

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/319015766>

Excel Spreadsheet in Teaching Numerical Methods

Conference Paper · August 2017

CITATION

1

READS

5,112

1 author:



Harimi Djamila

71 PUBLICATIONS 415 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Investigation on the Dynamic Aspect of Natural Air Flow ...for Occupants Thermal Comfort [View project](#)



Investigating Indoor Environmental Variables on Thermal Perception and Comfort [View project](#)

PAPER • OPEN ACCESS

Excel spreadsheet in teaching numerical methods

To cite this article: Harimi Djamila 2017 *J. Phys.: Conf. Ser.* **890** 012093

View the [article online](#) for updates and enhancements.

Related content

- [Flying lantern sheds a brighter light on Archimedes' principle](#)
Mak Se-yuen and Leu Chun-lun
- [Software validation applied to spreadsheets used in laboratories working under ISO/IEC 17025](#)
J M Banegas and M W Orué
- [Modelling the motion of meteors in the Earth's atmosphere](#)
Hilário Rodrigues

Excel spreadsheet in teaching numerical methods

Harimi Djamila

Faculty of Engineering, Universiti Malaysia Sabah, Jalan UMS, Kota Kinabalu, Sabah Malaysia

E-mail: harimidjamila@gmail.com

Abstract. One of the important objectives in teaching numerical methods for undergraduates' students is to bring into the comprehension of numerical methods algorithms. Although, manual calculation is important in understanding the procedure, it is time consuming and prone to error. This is specifically the case when considering the iteration procedure used in many numerical methods. Currently, many commercial programs are useful in teaching numerical methods such as Matlab, Maple, and Mathematica. These are usually not user-friendly by the uninitiated. Excel spreadsheet offers an initial level of programming, which it can be used either in or off campus. The students will not be distracted with writing codes. It must be emphasized that general commercial software is required to be introduced later to more elaborated questions. This article aims to report on a teaching numerical methods strategy for undergraduates engineering programs. It is directed to students, lecturers and researchers in engineering field.

1. Introduction

Numerical analysis is a subject of extreme importance to engineers in many applications. Numerical methods are the study of set of procedure that uses numerical approximation for the solution of the problems in mathematical analysis. This is by considering approximate error estimates. Numerical methods became an independent mathematical discipline in the 20th Century. However, numerical analysis originated in ancient times and slowly progressed and flourished when computers became available in the late 1940s and early 1950s [1]. Currently numerical analysis is linked with computing machines. At present, in various engineering courses, students are required to solve engineering problems using numerical methods. This is important in acquiring useful engineering implications [2]. A course in numerical methods is also important in learning finite element method. In fact, a finite element problem requires matrix using numerical methods. A mathematical software such Matlab provides a platform for the students for fast learning by altering initial data without repeating all the computations [2, 3].

Currently, many books on numerical methods using Matlab are available to the students such [4, 5, 6, 7] and many others. However, Matlab and other commercial software are costly. Further many software are not user-friendly for uninitiated [8]. An attractive alternative for the students who are not majoring in computer science is Excel spreadsheets. This alternative allows ready experimentation with numerical algorithms [9]. In fact, this strategy is already used for learning finite element methods [10]. Therefore, the objective of this article is to report on a teaching strategy, which could be implemented for teaching numerical methods for undergraduates in Civil and Mechanical Engineering degree programs. This article will illustrate the usefulness in using Excel spreadsheet for finding roots using numerical methods. The spreadsheet approach helps the students to concentrate most of their



efforts on thinking about the methods at hand, instead of turning their efforts to the need for programming [10]. The developed strategy will not only benefit the student but also the lecturer. The developed strategy is about learning by observation. Each time the student change for instance the numerical problem, the student will acquire new knowledge by observations.

The developed strategy helps the lecturers to detect the evolution of the thinking process of their students when applying numerical methods. The students also will certainly reach the conviction of the importance of continuous learning in the future. Further, their Excel files will be useful for small engineering applications or at least for checking the solution obtained via their own Matlab program when possible or needed. The present illustration is based only on finding roots numerically; the developed concept can be applied for other chapters with slight adjustment to the procedure used in each chapter

2. Illustrative Example

The purpose of the selected example is to illustrate a new strategy in teaching numerical methods. This strategy will be useful for both the lecturers and to the students. By using this strategy, the lecturers will not only refer to the books for the preparation of their notes. They will be also learning about the methods by observation while teaching their students. For the suggested strategy, the lecturers are required to develop their own questions to be solved using numerical methods. These questions will be solved using Excel spreadsheet. Such procedure will helps in questioning for instance the advantages and limitation of the learned methods. Thus, the lecturer will be the author for his/her own tutorials. Thus, the lecturer will be independently learning by observation. Further, this procedure will also help the students to produce useful knowledge instead of consuming knowledge from books only. This is important when the lecturers are at the beginning of their carriers specifically with engineering backgrounds.

