Math 170C: Numerical Analysis of Ordinary Differential Equations
Spring 2018

Review for Final Exam

Note: You are responsible for double-checking the accuracy of all the formulas in this review, and making corrections if needed.

Chapter 4. Numerical Differentiation and Integration

Numerical Differentiation

1. Simple and useful numerical differentiation formulas:

Forward/backward difference $f'(x_0) = \frac{1}{h}[f(x_0+h) - f(x_0)] - \frac{h}{2}f''(\xi);$

Central-difference $f'(x_0) = \frac{1}{2h}[f(x_0+h) - f(x_0-h)] - \frac{h^2}{6}f'''(\xi);$

Central-difference $f''(x_0) = \frac{1}{h^2}[f(x_0-h) + f(x_0+h) - 2f(x_0)] - \frac{h^2}{12}f^{(4)}(\xi).$

Exercise. Use the Taylor expansion to derive these formulas.
Exercise. Forward/backward difference: If $f(0) = 1$ and $f(0.1) = 0.2$, what is an approximation of $f'(0)$? What is the maximal error (neglect the round-off error) of your approximation if $|f''(x)| \le 0.1$ for all x?
Exercise. Central difference for first-order derivative: If $f(0) = 1$, $f(-0.1) = 0.8$, and $f(0.1) = 0.2$, what is an approximation of $f'(0)$? If $|f'''(x)| \le 4$ for all x, what is a good error bound (neglect the round-off error)?
Exercise. If $f(0) = 0.4$, $f(-0.1) = 0.7$, and $f(0.1) = 0.2$, find an approximation of $f''(0)$.

2. The Lagrange interpolation leads to the $(n+1)$-point numerical differentiation formula

$$f'(x_j) = \sum_{k=0}^{n} f(x_k)L_k'(x_j) + \frac{1}{(n+1)!}f^{(n+1)}(\xi_j)\prod_{k=0,k\neq j}^{n}(x_j - x_k), \qquad j = 0, 1, \ldots, n,$$

where $L_k(x) = (k = 0, \ldots, n)$ are the basic Lagrange polynomials associated with $x_0, x_1, \ldots, x_n$.
Exercise. Derive the three-point ($n = 2$) endpoint formula (4.4) on page 175.
Exercise. Suppose $f(0) = 0$, $f(0.1) = 0.2$, and $f(0.3) = 0.4$. Find an approximate of $f'(0)$ using the formula (4.4) on page 175.

Numerical Integration

1. Some concepts:
   (1) What is a basic, and a composite, numerical integration rule?
   (2) What is the degree of precision for a numerical integration rule?

Exercise. Find the numbers a, b, and c so that the numerical quadrature

$$\int_{-2}^{2} f(x)\,dx \approx af(-1) + bf(0) + cf(1)$$

has the degree of precision as high as possible. What is the degree of precision of this quadrature rule with your values of a, b, and c?

2. Newton–Cotes formulas (based on the Lagrange interpolation):

$$\int_a^b f(x)\,dx \approx \int_a^b p_n(x)\,dx = \sum_{i=0}^{n} a_i f(x_i), \qquad \text{where} \quad a_i = \int_a^b L_i(x)\,dx \quad (i = 1, \dots, n).$$

Here, $p_n(x) = \sum_{i=0}^n f(x_i)L_i(x)$ is the Lagrange interpolation of $f(x)$ at $x_0, x_1, \dots, x_n$ in $[a, b]$ and $L_i(x)$ $(i = 0, 1, \dots)$ are the basic Lagrange polynomials associated with $x_0, x_1, \dots, x_n$. The closed Newton–Cotes formulas are associated with $x_i = a + ih$ $(i = 0, 1, \dots, n)$, where $h = (b - a)/n$ and $n \geq 0$ is an integer.
Exercise. Why this formula is exact for a polynomial of degree $\leq n$?
Exercise. What is the difference between an open and a closed Newton–Cotes formula?

