ME 391 Final Exam Equation Sheet - Samantha Ramsey

Taylor Series Expansion: $f(x_0+\Delta x) = f(x_0) + \frac{df}{dx}\Big|_{x_0}\Delta x \quad + \quad \frac{1}{2!}\frac{d^2f}{dx^2}\Big|_{x_0}\Delta x^2 \quad + \quad \ldots \quad \frac{1}{n!}\frac{d^nf}{dx^n}\Big|_{x_0}\Delta x^n$

Initial Value Problem Methods:

Modified Euler Method (Predictor - Corrector):
2^{nd} order, explicit, conditionally stable & convergent
$y^p_{n+1} = y_n + \Delta t f_n$
$y^c_{n+1} = y_n + \Delta t(f_n + f^p_{n+1})$

Runge-Kutta 4:
Most accurate method, consistent, conditionally stable
$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$
$k_1 = \Delta t f(t_n, y_n)$
$k_2 = \Delta t f(t_n + \frac{\Delta t}{2}, y_n + \frac{k_1}{2})$
$k_3 = \Delta t f(t_n + \frac{\Delta t}{2}, y_n + \frac{k_2}{2})$
$k_4 = \Delta t f(t_n + \Delta t, y_n + k_3)$

Explicit Euler:
1^{st} order, explicit
$y_{n+1} = y_n + \Delta t f_n$

Implicit Euler:
1^{st} order, $f_{n+1}$is dependent on y_{n+1} - use solver!
$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$
$y_{n+1} = y_n + \Delta t f_{n+1}$

Boundary Value Problem Methods:

Equillibrium Method:
1) Construct a uniform finite difference grid from x_1 to x_{final}
2) Develop the centered FDA's using:
$y_i' = \frac{y_{i+1} - y_{i-1}}{2\Delta x}; \qquad y_i" = \frac{y_{i+1} - 2y_i + y_{i-1}}{\Delta x^2}$
3) Substitude into the ODE and solve the coupled system for $i = 1, 2, etc.$

Numerical Integration Methods:

Trapezoidal Rule for Equally Spaced Points: 2^{nd} order accurate
$I = \frac{h}{2}[f_0 + 2f_1 + 2f_2 + \ldots + 2f_{n-1} + f_n]$

Simpson's $1/3$ Rule: 4^{th} order accurate
Must have equally spaced data points and the total nuber of intervals must be even, which means the number of data points must be odd.
$I = \frac{1}{3}h[f_0 + 4f_1 + 2f_2 + 4f_3 + \ldots + 4f_{n-1} + f_n]$

Simpson's $3/8$ Rule: 4^{th} order accurate
Data points must be equally spaced, total number of intervals must be a multiple of three.
$I = \frac{3}{8}h[f_0 + 3f_1 + 3f_2 + 2f_3 + 3f_4 + \ldots + 3f_{n-1} + f_n]$

Order:
$\frac{Error(\Delta t_1)}{Error(\Delta t_2)} = \frac{\Delta t_1}{\Delta t_2}^n$

Linearize a Function:
$f(t,y) \backsim f(t_0,y_0) + \frac{\partial f}{\partial y}|_{t_0,y_0}(y - y_0) + \frac{\partial f}{\partial t}|_{t_0,y_0}(t - t_0)$

Stability:
$y_{n+1} = Gy_n$ where: $|G| \le 1$
G is the amplification factor

Root - Finding Methods:

Newton Method: $x_1 = x_0 + \frac{\alpha - f(x_0)}{f'(x_0)}$
where the function $f(x) = \alpha$
2^{nd} order, linearly convergent at double root

Quasi-Newton Method: $f'(x_0) = \frac{f(x_0+\delta) - f(x_0)}{\delta}$
if unable to differentiate, where δ is a very small step size

Secant Method: $x_2 = x_1 + \frac{(\alpha - f(x_1))(x_1 - x_0)}{f(x_1) - f(x_0)}$
where x_0 and x_1 are two initial guesses
1.62 order

Newtons Method for multiple equations and multiple unknowns:

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x}\Big|_{x_0y_0} & \frac{\partial f}{\partial y}\Big|_{x_0y_0} \\ \frac{\partial g}{\partial x}\Big|_{x_0y_0} & \frac{\partial g}{\partial y}\Big|_{x_0y_0} \end{bmatrix}^{-1} \begin{bmatrix} 0 - f(x_0,y_0) \\ 0 - g(x_0,y_0) \end{bmatrix}$$

where $\Delta x = x_1 - x_0$ and $\Delta y = y_1 - y_0$

Matrix Methods:

Finding the Inverse of a Matrix:
$[A|I] \quad \text{rref} \Longrightarrow \quad [I|A^{-1}]$

