

Teaching and Learning Numerical Analysis and Optimization: A Didactic Framework and Applications of Inquiry-based Learning

Pantelis Z. Lappas¹ & Manolis N. Kritikos¹

¹School of Business, Athens University of Economics and Business, Athens, Greece

Correspondence: Dr. Pantelis Z. Lappas, Management Science Laboratory (MSL), Athens University of Economics and Business, 47a Evelpidon Str. and 33 Lefkados Str., GR-113 62, Athens, Greece. Tel: 30-210-8203-659. E-mail: pzlappas@aueb.gr

Received: December 18, 2017

Accepted: January 5, 2018

Online Published: February 7, 2018

doi:10.5539/hes.v8n1p42

URL: <https://doi.org/10.5539/hes.v8n1p42>

Abstract

The main objective of this paper is to propose a didactic framework for teaching Applied Mathematics in higher education. After describing the structure of the framework, several applications of inquiry-based learning in teaching numerical analysis and optimization are provided to illustrate the potential of the proposed framework. The framework is based on the Process of Scientific Inquiry (PoSI), while it consists of three pillars, each characterized by the use of a particular cognitive tool: Algorithm for presenting a list of steps to follow in order to solve a problem, CMAP software for constructing concept maps and MATLAB software for computer programming. In addition to this, a WebQuest Scenario can be used as an “auxiliary” cognitive tool by providing students with the opportunity to combine technology (e.g., MATLAB and CMAP software) with educational concepts (e.g., optimization), and to incorporate inquiry-based learning (i.e., PoSI). Introducing these cognitive tools to the design of the proposed didactic framework provides considerable potential of knowledge consolidation with reference to solving complex numerical problems using efficient algorithms.

Keywords: process of scientific inquiry, numerical methods, numerical analysis, optimization, concept maps, computer programming, algorithms

1. Introduction

In recent years, the inquiry-based learning has received a great deal of attention from researchers who appear to focus mostly on the students' engagement in active discovery learning and learning in environments that include characteristics of active participation, self-action, observation, exploration and experimentation (Psycharis, 2011; Kyriazis et al., 2009; Gormally et al., 2009; Richardson & Liang, 2008; Justice et al., 2007). Teaching Applied Mathematics in higher education based on the Process of Scientific Inquiry (PoSI) (National Institutes of Health, 2005) poses a challenge due to its multidisciplinary character (Lappas & Kritikos, 2015). Applied Mathematics lies at the intersection of many scientific disciplines such as mathematics, computing, physical and engineering sciences (Nocedal & Wright, 2006). In particular, Applied Mathematics deals with the application of advanced numerical methods and optimization techniques for solving complex numerical problems. The opportunity to learn numerical analysis and optimization, effectively, is dependent upon wide range of factors, but among the most important are those which are associated with activities and practices within the educational process (Martín-Caraballo & Tenorio-Villalón, 2015; Olagide, 2014; Cochran, 2009).

This paper extends the work of Lappas and Kritikos (2015) by providing several applications of inquiry-based learning in teaching numerical analysis and optimization. A didactic framework based on the PoSI is presented for solving (a) nonlinear equations, (b) systems of linear equations and (c) constrained/unconstrained optimization problems. The aim of the framework is threefold: (i) to help students to understand the basic aspects of numerical analysis and optimization, (ii) to provide students with the opportunity to practice and refine their critical-thinking skills and (iii) to convey to students the purpose of scientific research. The above framework proposes the use of cognitive computer tools such as MATLAB software for computer programming, CMAP software for constructing concept maps and WebQuest scenario for presenting students with challenging problems to solve, involving internet resources.

The remainder of the paper is organized as follows. Section 2 proposes the aforementioned didactic framework based on the PoSI. Applications of inquiry-based learning are provided in Section 3. Finally, in Section 4 some

concluding remarks are discussed.