3. The following are three very useful formulas (in their basic forms):

Mid-point rectangle $$\int_\alpha^\beta f(x)\,dx = f\left(\frac{\alpha+\beta}{2}\right)(\beta-\alpha) + \frac{1}{24}f''(\xi)(\beta-\alpha)^3;$$

Trapezoid $$\int_\alpha^\beta f(x)\,dx = \frac{1}{2}\left[f(\alpha)+f(\beta)\right](\beta-\alpha) - \frac{1}{12}f''(\xi)(\beta-\alpha)^3;$$

Simpson's $$\int_\alpha^\beta f(x)\,dx = \frac{\beta-\alpha}{6}\left[f(\alpha) + 4f\left(\frac{\alpha+\beta}{2}\right) + f(\beta)\right] - \frac{1}{2880}f^{(4)}(\xi)(\beta-\alpha)^5.$$

Exercise. Derive each of these formulas using the Lagrange interpolation.
Exercise. What is the degree of precision of each of these formulas? Why?
Exericse. Derive the composite rule for each of these rules.
Exercise. If $f(0) = 1$, $f(0.1) = 1.2$, $f(0.2) = 1.4$, $f(0.3) = 2$, and $f(0.4) = 1.8$. Use the composite midpoint rectangle rule to approximate the integral of f over $[-0.05, 0.35]$. Use the composite trapezoidal and Simpson's rules to approximate the integral of f over $[0, 0.4]$.

4. For any $n \geq 1$, the n-point Gaussian quadrature (with error) is

$$\int_{-1}^{1} f(x)\,dx \approx \sum_{i=1}^{n} w_i f(x_i),$$

where $x_1, \dots, x_n \in (-1, 1)$ are the n distinct roots of the Legendre polynomial $P_n(x)$ and w_i are the integrals over $[-1, 1]$ of the basic Lagrange interpolation polynomials $L_i(x)$ associated with these roots. Note that $\{P_k(x)\}_{k=0}^\infty$ are orthogonal polynomials.
Exercise. What is the degree of precision of this formula?
Exercise. Use the composite 2-point Gaussian quadrature to approximate the integral of $f(x) = x^4$ on $[0, 2]$ with 2 subintervals.

Richardson's Extrapolation

1. Suppose $M = N(h) + K_2h^2 + K_4h^4 + \cdots$. Richardson's extrapolation: $\tilde{N}(h) := [4N(h/2) - N(h)]/3$.
Exercise. Expand $M = \tilde{N}(h) + \tilde{K}_4h^4 + \cdots$. What is $\tilde{K}_4$ in terms of K_4?
Exercise. What is Richardson's extrapolation formula using $N(h)$ and $N(h/3)$?
2. **Exercise.** Show that one step of Richardson's extrapolation of the trapezoidal rule of numerical integration is exactly Simpson's rule.

Chapter 5. Initial-Value Problems for Ordinary Differential Equations

Consider the initial-value problem for ODE: $y' = f(t, y)$ ($a \le t \le b$) and $y(a) = \alpha$. Let $N \ge 1$ be an integer, $h = (b-a)/N$, and $t_i = a + ih$ ($i = 0, 1, \ldots, N$). We try to find $w_i \approx y_i = y(t_i)$ for all i.

One-Step Method: Euler's, General Taylor's, and Runge-Kutta Method

1. The general form of a one-step method is

$$\begin{aligned} &w_0 = \alpha, \\ &w_{i+1} = w_i + h\phi(t_i, w_i), \qquad i = 0, 1, \ldots, N-1. \end{aligned}$$

   The truncation error for such a method is defined by

$$\tau_{i+1}(h) = \frac{y(t_{i+1}) - y(t_i)}{h} - \phi(t_i, y(t_i)), \qquad i = 0, 1, \ldots, N-1.$$

   Note that the truncation error is defined using the exact solution $y = y(t)$.
2. Euler's method is given by

$$\begin{aligned} &w_0 = \alpha, \\ &w_{i+1} = w_i + hf(t_i, w_i), \qquad i = 0, 1, \ldots, N-1. \end{aligned}$$

   If L is the Lipschitz constant for $f(t, y)$ in y and $|y''(t)| \le M$ ($a \le t \le b$), then error bound is

$$|y_i - w_i| \le \frac{hM}{2L}\left[e^{L(t_i-a)} - 1\right], \qquad i = 0, 1, \ldots, N.$$

