AME 60614: Numerical Methods
Fall 2021

Problem Set 1

Qihao Zhuo

All codes are submitted to Sakai, and only their file names will be mentioned in this report when they occurs.

1 Finite-Difference Schemes

1.1 $f_{i-2}, f_{i-1}, f_i, f_{i+1}, f_{i+2}$

With the MATLAB scripts $p1\_1.m$, coefficients solved by corresponding matrix equation are shown in the Tab. 1.

Table 1: Output of First Group

	a_1	a_2	a_3	a_4	a_5	$Truncation Error$	$Accuracy$
$f_i^{''}$	$\frac{-1}{12h^2}$	$\frac{4}{3h^2}$	$\frac{-5}{2h^2}$	$\frac{4}{3h^2}$	$\frac{-1}{12h^2}$	$\frac{h^4}{90} f_i^6$	$O\left(h^4\right)$
f_i^{iv}	$\frac{1}{h^4}$	$\frac{-4}{h^4}$	$\frac{6}{h^4}$	$\frac{-4}{h^4}$	$\frac{1}{h^4}$	$\frac{-h^2}{6} f_i^6$	$O\left(h^2\right)$
$f_i^{'''} - 3f_i^{'}$	$\frac{-\left(h^2+2\right)}{4h^3}$	$\frac{2h^2+1}{h^3}$	0	$\frac{-\left(h^2+1\right)}{h^3}$	$\frac{h^2+2}{4h^3}$	$\left(\frac{2h^2(h^2+2)}{15} - \frac{h^2(2h^2+1)}{60}\right) f_i^5$	$O\left(h^2\right)$

1.2 $f_i, f_{i+1}, f_{i+2}, f_{i+3}, f_{i+4}$

With the MATLAB scripts $p1\_2.m$, coefficients are shown in the Tab. 2.

Table 2: Output of Second Group

	a_1	a_2	a_3	a_4	a_5	Truncation Error	Accuracy
$f_i^{''}$	$\frac{35}{12h^2}$	$\frac{-26}{3h^2}$	$\frac{19}{2h^2}$	$\frac{-14}{3h^2}$	$\frac{11}{12h^2}$	$-\frac{5h^3}{6} f_i^5$	$O\left(h^3\right)$
f_i^{iv}	$\frac{1}{h^4}$	$\frac{-4}{h^4}$	$\frac{6}{h^4}$	$\frac{-4}{h^4}$	$\frac{1}{h^4}$	$-2h f_i^5$	$O\left(h\right)$
$f_i^{'''} - 3f_i^{'}$	$\frac{5\left(5h^2-2\right)}{4h^3}$	$\frac{-3\left(4h^2-3\right)}{h^3}$	$\frac{3\left(3h^2-4\right)}{h^3}$	$\frac{-\left(4h^2-7\right)}{h^3}$	$\frac{3\left(h^2-2\right)}{4h^3}$	Eq. 1	$O\left(h^2\right)$

$$TE = -\left(\frac{32h^2(h^2-2)}{5} + \frac{4h^2(3h^2-4)}{5} - \frac{h^2(4h^2-3)}{40} - \frac{81h^2(4h^2-7)}{40}\right) f_i^5 \tag{1}$$

1.3 $f_{i-4}, f_{i-3}, f_{i-2}, f_{i-1}, f_i$

With the MATLAB scripts $p1\_3.m$, coefficients are shown in the Tab. 3

$$TE = -\left(\frac{32h^2(h^2-2)}{5} + \frac{4h^2(3h^2-4)}{5} - \frac{h^2(4h^2-3)}{40} - \frac{81h^2(4h^2-7)}{40}\right) f_i^5 \tag{2}$$

