Ch. 3 Runge-Kutta Methods

Key Points:

- Quadrature (Newton-Cotes, Gaussian, ...)
  - Weights, nodes
  - Orthogonal polynomials (Jacobi, Legendre, Chebyshev, ...)

Runge-Kutta Schemes:

- RK matrix
- RK weights
- RK nodes
- Stages

Can we tie this back to ideas used in multistep case even though these will be single-step methods

(i.e., interpolation used there for \( \int f(t) dt \)?)
Numerical Quadrature (see also Heath, Ch. B)

We've talked a little about ways to approximate an integral

\[ I(f) \approx \int_a^b f(t) \, dt \]

**Left endpoint**

\[ \int_a^b f(t) \, dt \approx (b-a) \cdot f(a) \]

**Right endpoint**

\[ \int_a^b f(t) \, dt \approx (b-a) \cdot f(b) \]

**Midpoint rule**

\[ \int_a^b f(t) \, dt \approx (b-a) \cdot f \left( \frac{a+b}{2} \right) \]

**Trapezoid rule**

\[ \int_a^b f(t) \, dt \approx \frac{(b-a)}{2} \left[ f(a) + f(b) \right] \]

These are exact for the case where \( f(t) \) is a polynomial of degree 0 (i.e. a constant function).

... but midpoint is exact also for a polynomial of degree 1.

This is exact if the function is linear (polynomial of degree one).

What about more general methods?
Here is an example — **Simpson's Method**

Approximate the integral

\[ I(f) = \int_a^b f(t) \, dt \]

using three equally-spaced points

(3-pt. quadrature rule)

so

\[ \int_a^b f(t) \, dt \approx w_1 f_1 + w_2 f_2 + w_3 f_3 \]

where the weights \( w_1, w_2 \) and \( w_3 \) are chosen so that the quadrature rule integrates the first 3 polynomial basis functions exactly. We'll use monomial basis functions \((1, t, t^2)\) in this example.

\[ f(t) = 1 : \quad \int_a^b 1 \, dt = (b-a) = w_1 \cdot 1 + w_2 \cdot 1 + w_3 \cdot 1 \]

\[ f(t) = t : \quad \int_a^b t \, dt = \frac{1}{2} t^2 \bigg|_a^b = \frac{1}{2} (b^2-a^2) = w_1 a + w_2 \left( \frac{a+b}{2} \right) + w_3 \cdot b \]

\[ f(t) = t^2 : \quad \int_a^b t^2 \, dt = \frac{1}{3} t^3 \bigg|_a^b = \frac{1}{3} (b^3-a^3) = w_1 a^2 + w_2 \left( \frac{a+b}{2} \right)^2 + w_3 \cdot b^2 \]
This leads to the linear system

\[
\begin{bmatrix}
1 & 1 & 1 \\
\frac{a+b}{2} & b \\
\frac{(a+b)^2}{4} & b^2
\end{bmatrix}
\begin{bmatrix}
w_1 \\
w_2 \\
w_3
\end{bmatrix}
= 
\begin{bmatrix}
b-a \\
\frac{1}{2}(b^2-a^2) \\
\frac{1}{3}(b^3-a^3)
\end{bmatrix}
\]

Solving for the weights gives

\[w_1 = \frac{1}{2}(b-a),\quad w_2 = \frac{2}{3}(b-a),\quad w_3 = \frac{1}{6}(b-a)\]

This is Simpson’s Quadrature Rule.

(Order 3) For discussion of degree/order see p. 49.2 in notes.
Consider the numerical quadrature case where we approximate the integral
\[ I(f) = \int_a^b f(t) \, dt \]
by the n-point quadrature rule
\[ I(f) = Q_n(f) = \sum_{i=1}^{n} w_i f(t_i) \]
where \( a < t_1 < t_2 < \ldots < t_n < b \). The points \( t_i \) are called the nodes and the coefficients \( w_i \) are called the weights.

