553.481/681 Numerical Analysis

Homework 4 Solutions

Lowen Peng
Problem 1. (a) Derive Boole’s rule, the Newton-Cotes quadrature rule for $n=4$, so that, with $x_i=a+ih$, $i=0,\ldots,4$ for $h=(b-a)/4$, and $\xi \in [a,b]$

\[
\int_a^b f(x)dx = \frac{2h}{45} \left[ 7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4) \right] - \frac{8h^7}{945} f^{(6)}(\xi)
\]

You do not need to evaluate by hand the integrals to determine the rational coefficients $\alpha_i$, $i=0,1,\ldots,4$ but instead you can use symbolic integration (for example, \texttt{int} in Matlab). To evaluate the error term, you should derive and use the expression

\[
E_4(f) = \int_a^b f[x_0, x_1, x_2, x_3, x_4, x] \prod_{i=0}^{4}(x-x_i)
\]

Then you can use without proof the fact that $w(x) := \int_0^x \prod_{i=0}^4(\bar{x} - x_i) d\bar{x} \geq 0$. The integral to determine the error coefficient can be again obtained symbolically. 

(b) Modify the course script \texttt{simpcc.m} to write a code \texttt{boolc.m} that implements the composite Boole’s rule.

(c) Use your code to compare the composite Simpson’s rule and composite Boole’s rule applied to the following two integrals:

\begin{itemize}
  \item[(i)] \int_0^5 dx \sin(x^2)
  \item[(ii)] \int_0^5 dx \frac{1}{1+x^6}
\end{itemize}

Use $n=8,16,32,64,128,256$ and make a log-log plot of the errors in the approximations versus $n$. Are the results consistent with the proven asymptotic order of convergence? Explain your answer. For (ii) it will help to consider even larger $n$.

Note: The exact value of integral (i) can be obtained using the Matlab function \texttt{fresnels} and integral (ii) is

\[
\int_0^5 dx \frac{1}{1+x^6} = \frac{1}{12} \left[ \sqrt{3} \ln \left( \frac{26+5\sqrt{3}}{26-5\sqrt{3}} \right) + 2\arctan(10+\sqrt{3}) + 2\arctan(10-\sqrt{3}) + 4\arctan(5) \right].
\]

(a) Since $\alpha_i := \int_a^b \prod_{j \neq i}(x-x_j)/(x_i-x_j) dx$, we have

\begin{align*}
\alpha_0 &= \frac{1}{4h^4} \int_a^b (x-x_1)(x-x_2)(x-x_3)(x-x_4) dx \\
\alpha_1 &= -\frac{1}{3h^4} \int_a^b (x-x_0)(x-x_2)(x-x_3)(x-x_4) dx \\
\alpha_2 &= \frac{1}{2h^4} \int_a^b (x-x_0)(x-x_1)(x-x_3)(x-x_4) dx \\
\alpha_3 &= -\frac{1}{3h^4} \int_a^b (x-x_0)(x-x_1)(x-x_2)(x-x_4) dx \\
\alpha_4 &= \frac{1}{4h^4} \int_a^b (x-x_0)(x-x_1)(x-x_2)(x-x_3) dx.
\end{align*}

Code:

```matlab
syms x
alp(1)=int((x-1)*(x-2)*(x-3)*(x-4),0,4)/factorial(4);
alp(2)=-int((x*(x-2)*(x-3)*(x-4),0,4)/factorial(3);
alp(3)=int(x*(x-1)*(x-3)*(x-4),0,4)/factorial(2)^2;
alp(4)=-int(x*(x-1)*(x-2)*(x-4),0,4)/factorial(3);
alp(5)=int(x*(x-1)*(x-2)*(x-3),0,4)/factorial(4);

beta=int(x^2*(x-1)*(x-2)*(x-3)*(x-4),0,4)/factorial(6)
```

We find:

\[
\alpha_0 = \frac{14h}{45}, \quad \alpha_1 = \frac{64h}{45}, \quad \alpha_2 = \frac{8h}{15}, \quad \alpha_3 = \frac{64h}{45}, \quad \alpha_4 = \frac{14h}{45}.
\]