Table 3: Output of Third Group

	a_1	a_2	a_3	a_4	a_5	Truncation Error	Accuracy
$f_i^{''}$	$\frac{11}{12h^2}$	$\frac{-14}{3h^2}$	$\frac{19}{2h^2}$	$\frac{-26}{3h^2}$	$\frac{35}{12h^2}$	$\frac{5h^3}{6}f_i^5$	$O\left(h^3\right)$
f_i^{iv}	$\frac{1}{h^4}$	$\frac{-4}{h^4}$	$\frac{6}{h^4}$	$\frac{-4}{h^4}$	$\frac{1}{h^4}$	$2hf_i^5$	$O\left(h\right)$
$f_i^{'''}-3f_i^{'}$	$\frac{-3\left(h^2-2\right)}{4h^3}$	$\frac{4h^2-7}{h^3}$	$\frac{-3\left(3h^2-4\right)}{h^3}$	$\frac{3\left(4h^2-3\right)}{h^3}$	$\frac{-5\left(5h^2-2\right)}{4h^3}$	Eq. 2	$O\left(h^2\right)$

1.4 $f_{i-1}, f_i, f_{i+1}, f_{i-1}^{'}, f_i^{'}, f_{i+1}^{'}$

With the MATLAB scripts $p1\_4.m$, coefficients are shown in the Tab. 4.

Table 4: Output of Fourth Group

	a_1	a_2	a_3	a_4	a_5	a_6	Truncation Error	Accuracy
$f_i^{''}$	$\frac{2}{h^2}$	$\frac{-4}{h^2}$	$\frac{2}{2h^2}$	$\frac{1}{2h}$	0	$\frac{-1}{2h}$	$\frac{h^4}{360}f_i^6$	$O\left(h^4\right)$
f_i^{iv}	$\frac{-12}{h^4}$	$\frac{24}{h^4}$	$\frac{-12}{h^4}$	$\frac{-6}{h^3}$	0	$\frac{6}{h^3}$	$-\frac{h^2}{15}f_i^6$	$O\left(h^2\right)$
$f_i^{'''}-3f_i^{'}$	$\frac{15}{2h^3}$	0	$\frac{15}{2h^3}$	$\frac{-3}{2h^2}$	$\frac{-3\left(h^2+4\right)}{h^2}$	$\frac{-3}{2h^2}$	$\frac{h^4}{840}f_i^7$	$O\left(h^4\right)$

2 Richardson Extrapolation

For fourth-order central-difference scheme, fivr points are required. Modifying codes in Sec. 1 to solve the scheme with points of $x-2h, x-h, x, x+h, x+2h$.

$$\frac{1}{12h}f_{i-2}-\frac{2}{3}f_{i-1}+\frac{2}{3}f_{i+1}-\frac{1}{12}f_{i+2}=f_i^{'}-\frac{h^4}{30}-\frac{h^6}{252}-\frac{h^8}{4320}-\frac{17h^{10}}{1995840}$$

Under central-difference scheme, the odd-order terms are zero. So to get sixth-, eighth- and tenth- order schemes, solving corresponding linear equations need two, three and four expressions. Those expressions are from $p2\_0\_*.mlx$.

$$\begin{aligned}
f_1^{'}=\frac{1}{6h}f_{i-2}-\frac{4}{3h}f_{i-1}+\frac{4}{3h}f_{i+1}-\frac{1}{6h}f_{i+2}&=f_i^{'}-\frac{h^4}{480}-\frac{h^6}{16128}-\frac{h^8}{1105920}-\frac{17h^{10}}{2043740160}\\
f_2^{'}=\frac{1}{12h}f_{i-2}-\frac{2}{3h}f_{i-1}+\frac{2}{3h}f_{i+1}-\frac{1}{12h}f_{i+2}&=f_i^{'}-\frac{h^4}{30}-\frac{h^6}{252}-\frac{h^8}{4320}-\frac{17h^{10}}{1995840}\\
f_3^{'}=\frac{1}{24h}f_{i-2}-\frac{1}{3h}f_{i-1}+\frac{1}{3h}f_{i+1}-\frac{1}{24h}f_{i+2}&=f_i^{'}-\frac{8h^4}{15}-\frac{16h^6}{63}-\frac{8h^8}{135}-\frac{272h^{10}}{31185}\\
f_4^{'}=\frac{1}{48h}f_{i-2}-\frac{1}{6h}f_{i-1}+\frac{1}{6h}f_{i+1}-\frac{1}{48h}f_{i+2}&=f_i^{'}-\frac{128h^4}{15}-\frac{1024h^6}{63}-\frac{2048h^8}{135}-\frac{278528h^{10}}{31185}
\end{aligned}$$