**Plan:** Choose the nodes and weights for optimal performance (accuracy, computational cost, stability, ...).

Consider first the case where the nodes are specified (possibly equally-spaced but not necessarily).

Given \( t_1, t_2, \ldots, t_n \) find \( w_1, w_2, \ldots, w_n \) so that the quadrature rule integrates the first \( n \) basis functions exactly. Again, we'll use the monomial basis functions here \( 1, t, t^2, \ldots, t^{n-1} \).

(As we shall note below, this gives a quadrature rule of order \( n \).)
This leads to the linear system

\[
\begin{bmatrix}
1 & 1 & \cdots & 1 \\
t_1 & t_2 & \cdots & t_n \\
t_1^{n-1} & t_2^{n-1} & \cdots & t_n^{n-1}
\end{bmatrix}
\begin{bmatrix}
w_1 \\
w_2 \\
w_n
\end{bmatrix}
= 
\begin{bmatrix}
b-a \\
\frac{b-a}{2} \sum t_i^2 \left( \frac{1}{2} \right) (b-a)^2 \\
\frac{b-a}{n} \sum t_i^{n-1} (b-a)^n
\end{bmatrix}
\]

Vandermonde Matrix

- this nonsingular (although possibly not well-conditioned if \( n \) is large) can be solved to uniquely determine the weights \( w_1, w_2, \ldots, w_n \).
An alternative way to view this same problem is in terms of the Lagrangian basis functions. Basically, what we just did was to fit \( n \) points to a polynomial of degree \( n-1 \) and then integrated that polynomial. So in terms of the Lagrangian basis functions

\[
l_j(t) = \frac{\prod_{k=1, k \neq j}^{n} (t-t_k)}{\prod_{k=1}^{n} (t_j-t_k)} \quad j = 1, 2, \ldots, n
\]

So

\[
f(t) \approx p(t) = f_1 l_1(t) + f_2 l_2(t) + \ldots + f_n l_n(t)
\]

Note: \( l_j(t_i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \quad f_j = f(t_j) \)

So

\[
\int_{a}^{b} f(t) \, dt \approx \int_{a}^{b} p(t) \, dt = \sum_{j=1}^{n} f_j \int_{a}^{b} l_j(t) \, dt
\]

That is,

\[
w_j = \int_{a}^{b} l_j(t) \, dt \quad j = 1, 2, \ldots, n
\]
EX (Simpson's Rule)

\[ l_1(t) = \frac{(t-m)(t-b)}{(a-m)(a-b)} \]

\[ l_2(t) = \frac{(t-a)(t-b)}{(m-a)(m-b)} \]

\[ l_3(t) = \frac{(t-a)(t-m)}{(b-a)(b-m)} \]

... after some effort ...

\[ W_1 = \int_a^b l_1(t) \, dt = \frac{1}{6} (b-a) \]

\[ W_2 = \int_a^b l_2(t) \, dt = \frac{2}{3} (b-a) \]

\[ W_3 = \int_a^b l_3(t) \, dt = \frac{1}{6} (b-a) \]

\[ \int_a^b f(t) \, dt \approx W_1 f_1 + W_2 f_2 + W_3 f_3 \]
Error in Quadrature Rules  

\[ I(f) = \int_a^b f(t) \, dt \quad \text{E}(f) = I(f) - Q_n(f) \]

\[ Q_n(f) = \sum_{i=1}^n w_i f(t_i) \]

Suppose the quadrature rule is exact for a polynomial of degree \( n-1 \). That is,

\[ Q_n(f) = I(P_{n-1}) \]

Then

\[ |E(f)| = |I(f) - Q_n(f)| \]

\[ = |I(f) - I(P_{n-1})| \]

\[ = |I(f - P_{n-1})| \]

\[ \leq (b-a) \| f - P_{n-1} \|_{\text{sup}} \]

\[ \leq \frac{(b-a)h^n}{4n} \| f^{(n)} \|_{\text{sup}} \]

\[ \leq \frac{1}{4} h^{n+1} \| f^{(n)} \|_{\text{sup}} \]

\[ h = \frac{b-a}{n} \]

Def: A quadrature rule is order \( n \) if it is exact for every polynomial of degree \( n-1 \) but not exact for a polynomial of degree \( n \).
Note: (p. 344)

Heath defines the degree of a quadrature rule...

