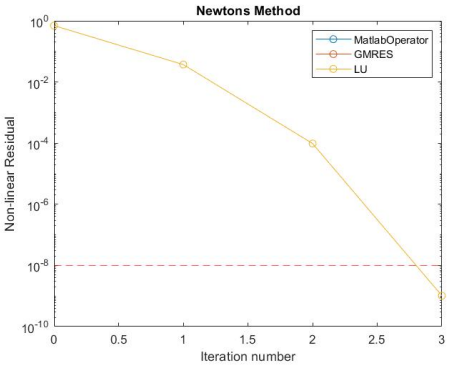
Advanced Numerical Analysis Homework 4

Keerthana C J

1 Exercise 2

As a part of Exercise 1, the Newton's method has been implemented. We solve the intersection of the given quadratic equations in 2D using both direct and iterative methods for inner iterations. The time for solution is about 0.0015s and the non-linear residual norm and the norm of the step values for each iteration are tabulated in the table in figure 1 . The convergence profiles of the direct and iterative methods overlap and is shown in the figure 1



Iter_No	x1	x2	resvec1	stepdiff1
1	0.5	1.75	0.69877	0.90139
2	0.51667	1.9417	0.037151	0.19239
3	0.51763	1.9319	9.7152e-05	0.0098335
4	0.51764	1.9319	1.0228e-09	3.098e-05

Figure 1: Using Newton's method to solve intersection two curves. The profiles on the right indicate the methods used for solving the non-linear system in each iteration. The table on the left shows the values at each iteration

2 Exercise 3

In a similar manner of Exercise 2, here also we a non-linear equation involving a Laplacian matrix. Similarly we plot convergence profiles which overlap for all the methods of solving the inner iterations, which are shown in figure 2. The iterative technique (GMRES) takes about 2.45s while the direct methods take roughly 3.6s. The convergence profile is plotted for the non-linear residual norm and the exit test is done on a relative residual norm.

3 Exercise 4

The eigen value problem is solved using Newton's method in Exercise 4. For a finding an eigenvalue pair (u, λ), given the constraint $||u|| = 1$ and an initial guess for $u\&\lambda$, we formulate a non-linear problem to be solved by the newton's method. As far the eigen value problem is given by the equation $(A - \lambda I)u = 0$ is linear, but if λ is unknown, we get a system of n non-linear equations with $n + 1$ unknowns for a matrix of dimension n. Hence, we introduce the constraint $||u|| = 1$, which leads to system of $n + 1$ non-linear equations with $n + 1$ unknowns. In the non-linear eigen value problem we have (1)

$$A(\lambda)u = 0 \tag{1}$$

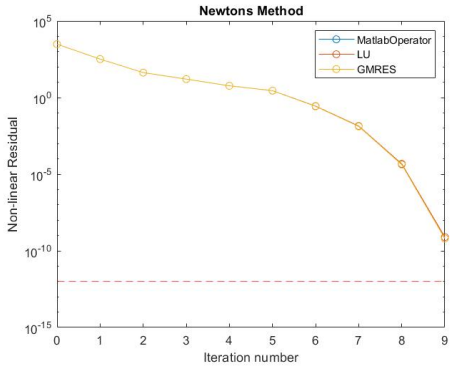


Figure 2: Solving a non-linear equation involving a Laplacian matrix

where, $A(\lambda)$ is a matrix of elements which depend on λ in a non-linear way. An example is the matrix polynomial, given as in (2)

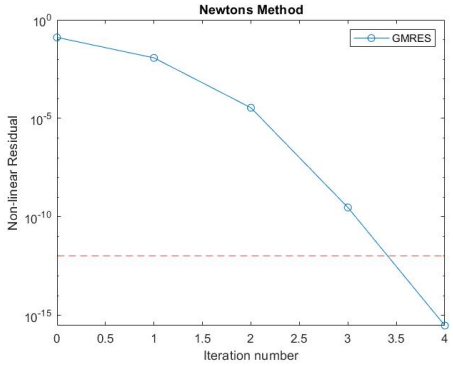


Figure 3: Eigen value problem solved using Newton's method

$$A(\lambda) = \sum_{k=0}^{d} \lambda^k A_k \tag{2}$$

For the special case $d = 1$, we have

$$A(\lambda) = A_0 - \lambda A_1; A_0 = A; A_1 = I \tag{3}$$

Hence, the non-linear system of equations are

$$(A - \lambda I)u = 0$$
$$u^T u - 1 = 0$$

Hence, the $F(x)$ and Jacobian of the systems are as follows

$$x = [u, \lambda]^T$$
$$F(x) := (A - \lambda I)u + u^T u - 1$$
$$F^{'}(x) = A - \lambda - u + 2u^T$$
$$F^{'}(x) = \begin{bmatrix} A - \lambda I & -u \\ 2u^T & 0 \end{bmatrix}$$

Following the above methods, $F(x)$ and Jacobian were computed and solved with an inital guess which is the original eigen value pair perturbed by a small value. The convergence profile of the system is obtained as in figure 3, where we see that the non-linear residual norm goes to 10^{-15}. This shows that the problem has been computed accurate to machine precision. The Jacobian becomes singluar if λ is not a simple eigen value. [1]

4 Exercise 5

The problem of exercise 2 is solved using QuasiNewton method with Bryoden's algorithm. The time for solution took about 0.103s.