Through $p2\_1.mlx$, Richardson Extrapolation Algorithms are derived.

$$\begin{aligned}
f^{'}\left(x\right)&=\frac{16}{15}f_4^{'}-\frac{1}{15}f_3^{'}+O\left(h^6\right)\\
f^{'}\left(x\right)&=\frac{1024}{945}f_4^{'}-\frac{16}{189}f_3^{'}+\frac{1}{945}f_2^{'}+O\left(h^6\right)\\
f^{'}\left(x\right)&=\frac{724762624}{665713083}f_4^{'}-\frac{385839104}{4279584105}f_3^{'}+\frac{1251248}{855916821}f_2^{'}-\frac{149297}{29957088735}f_1^{'}+O\left(h^6\right)
\end{aligned}$$

2.1 b

With $p2\_2.f90$, output of different schemes is shown in Tab. 5.

Table 5: Output of Richardson Extrapolation

	$I_{numerical}$	I_{exact}	$AbsoluteError$
4-th	-0.73598998070968713	-0.73575888234288467	2.3109836680246243E-004
6-th	-0.73600395924846174	-0.73575888234288467	2.4507690557706852E-004
8-th	-0.73600748822810591	-0.73575888234288467	2.4860588522124250E-004
10-th	-0.73600855032649926	-0.73575888234288467	2.4966798361458764E-004

It seems the higher the scheme, the output is approaching another value rather than the exact value. It might because of the range limits on complicated fractions.

3 Integral Equations

3.1 a

With the trapezoid method, the integral term could be approximated,

$$\int_0^x K(x,t) f(t)\,\mathrm{d}t \approx \frac{\Delta t}{2}\left[K(x,t_0) f(t_0) + 2K(x,t_1) f(t_1) + ... + 2K(x,t_{n-1}) f(t_{n-1} + K(x,t_n) f(t_n))\right]$$

Taking $f(t_i) = f_i$, $g(x_i) = g_i$ and $K_{ij} = K(x_i, t_j)$.

$$\begin{aligned} f_0 &= g_0 \\ f_1 + \frac{\Delta t}{2}(K_{10} f_0 + K_{11} f_1) &= g_1 \\ f_2 + \frac{\Delta t}{2}(K_{20} f_0 + 2K_{21} f_1 + K_{22} f_2) &= g_2 \\ &... \\ f_n + \frac{\Delta t}{2}(K_{n0} f_0 + 2K_{n1} f_1 + ... + 2K_{n,n-1} f_{n-1} + K_{n,n} f_n) &= g_n \end{aligned}$$

Those equations could be seen as a linear equations system.

$$\begin{aligned} \boldsymbol{f} + \boldsymbol{M}\boldsymbol{f} &= \boldsymbol{g} \\ (\boldsymbol{I} + \boldsymbol{M})\,\boldsymbol{f} &= \boldsymbol{g} \\ \boldsymbol{f} &= (\boldsymbol{I} + \boldsymbol{M})^{-1}\,\boldsymbol{g} \end{aligned}$$

Through solving such matrix equation, a discret set of approximate values of $f(x)$ will be given. The trapezoid method has a accuracy of $O(h^2)$. From the matrix equation, accuracy of $f(x)$ also has the accuracy of $O(h^2)$.

$$Error = \frac{h^2(b-a)}{12} f^2(x) \approx 0.0017$$

3.2 b

Using $p3\_1.mlx$, with 10 intervals in $[0, 1]$, the output is shown below.