2. Proposed Didactic Framework

2.1 The Process of Scientific Inquiry

Inquiry-based learning is related to the educational theory of constructivism, where students learn through a process of engagement. As a consequence, the structure of the proposed didactic framework is influenced by the Process of Scientific Inquiry (PoSI). The main components of the PoSI fall into five broad actions in the following order (National Institutes of Health, 2005):

- **Engage:** students are engaged in activities that encourage and facilitate learning situations with or without prior knowledge.
- **Explore:** students investigate the nature of the problems and try to construct their understanding.
- **Explain:** students try to connect their previous experiences with current learning.
- **Elaborate:** students apply or extend previously introduced concepts and experiences to new situations.
- **Evaluate:** a snapshot of student understanding is provided to assess whether the student acquired the intended knowledge and skills.

2.2 Cognitive Tools

Due to the increasing pressure on higher education institutes to keep up with the information age, educators are interested in exploring the potentials of using cognitive tools for teaching and learning (Dror et al., 2011; Kong, 2008; Kirschner & Erkens, 2006; Ozcelik & Yildirim, 2005). Toward the trend of cognitive tools, a primary purpose of this work is to explore the use of cognitive computer tools such as MATLAB software, CMAP software and WebQuest Scenario by undergraduate students to engage and facilitate critical thinking and higher order learning. MATLAB software is a high-level programming language in which students implement a range of numerical methods and optimization techniques through group-based projects (team works) to get hands-on experience with modern scientific computing for solving numerical problems (Hanselman & Littlefield, 2012). Several research papers in literature recognize the significance of using MATLAB for solving numerical problems (Costică, 2015; Jebamalar-Leavline & Gnana-Singh, 2014; Jain et al., 2010; Davidovich & Ribakov, 2010; Liang & Pan, 2009). In addition, CMAP software has been selected for constructing and modifying concept maps. Concept maps are tools for organizing and representing knowledge, while they enable the engagement of educators and students in the process of discovery (Kinchin, 2014; Vázquez-Cano et al., 2013; Pi áet al., 2011; Hay et al., 2008; Novak & Gowin, 1984).

According to Vanguri et al. (2004), WebQuest scenario provides students with the opportunity to combine technology with educational concepts, and to incorporate inquiry-based learning. Actually, WebQuest scenario reflects problem solving activities for students that incorporate the Internet, computer-based materials, computer-based tools (e.g., MATLAB and CMAP software) and other available resources. A WebQuest scenario is typically associated with the following sections (Göktepe, 2014):

- **Introduction:** a section that orientates students and raises awareness of the topic.
- **Task:** a section that describes the problem solving activity and the final outcomes.
- **Resources:** a section that provides url links (e.g., websites) or other sources (e.g., PDF files) determined by the educator to help the student carry out the problem solving activity.
- **Process:** a section that outlines the steps of the problem solving activity, including recommendations for reading.
- **Evaluation:** a section that explains grading or assessment of the problem solving activity using grading scale (e.g., rubrics).
- **Conclusion:** a section that reminds students of the purpose and the outcomes.
- **Educator Page:** a section that provides instructions to help other educators adapt the WebQuest.

2.3 Didactic Framework

Combining the PoSI with the use of cognitive tools is aimed to (i) help students to understand the basic aspects of applied mathematics (e.g., optimization), (ii) provide students with the opportunity to practice and refine their critical-thinking skills and (iii) convey to students the purpose of scientific research. From an abstract point of view, Figure 1 illustrates a didactic framework for teaching Applied Mathematics in higher education.

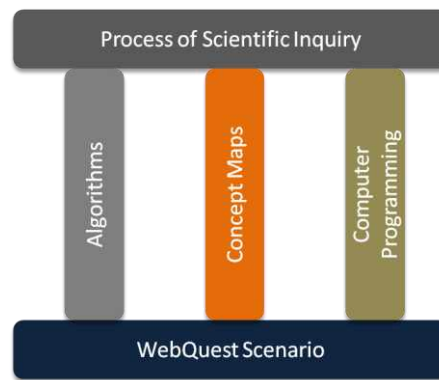


Figure 1. A didactic framework

The framework is based on the process of scientific inquiry, while it consists of three pillars: Algorithms, Concept Maps and Computer Programming. Algorithms allow students to present a list of steps to follow in order to solve a problem. In general, algorithms can be designed through the use of flowcharts or pseudo-codes. Concept maps help students to clarify their knowledge and make that knowledge explicit through the structural arrangements and linking of concepts and their relationships to generate meaningful propositions. Applied Mathematics practice is mostly through digital computation. Nowadays, the astonishing power and speed of personal computation allows the students to embed Applied Mathematics concepts in their practice. Once an algorithm has been formulated, computer programming leads from an original formulation of a computing problem to executable computer programs.