Recall that $f(x) = \sum_{i=0}^{4} \alpha_i f(x_i) + f[x_0, \ldots, x_4, x] \prod_{i=0}^{4}(x-x_i)$. Apply integration by parts, the mean value theorem, and Program 1.1 again to write
\[ E_4(f) = \int_a^b f[x_0, \ldots, x_4, x] \frac{d}{dx} w(x) \, dx \]

\[ = -\int_a^b w(x) \frac{d}{dx} f[x_0, \ldots, x_4, x] \, dx \]

\[ = -\int_a^b f[x_0, \ldots, x_4, x] w(x) \, dx \]

\[ = -f[x_0, \ldots, x_4, \xi, \xi] \int_a^b w(x) \, dx \]

\[ = -\frac{f^{(6)}(\xi)}{6!} \int_a^b \int_0^x \prod_{i=0}^4 (x' - x_i) \, dx' \, dx \]

\[ = \frac{f^{(6)}(\xi)}{6!} \int_a^b (x - x_0) \prod_{i=0}^4 (x - x_i) \, dx \]

\[ = 8h^7 945f^{(6)}(\xi). \]

It's just algebra to see that assembling the collected \(a_i, f(x_i)\), and \(E_4(f)\) fits the desired form.

(b) See

\begin{verbatim}
function I=boolc(f,a,b,n)
  if mod(n,4)~=0
    'n not a multiple of 4 in Boole'
    return
  end
  h=(b-a)/n;
  x1=a+h:4*h:b-3*h;
  x2=a+2*h:4*h:b-2*h;
  x3=a+3*h:4*h:b-h;
  x4=a+4*h:4*h:b-4*h;
  I=7*(f(a)+f(b))+32*(sum(f(x1))+sum(f(x3)))+12*sum(f(x2))+14*sum(f(x4));
  I=2*h*I/45.;
end
\end{verbatim}

(c) Note the results for the composite Simpson’s rule are consistent with our expectations because its error is of order 4. By a similar argument to that of the composite Simpson’s rule error, we expect the error of the composite Boole’s rule to be of order 6. We see this is roughly the case.

\begin{verbatim}
f=@(x) sin(x.^2); a=0; b=5; c=sqrt(pi/2); It=c*fresnels(5/c);
for ii =1:6
  n=4*2^ii;
  nn(ii)=n;
  simp(ii)=simpf(a,b,n);
  bool(ii)=boolc(f,a,b,n);
end
Is=simp(6)
Ib=bool(6)
errs = abs(simp-It); errb = abs(bool-It);
figure; Ps=polyfit(log(nn(2:6)),log(errs(2:6)),1);
ords=-Ps(1)
loglog(nn,errs,'b*'); nn,exp(Ps(1)*log(nn)+Ps(2)),'--r')
legend('error','fit')
xlabel('n'); ylabel('err')
title(['(i) Error in Composite Simpson and Linear Fit with Slope = -', num2str(ords)])
figure; Pb=polyfit(log(nn(4:6)),log(errb(4:6)),1);
ordb=-Pb(1)
loglog(nn,errb,'b*'); nn,exp(Pb(1)*log(nn)+Pb(2)),'--r')
legend('error','fit')
xlabel('n'); ylabel('err')
\end{verbatim}
f=@(x) 1./(1+ x .^6); a=0; b=5; c=sqrt(3); It=(c*log((26+5*c)/(26-5*c))+2*atan(10+c)+2*atan(10-c)+4*atan(5))/12

for ii =1:8
    n=4*2^ii;
    nn(ii)=n;
    simp(ii)=simpc(f,a,b,n);
    bool(ii)=boolc(f,a,b,n);
end
Is=simp(8)
Ib=bool(6)
errs=abs(simp-It); errb=abs(bool-It);

figure; Ps=polyfit(log(nn(5:8)),log(errs(5:8)),1);
ords=-Ps(1)
loglog(nn,errs,'b*','nn',exp(Ps(1)*log(nn)+Ps(2)),'--r')
legend('error','fit')
xlabel('n'); ylabel('err')
title(['(ii) Error in Composite Simpson and Linear Fit with Slope= -', num2str(ords)]

figure; Pb=polyfit(log(nn(5:2:7)),log(errb(5:2:7)),1);
ordb=-Pb(1)
loglog(nn,errb,'b*','nn',exp(Pb(1)*log(nn)+Pb(2)),'--r')
legend('error','fit')
xlabel('n'); ylabel('err')
title(['(ii) Error in Composite Boole and Linear Fit with Slope= -', num2str(ordb)])
Problem 2. (a) Prove that the Bernoulli polynomials satisfy the reflection property:

\[ (-1)^j B_j(1 - x) = B_j(x), \quad j \geq 2. \]

(b) Prove this identity relating the Bernoulli polynomials and Bernoulli numbers:

\[ B'_j(x) = j[B_{j-1}(x) + B_{j-1}], \quad j \geq 2. \]

Note: These results can be used to give a general proof of the Euler-MacLaurin formula. See Ralston, A First Course in Numerical Analysis (McGraw-Hill, 1965).