   Exercise. If $f(t, y) = y + t$ and $y(0) = 0$, find approximations of $y(0.1)$ and $y(0.2)$.
   Exercise. Derive the local truncation error.
3. The general Taylor method of order n is given by

$$\begin{aligned} &w_0 = \alpha, \\ &w_{i+1} = w_i + hT^{(n)}(t_i, w_i), \qquad i = 0, 1, \ldots, N-1), \end{aligned}$$

   where

$$T^{(n)}(t, w) = f(t, w) + \frac{h}{2!}f'(t, w) + \cdots + \frac{h^{n-1}}{n!}f^{(n-1)}(t, w).$$

   Exercise. Derive this method using the Taylor expansion.
   Exercise. What is the local truncation error for this method?
   Exercise. If $f(t, y) = t^2 + \sin y$, what is $f'(t, y)$?
4. The midpoint method is a special Runge-Kutta method of order 2. It is given by

$$\begin{aligned} &w_0 = \alpha, \\ &w_{i+1} = w_i + hf\left(t_i + h/2, w_i + (h/2)f(t_i, w_i)\right), \quad i = 0, 1, \ldots, N-1. \end{aligned}$$

   Exercise. Let $f(t, y) = t + y$ ($0 \le t \le 3$), $\alpha = 0$, $h = 1$. Compute w_0, w_1 and w_2.
   Exercise. What is the order of the local truncation error of this method?
5. The modified Euler's method is also a Runge-Kutta method of order 2. It is given by

$$\begin{aligned} &w_0 = \alpha, \\ &w_{i+1} = w_i + \frac{h}{2}\left[f(t_i, w_i) + f(t_{i+1}, w_i + hf(t_i, w_i))\right], \quad i = 0, 1, \ldots, N-1. \end{aligned}$$

   Exercise. Let $f(t, y) = t + y$ ($0 \le t \le 3$), $\alpha = 0$, $h = 1$. Compute w_0, w_1 and w_2.
   Exercise. What is the order of the local truncation error of this method?

6. A commonly used Runge-Kutta method of order 4 is given by

$$
\begin{aligned}
w_0 &= \alpha, \\
k_1 &= hf(t_i, w_i), \\
k_2 &= hf(t_i + h/2, w_i + k_1/2), \\
k_3 &= hf(t_i + h/2, w_i + k_2/2), \\
k_4 &= hf(t_{i+1}, w_i + k_3), \\
w_{i+1} &= w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \quad i = 0, 1, \ldots, N-1.
\end{aligned}
$$

Exercise. Write a pseudocode for this method.
Exercise. Let $f(t, y) = t + y$ $(0 \le t \le 3)$, $\alpha = 0$, $h = 1$. Compute w_0 and w_1.

Multistep Methods for Solving Initial-Value Problems of ODE

1. An m-step (linear) multistep method is given by

$$
\begin{aligned}
& w_0 = \alpha,\ w_1 = \alpha_1,\ \ldots, w_{m-1} = \alpha_{m-1}, \\
& w_{i+1} = a_{m-1} w_i + a_{m-2} w_{i-1} + \cdots + a_0 w_{i+1-m} \\
& \qquad + h\left[b_m f(t_{i+1}, w_{i+1}) + b_{m-1} f(t_i, w_i) + \cdots + b_0 f(t_{i+1-m}, w_{i+1-m})\right] \\
& \qquad (i = m-1, m, \ldots, N-1).
\end{aligned}
$$

The local truncation error of such a method is defined by

$$
\begin{aligned}
\tau_{i+1}(h) = & \frac{1}{h}\left[y(t_{i+1}) - a_{m-1} y(t_i) - \cdots - a_0 y(t_{i+1-m})\right] \\
& - \left[b_m f(t_{i+1}, y(t_{i+1})) + b_{m-1} f(t_i, y(t_i)) + \cdots + b_0 f(t_{i+1-m}, y(t_{i+1-m}))\right] \\
& (i = m-1, m, \ldots, N-1).
\end{aligned}
$$

Exercise. What is an implicit multistep method? An explicit multistep method?