For illustration purpose, the chapter roots of equation were selected. The aim in teaching numerical methods is not only learning numerical methods. It is also important that the students should be involved in the thinking process when solving problems. For instance when using a bisection method, the student can easily observe that the root can be found when the function cross the x-axis. Now the lecturer may ask further question to the students how to write this information in an easy language, which may be accepted by a computer. The computer has no eyes to observe the situation. The lecturer can further inform the students that in many cases, when there is a root, the function cross the x-axis, the function will change the sign from positive to negative value and vis versa. Thus if we randomly select two initial values, x_l located before the root and x_u located after the root, in many cases $f(x_l).f(x_u) < 0$. Of course, there are exceptions, those are supposed to be given later to the student. The procedure of thinking prior application of any numerical method is very important in grasping the concepts and in understanding how the methods were developed. The strategy is for opening the students' creativity by observing first the simplicity of many numerical methods specifically at undergraduate level. Probably, by following this strategy, a few students may suggest additional requirements from any learned method in class or suggesting new methods.

Returning to our illustrative example, let us found the roots of a polynomial $f(x) = -10x^2 + 200x + 210$. Any polynomial function of second order is recommended to be used firstly for illustration purpose. This is to let most of students comfortable with the procedure. The students are expected also to learn the difference between the exact and approximate solution. Additionally, they are supposed to learn advantages and limitation of each method. The plot of the selected function is shown in Figure 1. It is apparent from the figure that the present function has two real roots: (-1) and (21). It is important to emphasise that the example was suggested without referring to any textbook and without an initial knowledge of the expected results when using numerical methods. The suggested equation was only checked to make sure that the function has two real roots to avoid exceptions. In the present illustration, four basic numerical methods for finding roots were considered. These are Bisection method, False-position method, Newton-Raphson method and Secant method. For the development of the algorithm, the author referred to Chapra and Canale [11] Book. The first known drawback of

numerical methods in finding roots is that, only one root can be estimated at a time within the selected range. The solution will be mostly approximate. It is apparent from figure 1 that the two roots are located within the interval $[-6, 22]$.

3. Prediction of the first root of an equation using various numerical methods

For the estimation of the first root which is (-1) by using the bisection method, two initial guesses $[-6, 4]$ were selected. It is apparent that the root is located within the selected interval. These two values are initial guesses. Although, the bisection method is known to converge very slowly to the solution, for the present case (figure 2), the method converged to the exact solution after one iteration only. This situation occurred because the two selected initial guesses when using Bisection method produced exact solution.

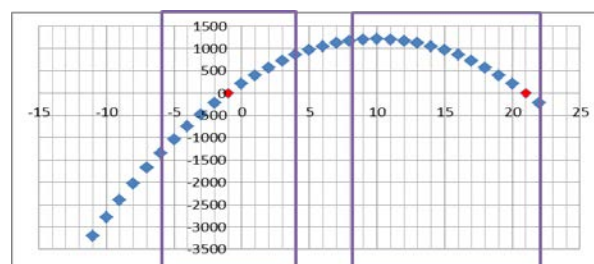


Figure 1. Selected function for finding roots.

The situation was different with the False-position method. It required 8 iterations to converge to the exact solution. The results are illustrated in figure 3. The Excel file used for all the calculation will be made available via ResearchGate. From the obtained results, it is apparent that the Bisection method converged faster than the False-position method.

i	Xl	Xu	Xr	f(xl)	f(xu)	f(xr)	f(xl)*f(xr)	Approximate Error %	Exact Error %
1	-6.0000	4.0000	-1.0000	-1350.0000	850.0000	0.0000	0.0000	/	0.00

Figure 2. Bisection method within the Interval $[-6, 4]$.