(a) Write

\[
\sum_{j=1}^{\infty} B_j(1 - x) \frac{(-t)^j}{j!} = \frac{-t(e^{tx} - t - 1)}{e^t - 1} \\
= \frac{-t(e^{tx} - e^t)}{1 - e^t} \\
= \frac{t(e^{tx} - e^t)}{e^t - 1} \\
= \frac{t(e^{tx} - 1)}{e^t - 1} + \frac{t(1 - e^t)}{e^t - 1} \\
= \sum_{j=1}^{\infty} B_j(x) \frac{t^j}{j!} - t.
\]

Thus, by comparing coefficients of powers of \( t \), we find that \( B_1(1 - x) = 1 - B_1(x) \) and \((-1)^j B_j(1 - x) = B_j(x)\) for \( j \geq 2 \).

(b) Write

\[
\sum_{j=1}^{\infty} \frac{d}{dx} B_j(x) \frac{t^j}{j!} = \frac{d}{dx} \frac{t(e^{tx} - 1)}{e^t - 1} = \frac{t^2 e^{tx}}{e^t - 1} \\
= \frac{t^2(e^{tx} - 1)}{e^t - 1} + \frac{t^2}{e^t - 1} \\
= \sum_{j=1}^{\infty} [B_j(x) + B_j] \frac{t^{j+1}}{j!} + B_0 t \\
= \sum_{j=1}^{\infty} (j + 1) [B_j(x) + B_j] \frac{t^{j+1}}{(j + 1)!} + t \quad (\text{since } B_0 = 1) \\
= t + \sum_{j=2}^{\infty} j [B_{j-1}(x) + B_{j-1}] \frac{t^j}{j!}.
\]

Thus, \( B'_1(x) = 1 \) and for \( j \geq 2 \), \( B'_j(x) = j[B_{j-1}(x) + B_{j-1}] \), as desired.
Problem 3. (a) The midpoint rule \( I_M(f) = hf \left( \frac{a+b}{2} \right) \), \( h = b-a \) for evaluating the integral \( I(f) = \int_a^b f(x) \, dx \) can be shown to have the asymptotic error formula

\[
I(f) = I_M(f) + \frac{h^2}{12} f'(b) - f'(a) + O(h^4).
\]

Using this information, obtain a new numerical integration formula \( \tilde{I}(f) \) with a higher order of convergence by making a linear combination of \( I_M(f) \) and the trapezoidal rule \( I_T(f) = \frac{h}{2} [f(a) + f(b)] \). Write out the weights for this new formula \( \tilde{I}(f) \),

(b) Show that in Romberg integration, with \( T^{(0)}(h_k) \) the composite trapezoidal rule,

\[ T^{(1)}(h_k) = \frac{1}{3} \left[ 4T^{(0)}(h_k) - T^{(0)}(h_{k-1}) \right] \]

is the composite Simpson rule and

\[ T^{(2)}(h_k) = \frac{1}{15} \left[ 16T^{(1)}(h_k) - T^{(1)}(h_{k-1}) \right] \]

is the composite Boole rule.

(a) Recall that

\[
I(f) = I_T(f) - \frac{h^2}{12} [f'(b) - f'(a)] + O(h^4).
\]

Thus \( 3I(f) = 2I(f) + I(f) = 2I_M(f) + I_T(f) + O(h^4) \) and \( I(f) = [I_T(f) + 2I_M(f)]/3 + O(h^4) \) where \( I_T(f) + 2I_M(f) = h[f(a) + f(b)]/2 + 2hf(c) = h[f(a) + f(b)]/2 \). This is Simpson’s rule with increments of size \( h/2 \).