5 Exercise 6

The Bratu problem is solved using Newton's, Broyden's and Inexect newton's method. The inexact newton's method has been solved using the four forcing term options which is specified as options 1,2,3 and 4. The convergence profiles are shown in the figure 4. The comparison of the four forcing terms are provided in the tables in the figures 5 & 6 The four options take a timing as in table 1 Hence option 4 forms the best option

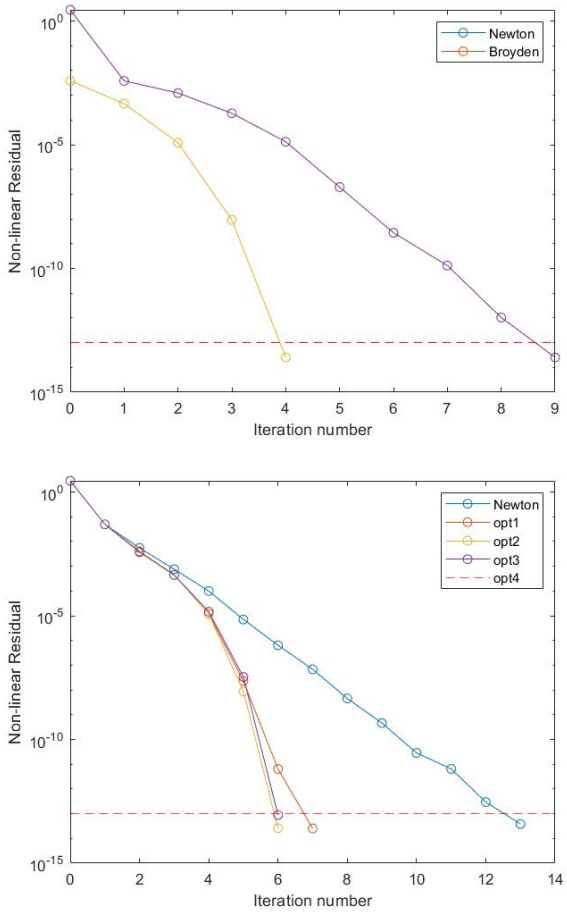


Figure 4: Bratu's problem convergence profiles

in terms of time and total number of linear iterations also.

Methods	Time(s)
Newton	10.83
Broyden	2.43
Inexact Newton	
option 1	4.03
option 2	2.62
option 3	2.68
option 4	2.34

Table 1: Comparison of times of different methods

k	forcingterm	residual	liniter
0	0.1	2.8307	0
1	0.1	0.049318	4
2	0.1	0.0055021	22
3	0.1	0.00076142	24
4	0.1	9.9688e-05	31
5	0.1	7.0708e-06	34
6	0.1	6.3989e-07	34
7	0.1	6.8585e-08	34
8	0.1	4.541e-09	32
9	0.1	4.609e-10	33
10	0.1	2.8569e-11	30
11	0.1	6.4898e-12	30
12	0.1	2.9415e-13	30
13	0.1	3.8281e-14	32

k	forcingterm	residual	liniter
0	0.1	2.8307	0
1	0.033333	0.049318	4
2	0.011111	0.0039798	26
3	0.0037037	0.00044814	32
4	0.0012346	1.193e-05	37
5	0.00041152	2.295e-08	44
6	0.00013717	6.3931e-12	62
7	4.5725e-05	2.534e-14	66

Figure 5: Left table is for option 1 and the right table is for option 2

References

[1] L.Ridgway Scott. *Numerical Analysis*. Princeton University Press, 2011. ISBN: 9780691146867.

k	forcingterm	residual	liniter
0	0.1	2.8307	0
1	0.046852	0.049318	4
2	0.0039579	0.0041663	25
3	0.00042599	0.00044842	39
4	1.1079e-05	1.1662e-05	44
5	8.1868e-09	8.6177e-09	64
6	2.4494e-14	2.5783e-14	129

k	forcingterm	residual	liniter
0	0.1	2.8307	0
1	0.00028837	0.049318	4
2	0.0054174	0.0037242	48
3	0.013368	0.00044179	29
4	0.0010363	1.4592e-05	30
5	5.0202e-06	3.3543e-08	46
6	6.6626e-12	8.883e-14	90

Figure 6: Left table is for option 3 and the right table is for option 4