i	Xl	Xu	Xr	f(xl)	f(xu)	f(xr)	f(xl)*f(xr)	Approximate Error %	Exact Error %
1	-6.0000	4.0000	0.1364	-1350.0000	850.0000	237.0868	-320067.1488	/	113.64
2	-6.0000	0.1364	-0.7803	-1350.0000	237.0868	47.8478	-64594.5234	117.48	21.97
3	-6.0000	-0.7803	-0.9590	-1350.0000	47.8478	9.0067	-12159.0057	18.63	4.10
4	-6.0000	-0.9590	-0.9924	-1350.0000	9.0067	1.6730	-2258.5244	3.37	0.76
5	-6.0000	-0.9924	-0.9986	-1350.0000	1.6730	0.3100	-418.4810	0.62	0.14
6	-6.0000	-0.9986	-0.9997	-1350.0000	0.3100	0.0574	-77.5046	0.11	0.03
7	-6.0000	-0.9997	-1.0000	-1350.0000	0.0574	0.0106	-14.3530	0.02	0.00
8	-6.0000	-1.0000	-1.0000	-1350.0000	0.0106	0.0020	-2.6580	0.00	0.00
9	-6.0000	-1.0000	-1.0000	-1350.0000	0.0020	0.0004	-0.4922	0.00	0.00

Figure 3. False-Position method within the Interval $[-6, 4]$.

For the next step, the derivative of the equation was determined. This was required for estimating the root by using the Newton-Raphson method. The method requires only one initial guess. For estimating the approximate root, the first selected initial guess was (-6) . The procedure was also repeated with an initial guess equal to (4) . These two initial guesses were selected from the Bisection method. This is without any consideration of other factors. Newton-Raphson method is a popular method for finding the roots of an equation. It is widely known to converge fast to the solution. However, it is an open method, so it may diverge in some cases; thus, no solution will be found. Usually, when using the Newton-Raphson method, the initial guess is recommended to be close to the root when possible. The obtained results from the present cases are illustrated in figure 4. It is apparent

that the Newton-Raphson method converged faster than the False-position method. The situation is different with the Bisection method.

For the next step, the Secant method was selected to find the root within the range of $[-6, 4]$. The main difference between Secant and Newton method is this; the derivative of the function in Newton-Raphson method is replaced by numerical differentiation equation. This is very useful when the derivative of a function is difficult. Although, Secant method is an open method, it requires two initial guesses. The method may not necessary converge to the solution. The obtained results are shown in figure 5. From the figure, it is apparent that this method converged to the exact solution after 7 iteration.

i	x_i	$f(x_i)$	$f'(x_i)$	Approximate Error %	Exact Error %
0	-6.0000	-1350.0000	320.0000		500.00
1	-1.7813	-177.9785	235.6250	236.84	78.13
2	-1.0259	-5.7055	220.5181	73.63	2.59
3	-1.0000	-0.0067	220.0006	2.59	0.00
4	-1.0000	0.0000	220.0000	0.00	0.00

i	x_i	$f(x_i)$	$f''(x_i)$	Approximate Error %	Exact Error %
1	4.0000	850.0000	120.0000		500.00
2	-3.0833	-501.7361	261.6667	229.73	208.33
3	-1.1659	-36.7666	223.3174	164.47	16.59
4	-1.0012	-0.2711	220.0246	16.44	0.12
5	-1.0000	0.0000	220.0000	0.12	0.00
6	-1.0000	0.0000	220.0000	0.00	0.00

Figure 4. Newton-Raphson method with an initial guess of (-6) and (4) .

i	x_{i-1}	x_i	$f(x_{i-1})$	$f(x_i)$	Approximate Error %	Exact Error %
1	-6.0000	4.0000	-1350.0000	850.0000		
2	4.0000	0.136364	850.0000	237.0868	2833.33	500.00
3	0.1364	-1.358166	237.0868	-80.0794	110.04	113.64
4	-1.3582	-0.980821	-80.0794	4.2156	38.47	-35.82
5	-0.9808	-0.999693	4.2156	0.0676	1.89	1.92
6	-0.9997	-1	0.0676	-0.0001	0.03	0.03
7	-1.0000	-1	-0.0001	0.0000	0.00	0.00

Figure 5. Secant method within the Interval $[-6, 4]$.