Matrix Multiplication:
$AB = [a_{ik}][b_{kj}] = [c_{ij}] = C$
$c_{ij} = \sum_{k=1}^{m} a_{ik}b_{kj}$

LU Decomposition:
1) Take A and decompose to $A = LU$ where U is the upper triangular matrix formed through Gauss elimination and L is the lower triangular matrix formed using the record of eliminiation multipliers
2) $Ax = b \rightarrow \frac{LUx}{y} = b$ solve $Ly = b$ for y
3) $Ux = y$ solve for x

Jacobi Iteration*:
$x_i^{(k)} = x_i^{(k-1)} + \frac{1}{a_{ii}}\left[b_i - \sum_{j=1}^{n} a_{ij}x_j^{(k-1)}\right]$
$x_1^{(1)} = x_1^{(0)} + \frac{1}{a_{11}}\left[b_1 - a_{11}x_1^{(0)} - a_{12}x_2^{(0)} - a_{13}x_3^{(0)}\right]$
$x_2^{(1)} = x_2^{(0)} + \frac{1}{a_{22}}\left[b_2 - a_{21}x_1^{(0)} - a_{22}x_2^{(0)} - a_{23}x_3^{(0)}\right]$
$x_3^{(1)} = x_3^{(0)} + \frac{1}{a_{33}}\left[b_3 - a_{31}x_1^{(0)} - a_{32}x_2^{(0)} - a_{33}x_3^{(0)}\right]$

Gauss-Seidel*:
for SOR - multiply fraction by ω
$x_i^{(k)} = x_i^{(k-1)} + \frac{1}{a_{ii}}\left[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i}^{n} a_{ij}x_j^{(k-1)}\right]$
$x_1^{(1)} = x_1^{(0)} + \frac{1}{a_{11}}\left[b_1 - a_{11}x_1^{(0)} - a_{12}x_2^{(0)} - a_{13}x_3^{(0)}\right]$
$x_2^{(1)} = x_2^{(0)} + \frac{1}{a_{22}}\left[b_2 - a_{21}x_1^{(1)} - a_{22}x_2^{(0)} - a_{23}x_3^{(0)}\right]$
$x_3^{(1)} = x_3^{(0)} + \frac{1}{a_{33}}\left[b_3 - a_{31}x_1^{(1)} - a_{32}x_2^{(1)} - a_{33}x_3^{(0)}\right]$
*assumes Diagonal Dominance

Interpolation Methods - used to determine values of a function other than the discrete data set.

Direct Fit Polynomials (does not require equally spaced data):

for n > 4, system of equations may be ill-conditioned

$P_n(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n$

make matrix using table of f and x values and solve for the a coefficients

to approximate derivatives, differentiate polynomial after solving for a

Lagrange Polynomial through points: $[a, f(a)], [b, f(b)]$

does not require equally spaced data, does not require solution of linear system

$P_1(x) = \frac{(x-b)}{(a-b)}f(a) + \frac{(x-a)}{(b-a)}f(b)$

$P_2(x) = \frac{(x-b)(x-c)}{(a-b)(a-c)}f(a) + \frac{(x-a)(x-c)}{(b-a)(b-c)}f(b) + \frac{(x-a)(x-b)}{(c-a)(c-b)}f(c)$

Where x is the data point we are trying to solve the function for.

Ordinary Differences:

Just subtract upper value from lower value in each column!

Central Difference Formulas:

Considering the general form of the Newton Polynomial for the first derivative,

if x_1 is the centerpoint of x_0 and x_1 at $x = x_1, s+1$. Therefore,

$P'_n(x_1) = \frac{1}{\Delta x}\left(\Delta f_0 + \frac{1}{2}\Delta^2 f_0 - \frac{1}{6}\Delta^3 f_0 + \ldots\right)$

Divided Differences:

$$f[x_0, x_1, \ldots, x_n] = \frac{f[x_1, x_2, \ldots, x_n] - f[x_0, x_1, \ldots, x_{n-1}]}{x_n - x_{n-0}}$$

$f(x) \approx P_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x0)(x - x_1) + \ldots$

$f'(x) \approx P'_n(x) = f[x_0] + f[x_0, x_1] + f[x_0, x_1, x_2][(x - x_1) + (x - x_0)] + \ldots$

Newton Forward Difference Polynomial:

$$P_n(x) = f_0 + s\Delta f_0 + \frac{s(s-1)}{2!}\Delta^2 f_0 + \ldots + \frac{s(s-1)(s-2)\ldots(s-n)}{(n+1)!}\Delta^{n+1} f_0$$

$h = \text{step size} \quad \Longrightarrow \quad s_1 = \frac{x - x_0}{h}$

where x is the value you're approximating for and x_0 is the selected basepoint

$$f'(x) \approx P'_n(x) = \frac{1}{\Delta x}\Big(\Delta f_0 + \frac{1}{2}[(s-1) + s]\Delta^2 f_0 + \frac{1}{6}[(s-1)(s-2) + s(s-2) + s(s-1)]\Delta^3 f_0 + \ldots\Big)$$

in the case $x = x_0 (s = 0)$, the derivative approximation can be simplified to:

