AMSC 660 Assignment 4

Jiaqi Leng

September 27, 2019

1 Problem 1

(a) From the discrete equations $\frac{1}{2}\frac{1}{\Delta x^2}(U_{j+1} - 2U_j + U_{j-1}) = F_j$ for $j = 1, 2, ..., n-1$, we can write the matrix A as

$$A = \frac{1}{2\Delta x^2}\begin{pmatrix} -2 & 1 & 0 & 0 & ... & 0 & 0 \\ 1 & -2 & 1 & 0 & ... & 0 & 0 \\ 0 & 1 & -2 & 1 & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & ... & 1 & -2 \end{pmatrix}.$$

In other words, A is a $(n-1)\times(n-1)$ matrix with all diagonal entries being $-\frac{1}{\Delta x^2}$, and $A_{j,j+1} = A_{j+1,j} = \frac{1}{2\Delta x^2}$ for all $j = 1, 2, ..., n-2$.

For a vector $r_k = (r_{kj})^T$ with $r_{kj} = \sin k\pi x_j$ for $j = 1, 2, ..., n-1$, we can check

$$Ar_k = \frac{1}{2\Delta x^2}(r_{k,j+1} - 2_{k,j} + r_{k,j-1})^T, \text{ where}$$

$$\begin{aligned} r_{k,j+1} - 2_{k,j} + r_{k,j-1} &= \sin\big[k\pi(x_j + \Delta x)\big] - 2\sin(k\pi x_j) + \sin\big[k\pi(x_j - \Delta x)\big] \\ &= 2\big[\cos(k\pi\Delta x) - 1\big]\sin(k\pi x_j) = 2\big[\cos(k\pi\Delta x) - 1\big]r_{k,j}, \end{aligned}$$

it follows that

$$Ar_k = \frac{\cos(k\pi\Delta x) - 1}{\Delta x^2} r_k.$$

We conclude that each r_k is an eigenvector of A with eigenvalues $\lambda_k = \frac{\cos(k\pi\Delta x)-1}{\Delta x^2}$.

(b) As A has $n-1$ linearly independent eigenvectors (because their corresponding eigenvalues are different), A is invertible. Multiply A^{-1} to $Ar_k =$

$\lambda_k r_k$, we have $A^{-1}Ar_k = \lambda_k A^{-1}r_k$, or equivalently, $A^{-1}r_k = (1/\lambda_k)r_k$. It turns out that r_k are also eigenvectors of A^{-1}, with eigenvalue $1/\lambda_k$. Note that $\{r_k\}_k$ form a basis for $\mathbb{R}^{n-1}$, then any vector $u \in \mathbb{R}^{n-1}$ can be represented as $u = \sum_{k=1}^{n-1} a_k r_k$, hence

$$A^{-1}u = A^{-1}\big(\sum_{k=1}^{n-1} a_k r_k\big) = \sum_{k=1}^{n-1} a_k A r_k = \sum_{k=1}^{n-1} \lambda_k^{-1} a_k r_k.$$

Note that A is a symmetric matrix, so by the spectral theorem, each two linearly independent eigenvectors are orthogonal to each other, then by the Pythagorean theorem, we have

$$\|A^{-1}u\|_{l^2} = \|\sum_{k=1}^{n-1} \lambda_k^{-1} a_k r_k\|_{l^2} = \sum_{k=1}^{n-1} |\lambda_k^{-1}| \|a_k r_k\|_{l^2}.$$

Therefore, we see

$$\frac{\|A^{-1}u\|_{l^2}}{\|u\|_{l^2}} = \frac{\sum_{k=1}^{n-1} |\lambda_k^{-1}| \|a_k r_k\|_{l^2}}{\sum_{k=1}^{n-1} \|a_k r_k\|_{l^2}} \leq \max_k\{|\lambda_k^{-1}|\} = \frac{\Delta x^2}{1-\cos(\pi \Delta x)},$$

and the maximal value is sharp: just choose $u = r_1$, and the eigenvalue $|\lambda_1^{-1}|$ is the largest. It follows that $\|A^{-1}\|_{l^2} = \frac{\Delta x^2}{1-\cos(\pi \Delta x)}$.