2. The m-step Adams–Bashforth method (ABm) and the m-step Adams–Moulton method (AMm) are derived by replacing $f(t, y(t))$ in

$$
y(t_{i+1}) = y(t_i) + \int_{t_i}^{t_{i+1}} f(t, y(t))\, dt
$$

by its Lagrange interpolation at m points $(t_i, f(t_i, y(t_i))), \ldots, (t_{i+1-m}, f(t_{i+1-m}, y(t_{i+1-m})))$ and the $m+1$ points $(t_{i+1}, f(t_{i+1}, y(t_{i+1}))), \ldots, (t_{i+1-m}, f(t_{i+1-m}, y(t_{i+1-m})))$, respectively, and then change $y(t_j)$ to w_j.
Exercise. Is ABm explicit or implicit? What about AMm?
Exercise. Derive AB2.
Exercise. Solve $y' = y - t^2 + 1$ $(0 \le t \le 2)$ and $y(0) = 0.5$ with $h = 0.2$ using AB2, AB3, and AB4 (see (5.33), (5.34), and (5.35) in the book). Use RK4 to find the initial values.

Predictor-Corrector Method

1. A predictor-corrector method is to combine an explicit method as a predictor, say,

$$
w_{i+1} = F(w_i, w_{i-1}, \ldots, w_{i+1-m}; h), \qquad i = m-1, \ldots, N-1),
$$

with an implicit method as a corrector, say,

$$
w_{i+1} = G(w_{i+1}, w_i, w_{i-1}, \ldots, w_{i+1-m}; h), \qquad i = m-1, \ldots, N-1.
$$

To compute w_{i+1}, we first compute $\tilde{w}_{i+1}$, a predicted value, by the predictor:

$$\tilde{w}_{i+1} = F(w_i, w_{i-1}, \ldots, w_{i+1-m}; h), \qquad i = m-1, \ldots, N.$$

We then compute w_{i+1} by the corrector:

$$w_{i+1} = G(\tilde{w}_{i+1}, w_i, w_{i-1}, \ldots, w_{i+1-m}; h), \qquad i = m-1, \ldots, N.$$

Exercise. Solve the problem in the previous exercise using the AB4-AM3 predictor-corrector method, with RK4 for the initial values. Why AB4-AM3 not AB4-AM4?
Exercise. Solve the problem in the previous exercise using Milne's method as a predictor and the implicit Simpson's method as the corrector. (See page 313 for these two methods.)

Higher-Order Equations and Systems of First-Order Equations

1. To solve the initial-value problem of a higher-order equation, you need to convert such a problem to the initial-value problem of a system of first-order equations.
   Exercise. Convert the initial-value problem $y'' = 2y' - e^t y + t$ $(0 \le t \le 2)$, $y(0) = 3$, and $y'(0) = 4$ into the initial-value problem of a system of first-order equations. Then solve it using a simple one-step method with $h = 0.5$

Consistence, Convergence, and Stability. Stiff Differential Equations.

1. The one-step method

$$w_0 = \alpha,$$
$$w_{i+1} = w_i + h\phi(t_i, w_i; h), i = 0, \ldots, N-1,$$

   is consistent, if

$$\lim_{h\to 0} \max_{1\le i\le N} |\tau_i(h)| = 0,$$

   and is convergent, if

$$\lim_{h\to 0} \max_{1\le i\le N} |y(t_i) - w_i| = 0.$$

   Under some assumptions, the method is stable. Moreover, the method is consistent if and only if it is convergent; and if and only if $\phi(t, y, 0) = f(t, y)$ for all $t \in [a, b]$. The convergence rate is the same as that for the local truncation error.
   Exercise. Show that the modified Euler's method is consistent and convergent.
2. Consider an m-step (linear) multistep method defined above. It is consistent, if

$$\lim_{h\to 0} \max_{m\le i\le N} |\tau_i(h)| = 0 \quad \text{and} \quad \lim_{h\to 0} \max_{0\le i\le m-1} |y(t_i) - w_i| = 0,$$