(b) Write composite trapezoidal with stepsize \( h_k \) as

\[
T^{(0)}(h_k) = \sum_{j=0}^{2^k-1} h_k f(x_{j}) + f(x_{j+1}) \]

and see that

\[
T^{(1)}(h_k) = \frac{1}{3} \left[ 4T^{(0)}(h_k) - T^{(0)}(h_{k-1}) \right] = \frac{1}{3} \sum_{j=0}^{2^k-1} h_k [f(x_{2j}) + f(x_{2j+1}) + f(x_{2j+2})],
\]

which is composite Simpson. Similarly, we further expand

\[
T^{(1)}(h_k) = \frac{1}{3} \sum_{j=0}^{2^k-2} h_k [f(x_{4j}) + 4f(x_{4j+1}) + 2f(x_{4j+2}) + 4f(x_{4j+3}) + f(x_{4j+4})],
\]

and write

\[
T^{(1)}(h_{k-1}) = \frac{1}{3} \sum_{j=0}^{2^k-2} 2h_k [f(x_{4j}) + 4f(x_{4j+2}) + f(x_{4j+4})],
\]

so that

\[
T^{(2)}(h_k) = \frac{16T^{(1)}(h_k) - T^{(1)}(h_{k-1})}{15} = \frac{2}{45} \sum_{j=0}^{2^k-2} h_k [7f(x_{4j}) + 32f(x_{4j+1}) + 12f(x_{4j+2}) + 32f(x_{4j+3}) + 7f(x_{4j+4})],
\]

which is composite Boole.
Problem 4. Use the MATLAB script romberg.m and the intrinsic function quad in order to compare Romberg integration and adaptive extrapolated Simpson's rule applied to the following integrals:

\[(i) \int_0^{2\pi} dx \exp(\cos x) \quad (ii) \int_0^1 \frac{dx}{1+\exp(-x^{200})}\]

Calculate each integral to a tolerance of $tol = 10^{-14}$ and record the number of function calls made by both algorithms. Explain your results, using the geometric and smoothness properties of the integrands.

Running the program for (i) evaluates the integral to about 7.954926521012845 for both methods, with 1025 function calls in Romberg and 2993 in adaptive Simpson. Clearly, Romberg is much superior to adaptive Simpson. This should be expected because the integrand is quite smooth, and Romberg works well to get highly accurate integrals of smooth functions.

Running the program for (ii) evaluates the integral to about 0.501211058377616 for both methods, with 8193 function calls in Romberg and 565 in adaptive Simpson. In contrast to the first case, adaptive Simpson is now greatly superior. The plot of the integrand shows a very sharp jump at $x = 1$, or a “near-discontinuity” where derivatives are very large. Adaptive methods handle this type of integrand more efficiently, by using a small $h$ only near the point $x = 1$, whereas Romberg with a uniform grid is forced to use a small $h$ everywhere.
Problem 5*. Show that the cubic Hermite interpolating polynomial on the interval \( x_k < x < x_{k+1} \) is given by

\[
H_2(x) = \frac{3hs^2 - 2s^3}{h^3}y_{k+1} + \frac{h^3 - 3hs^2 + 2s^3}{h^3}y_k + \frac{s^2(s - h)}{h^2}y_{k+1} + \frac{s(s - h)^2}{h^2}y_k
\]

with \( s = x - x_k \) and \( y_i = f(x_i), \ y'_i = f'(x_i), \ i = k, k+1 \) and that its integral over that interval is given by

\[
\int_{x_k}^{x_{k+1}} dx H_2(x) = \frac{h}{2}(y_{k+1} + y_k) - \frac{h^2}{12}(y'_{k+1} - y'_k).
\]

Thus, integrating the piecewise cubic Hermite interpolating polynomial yields the composite corrected trapezoidal rule.

First, we write

\[
H_2(x) = f(a) + (x - a)f'(a) + (x - a)^2 f[a, a, b] + (x - a)^3 (x - b) f[a, a, b, b],
\]

where

\[
f[a, a, b] = \frac{f[a, b] - f'(a)}{b - a}, \quad f[a, a, b, b] = \frac{f'(a) - 2f[a, b] + f'(b)}{(b - a)^2}.
\]

Thus

\[
H_2(x) := f(a) + sf'(a) + s^2 \left[ \frac{f(b) - f(a)}{h^2} - \frac{f'(a)}{h} \right] + s^2(s - h) \left[ \frac{f'(a) + f'(b) - 2(f(b) - f(a))}{h^3} \right] \]

\[
= \left[ 1 - \frac{s^2}{h^2} + \frac{2s^2(s - h)}{h^3} \right] f(a) + \left[ \frac{s^2}{h^2} - \frac{2s^2(s - h)}{h^3} \right] f(b) + \left[ s - \frac{s^2}{h} + \frac{s^2(s - h)}{h^2} \right] f'(a) + \frac{s^2(s - h)}{h^2} f'(b).
\]

