**ECSE 543**

**Assignment 3**

**Numerical Methods in Electrical Engineering**

**AKEEL ALI: 260275389**

Table of Contents

[Question 1 3](#_Toc309389118)

[(a) 3](#_Toc309389119)

[Question 2 3](#_Toc309389120)

[(a) 3](#_Toc309389121)

[Question 3 3](#_Toc309389122)

[(a) 3](#_Toc309389123)

[Appendix 3](#_Toc309389124)

## Question 1

### (a)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Point 1 | Point 2 | Point 3 | Point 4 | Point 5 | Point 6 |
| B (T) | 0.0 | 0.2 | 0.4 | 0.6 | 0.8 | 1.0 |
| H (A/m) | 0.0 | 14.7 | 36.5 | 71.7 | 121.4 | 197.4 |

Table : The first 6 points to be interpolated

Using Wolfram Alpha to expand this expression gives:

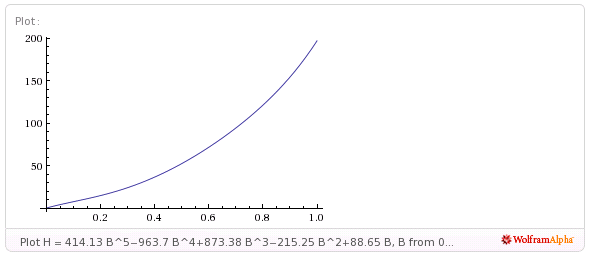


Figure : Plot of H (A/m) versus B (T) Interpolation (first 6 points)

Yes, this interpolation and its associated plot lie close to the true B versus H over this range as the chosen 6 adjacent points are closely spaced. Moreover, the plot resembles a theoretical H versus B plot in a hysteresis loop.

### (b)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Point 1 | Point 2 | Point 3 | Point 4 | Point 5 | Point 6 |
| B (T) | 0.0 | 1.3 | 1.4 | 1.7 | 1.8 | 1.9 |
| H (A/m) | 0.0 | 540.6 | 1062.8 | 8687.4 | 13924.3 | 22650.2 |

Table 2: The given 6 points to be interpolated

The Lagrange polynomial for the points above is given by:

Using Wolfram Alpha to expand this expression gives:

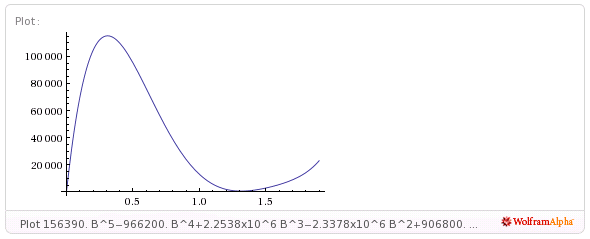


Figure 2: Plot of H (A/m) versus B (T) Interpolation (given 6 points)

No, this interpolation and its associated graph are not plausible H versus B relations as the chosen points to generate the Lagrange polynomial were widely spaced (leading to a divergent polynomial).

### (c)

EXPLAIN HERMITE POLYNOMIALS?

## Question 2

### (a)

The magnetic circuit suggests the following relation:

The reluctance of a magnetically uniform magnetic circuit element can be calculated as:

Thus, substituting the reluctance expressions into our original relation, we obtain:

Then, substituting the given values in the problem statement yields:

### (b)

We are to use the relation between B and H from Table 1 in the assignment instructions. To link to H, we know that , hence, we need only divide to get to B. In other words, we can use our table with:

For the Newton-Raphson method, we solve for in the following relation:

Where:

Since , , then:

The relation between H and B in the steel core is described with a piecewise interpolation of the following points:



is given by:

Our initial guess for the flux is . Linear interpolation (and extrapolation) is used for the B-H relation. The code to solve for the flux was written in C and is included in the appendix ().

The table below shows the results after each iteration of the Newton-Raphson algorithm until the residue drops below .

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| k | flux (Wb) | f | f' | |residue| |
| 0 | 0 | -8000 | 40009240 | 1.0056E-304 |
| 1 | 0.000199954 | 9356.654 | 301565700 | 1 |
| 2 | 0.000168927 | 1201.881 | 156953700 | 1.169582 |
| 3 | 0.000161269 | 1.42153E-12 | 156953700 | 0.1502351 |

Table : Results of the NR method after each iteration

The number of iterations are and the final flux value is (Wb).

### (c)

Successive substitution was implemented to solve our problem, but the method does not converge. The divergence is shown in the partial results below:

|  |  |  |
| --- | --- | --- |
| itr | flux (Wb) | f |
| 0 | 0 | -8000 |
| 1 | 8000 | 2.41253E+12 |
| 2 | -2.41253E+12 | -9.65233E+19 |
| 3 | 9.65233E+19 | 2.91081E+28 |
| 4 | -2.91081E+28 | -1.16459E+36 |
| 5 | 1.16459E+36 | 3.51202E+44 |
| … | … | … |
| 35 | 1.9467E+277 | 5.8707E+285 |
| 36 | -5.8707E+285 | -2.3488E+293 |
| 37 | 2.3488E+293 | 7.0832E+301 |
| 38 | -7.0832E+301 | -inf |
| 39 | inf | inf |
| 40 | nan | nan |

Table : Results for non-converging successive substitution

HOW DO I SOLVE SUCCESSIVE SUBSTITUTION DIVERGENCE?