$$
\begin{aligned}
\boldsymbol{I}+\boldsymbol{M} = &1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \\
&0.0005\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \\
&0.002\ 0.001\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \\
&0.0045\ 0.004\ 0.001\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \\
&0.008\ 0.009\ 0.004\ 0.001\ 1\ 0\ 0\ 0\ 0\ 0\ 0 \\
&0.0125\ 0.016\ 0.009\ 0.004\ 0.001\ 1\ 0\ 0\ 0\ 0\ 0 \\
&0.018\ 0.025\ 0.016\ 0.009\ 0.004\ 0.001\ 1\ 0\ 0\ 0\ 0 \\
&0.0245\ 0.036\ 0.025\ 0.016\ 0.009\ 0.004\ 0.001\ 1\ 0\ 0\ 0 \\
&0.032\ 0.049\ 0.036\ 0.025\ 0.016\ 0.009\ 0.004\ 0.001\ 1\ 0\ 0 \\
&0.0405\ 0.064\ 0.049\ 0.036\ 0.025\ 0.016\ 0.009\ 0.004\ 0.001\ 1\ 0 \\
&0.05\ 0.081\ 0.064\ 0.049\ 0.036\ 0.025\ 0.016\ 0.009\ 0.004\ 0.001\ 1
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{g} &= (1.0000\ 0.8010\ 0.2969\ -0.2824\ -0.6894\ -0.7788\ -0.5644\ -0.1893\ 0.1629\ 0.3599\ 0.3679) \\
\boldsymbol{f} &= (1.0000\ 0.8005\ 0.2941\ -0.2904\ -0.7055\ -0.8049\ -0.6009\ -0.2352\ 0.1096\ 0.3010\ 0.3050)
\end{aligned}
$$

From $\boldsymbol{g}$ and $\boldsymbol{f}$, the absolute error magnitude is of 0.001. So the accuracy approximation in Problem 3.a is valified.

4 Gauss-Hermite Quadrature

4.1 a

With a simple code $p4\_0.f90$, $x^4\left[\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}\right] \approx 7.6945986\times10^{-19}$ when $x = 10$. Therefore the integral on $[-\infty, \infty]$ could be replaced by integral on $[-10, 10]$.

With code of Simpson's rule, $p4\_1.f90$, the three numerical integrals are shown in Tab. 6.

Table 6: Simpson method for different moments

	$I_{Simpson}$	I_{exact}	$AbsoluteError$	$RelativeError$	h
1	0.99999999945724893	1	5.4275106631251901E-010	5.4275106631251901E-010	0.1
x^2	0.99999999945724838	1	5.4275162142403133E-010	5.4275162142403133E-010	0.1
x^4	2.9999999983717434	3	1.6282566406289334E-009	5.4275221354297776E-010	0.1

For $f(x) = x^4$, $f^4 = 24$.

$$
\begin{aligned}
\frac{24\times20}{180n^4} &\le 0.000001 \\
n &\ge \left(\frac{8}{3}\times10^6\right)^{1/4} \approx 40.41
\end{aligned}
\tag{3}
$$

So to get an absolute error of 10^{-6}, in my case 41 points is the least.

4.2 b

The general form of Gauss-Hermite Quadrature is using e^{-x^2}, so a variable changing of $x = \sqrt{2}t$ is used to modify the function form.

$$\int_{-\infty}^{\infty}\left[\frac{1}{\sqrt{2\pi}}e^{\frac{-x^2}{2}}\right]\mathrm{d}x = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty}e^{-t^2}\mathrm{d}t$$
$$\int_{-\infty}^{\infty}x^2\left[\frac{1}{\sqrt{2\pi}}e^{\frac{-x^2}{2}}\right]\mathrm{d}x = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty}2t^2e^{-t^2}\mathrm{d}t$$
$$\int_{-\infty}^{\infty}x^4\left[\frac{1}{\sqrt{2\pi}}e^{\frac{-x^2}{2}}\right]\mathrm{d}x = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty}4t^4e^{-t^2}\mathrm{d}t$$
$$\int_{-\infty}^{\infty}\cos x\left[\frac{1}{\sqrt{2\pi}}e^{\frac{-x^2}{2}}\right]\mathrm{d}x = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty}\cos\left(\sqrt{2}t\right)e^{-t^2}\mathrm{d}t$$

For $f(x) = x^2$, $2N+1 = 2 \Rightarrow N+1 = 1.5$. So 2 points would be required to exactly compute it.