Recall the Taylor expansion of $\cos(x) = 1 - \frac{x^2}{2} + \frac{x^4}{4!} - \frac{x^6}{6!} + ...$, and $\Delta x = 1/n$, we get

$$\|A^{-1}\|_{l^2} = \frac{1/n^2}{\frac{\pi^2}{2n^2} - \frac{\pi^4}{4! n^4} + ...} = \frac{1}{\pi^2/2 + \frac{\pi^2}{n^2}\sum_i^\infty \left((-1)^i \frac{(\pi/n)^{2i}}{(2+2i)!}\right)} = \frac{1}{\pi^2/2 + \frac{\pi^2}{n^2} S},$$

the series $S = \sum_i^\infty (-1)^i \frac{(\pi/n)^{2i}}{(2+2i)!}$ is a convergent geometric series (for each fixed $n > 4 > \pi$), and S decreases as n increases. As $n \to \infty$, $\frac{\pi^2}{n^2} S \to 0$, we have

$$\lim_{n\to\infty} \|A^{-1}\|_{l^2} = \frac{1}{\pi^2/2} = \frac{2}{\pi^2}.$$

Using a similar argument, we can show that

$$\|A\|_{l^2} = \max_k\{|\lambda_k|\} = |\lambda_{n-1}| = \frac{1-\cos(\pi - \pi\Delta x)}{\Delta x^2} = \frac{1+\cos(\pi \Delta x)}{\Delta x^2}.$$

As $n \to \infty$, $\Delta x \to 0$, so $\cos(\pi \Delta x) \to 1$, and when n is large, $\lim_{n\to\infty} \frac{1+\cos(\pi\Delta x)}{\Delta x^2} \approx 2n^2$, thus the condition number

$$\lim_{n\to\infty} \kappa_{l^2}(A) = \|A\|_{l^2}\|A^{-1}\|_{l^2} \approx \frac{4}{\pi^2} n^2 = O(n^2).$$

(c) As u is smooth, we have the Taylor expansion:

$$\tilde{U}_{j+1} = u(x_j+\Delta x) = u(x_j)+u'(x_j)\Delta x+\frac{1}{2}u''(x_j)(\Delta x)^2+\frac{1}{3!}u'''(x_j)(\Delta x)^3+O(\|\Delta x\|^4),$$

$$\tilde{U}_{j-1} = u(x_j+\Delta x) = u(x_j)-u'(x_j)\Delta x+\frac{1}{2}u''(x_j)(\Delta x)^2-\frac{1}{3!}u'''(x_j)(\Delta x)^3+O(\|\Delta x\|^4),$$

use the relation $\tilde{U}_j = u(x_j)$, we have

$$\frac{1}{2(\Delta x)^2}(\tilde{U}_{j+1} - 2\tilde{U}_j + \tilde{U}_{j-1}) = \frac{1}{2}u''(x_j) + O(\|\Delta x\|^2)$$

$$= F_j + O(\|\Delta x\|^2). \text{ (Use } F(x_j) = \frac{1}{2}u''(x_j).)$$

Consequently, we have

$$\|R\|^2_{\Delta x} = \Delta x \sum_{j=1}^{n-1} R_j^2 = \Delta x(n-1)O(\|\Delta x\|^4) = O(\Delta x^4),$$

the last step following from $\Delta x(n-1) = (n-1)/n \leq 1$.

We conclude that $\|R\|_{\Delta x} = O(\Delta x^2) = O(1/n^2)$.

(d) $\|R\|_{l^2}$ is the l^2 vector norm, which can be computed as

$$\|R\|^2_{l^2} = \sum_{j=1}^{n-1} R_j^2 = \frac{1}{\Delta x}\|R\|^2_{\Delta x} \implies \|R\|_{\Delta x} = (\Delta x)^{1/2}\|R\|_{l^2}.$$

(e) Since $F = AU$, and $R = A\tilde{U}-F$, so $R = A(\tilde{U}-U)$. Hence, $\tilde{U}-U = A^{-1}R$, and

$$\|U - \tilde{U}\|_{l^2} \leq \|A^{-1}\|_{l^2}\|R\|_{l^2} \approx \frac{2}{\pi^2}\|R\|_{l^2}, \text{ (the last step from (b))}$$

thus,

$$\|U - \tilde{U}\|^2_{\Delta x} \approx (\frac{2}{\pi^2})^2 \Delta x \|R\|^2_{l^2} = O(\Delta x^4),$$