4. Prediction of the second root of an equation using various numerical methods

The procedure of calculation is repeated for finding the second root of the equation, which is (21). The Bisection method was firstly used. The selected initial guesses are $[8, 22]$. The obtained results are shown in figure 6. This time, the method was only close to the solution. It required several iterations. Given that, the approximate absolute percent relative error is only an approximate estimation of the error, it is important to check that the function $f(x)$ value is close to 0 at the final approximate root. After completing the Bisection method, the procedure of calculation using the False-position method was repeated with the selected interval $[8, 22]$. The obtained results are shown in figure 7. Five iterations were required to reach the solution. In this illustration, the False-position method reached the exact solution. This was not the case of the Bisection method.

i	X_l	X_u	X_r	$f(x_l)$	$f(x_u)$	$f(x_r)$	$f(x_l)*f(x_r)$	Approximate Error %	Exact Error %
1	8.0000	22.0000	15.0000	1170.0000	-230.0000	960.0000	1123200.0000	/	28.57
2	15.0000	22.0000	18.5000	960.0000	-230.0000	487.5000	468000.0000	18.92	11.90
3	18.5000	22.0000	20.2500	487.5000	-230.0000	159.3750	77695.3125	8.64	3.57
4	20.2500	22.0000	21.1250	159.3750	-230.0000	-27.6563	-4407.7148	4.14	0.60
5	20.2500	21.1250	20.6875	159.3750	-27.6563	67.7734	10801.3916	2.11	1.49
6	20.6875	21.1250	20.9063	67.7734	-27.6563	20.5371	1391.8705	1.05	0.45
7	20.9063	21.1250	21.0156	20.5371	-27.6563	-3.4399	-70.6465	0.52	0.07
8	20.9063	21.0156	20.9609	20.5371	-3.4399	8.5785	176.1774	0.26	0.19
9	20.9609	21.0156	20.9883	8.5785	-3.4399	2.5768	22.1046	0.13	0.06
10	20.9883	21.0156	21.0020	2.5768	-3.4399	-0.4297	-1.1073	0.07	0.01
11	20.9883	21.0020	20.9951	2.5768	-0.4297	1.0740	2.7674	0.03	0.02
12	20.9951	21.0020	20.9985	1.0740	-0.4297	0.3222	0.3461	0.02	0.01
13	20.9985	21.0020	21.0002	0.3222	-0.4297	-0.0537	-0.0173	0.01	0.00
14	20.9985	21.0002	20.9994	0.3222	-0.0537	0.1343	0.0433	0.00	0.00

Figure 6. Bisection method within the Interval $[8, 22]$.

i	x_l	x_u	x_r	$f(x_l)$	$f(x_u)$	$f(x_r)$	$f(x_l)*f(x_r)$	Approximate Error %	Exact Error %
1	8.0000	22.0000	19.7000	1170.0000	-230.0000	269.1000	314847.0000	/	6.19
2	19.7000	22.0000	20.9401	269.1000	-230.0000	13.1438	3537.0057	5.92	0.29
3	20.9401	22.0000	20.9974	13.1438	-230.0000	0.5745	7.5506	0.27	0.01
4	20.9974	22.0000	20.9999	0.5745	-230.0000	0.0250	0.0144	0.01	0.00
5	20.9999	22.0000	21.0000	0.0250	-230.0000	0.0011	0.0000	0.00	0.00

Figure 7. False-position method within the Interval [8, 22].

The procedure for finding the root (21) was also made with the Newton-Raphson method. For the first calculation, the selected initial guess was (8). The second selected initial guess was (22). The results of both cases are plotted in figure 8. When the value of the initial guess was (8), the method failed to find the expected root which (21). However, the first root was found. This case is widely known when using Newton-Raphson method. Interestingly, the approximate error converged to (0) after 8th iteration. When an initial guess of (22) was selected, the Newton-Raphson method required little iteration to converge to the exact expected solution.

i	x_i	$f(x_i)$	$f'(x_i)$	Approximate Error %	Exact Error %
1	8.0000	1170.0000	40.0000		61.90
2	-21.2500	-8555.6250	625.0000	137.65	201.19
3	-7.5610	-1873.8872	351.2200	181.05	136.00
4	-2.2256	-284.6613	244.5127	239.72	110.60
5	-1.0614	-13.5536	221.2287	109.68	105.05
6	-1.0002	-0.0375	220.0034	6.13	104.76
7	-1.0000	0.0000	220.0000	0.02	104.76
8	-1.0000	0.0000	220.0000	0.00	104.76

i	x_i	$f(x_i)$	$f'(x_i)$	Approximate Error %	Exact Error %
1	22.0000	-230.0000	-240.0000		2300.00
2	21.0417	-9.1840	-220.8333	4.55	2204.17
3	21.0001	-0.0173	-220.0016	0.20	2200.01
4	21.0000	0.0000	-220.0000	0.00	2200.00
5	21.0000	0.0000	-220.0000	0.00	2200.00
6	21.0000	0.0000	-220.0000	0.00	2200.00
7	21.0000	0.0000	-220.0000	0.00	2200.00
8	21.0000	0.0000	-220.0000	0.00	2200.00
9	21.0000	0.0000	-220.0000	0.00	2200.00

Figure 8. Newton-Raphson method with an initial guess of (8) and (22).