$f'(x_0) \approx P'_n(x_0) = \frac{1}{\Delta x}\left(\Delta f_0 - \frac{1}{2}\Delta^2 f_0 + \frac{1}{3}\Delta^3 f_0 + \ldots\right)$

$f''(x_0) \approx P''_n(x_0) = \frac{1}{\Delta x^2}\left(\Delta^2 f_0 - \Delta^3 f_0 + \ldots\right)$

Newton Backward Difference Polynomial:

$$P_n(x) = f_0 + s\Delta f_0 + \frac{s(s+1)}{2!}\Delta^2 f_0 + \ldots + \frac{s(s+1)(s+2)\ldots(s+n)}{(n+1)!}\Delta^{n+1} f_0$$

An FDE is consistent with an ODE if the difference (truncation error) between them vanishes as $\Delta t \to 0$.

ORDER of a FD solution of an ODE is the rate at which the global error approaches zero as $\Delta t \to 0$.

An FDE is STABLE if it produces a bounded solution when the exact solution is bounded.

A finite difference method is CONVERGENT if the numerical solution approaches the exact solution of the ODE as $\Delta t \to 0$.

Least Squares Approximation:

Yields a polynomial that passes through the data set in the best possible manner.

$$\begin{bmatrix} \sum_{i=1}^{N} x_i^2 & \sum_{i=1}^{N} x_i \\ \sum_{i=1}^{N} x_i & \sum_{i=1}^{N} 1 \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{N} Y_i x_i \\ \sum_{i=1}^{N} Y_i \end{bmatrix}$$

for $f(x) = ax + b$. Must linearize the function, perform linearization calculation on Y before plugging into matrix.

Using Taylor Series, determine the i^{th} order forward difference polynomial for the j^{th} derivative of an arbitrary function. Add $i + j$ to determine the number of columns in matrix. Example - third order, second derivative:

$\alpha \quad f(x) = f(x)$

$\beta \quad f(x + \Delta x) = f(x) + f'(x)\Delta x + \frac{1}{2!}f''(x)\Delta x^2 + \frac{1}{3!}f'''(x)\Delta x^3 + \frac{1}{4!}f''''(x)\Delta x^4$

$\gamma \quad f(x + 2\Delta x) = f(x) + f'(x)2\Delta x + \frac{1}{2!}f''(x)2\Delta x^2 + \frac{1}{3!}f'''(x)2\Delta x^3 + \frac{1}{4!}f''''(x)2\Delta x^4$

$\delta \quad f(x + 3\Delta x) = f(x) + f'(x)3\Delta x + \frac{1}{2!}f''(x)3\Delta x^2 + \frac{1}{3!}f'''(x)3\Delta x^3 + \frac{1}{4!}f''''(x)3\Delta x^4$

$\epsilon \quad f(x + 4\Delta x) = f(x) + f'(x)4\Delta x + \frac{1}{2!}f''(x)4\Delta x^2 + \frac{1}{3!}f'''(x)4\Delta x^3 + \frac{1}{4!}f''''(x)4\Delta x^4$

Group terms into matrix with i^{th} term $= 1$ and solve.

$f''(x) = \alpha f(x) + \beta f(x + \Delta x) + \gamma f(x + 2\Delta x) + \ldots$

INTEGRATION RULES:

$\int x^n dx = \frac{x^{n+1}}{n+1} + C$

$\int e^x dx = e^x + C$

$\int \frac{1}{x} dx = ln(x) + C$

$\int n^x dx = \frac{n^x}{ln(n)} + C$

$\int cos(x)dx = sin(x) + C$

$\int sin(x)dx = -cos(x) + C$

$\int sec^2(x)dx = tan(x) + C$

$\int csc^2(x)dx = -cot(x) + C$

$\int tan(x)sec(x)dx = sec(x) + C$

$\int cot(x)csc(x)dx = -csc(x) + C$

$\int \frac{du}{\sqrt{a^2-u^2}} = arcsin\frac{u}{a} + C$

$\int -\frac{du}{\sqrt{a^2-u^2}} = arccos\frac{u}{a} + C$

$\int \frac{du}{a^2+u^2} = \frac{1}{a}arctan\frac{u}{a} + C$

$\int \frac{du}{u\sqrt{u^2-a^2}} = \frac{1}{a}arcsec\frac{u}{a} + C$

$\int \frac{du}{u^2-a^2} = \frac{1}{2a}ln|\frac{u-a}{u+a}| + C$

Integration by Parts:

$$= uv - \int u'v dx = \int uv' dx$$