## Question 3

### (a)

COPY ONENOTE

### (b)

FINISH : How to compute dJ/dv & AdJ/dvA\_T (see onenote)

COPY ONENOTE

ISSUE IN CODE: why do I have to define f as –f to get right answer (maybe I’m wrong with the sign in f to begin with?)

The algorithm implemented in question 2 for the Newton-Raphson method was adapted for a vectors in file (see appendix). The error measure was defined as the absolute average of . The algorithm was stopped when this error dropped below

The results of running the program are listed in the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| k | Voltage Diode A (V) | Voltage Diode B (V) | f[0] | f[1] | Error |
| 0 | 0 | 0 | -4.40E-04 | 0.00E+00 | 2.20E-04 |
| 1 | 0.145503 | 0.072751 | 1.98E-04 | 1.81E-04 | 1.89E-04 |
| 2 | 0.124114 | 0.081581 | 5.67E-05 | 5.52E-05 | 5.60E-05 |
| 3 | 0.11086 | 0.08925 | 1.02E-05 | 8.56E-06 | 9.38E-06 |
| 4 | 0.107695 | 0.090516 | 3.89E-07 | 3.33E-07 | 3.61E-07 |
| 5 | 0.107564 | 0.090571 | 6.18E-10 | 5.10E-10 | 5.64E-10 |
| 6 | 0.107563 | 0.090571 | 1.51E-15 | 1.28E-15 | 1.39E-15 |

Table : Results of the Newton Raphson method in solving f = 0

The voltage across diode A is found to be equal to 0.107563 (V) and is equal to 0.090671 (V) across diode B.

We observe that the method converges very rapidly. In fact, since:

Then the convergence is indeed quadratic.

## Question 4

### (a)

Add OneNote notes.

The table below summarizes the results obtained using one-point Gauss-Legendre integration as described above. The following plot displays the error versus N on a logarithmic scale.

|  |  |  |
| --- | --- | --- |
| N | Integral | |Error| |
| 1 | 0.479425539 | 0.019727845 |
| 2 | 0.46452136 | 0.004823666 |
| 3 | 0.461832842 | 0.002135147 |
| 4 | 0.460897009 | 0.001199315 |
| 5 | 0.460464752 | 0.000767058 |
| 6 | 0.460230183 | 0.000532489 |
| 7 | 0.460088826 | 0.000391132 |
| 8 | 0.459997113 | 0.000299419 |
| 9 | 0.459934249 | 0.000236555 |
| 10 | 0.459889291 | 0.000191597 |
| 11 | 0.45985603 | 0.000158336 |
| 12 | 0.459830736 | 0.000133041 |
| 13 | 0.459811051 | 0.000113357 |
| 14 | 0.459795434 | 9.77394E-05 |
| 15 | 0.459782834 | 8.51402E-05 |
| 16 | 0.459772523 | 7.48291E-05 |
| 17 | 0.459763978 | 6.62838E-05 |
| 18 | 0.459756817 | 5.91228E-05 |
| 19 | 0.459750757 | 5.30627E-05 |
| 20 | 0.459745583 | 4.78887E-05 |

Table : Results for one-point Gauss-Legendre Integral of sin(x) from 0 to 1 divided into N segments

Figure 3: Absolute Error in the Integral versus N (logarithm scales)

As can be seen from the table above, the value gets really close to the actual integral of as N increases. Moreover, the error decreases logarithmically with N.

### (b)

Add OneNote notes.

The same code as part (a) was run for , but with N going from 10 to 200 (by increments of 10).

|  |  |  |
| --- | --- | --- |
| N | Integral | |Error| |
| 10 | -0.965759065 | 0.034240935 |
| 20 | -0.982775472 | 0.0172245280 |
| 30 | -0.988493840 | 0.011506160 |
| 40 | -0.991361701 | 0.008638299 |
| 50 | -0.993085195 | 0.006914806 |
| 60 | -0.994235347 | 0.005764653 |
| 70 | -0.995057452 | 0.004942548 |
| 80 | -0.995674341 | 0.004325660 |
| 90 | -0.996154326 | 0.003845674 |
| 100 | -0.996538431 | 0.003461569 |
| 110 | -0.996852775 | 0.003147226 |
| 120 | -0.997114780 | 0.002885220 |
| 130 | -0.997336515 | 0.002663485 |
| 140 | -0.997526600 | 0.002473400 |
| 150 | -0.997691361 | 0.002308639 |
| 160 | -0.997835543 | 0.002164457 |
| 170 | -0.997962774 | 0.002037226 |
| 180 | -0.998075877 | 0.001924123 |
| 190 | -0.998177083 | 0.001822917 |
| 200 | -0.998268174 | 0.001731826 |

Table : Results for one-point Gauss-Legendre Integral of ln(x) from 0 to 1 divided into N segments

Table 4: Absolute Error in the Integral versus N (logarithm scales)

The tabulated results likewise show a convergence towards the actual value of -1. The error decreases logarithmically too, but not as fast as in (a). There is about a factor of 10 difference between the two convergences (for (a) we went from N=1 to 20, whereas here we go from N=10 to 200 to achieve similar error results).