The selected initial guess in this case was close to the root. The approximate root was (21). It is worth making comparison between False-position and Newton-Raphson methods by considering the approximate error. It is apparent that the Newton-Raphson method with an initial guess of (22) reached faster the root than the False-position method. So, what will be the case with the Secant method? The procedure was repeated to determine the root by using two initial guesses. Those are (8) and (22). The results are plotted in figure 9. It is apparent that the method converge to the exact solution after 6 iteration. It is obvious that the initial guess (22) is closer to the solution than (8). The convergence was faster than the Newton-Raphson method with an initial guess equal to (8) but slower than the Newton-Raphson method with an initial guess equal to (22).

i	x_{i-1}	x_i	$f(x_{i-1})$	$f(x_i)$	Approximate Error %	Exact Error %
1	8.0000	22.0000	1170.0000	-230.0000		
2	22.0000	19.7	-230.0000	269.1000	11.68	4.76
3	19.7000	20.94009	269.1000	13.1438	5.92	6.19
4	20.9401	21.00377	13.1438	-0.8303	0.30	0.29
5	21.0038	20.99999	-0.8303	0.0023	0.02	0.02
6	21.0000	21	0.0023	0.0000	0.00	0.00

Figure 9. Secant method within the Interval [8, 22].

5. Further numerical analysis

In the last section, two initial guesses were selected to bracket both roots. This is by using Bisection and False position methods. The selected range was [-2, 22]. The results are plotted in figures 10 and 11. Although, the Bisection method converged slowly to the first root, the situation was different with False-position method. For this last case, it was expected that the method will converge to either (-1) or (21). However, this was not the case. The $f(x)$ values of the two selected initial guesses were the same. Therefore, it was not possible to find out the approximate solution given there is a division by zero.

i	Xl	Xu	Xr	f(xl)	f(xu)	f(xr)	f(xl)*f(xr)	Approximate Error %
1	-2.0000	22.0000	10.0000	-230.0000	-230.0000	1210.0000	-278300.0000	/
2	-2.0000	10.0000	4.0000	-230.0000	1210.0000	850.0000	-195500.0000	150.00
3	-2.0000	4.0000	1.0000	-230.0000	850.0000	400.0000	-92000.0000	300.00
4	-2.0000	1.0000	-0.5000	-230.0000	400.0000	107.5000	-24725.0000	-300.00
5	-2.0000	-0.5000	-1.2500	-230.0000	107.5000	-55.6250	12793.7500	-60.00
6	-1.2500	-0.5000	-0.8750	-55.6250	107.5000	27.3438	-1520.9961	-42.86
7	-1.2500	-0.8750	-1.0625	-55.6250	27.3438	-13.7891	767.0166	-17.65
8	-1.0625	-0.8750	-0.9688	-13.7891	27.3438	6.8652	-94.6651	-9.68
9	-1.0625	-0.9688	-1.0156	-13.7891	6.8652	-3.4399	47.4336	-4.62
10	-1.0156	-0.9688	-0.9922	-3.4399	6.8652	1.7181	-5.9103	-2.36
11	-1.0156	-0.9922	-1.0039	-3.4399	1.7181	-0.8595	2.9567	-1.17
12	-1.0039	-0.9922	-0.9980	-0.8595	1.7181	0.4296	-0.3693	-0.59
13	-1.0039	-0.9980	-1.0010	-0.8595	0.4296	-0.2149	0.1847	-0.29
14	-1.0010	-0.9980	-0.9995	-0.2149	0.4296	0.1074	-0.0231	-0.15
15	-1.0010	-0.9995	-1.0002	-0.2149	0.1074	-0.0537	0.0115	-0.07

Figure 10. Bisection method with initial guesses [-2, 22].

i	Xl	Xu	Xr	f(xl)	f(xu)	f(xr)	f(xl)*f(xr)
1	-2.0000	22.0000	#DIV/0!	-230.0000	-230.0000	#DIV/0!	#DIV/0!
2	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!
3	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!