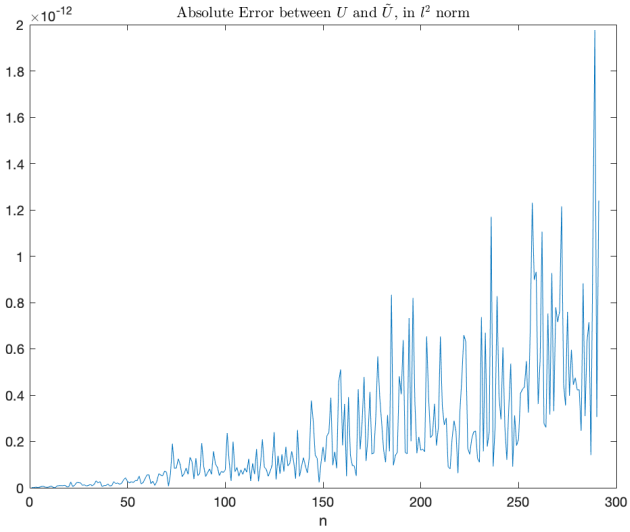

Figure 1: Problem 1 (f): Absolute errors between U_n and U, for $10 \leq n \leq 300$. The errors are always very small, bounded by 2×10^{-12}.

equivalently, $\|U - \tilde{U}\|_{\Delta x} = O(\Delta x^2)$.

(f) We define two MATLAB functions: "my_A.m" outputs the matrix A with dimension n, and "solver.m" solves the linear equation $AU = F$ for given A and F.

1.1 Test 1: the accuracy of the solver function

We will run "solver.m" for F with different sizes (from $n = 10$ to 300). At each fixed size n, we generate a $n \times 1$ random vector U_n, and let $F_n = AU_n$. Input F_n into the solver function and we get the computed value $\tilde{U}_n$. The error is computed as $\|U_n - \tilde{U}_n\|_{l^2}$. We plot the errors in Figure 1.

1.2 Test 2: the convergence rate of the numerical method

Now, we use the numerical method in problem 1 to solve differential equations. We choose two test cases:

$$u_1(x) = \sin(3\pi x), f_1(x) = -\frac{9\pi^2}{2}\sin(3\pi x);$$

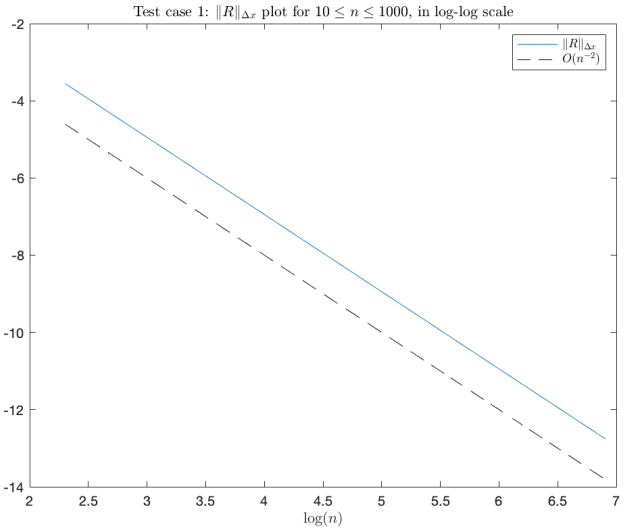

Figure 2: Problem 1 (f): Test Case 1: $u(x) = \sin(3\pi x)$, $f(x) = -\frac{9\pi^2}{2}\sin(3\pi x)$, in log-log scale.

$$u_2(x) = 2e^x + (2-2e)x - 2, f_1(x) = e^x.$$

And we run from $n = 10$ to $n = 1000$. The error $\|R\|_{\Delta x}$ for each test case is plotted in Figure 2 and Figure 3, respectively.

We can see clearly from Figure 2 and 3 that the convergence rate for $\|R\|_{\Delta x}$ is $O(n^{-2})$, the same as our theoretical results.

2 Problem 2

(a) We check $p(t) = p(0)S(t)$ satisfies the equation $\dot{p} = pA$:

$$\dot{p} = \frac{\mathrm{d}}{\mathrm{d}t}(p(0)S(t)) = p(0)\dot{S} = p(0)SA = p(t)A.$$

And $p(0) = p(0)S(0) = p(0)I = p(0)$, so $p(t) = p(0)S(t)$ is a solution to the IVP. It follows from the existence and uniqueness theorem for systems of ODE that this is the unique solution.