**Def:** A quadrature rule is said to be of degree \( d \) if it is exact for every polynomial of degree \( d \) but is not exact for some polynomial of degree \( d + 1 \).

I.e., the largest positive integer \( n \) such that the formula is exact for \( f \) of degree at least \( n - 1 \).

**Observe:**

- The Midpoint Rule is a 1-pt. interpolatory quadrature rule. Note that it gets both constant functions (polynomial of degree 0) and linear functions (polynomial of degree 1) exact [Degree 1].

- The Trapezoid Rule is a 2-pt. interpolatory quadrature rule. It gets constant and linear functions exact. It does not get quadratic functions exact. [Degree 1]

Iserles refers to these as **Order 2**. ["Order" = "Degree" + 1]
(see Heath, p. 324)

Note: If a polynomial of degree $n-1$, $P_{n-1}(t)$, interpolates a sufficiently smooth function $f(t)$ at $t = t_1, t_2, \ldots, t_n$ distinct points, then

$$f(t) - P_{n-1}(t) = \frac{f^{(n)}(\theta)}{n!} (t-t_1)(t-t_2) \cdots (t-t_n)$$

where $\theta \in [t_i, t_{i+1}]$.

If $|f^{(n)}(t)| < M$ for $t \in [t_1, t_n]$ and $h = \max \{ |t_{i+1} - t_i| \mid i = 1, 2, \ldots, n-1 \}$ then

$$\|f(t) - P_{n-1}(t)\|_{\sup} \leq \frac{Mh^n}{4n}$$

$$\|1\|_{\sup} = \sup \{ |1| \mid t \in [t_1, t_n] \} \quad \text{least upper bound}$$

---

See also Isersels result p. 33 (written here for weighting function $w(t) = 1$).

$$\left| \int_a^b f(t) dt - \sum_{i=1}^n w_i f(t_i) \right| \leq C \max_{a \leq t \leq b} |f^{(n)}(t)|$$

(Proved using Peano kernel theorem)
Summarizing so far... (Iserles Lemma 3.1)

Lemma 3.1

Given any distinct set of nodes $t_1, t_2, \ldots, t_n$, it is possible to find a set of weights $w_1, w_2, \ldots, w_n$ such that

$$
\int_a^b f(t) \, dt \approx \sum_{i=1}^{n} w_i f(t_i)
$$

is of order $p \geq n$. (so degree $\geq n-1$)

Proof: this uses the construction we've already examined using basis functions $\{1, t, t^2, \ldots, t^{n-1}\}$, Vandermonde matrix, etc.

Remark:

- Again recall the example of the midpoint rule, that is an interpolatory quadrature rule (1-pt) (i.e. $n=1$) but has order 2 (i.e. $p=2$, $n=1$) (degree 1)

\[ p \geq n. \]

- Simpson's Rule, a 3-pt interpolatory quadrature rule, (i.e. $n=3$) has order 4 (degree 3) (i.e. $p=4$, $n=3$)

- Trapezoid Rule, a 2-pt interpolatory quadrature rule, (i.e. $n=2$) has order 2 (so here $p=2$, $n=2$)
Error in Midpoint Rule

\[ I(f) = \int_{a}^{b} f(t) \, dt \]

\[ Q_{\text{mid}}(f) = (b-a) f(m) \quad m = \frac{a+b}{2} = h \]

\[ E(f) = I(f) - Q_{\text{mid}}(f) = \int_{a}^{b} f(t) \, dt - (b-a) f(m) \]

