<u>1. Root finding</u>

Formulate Newton's method for solving the nonlinear $2 \; 2$ system of equations

$$\begin{aligned} f(x,y) &= 0 \\ g(x,y) &= 0 \end{aligned} .$$

In the same style as how one proves quadratic convergence in the scalar case for $f(x) = 0$, show quadratic convergence (assuming sufficient smoothness of *f*, *g*, root being simple, etc.) in the $2 \; 2$ case. Assuming the root $x = \alpha, y = \beta$ to be of multiplicity one, define ${}_n = x_n - \alpha, \; {}_n = y_n - \beta$, and show that both ${}_{n+1}$ and ${}_{n+1}$ are of size $O({}_n^2, {}_n^2)$..

<u>Solution:</u>

We first recall the proof for the scalar case $f(x) = 0$

2. Quadrature

(1) Consider quadrature

$$I_{quad} = \sum_{i=0}^{n} \alpha_i f(x_i), \quad x_i \in [-1, 1] \tag{0.1}$$

for the integral

$$I = \int_{-1}^{1} f(x)\, w(x)\, dx,$$

where w is a positive weight in $(-1, 1)$. Let

$$\Omega_{n+1}(x) = \prod_{i=1}^{n} (x - x_i)$$

denote the polynomial of degree $n+1$ associated with the (distinct) quadrature nodes $x_0, x_1, \ldots, x_n$. Prove that

$$\int_{-1}^{1} \Omega_{n+1}(x)\, p(x)\, w(x)\, dx = 0 \tag{0.2}$$

for any polynomial of degree less or equal to $m-1$ if and only if the ...

Proof:

(1) If the quadrature formula (0.1) is exact for all polynomials of degree less or equal to $n+m$, then

$$\int_{-1}^{1} \Omega_{n+1}(x)\, p(x)\, w(x)\, dx = \sum_{i=0}^{n} \alpha_i \Omega_{n+1}(x_i)\, p(x_i)\, w(x_i) = 0$$

Let us consider polynomial $f(x)$ of degree less or equal to $n+m$. We can write

$$f(x) = \Omega_{n+1}(x)\, \pi_{m-1}(x) + q_n(x),$$

where q_n is the remainder of the division of f by the polynomial Ω_{n+1}. We ... using (0.2)

$$I = \int_{-1}^{1} f(x)\, w(x)\, dx$$

$$= \int_{-1}^{1} q_n(x)\, w(x)\, dx$$

and, by the direct evaluation,

$$I_{quad} = \sum_{i=0}^{n} \alpha_i f(x_i)$$

$$= \sum_{i=0}^{n} \alpha_i \Omega_{n+1}(x_i)\, p(x_i)\, w(x_i) + \sum_{i=0}^{n} \alpha_i q_n(x_i)$$

$$= \sum_{i=0}^{n} \alpha_i q_n(x_i).$$

We obtain the result by observing that the quadrature weights α_i can always be chosen to satisfy

$$I = I_{quad}$$

for an arbitrary polynomial of degree less or equal to n.

3. Interpolation / Approximation

Assuming that $_{n}$, $n = 0,1,2,\ldots$ form a set of orthogonal polynomials of degrees n over some interval $[a,b]$ with weight function $w(x) > 0$, show that they obey a three-term recursion relation of the form

$$ {}_{1}(\ )\quad (\qquad\qquad)\quad (\ )\qquad\qquad {}_{1}(\ )\quad 0,\qquad\qquad 1,2,3, $$

4. Linear Algebra

Let $A \in \mathbb{C}^{n\times n}$ be a symmetric complex valued matrix, $A = A^t$. It is possible to...

[redacted]

Proof:

[illegible]

$Au = \mu u$,

we also have

$Au = \mu u$

as well as

$$A\overline{A}u = \mu A\overline{u} = \mu^2 u.$$

$$\overline{A}A\overline{u} = \mu\overline{A}u = \mu^2\overline{u}.$$

We recognize that u and $\overline{u}$ are the singular vectors and, thus, they are orthonormal.

[illegible]

[redacted]

5. ODE

Consider the 4th order Adams-Bashforth scheme (AB4) for solving the ODE $y' = f(x, y)$:

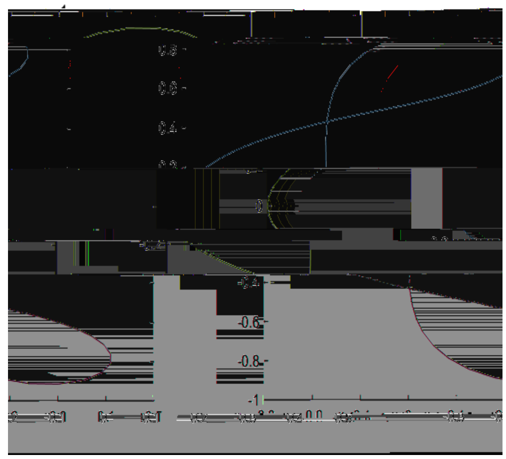
$$y_{n+1} = y_n + \frac{h}{24}\left(55 f_n - 59 f_{n-1} + 37 f_{n-2} - 9 f_{n-3}\right) \ .$$

a. Apply the *root condition* to this scheme. Explain the outcome of the test, and explain what information this provides regarding the scheme.

b. The Matlab code

```
r  = e p(comple (0,linspace(0,2*pi)));
 i = 24*(r.^4-r.^3)./(55*r.^3-59*r.^2+37*r-9);
plot( i);
```

generates the figure shown to the right.



i. Derive the relation used in the code.

ii. Explain (no need to do the algebra) how you

ii. The generated curve marks all possible ˎ -values for when a root *r* is on the periphery of the unit circle. If ˎ crosses a curve segment, a root moves between the inside and the outside of the unit circle. The stability

Proof.

(1) The even derivatives of the sine series

$$\partial_x^{2k}\partial_y^{2l}\sum_{m,n=1}^{\infty} u_{mn}\sin(\pi m x)\sin(\pi n y)$$

and, thus, is zero on the boundary. However, it is easy to see that this is ... This mismatch ... the solution near the boundary ... polynomials, e.g. Legendre polynomials, to ap-

(2) One can use orthogonal polynomials ... to approximate the solution,

$$u(x,y)=\sum_{m,n=0}^{\infty} u_{mn}P_m(2x-1)P_n(2y-1).$$

This series converges rapidly since the boundary conditions for the even derivatives are not forced on the solution.

(3) We can truncate the Legendre series so that

$$u_N(x,y)=\sum_{m,n=0}^{N} u_{mn}P_m(2x-1)P_n(2y-1)$$

is an accurate approximation of the solution. We use the Legendre series for ... numerical scheme using such nodes in both x and y directions.