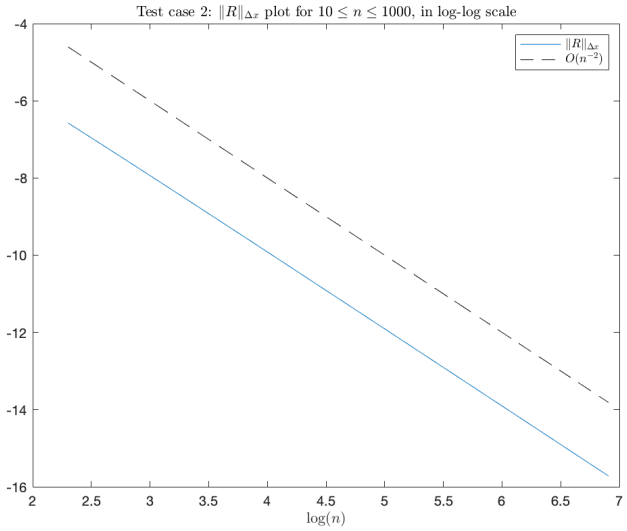

Figure 3: Problem 1 (f): Test Case 2: $u(x) = 2e^x + (2-2e)x - 2$, $f(x) = e^x$, in log-log scale.

(b) Here $\|\cdot\|$ is an arbitrary matrix norm.

$$\|\exp(B)\| = \|\sum_{k=0}^{\infty} \frac{1}{k!} B^k\| \leq \sum_{k=0}^{\infty} \frac{1}{k!} \|B^k\| \text{ (by the triangle inequality)}$$

$$\leq \sum_{k=0}^{\infty} \frac{1}{k!} \|B\|^k = \exp(\|B\|),$$

as long as $\|B\|$ is finite, we show that $\|\exp(B)\| \leq \exp(\|B\|) < \infty$, so the infinite sum of matrices indeed converges.

(c) We first check the initial condition. Let $O = (0)_{i,j}$, the zero matrix. It is obvious that $O^k = O$ for $k \geq 1$ and $O^0 = I$. It turns out that

$$S(0) = \exp(0A) = \exp(O) = \sum_{k=0}^{\infty} \frac{1}{k!} O^k = I + O + \frac{1}{2} O^2 + ... = I.$$

Next, we show $S = \exp(tA)$ satisfies the differential equation:

$$\frac{\mathrm{d}}{\mathrm{d}t} \exp(tA) = \frac{\mathrm{d}}{\mathrm{d}t} \Big(\sum_{k=0}^{\infty} \frac{1}{k!} (tA)^k \Big) = \sum_{k=0}^{\infty} \frac{1}{k!} k t^{k-1} A^k$$

$$= \sum_{k=0}^{\infty} \frac{1}{k!} t^k A^{k+1} = \big(\sum_{k=0}^{\infty} \frac{1}{k!} t^k A^k \big) A = \exp(tA)A = S(t)A.$$

(d) If A has the eigenvalue and eigenvector decomposition $A = R\Lambda L$, we have $LR = I$. Hence, $A^k = (R\Lambda L)(R\Lambda L)...(R\Lambda L) = R\Lambda^k L$. Thus,

$$\exp(tA) = \sum_{k=0}^{\infty} \frac{1}{k!}(tA)^k = \sum_{k=0}^{\infty} \frac{1}{k!}(Rt^k \Lambda^k L) = R\big(\sum_{k=0}^{\infty} \frac{1}{k!}(t\Lambda)^k \big) L = R \exp(t\Lambda) L.$$

(e) We choose parameters $n = 4, \lambda = 1, \mu = 4$. The diagonalization of A is $L\Lambda R$, r_k is the $k-$th column of the right eigenvector matrix R, and λ_k is the $k-$eigenvalue. The computed results are

$$r_1 = (-0.0431075696975893, 0.294356898204907, -0.660136756448556, 0.689721115161435)^T,$$

$$Ar_1 = (0.337464467902497, -2.30435152626345, 5.16783249022384, -5.39943148643996)^T,$$

$$\lambda_1 r_1 = (0.337464467902494 - 2.304351526263455.16783249022383 - 5.39943148643996)$$

$$\|Ar_1 - \lambda_1 r_1\|_{l^2} = 1.10046569643524e - 14.$$

$$r_2 = (-0.500000000000000, -0.500000000000000, -0.500000000000000, -0.500000000000000)^T,$$

$$Ar_2 = (0, 1.66533453693773e{-}16, -1.16573417585641e{-}15, 1.99840144432528e{-}15)^T,$$

$$\lambda_2 r_2 = (-1.66533453693773e - 16, -1.66533453693773e - 16,$$
$$-1.66533453693773e - 16, -1.66533453693774e - 16)^T$$

$$\|Ar_2 - \lambda_2 r_2\|_{l^2} = 2.41329901832056e - 15.$$

$$r_3 = (0.0588235294117646, -0.235294117647058, -0.235294117647060, 0.941176470588235)^T,$$

$$Ar_3 = (-0.294117647058822, 1.17647058823529, 1.17647058823530, -4.70588235294118)^T,$$

$$\lambda_3 r_3 = (-0.294117647058823, 1.17647058823529, 1.17647058823530, -4.70588235294118)^T$$

$$\|Ar_3 - \lambda_3 r_3\|_{l^2} = 4.68173296024904e - 15.$$

$$r_4 = (-0.0566262931618537, 0.0663418290945981, 0.414148201563851, 0.906020690589657)^T,$$