Let us expand \( f(t) \) in a Taylor series about \( t = m \)

\[ f(t) = f(m) + f'(m)(t-m) + \frac{1}{2} f''(m)(t-m)^2 + \frac{1}{3!} f'''(m)(t-m)^3 + \frac{1}{4!} f^{(4)}(m)(t-m)^4 + \cdots \]

\[ \int_{a}^{b} f(t) \, dt = f(m)(b-a) + f'(m) \left( \int_{a}^{b} (t-m) \, dt \right) + \frac{1}{2} f''(m) \int_{a}^{b} (t-m)^2 \, dt + \frac{1}{3!} f'''(m) \int_{a}^{b} (t-m)^3 \, dt + \frac{1}{4!} f^{(4)}(m) \int_{a}^{b} (t-m)^4 \, dt + \cdots \]

\[ = f(m)(b-a) + \frac{1}{2} f''(m) \left[ \frac{1}{3} (b-m)^3 - \frac{1}{3} (a-m)^3 \right] + \frac{f^{(4)}(m)}{4!} \frac{1}{5} (b-m)^5 \]

\[ = f(m)(b-a) + \frac{1}{2} f''(m) \left[ \frac{1}{3} h^3 - \frac{1}{3} (ch)^3 \right] + O(h^5) \]

\[ = f(m)(b-a) + \frac{1}{3} h^3 f''(m) + O(h^5) \]
So

\[ E(f) = I(f) - Q_{\text{mid}}(f) \]

\[ = \frac{1}{3} h^3 f''(m) + O(h^5) \]

---

this formula suggests that \( E = 0 \) (method is exact)
for constant and linear functions \( f \) (for which \( f'' = 0 \)).

(i.e. midpoint \( \rightarrow \) degree 1, order 2)

- A similar calculation can be made (see Burden & Faires)
to show for Simpson's Rule

\[ E(f) = I(f) - Q_{\text{simp}}(f) = h^5 f^{(4)}(\theta) \]

Simpson (3-pt quad.)
is exact for polynomials
of degree 0, 1, 2, 3.

(i.e. Simpson's Rule \( \rightarrow \) degree 3, order 4)

- Similarly, Trapezoid Rule has \( (\text{Burden} + \text{Faires}) \)

\[ E(f) = I(f) - Q_{\text{trap}}(f) = h^3 f''(\theta) \]

\( \text{Interpolating polynomial is degree 1} \)

\( \text{Interpolating polynomial is degree 2} \)
Stability in Quadrature Rules

Let \( Q_n(f) = \sum_{i=1}^{n} w_i f(t_i) \)

be some quadrature rule \((n\text{-pt.})\).

If there is a perturbation in the function \( f \),

\[
|Q_n(f) - Q_n(\hat{f})| = \left| \sum_{i=1}^{n} w_i \hat{f}(t_i) - \sum_{i=1}^{n} w_i f(t_i) \right|
\]

\[
= \left| \sum_{i=1}^{n} w_i (\hat{f}(t_i) - f(t_i)) \right|
\]

\[
\leq \sum_{i=1}^{n} |w_i| |\hat{f}(t_i) - f(t_i)|
\]

\[
\leq \left( \sum_{i=1}^{n} |w_i| \right) \| \hat{f} - f \|_{\text{sup}}
\]

For stability \((\text{i.e., output relatively insensitive to input})\),

we hope for \( \sum_{i=1}^{n} |w_i| \) to not be large.

Note: if \( w_i \)'s are all positive then \( \sum_{i=1}^{n} w_i \) and we know from the Vandermonde matrix \((\text{row 1 e.g.})\) that \( \sum_{i=1}^{n} w_i = (b-a)^n \).

\( \sum_{i=1}^{n} w_i \) is as good as you can expect

as this is the condition\# for integration. \( \text{(see Heath, p. 341)} \)
If the $w_i$'s have different signs then the quantity $\sum_{i=1}^{n} |w_i|$ could be considerably larger than $b-a$.