For $f(x) = x^4$, $2N+1 = 4 \Rightarrow N+1 = 2.5$. So 3 points would be required to exactly compute it.

The output of Gauss-Hermite Quadrature of second moments with 2 points and fourth moments with 4 points is shown in Tab. 7. The conclusion is verified.

Table 7: Gauss-Hermite Quadrature for different moments

	I_{G-H}	I_{exact}	$AbsoluteError$	$QuadraturePoints$
x^2	1	1	0	2
x^4	3	3	0	3

4.3 c

For $f(x) = \cos x,\ f^4(x) = \cos x$.

$$\frac{1\times 20}{180n^4} \leq 10^{-6}$$
$$n \geq \left(\frac{10^6}{9}\right)^{1/4} \approx 18.26 \tag{4}$$

So to get an absolute error of 10^{-6}, in my case 19 points is the least.

For $f(x) = \cos x,\ f(t) = \frac{\cos\left(\sqrt{2}t\right)}{\sqrt{\pi}}$. And the error formula for Gauss-Hermite is $error = \frac{n!\sqrt{\pi}}{2^n(2n)!}f^{2n}(\epsilon)$. to get an absolute error of 10^{-6}, 6 points is the least.

Modifying $p4\_2.mlx$ and $p4\_1.f90$, output is shown in Tab. 8 and the two conclusions are valified.

Table 8: Two Numerical Integrals of $\cos x$

	$I_{numerical}$	I_{exact}	$AbsoluteError$	$QuadraturePoints$
Simpson	0.60652631055192174	$e^{-1/2}$	4.3557993966159003E-006	19
Gauss-Hermite	0.606556817612611	$e^{-1/2}$	1.18710329088945E-006	6

5 Appendix

$p1\_1.m, p1\_2.m, p1\_3.m, p1\_4.m$ are four Matlab scripts to solve matrix equations for coefficients and truncation errors in Problem 1. The matrix equations come from Taylor series. In $p1\_4.m$ the matrices are modified because of the first derivative terms. Scripts and output are included in $p1$ folder.

$p2\_0\_1h.mlx, p2\_0\_2h.mlx, p2\_0\_4h.mlx, p2\_0\_8h.mlx$ are modified from scripts used in Problem 1 and used to solve expressions of fourth central differential expressions with different stencils. $p2\_1.mlx$ is to solve different matrix equations to get expressions of Richardson Extrapolation with different accuracy. $p2\_2.f90$ is a Fortran code used to calculate numerical derivatives with expressions from $p2\_1.mlx$. In the Fortran code, there are two user-defined functions to solve function value and central differential values.

$p3\_1.mlx$ is a Matlab script used to realize the algorithm of solving the Volterra integral equation with trapezoid method. $\boldsymbol{I}+\boldsymbol{M}$ and $\boldsymbol{g}$ are set up through loops and the values come from the trapezoid method. In the end, the script solve the matrix equation to get a set of discrete values of $\boldsymbol{f}$.

$p4\_0.f90$ is a simple code to test value of $x^4\left[\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}\right]$ to determine integral intervals. $p4\_1.f90$ is a Fortran code for Simpson rule. For different functions, the expressions could be modified directly. $p4\_2.mlx$ is a Matlab script of Gauss-Hermite Quadrature, in which there is a function module to solve weights and abscissas. In the main part, number of quadrature points and function expressions could be user-defined.

Latex files are include in the *tex* folder.