$Ar_4 = (0.122968122256452 - 0.144066116556554 - 0.899353000851206 - 1.96748995610322)^T$,

$\lambda_4 r_4 = (0.122968122256452 - 0.144066116556553 - 0.899353000851206 - 1.96748995610322)^T$

$$\|Ar_4 - \lambda_4 r_4\|_{l^2} = 1.20481259915584e - 15.$$

(f) For the $k-$th row of L, i.e., l_k, suppose $l_k A = p_k = (p_k^i)_{i=1,2,3,4}$. We compute $E(k) := \|l_k A - \frac{p_k^1}{l_k^1} l_k\|_{l^2}$ (we already checked all $l_k^1 \neq 0$). The computed values are

$$E(1) = 1.70208450860237e - 14, E(2) = 4.47840023880487e - 15,$$

$$E(3) = 5.79021488995814e - 15, E(4) = 1.91009991535709e - 15.$$

$E(k)$ turns out to be extremely small, which indicates that $l_k A = \frac{p_k^1}{l_k^1} l_k \|_{l^2}$, i.e., l_k are left eigenvectors of A.

Corresponding to $\lambda = 0$ is the left eigenvector l_2:

$l_2 = (-1.50588235294118, -0.376470588235294, -0.0941176470588238, -0.0235294117647057)$.

From part (e),

$p_\infty = (0.752941176470588, 0.188235294117647, 0.0470588235294118, 0.0117647058823529)$,

it is obvious that $l_2 = -2p_\infty$.

(g) We define the function '**fund_soln**', with two inputs A, the matrix, and t, the time. Output of this function is the computed result for u, solution to the differential equation. The matrix A is given by the function **mctr**, with input n (the size of the matrix), λ and μ (two parameters).

Set parameters $n = 4, \lambda = 1, \mu = 4$, $t = 1$, and n ranging from 4 to 80. The printed values are in Figure 4. We also plot the errors and condition numbers in log scale, see Figure 5. We can find in the plot that the errors and condition numbers both increases in exponential speed as n increases, and their exponential factors are almost the same (so we see two parallel lines).

(h) For the $k-$th column of A, it is $r_k = (0, ..., \lambda, -(\lambda + \mu), \mu)^T$ (for $1 < k < n$), and the first non-zero entry λ appears at the $(k-1)-$th component. So we may compute

$$(1, \frac{\lambda}{\mu}, \frac{\lambda^2}{\mu^2}, ..., \frac{\lambda^{n-1}}{\mu^{n-1}}) r_k = \frac{\lambda^{k-1}}{\mu^{k-1}}\lambda - \frac{\lambda^k}{\mu^k}(\lambda + \mu) + \frac{\lambda^{k+1}}{\mu^{k+1}}\mu = 0.$$