Types of Quadrature

- **Newton-Cotes Quadrature**:
  - Here the nodes are equally spaced on the interval $[a, b]$.
  - These can be "closed"

  \[ h = \frac{b-a}{n-1} \]

  e.g. Trapezoid — 2pt. closed N-C.
  Simpson — 3pt. closed N-C

  or "open"

  \[ h = \frac{(b-a)}{(n+1)} \]

  e.g. Midpoint Rule — 1pt. open N-C
Further comments on Newton-Cotes

- Polynomial "wiggle" becomes a problem as \( n \) increases

- In fact, according to Heath (p. 350)
  \[
  \sum_{i=1}^{n} |w_i| \to \infty \text{ as } n \to \infty
  \]
  indicating loss of stability.

- From an error point of view, old # of points (\( n \)) seems better than even.

  Go clue: maybe a careful choice of nodes can make things better!

Some of these issues can be addressed using quadrature rules involving

- Orthogonal polynomials

- Gaussian quadrature
Orthogonal Polynomials

Suppose \( p \) and \( q \) are polynomial functions. Define the inner product of \( p \) and \( q \) with respect to a weight function \( w(t) \) (nonnegative):

\[
\langle p, q \rangle = \int_a^b w(t)p(t)q(t)\,dt
\]

with \( \int_a^b t^j w(t)\,dt < \infty \) for \( j = 0, 1, 2, \ldots \) and \( \int_a^b w(t)\,dt > 0 \).

\( p \) and \( q \) are orthogonal if \( \langle p, q \rangle = 0 \)

with respect to the weighting function \( w(t) \).

A set of polynomials \( \{ p_j \} \) is orthonormal if

\[
\langle p_i, p_j \rangle = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}
\]

EX

Legendre Polynomials \( w(t) = 1 \) on \([-1, 1]\).

- \( P_0(t) = 1 \)
- \( P_1(t) = t \)
- \( P_2(t) = \frac{1}{2}(3t^2 - 1) \)
- \( P_3(t) = \frac{1}{2}(5t^3 - 3t) \)

Recurrence

\[
(k+1)P_{k+1}(t) = (2k+1)tP_k(t) - kP_{k-1}(t)
\]
Orthogonal Polynomials \( \{P_0(t), P_1(t), P_2(t), P_3(t), \ldots \} \)

EX: Legendre \([-1, 1] \quad w(t) = 1 \quad P_n(1) = 1 \)

\( P_0(t) = A = 1 \)

\( P_1(t) = A + B \)

\[ 0 = \int_{-1}^{1} (A + B) 1 \cdot 1 \, dt = \frac{1}{2} A + B \]

\[ (\frac{1}{2} A + B) - (\frac{1}{2} A - B) = 2B \quad B = 0 \]

\( P_1(t) = A = 1 \)

\( P_2(t) = A + B + C \)

\[ 0 = \int_{-1}^{1} (A + B + C) \cdot 1 \, dt = \frac{1}{3} A + \frac{1}{2} B + \frac{1}{2} C \]

\[ (\frac{1}{3} A + \frac{1}{2} B + C) - (\frac{1}{3} A + \frac{1}{2} B - C) = \frac{2}{3} A + 2C \]

\[ 0 = \frac{2}{3} A + 2C \]

\( t: 0 = \int_{-1}^{1} (A + B + C) t \, dt = \frac{1}{4} A + \frac{3}{4} B + \frac{1}{2} C \]

\[ (\frac{1}{4} A + \frac{3}{4} B + C) - (\frac{1}{4} A + \frac{3}{4} B - C) = \frac{2}{3} B \]

\[ B = 0 \]

\( P_2(t) = 1 = A + C \quad A = -3C \)

\( 1 = -2C \quad C = -\frac{1}{2} \quad A = \frac{3}{2} \)

\[ P_2(t) = \frac{3}{2} t^2 - \frac{1}{2} = \frac{1}{2} (3t^2 - 1) = P_2(t) \]
So in general, we can generate \( P_k(t) \) by requiring

\[
0 = \sum_{i=1}^{j} t^i P_k(t) \, dt 
\]

for \( j = 0, 1, 2, 3, \ldots, k-1 \). 