Some algebra will get us our desired form:

\[
1 - \frac{s^2}{h^2} + \frac{2s^2(s - h)}{h^3} = \frac{h^3 - 3hs^2 + 2s^3}{h^3},
\]

\[
\frac{s^2}{h^2} - \frac{2s^2(s - h)}{h^3} = \frac{3hs^2 - 2s^3}{h^3},
\]

\[
s - \frac{s^2}{h} + \frac{s^2(s - h)}{h^2} = \frac{s[h^2 - 2hs + s^2]}{h^2}.
\]

Let \( a = x_k \) and \( b = x_{k+1} \), then we’re done. For computing the integral, use the obvious substitution \( x \mapsto s \) and write

\[
\int_{x_k}^{x_{k+1}} H_2(x) \, dx = \int_0^h H_2(s + x_k) \, ds
\]

\[
= \frac{h^4 - h^4/2}{h^3}y_{k+1} + \frac{h^4 - h^4/2}{h^3}y_k + \frac{h^4/4 - h^4/3}{h^2}y_{k+1} + \frac{h^4/4 - 2h^4/3 + h^4/2}{h^2}y_k
\]

\[
= \frac{h}{2}(y_{k+1} + y_k) - \frac{h^2}{12}(y'_{k+1} - y'_k).
\]
Problem 6*. (a) If a set of data \((x_i, y_i), i = 0, 1, 2, \ldots\) are given, prove inductively that

\[
p_\ell(0; x_0, \ldots, x_\ell) = \frac{x_0 p_{\ell-1}(0; x_1, \ldots, x_\ell) - x_\ell p_{\ell-1}(0; x_0, \ldots, x_{\ell-1})}{x_0 - x_\ell}
\]

where \(p_\ell(x; x_0, \ldots, x_\ell)\) is the polynomial of degree \(\leq \ell\) that interpolates the data on the points \(x_0, \ldots, x_\ell\).

(b) Use part (a) to show that if \(h_k = 2^{-k} h\), then the Romberg approximate integrators are given by

\[
T^{(\ell)}(h_k) = p_\ell(0; h_{k-\ell}, \ldots, h_k), \quad k \geq \ell
\]

where the polynomials interpolate the data \((h_k, T^{(0)}(h_k)), k = 0, 1, 2, \ldots\).

(a) From the course notes on Newton divided differences recall that

\[
P_\ell(x; x_0, \ldots, x_\ell) = \left(\frac{x - x_\ell}{x_\ell - x_0}\right)P_{\ell-1}(x; x_0, \ldots, x_{\ell-1}) + \left(\frac{x - x_0}{x_\ell - x_0}\right)P_{\ell-1}(x; x_1, \ldots, x_\ell).
\]

Take \(x = 0\) and we’re done.

(b) Using part (a),

\[
P_\ell(0; h_{k-\ell}, \ldots, h_k) = \frac{h P_{\ell-1}(0; h_{k-\ell+1}, \ldots, h_k) - 2^{-\ell} h P_{\ell-1}(0; h_{k-\ell}, \ldots, h_{k-1})}{h - 2^{-\ell} h}
\]

\[
= \frac{2^\ell P_{\ell-1}(0; h_{k-\ell+1}, \ldots, h_k) - P_{\ell-1}(0; h_{k-\ell}, \ldots, h_{k-1})}{2^\ell - 1},
\]

where \(P_0(0; h_k) = T^{(0)}(h_k)\). We thus see that \(P_\ell(0; h_{k-\ell}, \ldots, h_k)\) and \(T^{(\ell)}(h_k)\) satisfy identical recursion relations in \(\ell\) and, furthermore, have identical starting values for \(\ell = 0\). Thus we conclude that

\[
P_\ell(0; h_{k-\ell}, \ldots, h_k) = T^{(\ell)}(h_k).
\]

This result gives a simple interpretation of the Romberg approximation \(T^{(\ell)}(h_\ell)\) as the result of using the polynomial interpolant on the data \((h_0, T^{(0)}(h_0)), \ldots, (h_\ell, T^{(0)}(h_\ell))\) to extrapolate to \(h = 0\).