n	error	condition number		n	error	condition number
4	2.653884333	5.664158009		42	70187598992	1.5057E+12
5	2.971294744	10.91489442		43	1.38734E+11	3.01214E+12
6	3.252636772	21.54403429		44	2.74295E+11	6.02202E+12
7	5.339020638	42.93799961		45	5.42456E+11	1.20491E+13
8	9.936940341	85.8849035		46	1.07308E+12	2.41028E+13
9	17.5194953	171.9863739		47	2.12313E+12	4.81259E+13
10	33.26870315	344.5035162		48	4.20225E+12	9.66787E+13
11	63.63120933	690.0556443		49	8.32383E+12	1.93489E+14
12	122.5739325	1382.05855		50	1.65151E+13	3.88757E+14
13	235.2409035	2767.658858		51	3.29203E+13	8.20315E+14
14	452.8901403	5541.722674		52	7.31031E+13	1.88E+15
15	874.9593481	11095.02455		53	1.36613E+14	3.36E+15
16	1695.144957	22211.02358		54	2.64937E+14	7.05E+15
17	3288.069643	44460.14701		55	6.76875E+14	4.26E+16
18	6390.702483	88989.71962		56	1.41E+15	5.30E+16
19	12441.26436	178106.3535		57	3.58015E+14	9.94E+15
20	24251.31588	356445.45		58	8.25588E+14	4.73E+16
21	47332.34787	713319.0285		59	1.40E+16	2.38E+18
22	92486.80479	1427427.91		60	2.50E+17	2.42E+17
23	180906.4981	2856317.097		61	6.05E+16	4.46E+17
24	354193.7952	5715344.815		62	1.09E+17	1.98E+17
25	694073.863	11435730.37		63	1.95E+16	1.12E+17
26	1361189.53	22880856.9		64	1.29E+17	3.95E+17
27	2671487.789	45779267.95		65	8.64E+15	6.61E+17
28	5246696.475	91591400.56		66	1.02E+17	1.61E+18
29	10310883.83	183244452.6		67	1.08E+16	1.13E+18
30	20275157.35	366604779.2		68	1.70E+17	1.14E+18
31	39890914.84	733427738.3		69	5.63E+16	1.12E+18
32	78525343.14	1467266969		70	2.49E+16	1.41E+18
33	154652823.9	2935311576		71	1.35E+17	1.97E+18
34	304723083.9	5872091974		72	1.23E+17	1.13E+18
35	600676661.3	11746967211		73	1.59E+17	9.54E+17
36	1184550198	23499245288		74	6.70E+16	7.18E+17
37	2336866709	47008442583		75	1.33E+17	3.40E+18
38	4611826267	94036097898		76	1.56E+21	7.03E+18
39	9104636358	1.88108E+11		77	1.26E+21	4.75E+18
40	17980209209	3.76292E+11		78	4.27E+20	6.45E+18
41	35519186073	7.52715E+11		79	1.57E+20	8.76E+18
				80	5.21E+20	1.22E+19

Figure 4: Problem 2 (g): n, errors, and condition numbers.

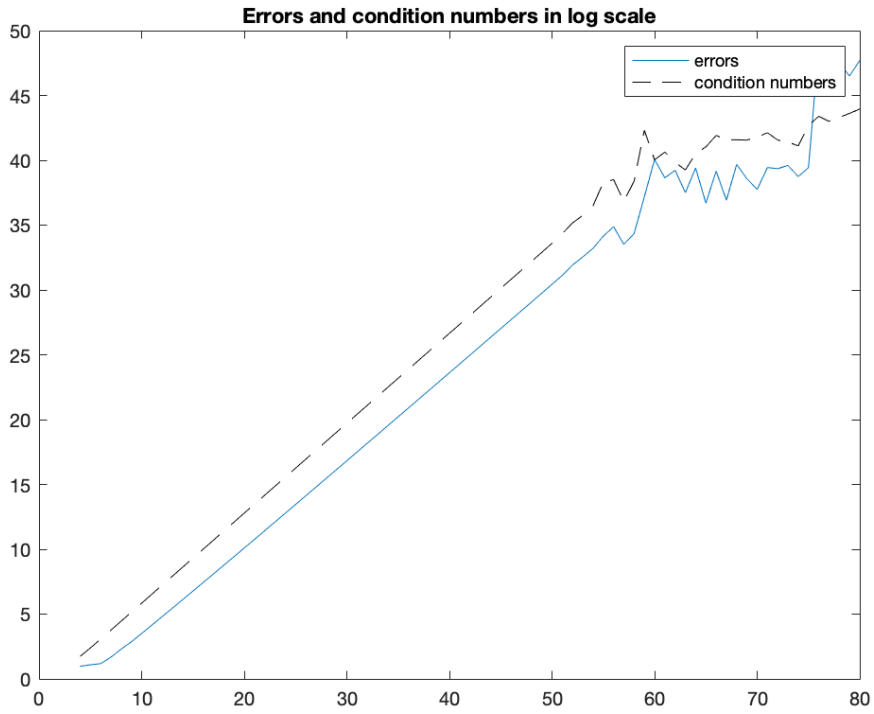

Figure 5: Problem 2 (g): errors and condition in log scale.

For $k = 1$,

$$(1, \frac{\lambda}{\mu}, \frac{\lambda^2}{\mu^2}, ..., \frac{\lambda^{n-1}}{\mu^{n-1}})(-\lambda, \mu, 0, ...0)^T = -\lambda + \lambda = 0;$$

for $k = n$,

$$(1, \frac{\lambda}{\mu}, \frac{\lambda^2}{\mu^2}, ..., \frac{\lambda^{n-1}}{\mu^{n-1}})(0, 0, ..., \lambda, -\mu)^T = \frac{\lambda^{n-1}}{\mu^{n-2}} - \frac{\lambda^{n-1}}{\mu^{n-1}}\mu = 0.$$

So we checked that $(1, \frac{\lambda}{\mu}, \frac{\lambda^2}{\mu^2}, ..., \frac{\lambda^{n-1}}{\mu^{n-1}})r_k = 0$ for all $0 \leq k \leq n$.