7th degree is a polynomial.

and \( P_k(1) = 1 \) (i.e., \( P_k \) is a monic or

weight function \( w(t) = 1 \) to

all lower degree polynomials)

... but there is also a recurrence relation:

\[
(k+1) P_{k+1}(t) = (2k+1) t P_k(t) - k P_{k-1}(t)
\]

so for example since we already have

\[
P_0(t) = 1
\]

\[
P_1(t) = t
\]

\[
P_2(t) = \frac{1}{2}(3t^2 - 1)
\]

\[
3 \cdot P_3(t) = 5t P_2(t) - 2P_1(t)
\]

\[
= 5t \left( \frac{1}{2}(3t^2 - 1) \right) - 2t
\]

\[
= \frac{5}{2} \cdot (3t^2 - t) - 2t
\]

\[
P_3(t) = \frac{5}{2} t^3 - \frac{3}{2} t
\]
More generally, \((\text{with } p_m \neq 0)\) \(p_m \in \mathbb{P}_m\) (\(\mathbb{P}_m = \text{set of all real polynomials of degree } m\))

If \(\langle p_m, p \rangle = 0\) for every \(p \in \mathbb{P}_{m-1}\),

This \(p_m\) is orthogonal to all lower degree polynomials.

\(\text{EX}\)

Note: \(\langle p_2, p_j \rangle = 0\) on \([-1, 1]\) w.r.t. \(w(t) = 1\), \(j = 0, 1\).

\[\int_{-1}^{1} \frac{1}{2} (3t^2 - 1) t \, dt = \frac{1}{2} \int_{-1}^{1} (3t^3 - t) \, dt = 0\]

Odd function

\[\int_{-1}^{1} \frac{1}{2} (3t^2 - 1) \cdot |t| \, dt = \frac{1}{2} \int_{-1}^{1} (t^3 - t) \, dt = \frac{1}{2} (0 - 0) = 0\]

As noted on previous pages.

See comment on uniqueness of monic polynomials orthogonal polynomials.

(Iserles, p. 35)
EX Chebyshev

\[ W(t) = \frac{1}{(1-t^2)^{1/2}} \] on \([-1,1]\)

\[
\begin{align*}
T_0(t) &= 1 \\
T_1(t) &= t \\
T_2(t) &= 2t^2 - 1 \\
T_3(t) &= 4t^3 - 3t
\end{align*}
\]

Recurrence:

\[ T_{k+1}(t) = 2tT_k(t) - T_{k-1}(t) \]

Also,

\[ T_k(t) = \cos \left( k \arccos(t) \right) \]

Zeros at:

\[ -z_i = \cos \left( \frac{(2i-1)\pi}{2k} \right) \quad i = 1, 2, \ldots, k \]

The zeros tend to bunch up near the ends -1 and 1.

How are these related to Gaussian Quadrature?
Gaussian Quadrature

Basic idea: Previously, we started with a given set of nodes \( t_j \) and then selected the weights \( w_j \) in order to maximize the number of polynomials (maximize the degree) that we could represent exactly. If we used an \( n \)-pt quadrature rule we were, in general, able to get a quadrature method of degree \( n-1 \) (maybe \( n \) in certain cases).

Here, for Gaussian Quadrature, we shall use the nodes as unknowns as well and require that use the requirements that additional polynomial degrees be exactly represented to determine the weights + nodes.

**EX** (Heath, p.351)

\[ I(f) = \int f(t) dt \approx w_1 f(t_1) + w_2 f(t_2) \]

2-pt quadrature rule.