   and is convergent, if

$$\lim_{h\to 0} \max_{1\le i\le N} |y_i - w_i| = 0.$$

   The root condition: $|\lambda_j| \le 1$ $(j = 1, \ldots, m)$, and if $|\lambda_j| = 1$ $(1 \le j \le m)$ then λ_j is a simple root. Here, $\lambda_1, \ldots, \lambda_m$ are the m roots of the characteristic polynomial

$$P_m(\lambda) = \lambda^m - a_{m-1}\lambda^{m-1} - \cdots - a_1\lambda - a_0.$$

   Stability is equivalent to the root condition. If consistent, then stability is equivalent to the convergence. Strong stability: the root condition, and 1 is the only root of $P_m(\lambda)$ with

magnitude 1. Weak stability: the root condition, and there are more than one distinct roots of $P_m(\lambda)$ with magnitude 1.
Exercise. Is Milne's method (cf. page 313) stable? strongly stable?
Exercise. Is Simpson's method (cf. page 313) stable? strongly stable?
Exercise. Study the stability of ABm and AMm methods for $m = 2, 3, 4$.
Exercise. Study the stability of the method

$$\begin{aligned} &w_0 = \alpha,\ w_1 = \alpha_1, \\ &w_{i+1} = -4w_i + 5w_{i-1} + 2h\left[f(t_i, w_i) + 2hf(t_{i-1}, w_{i-1})\right], \qquad i = 1, \ldots, N-1. \end{aligned}$$

3. For the m-step multistep method defined above, we introduce the polynomial

$$Q_m(z, h\lambda) = (1 - h\lambda b_m)z^m - (a_{m-1} + h\lambda b_{m-1})z^{m-1} - \cdots - (a_0 + h\lambda b_0).$$

The region of absolute stability is $R = \{h\lambda \in \mathbb{C} : |\beta_k| \le 1, k = 1, \ldots, m\}$, where β_k ($k = 1, \ldots, m$) are all the roots of $Q_m(z, h\lambda)$. The method is A-stable, if R contains the entire left half-plane.
Exercise. Show that the backward Euler's method is A-stable.
Exercise. Show that the implicit trapezoidal method is A-stable.

Chapter 11. Boundary-Value Problems for Ordinary Differential Equations

1. The shooting method for the boundary-value problem of a second-order linear equation:

$$y'' = p(x)y' + q(x)y + r(x) \quad (a \le x \le b) \quad \text{and} \quad y(a) = \alpha, \quad y(b) = \beta.$$

The solution is

$$\begin{aligned} y(x) &= y_1(x) + \frac{\beta - y_1(b)}{y_2(b)} y_2(x) \qquad (a \le x \le b), \\ y_1'' &= p(x)y_1' + q(x)y_1 + r(x) \quad (a \le x \le b) \quad \text{and} \quad y_1(a) = \alpha, \quad y_1'(a) = 0, \\ y_2'' &= p(x)y_2' + q(x)y_2 \quad (a \le x \le b) \quad \text{and} \quad y_2(a) = 0, \quad y_2'(a) = 1. \end{aligned}$$

Exercise. Use the shooting method to solve the boundary-value problem $y'' = y + 2 - x^2$ ($0 \le x \le 1$), $y(0) = 0$, and $y(1) = 1$.

2. The finite-difference method for the boundary-value problem of a second-order linear equation is based on the approximations

$$\begin{aligned} y''(x_i) &\approx \frac{1}{h^2}\left[y(x_{i-1}) + y(x_{i+1}) - 2y(x_i)\right], \\ y'(x_i) &\approx \frac{1}{2h}\left[y(x_{i+1}) - y(x_{i-1})\right]. \end{aligned}$$

Exercise. Write down the matrix A and the vector $\mathbf{b}$ in the system of equations $A\mathbf{w} = \mathbf{b}$ obtained in using the finite-difference method to solve the boundary-value problem $y'' = y + 2 - x^2$ ($0 \le x \le 1$), $y(0) = 0$, and $y(1) = 1$ with $h = 0.25$.