### (c)

The natural logarithm function changes more rapidly as x gets closer to 0. Hence, the following experiment was attempted:

The number of segments over the integration range [0, 1] was limited to 10 and unevenly distributed to dedicate a greater number for ranges closer to 0. The range [0, 1] was divided into two: [0, w] & [w, 1]. The first range (closest to 0) was accorded segments out of the 10, and the second range was given the rest. is varies from 6 to 9 and w varies from 0.05 to 0.60 (in increments of 0.05).

A function was written to produce the effect of this experiment (see ). The following results are obtained from its output:

|  |  |  |  |
| --- | --- | --- | --- |
|  | w | Integral | |Error| |
| 6 | **0.05** | -0.96811 | 0.031885 |
| 6 | **0.10** | -0.97858 | 0.021423 |
| 6 | **0.15** | -0.98186 | 0.018141 |
| 6 | **0.20** | -0.9824 | 0.017603 |
| 6 | **0.25** | -0.98162 | 0.018382 |
| 6 | **0.30** | -0.98012 | 0.019885 |
| 6 | **0.35** | -0.97818 | 0.02182 |
| 6 | **0.40** | -0.97597 | 0.024032 |
| 6 | **0.45** | -0.97357 | 0.026427 |
| 6 | **0.50** | -0.97105 | 0.02895 |
| 6 | **0.55** | -0.96844 | 0.031563 |
| 6 | **0.60** | -0.96576 | 0.034241 |
| 7 | **0.05** | -0.95271 | 0.047287 |
| 7 | **0.10** | -0.96939 | 0.030613 |
| 7 | **0.15** | -0.97645 | 0.023549 |
| 7 | **0.20** | -0.9795 | 0.020501 |
| 7 | **0.25** | -0.9805 | 0.019501 |
| 7 | **0.30** | -0.98033 | 0.019669 |
| 7 | **0.35** | -0.97944 | 0.020556 |
| 7 | **0.40** | -0.97809 | 0.021912 |
| 7 | **0.45** | -0.97642 | 0.023583 |
| 7 | **0.50** | -0.97452 | 0.025475 |
| 7 | **0.55** | -0.97248 | 0.027525 |
| 7 | **0.60** | -0.97031 | 0.029688 |
| 8 | **0.05** | -0.91855 | 0.081446 |
| 8 | **0.10** | -0.94646 | 0.053539 |
| 8 | **0.15** | -0.96096 | 0.039043 |
| 8 | **0.20** | -0.96913 | 0.030873 |
| 8 | **0.25** | -0.97378 | 0.026225 |
| 8 | **0.30** | -0.97627 | 0.023725 |
| 8 | **0.35** | -0.97738 | 0.022616 |
| 8 | **0.40** | -0.97755 | 0.022445 |
| 8 | **0.45** | -0.97707 | 0.022929 |
| 8 | **0.50** | -0.97612 | 0.023878 |
| 8 | **0.55** | -0.97483 | 0.025166 |
| 8 | **0.60** | -0.9733 | 0.0267 |
| 9 | **0.05** | -0.81003 | 0.189974 |
| 9 | **0.10** | -0.86451 | 0.135488 |
| 9 | **0.15** | -0.89925 | 0.100754 |
| 9 | **0.20** | -0.92295 | 0.077051 |
| 9 | **0.25** | -0.93958 | 0.060422 |
| 9 | **0.30** | -0.95134 | 0.048658 |
| 9 | **0.35** | -0.95962 | 0.040383 |
| 9 | **0.40** | -0.96532 | 0.034676 |
| 9 | **0.45** | -0.9691 | 0.030898 |
| 9 | **0.50** | -0.97142 | 0.028582 |
| 9 | **0.55** | -0.97262 | 0.027385 |
| 9 | **0.60** | -0.97296 | 0.027044 |

Table : Integral results for different segment divisions

As can be seen from the table above, the experiment yields a higher accuracy with this scheme when compared to the equally distributed segments, where at N=10 the integral was -0.965759065.

In our case above, the highest accuracy was obtained when 6 of the 10 segments were allocated to the range [0, 0.20] and the rest to [0.20, 1]. The integral result was -0.9824. With a more rigorous experiment, one could pinpoint the exact value of and w that give the highest accuracy of the integral at N=10.

## Appendix