We compute the errors between $S(3n)$ and $\mathbf{1}p_\infty$ for $S(3n)$ given by different methods, the results are presented in Figure 6. It turns out that the build-in function **expm** gives better results, since the errors are significantly smaller than the eigenvalue method.

The stability of computing the matrix exponential using the ill-conditioned eigenvalue/eigenvector problem is poor. In our case, since the matrix is ill-conditioned, the computing of eigenvalue/eigenvectors does not perform well, leading to the failure of computing $S(3 * n)$.

3 Problem 3

(a) The function that prints the eigenvalues of $A(s)$ is 'ptd_eig(n, λ, μ, s)'. There are four inputs: n is the dimension of the matrix A, λ and μ are parameters that describe the matrix A, and s is the perturbation parameter. The output is a $n \times 1$ column vector, each entry is an eigenvector of $A(s)$.

The eigenvalues for $A(0)$ are 5.32907051820075e-15, -8.95075336237665, -8.80422606519425, -8.56402609673075, -8.23606797752564, -7.82842712472164, -7.35114100919281, -6.81596199893665, -6.23606797751829, -5.62573786014752, -5.00000000000794, -4.37426213983486, -3.76393202250390, -1.04924663761976, -1.19577393481836, -1.43597390324760, -3.18403800103640, -1.76393202250114, -2.17157287524956, -2.64885899083630.

The eigenvalues for $A(0.1)$ are -9.38143913049492 + 0.342830239868571i, -9.38143913049492 - 0.342830239868571i, -8.89545350458279 + 0.997861605100608i, -8.89545350458279 - 0.997861605100608i, -7.98783959879080 + 1.55211025804179i, -7.98783959879080 - 1.55211025804179i, -6.76122461439052 + 1.93735854009986i, -6.76122461439052 - 1.93735854009986i, -5.34634384079658 + 2.10454238371548i,

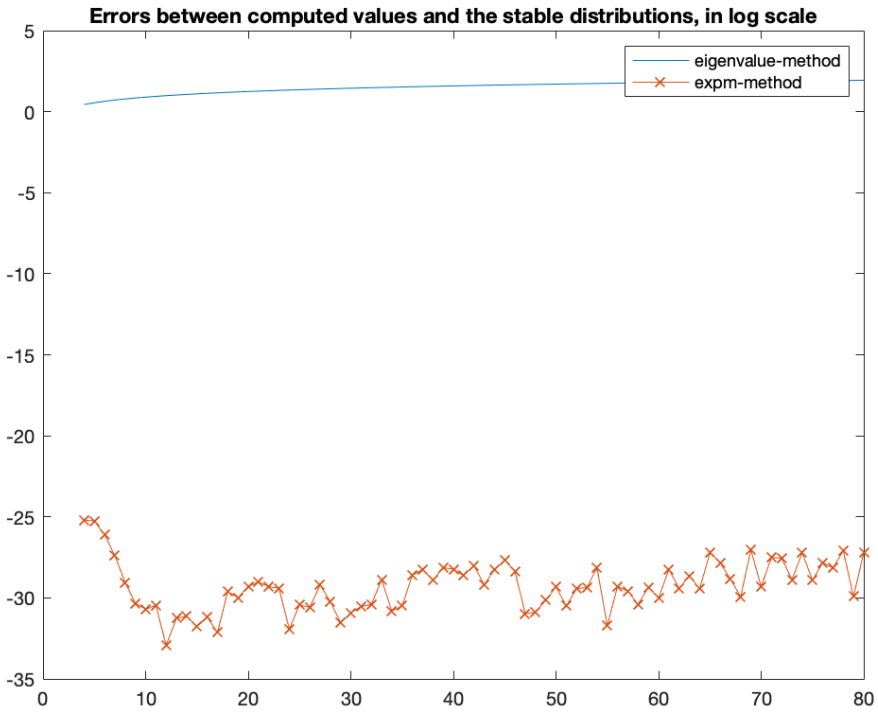

Figure 6: Problem 2 (h): errors comparision between the two methods, in log scale.