Treat \( w_1, w_2 \) and \( t_1, t_2 \) as "to be determined"

(using \( w(t) = 1 \))
Require

\[ p(t) = 1 \quad \int_1^t dt = w_1 f(t_1) + w_2 f(t_2) \]

\[ 2 = w_1 \cdot 1 + w_2 \cdot 1 \]

\[ p(t) = t \quad \int_1^t dt = w_1 f(t_1) + w_2 f(t_2) \]

\[ 0 = w_1 t_1 + w_2 t_2 \]

\[ p(t) = t^2 \quad \int_1^t t^2 dt = w_1 f(t_1) + w_2 f(t_2) \]

\[ \frac{2}{3} = w_1 t_1^2 + w_2 t_2^2 \]

\[ p(t) = t^3 \quad \int_1^t t^3 dt = w_1 f(t_1) + w_2 f(t_2) \]

\[ 0 = w_1 t_1^3 + w_2 t_2^3 \]

\[ \Rightarrow \begin{cases} t_1 = -\frac{1}{\sqrt[3]{3}} & w_1 = 1 \\ t_2 = \frac{1}{\sqrt[3]{3}} & w_2 = 1 \end{cases} \]

\[ n = 2 \text{ pts.} \]

\[ \text{i.e. } 2n-1 \]

In general, this method is degree 3 (order 4)

since 1, t, t^2, t^3 are integrated exactly.
Before connecting orthogonal polynomials to Gaussian Quadrature, let's go back and generalize slightly our definition of quadrature ... introducing a weight function \( w(t) \)

\[
(\star) \quad \int_a^b w(t)f(t)\,dt \approx \sum_{i=1}^{n} W_i f(t_i)
\]

where

\[
0 < \int_a^b w(t)\,dt < \infty, \quad \left| \int_a^b t^j w(t)\,dt \right| < \infty \quad j=1,2,\ldots
\]

Here's the main Gaussian Quadrature Theorem

**Thm 3.3** (orthogonal polynomial)

Let \( t_1, t_2, \ldots, t_n \) be the zeros of \( p_n \in P_n \)

(set of all polynomials of degree \( n \)), and let \( W_1, W_2, \ldots, W_n \)

be the solution to the Vandermonde system

\[
\sum_{j=1}^{n} W_j t^m = \int_a^b t^m w(t)\,dt \quad m=0,1,2,\ldots,n-1
\]

(requiring the quadrature rule be exact for polynomials \( 1, t, t^2, \ldots, t^{n-1} \)). Then

(i) The quadrature method \((\star)\) is of order \( 2n \) ("degree" \( 2n-1 \))

(ii) No other quadrature \((n-1 pt.)\) can exceed this order.
Proof of (i)

Idea: Show that the coefficients (weights) \( w_j \) and the nodes \( t_j \) are such that the quadrature rule

\[ Q_n(f) = \sum_{j=1}^{n} w_j f(t_j) \]

is exact for polynomials \( \hat{\phi} \in P_{2n-1} \) (i.e., \( \int_{a}^{b} \hat{\phi}(t)w(t)dt = \sum_{j=1}^{n} w_j \hat{\phi}(t_j) \)) (not just \( P_{n-1} \))

Let \( \hat{\phi} \in P_{2n-1} \). Note that \( \hat{\phi} \) can be expressed in terms of "quotient" and "remainder":

\[ \hat{\phi} = P_n \phi + r \]

where \( P_n \) is an orthonormal polynomial of degree \( n \) and \( \phi, r \) are polynomials of degree \( n-1 \). Consider

\[ \int_{a}^{b} \hat{\phi}(t)w(t)dt = \int_{a}^{b} P_n(t)\phi(t)w(t)dt + \int_{a}^{b} r(t)w(t)dt \]

\[ = \langle P_n, \phi \rangle + \int_{a}^{b} r(t)w(t)dt \]

since \( P_n \) is an \( n^{th} \) orthonormal polynomial and \( \phi \in \mathbb{P}_{n-1} \).