Figure 11. False-position method with initial guesses [-2, 22].

For the case of Newton-Raphson, the method converged to the root which was closer to the initial guess, which is (-1). The results are plotted in figure 12. For the Secant method, similar problem occurred as for False-position method. There is a division by zero. This is illustrated also in figure 12. Therefore, further calculation was not possible. It is important to mention such cases cannot be necessary learned from book. The benefit in using Excel is clearly shown in the present illustrative example which is learning by observation.

i	xi	f(xi)	f'(xi)	Approximate Error %
1	-2.0000	-230.0000	240.0000	
2	-1.0417	-9.1840	220.8333	92.00
3	-1.0001	-0.0173	220.0016	4.16
4	-1.0000	0.0000	220.0000	0.01
5	-1.0000	0.0000	220.0000	0.00

(a) Newton-Raphson method

i	xi-1	xi	f(xi-1)	f(xi)	Approximate Error %
1	-2.0000	22.0000	-230.0000	-230.0000	
2	22.0000	#DIV/0!	-230.0000	#DIV/0!	#DIV/0!
3	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!
4	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!
5	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!

(b) Secant method**Figure 12.** Newton-Raphson method with an initial guess (-2) & Secant method with initial guesses [-2, 22].

At this stage, students are ready for more advanced numerical methods in finding roots and for writing their first Matlab codes. For any unexpected issue with their Matlab program, the student can always refer back to Excel file to check carefully the solution.

6. Conclusion

This article focuses on using Excel spreadsheet in numerical methods as a teaching strategy for undergraduate students. The strategy is about learning many methods using one single equation to observe the advantages and limitations of each numerical method. The developed strategy helps the lecturers to detect the evolution of the thinking process of their students when applying numerical methods. An illustrative example about finding roots using four numerical methods was suggested. The developed concept can be applied for other chapters with slight adjustment to the procedure used in each chapter. The developed strategy goes beyond learning from textbooks only. The students will learn from their own experience with the first application used in numerical methods. The developed

strategy will be of great help to the new lecturers by acquiring new knowledge with each application. The students also will certainly reach the conviction of the importance of long life learning. Further, their Excel files will be useful for small engineering applications or at least for checking the solution obtained via their own Matlab program when needed.

References

- [1] Gutknecht M H 2010 *Seminar on Numerical analysis in Zurich – 50 years ago* (Seminar für Angewandte Mathematik, Eidgenössische Technische Hochschule, Rämistr)
- [2] Kim A S, Park C, Park SH 2003 Development of Web-based engineering numerical software (WENS) Using MATLAB *Applications to Linear Algebra* (Computer Applications in Engineering Education) pp 67-74
- [3] Martín-Caraballo A M & Tenorio-Villalón A F 2015. *Teaching Numerical Methods for Non-linear Equations with GeoGebra-Based Activities* (Mathematics Education vol 2) pp 53-65
- [4] Matthews JH. and Fink KD 2004 *Numerical Methods using MATLAB*. (New Jersey: Pearson.) 4th Edition.
- [5] Young T & Martin J M 2017 *Introduction to Numerical Methods and Matlab Programming for Engineers* (Department of Mathematics: Ohio University Athens)
- [6] Yang WY, Cao W, Chung T, Morris J 2005 *Applied Numerical Methods Using MATLAB* (John Wiley & Sons, Inc).
- [7] Chapra SC 2008 *Applied Numerical Methods with MATLAB for Engineers and Scientists*. (New York: McGraw Hill) 2nd Edition
- [8] Zuraini O, Abdul S b S 2010 *Proc. Conf. on Numerical Method Calculator for Engineering Education* (NUMCeE Malaysian Technical Universities Conference on Engineering and Technology Bayview Hotel, Melaka, Malaysia).
- [9] Wyk C J V 2006 *Using Spreadsheets to Learn Numerical Methods*. (eJSiE vol 1) pp148-157.
- [10] Kand T and L Morgan 2005 *Proc. On The Application of EXCEL in Teaching Finite Element Analysis to Final Year Engineering Students* (ASEE/A aeE 4th Global Colloquium on Engineering Education by Australasian Association for Engineering Education).
- [11] Chapra SC and Canale R P 2015 *Numerical Methods for Engineers* (New York: McGraw Hill) 7th Edition.