-5.34634384079658 - 2.10454238371548i, -3.89133004507189 + 2.03071403695992i, -3.89133004507189 - 2.03071403695992i, -2.54738357378247 + 1.72086666320877i, -2.54738357378247 - 1.72086666320877i, -1.45287447736953 + 1.20387989741754i, -1.45287447736953 - 1.20387989741754i, -0.0480368238569119 + 0.00000000000000i, -0.712267533372322 + 0.504881144798556i, -0.712267533372322 - 0.504881144798556i, -1.09965053883939 + 0.00000000000000i.

The condition number is defined to be the ratio between the change of output and the change of small input. By our perturbation $s = 0.1$, the resulting change in output is enormous, thus the eigenvalue eigenvector problem is ill-conditioned (with very large condition number).

(b) We formally differentiate $Ar_k = \lambda_k r_k$, and we get $\dot{A}r_k + A\dot{r}_k = \dot{\lambda_k} r_k + \lambda_k \dot{r}_k$. Multiply r_k^* to this equation and use the relation $r_k^* A = \lambda_k r_k^*$, we eventually get

$$\dot{\lambda}_k = \frac{r_k^* \dot{A} r_k}{\|r_k\|_{l^2}}.$$

Note that $\dot{A}(0) = B$, so $\dot{\lambda}_k = \frac{r_k^* B r_k}{\|r_k\|_{l^2}}$.

We use $\lambda_k(A(s))$ to denote the $k-$th eigenvalue of $A(s)$, hence $\Delta\lambda_k = \lambda_k(A(s)) - \lambda_j(A)$. To figure out the size of s that $\Delta\lambda_k$ can accurately approximate $s\dot{\lambda}_k$, we compute the quantity $E_k(s) = \|\Delta\lambda_k - s\dot{\lambda}_k\|_{l^2}$ for various scales of s.

We choose $n = 5$ and $n = 20$, and let s be $0.1/i$ for i running from 1 to 1000. The computed results are in Figure 7 and Figure 8. All the λ_k for $1 \leq k \leq n$ are plotted in the same figure. As we can see, when k is nearly 100 to 200, the errors reach their minimal. For larger k, i.e., for even smaller s, the approximation seems to have a systematic error, that converges to some non-zero constant for all k, though the convergence rates are different. It turns out that for larger n, the systematic error is also larger.

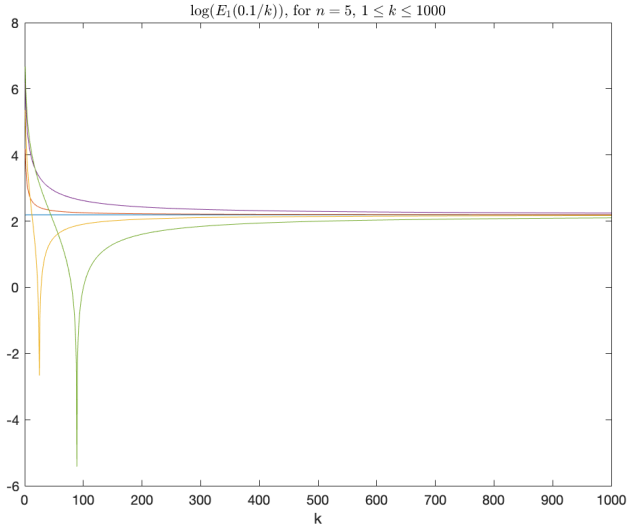

Figure 7: Problem 3 (b): When $n = 5$, $s = 0.1/k$ for $k \geq 1000$.

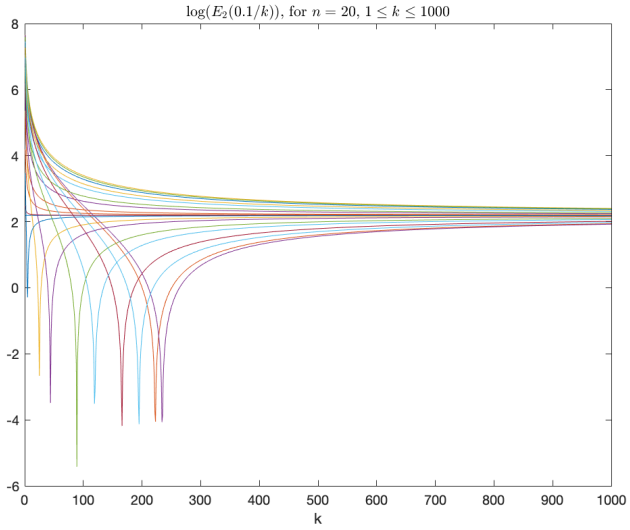

Figure 8: Problem 3 (b): When $n = 20$, $s = 0.1/k$ for $k \geq 1000$.