Also

\[ \sum_{j=1}^{n} w_j \hat{\phi}(t_j) = \sum_{j=1}^{n} w_j P_n(t_j)\phi(t_j) + \sum_{j=1}^{n} w_j r(t_j) = \sum_{j=1}^{n} w_j \phi(t_j) \]

\( P_n(t_j) = 0 \) by definition of \( t_j \).
We have already seen (Lemma 3.1), given nodes \( t_1, t_2, \ldots, t_n \) it is possible to choose weights \( w_j \) (\( j = 1, \ldots, n \)) such that \( i + r \in P_{n-1} \)

\[
\int_a^b w(t) r(t) \, dt = \sum_{j=1}^n w_j r(t_j)
\]

So, choose the weights \( w_j \) based on these conditions. Then

\[
\int_a^b w(t) \hat{r}(t) \, dt = \int_a^b w(t) r(t) \, dt = \sum_{j=1}^n w_j r(t_j) = \sum_{j=1}^n w_j \hat{r}(t_j)
\]

That is, our choice of \( t_j \) and \( w_j \) allow our quadrature scheme to integrate exactly any polynomial of degree \( 2n-1 \). (i.e. it works for \( \hat{f} \in P_{2n-1} \)). Therefore, the order of the quadrature method \( (*) \) is \( 2n \) (degree \( 2n-1 \)).

\[ \Rightarrow \] So with a carefully selected set of nodes, we basically double degree of the quadrature rule.
Proof of (ii)

Suppose the method integrates \( \hat{f} \in P_{2n} \) exactly.

In particular, consider the polynomial

\[
\hat{p}(t) = \prod_{i=1}^{n} (t - t_i)^2 > 0.
\]

(\( so \ \hat{p}(t_i) = 0 \) and \( \hat{p}(t) > 0 \)). Note

\[
\sum_{j=1}^{n} w_j \hat{p}(t_j) = 0
\]

but, recalling that \( w(t) \) is non-negative

\[
\int_{a}^{b} w(t) \hat{p}(t) dt > 0
\]

(Note, this integral could be zero only if \( w(t) = \begin{cases} 0 & t = t_i; \\ \ast0 & \text{elsewhere} \end{cases} \))

But this function — indicator function — would have \( \int_{a}^{b} w(t) dt = 0 \).

Therefore, it is not possible that

\[
\int_{a}^{b} w(t) \hat{p}(t) dt = \sum_{j=1}^{n} w_j \hat{p}(t_j)
\]

So, the method does not integrate \( \hat{f} \in P_{2n} \) exactly.

Therefore, the quadrature rule is order \( 2n \) (degree \( 2n - 1 \)).
A generalization of this Thm is...

**Thm 3.4 (Isaacs, p. 37)**

Let \( r \in \mathbb{P}_n \) obey the orthogonality conditions

\[
\langle r, p \rangle = 0 \quad \text{for every } p \in \mathbb{P}_{n-1}
\]

and

\[
\langle r, t^m \rangle \neq 0 \quad \text{for some } m \in \{0, 1, 2, \ldots, n\}
\]

Let \( t_1, t_2, \ldots, t_n \) be the zeros of the polynomial \( r \).

Choose \( w_1, w_2, \ldots, w_n \) as in

\[
\sum_{j=1}^{n} w_j t_j^m = \int_{a}^{b} t^m w(t) dt \quad m = 0, 1, 2, \ldots, n-1
\]

Then the quadrature formula

\[
\int_{a}^{b} w(t) f(t) dt \approx \sum_{j=1}^{n} w_j f(t_j)
\]

has order \( p = n + m \) (degree \( n + m - 1 \)).

**Comments:**
- \( m \in \{0, 1, 2, \ldots, n\} \) so \( w \leq n \).
- And the order \( p = n + m \leq 2n \).
- \( r \) does not satisfy "full" orthogonality conditions... only a partial set of conditions.
- This gives a quadrature scheme with an order reduced from the original Gaussian quadrature.