the midpoint rule $y_{n+1} = y_{n-1} + 2hf(x_n, y_n)$.

12.5.2. *Order, consistency, error constant, and local truncation error.* Associated with a given linear multistep method, we define a linear difference operator $\mathcal{L}$ by

$$\mathcal{L}[y(x);h] = y(x+h) - \sum_{i=0}^{p} a_i y(x-ih) - h\sum_{i=-1}^{p} b_i y'(x-ih).$$

Expanding $y(x+ih)$ and $y'(x+ih)$ in a Taylor series about x and collecting terms gives

$$\mathcal{L}[y(x);h] = C_0 y(x) + C_1 h y'(x) + \cdots + C_q h^q y^{(q)}(x) + \cdots,$$

where the C_i are constants. More specifically, we have

$$\mathcal{L}[y(x);h] = [1 - \sum_{i=0}^{p} a_i] y(x) + \sum_j y^{(j)}(x)\left[\frac{h^j}{j!} - \sum_{i=0}^{p} a_i(-i)^j \frac{h^j}{j!} - \sum_{i=-1}^{p} b_i (-i)^{j-1}\frac{h^j}{(j-1)!}\right].$$

Hence,

$$C_0 = 1 - \sum_{i=0}^{p} a_i, \qquad C_1 = 1 - \sum_{i=0}^{p} a_i(-i) - \sum_{i=-1}^{p} b_i,$$

$$C_j = \frac{1}{j!}\left[1 - \sum_{i=0}^{p} a_i(-i)^j - j\sum_{i=-1}^{p} b_i(-i)^{j-1}\right].$$

Definition: The linear difference operator $\mathcal{L}$ and associated linear multistep method are said to be of order r if $C_0 = C_1 = \cdots = C_r = 0$ and $C_{r+1} \neq 0$. Then C_{r+1} is called the error constant.

Note that $\mathcal{L}$ will have order r if and only if the multistep method is exact for all polynomials of degree $\leq r$, but not for polynomials of degree $r+1$, i.e., $\mathcal{L}[x^q;h] = 0$, $q = 0,\ldots,r$, $\mathcal{L}[x^{r+1};h] \neq 0$. Now

$$\begin{aligned}\mathcal{L}[x^q;h] &= \sum_{j=0}^{q} C_j \frac{d^j}{dx^j}(x^q)h^j \\ &= C_q q! h^q + \cdots + C_j q(q-1)\cdots(q-j+1)x^{q-j}h^{q-1} + \cdots + C_1 q x^{q-1} h + C_0 x^q.\end{aligned}$$

Hence if $\mathcal{L}[x^q;h] = 0$, $q = 0,\ldots,r$, then $q = 0$ implies $C_0 = 0$, $q = 1$ implies $C_1 = 0$, and finally $q = r$ implies $C_r = 0$.

Definition: A linear multistep method is consistent if it has order $r \geq 1$, i.e., if it is exact for linear polynomials, i.e., if

$$\sum_{i=0}^{p} a_i = 1, \qquad \sum_{i=0}^{p} a_i(-i) + \sum_{i=-1}^{p} b_i = 1.$$

If a linear multistep method has order r, then

$$\mathcal{L}[x^{r+1};h] = C_{r+1}(r+1)!h^{r+1}, \quad \text{so} \quad C_{r+1} = \mathcal{L}[x^{r+1};h]/[(r+1)!h^{r+1}].$$

Definition: The local truncation error in going from x_n to x_{n+1} is defined to be $\mathcal{L}[y(x_n);h]$.

If the method is of order r, then

$$\mathcal{L}[y(x_n);h] = C_{r+1}h^{r+1}y^{(r+1)}(x_n) + O(h^{r+2}).$$

The first term on the right side of the equation above is called the principle local truncation error.

To compute the local truncation error for methods defined by numerical integration, we recall that using the error formula for polynomial interpolation and again setting $g(x) = f(x, y(x))$, we get

$$y(x_{n+1}) - y(x_{n-j}) = \int_{x_{n-j}}^{x_{n+1}} y'(x)\,dx = \int_{x_{n-j}}^{x_{n+1}} f(x, y(x))\,dx$$

$$= \sum_{k=0}^{p} g[x_n, \ldots, x_{n-k}] \int_{x_{n-j}}^{x_{n+1}} \prod_{i=0}^{k-1}(x - x_{n-i})\,dx + \int_{x_{n-j}}^{x_{n+1}} g[x_n, \ldots, x_{n-p}, x] \prod_{i=0}^{p}(x - x_{n-i})\,dx.$$

Hence,

$$\mathcal{L}[y(x_n);h] = \int_{x_{n-j}}^{x_{n+1}} g[x_n, \ldots, x_{n-p}, x] \prod_{i=0}^{p}(x - x_{n-i})\,dx.$$

Example: Consider the case $j = 0$. Then

$$\prod_{i=0}^{p}(x - x_{n-i}) = (x - x_n)\cdots(x - x_{n-p}) \geq 0$$

for $x \in [x_n, x_{n+1}]$. Hence

$$\mathcal{L}[y(x_n);h] = g[x_n, \ldots, x_{n-p}, \xi] \int_{x_{n-j}}^{x_{n+1}} \prod_{i=0}^{p}(x - x_{n-i})\,dx$$

$$= \frac{g^{(p+1)}(\xi)}{(p+1)!} \int_{x_{n-j}}^{x_{n+1}} \prod_{i=0}^{p}(x - x_{n-i})\,dx = \frac{y^{(p+2)}(\xi)}{(p+1)!} \int_{x_{n-j}}^{x_{n+1}} \prod_{i=0}^{p}(x - x_{n-i})\,dx.$$

When $p = 0$, $\mathcal{L}[y(x_n);h] = h^2 y^{(2)}(\xi)/2$. This is Euler's method. When $p = 1$, we get

$$\mathcal{L}[y(x_n);h] = \frac{y^{(3)}(\xi)}{2!} \int_{x_n}^{x_{n+1}} (x - x_n)(x - x_{n-1})\,dx$$

$$= \frac{y^{(3)}(\xi)}{2!} \int_{x_n}^{x_{n+1}} [(x - x_n)^2 + h(x - x_n)]\,dx = \frac{5h^3}{12} y^{(3)}(\xi).$$