In fact, each pillar can be characterized by the use of a particular computer-based cognitive tool. Algorithms written in pseudo-code can be formulated using commercial or open-source word processor software such as Microsoft Office Word (<https://www.microsoft.com>) or Apache Open Office Writer (<http://www.openoffice.org/>). Alternatively, algorithms written in flowchart can be formulated using commercial software such as Microsoft Office Visio or open-source software like Inkscape (<https://inkscape.org/en/>). CMAP software (<http://cmap.ihmc.us/>) can be used for constructing concept maps, while MATLAB (commercial) software (<https://www.mathworks.com/>) or GNU Octave (open-source) software (<https://www.gnu.org/software/octave/>) can be deployed for computer programming. In addition to this, a WebQuest Scenario (e.g., <http://zunal.com/> or <http://webquest.org/>) can be used as an “auxiliary” cognitive tool by providing students with the opportunity to combine technology (e.g., MATLAB and CMAP software) with educational concepts (e.g., optimization), and to incorporate inquiry-based learning (i.e., PoSI).

Diagrams of the PoSI approach and WebQuest scenario are shown in Figure 2. Each diagram is effectively a cycle, reflecting the recurring nature of managing an educational process. It is important to take note the similarities between the two diagrams. This is not a coincidence.

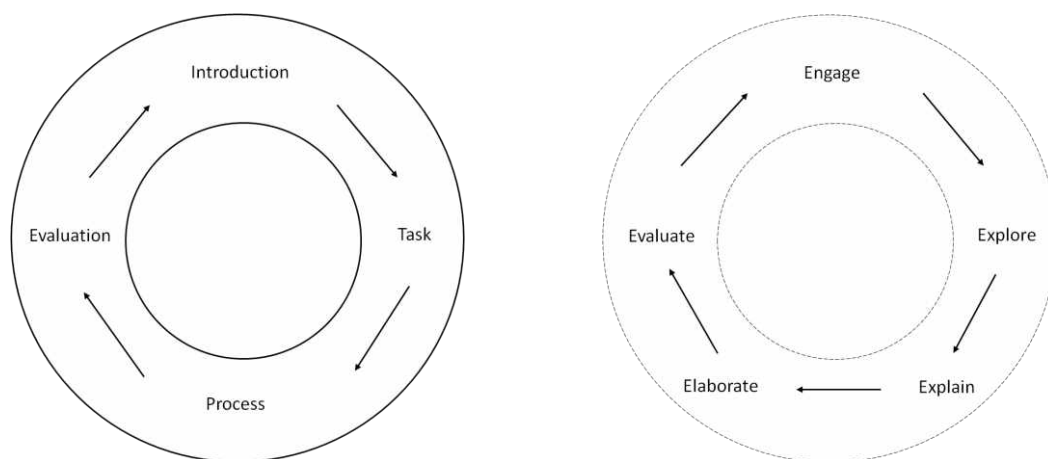


Figure 2. Side-by-side review of the webquest scenario and the process of scientific inquiry

Orientating students and raising awareness of the topic should be performed while students are engaged in activities that encourage and facilitate learning situations with or without prior knowledge. Therefore, it makes sense that these “process” activities are linked. Describing the problem solving activity and the final outcomes would be the ideal time for students to investigate the nature of the problems and try to construct their understanding. Thus, these two sets of “process” activities are obvious candidates to be linked. Once the steps of the problem solving activity are outlined, students should (a) try to connect their previous experiences with current learning and then (b) apply or extend previously introduced concepts and experiences to new situations. The action of evaluation is a vital part of assessing students’ understanding, which is a component in each diagram that can be linked together. Finally, the components of “Resources” and “Educator Page” related to WebQuest scenario are left due to their web-based nature. The result of linking these “process” activities together is a combined and integrated didactic framework, as shown in Figure 3.

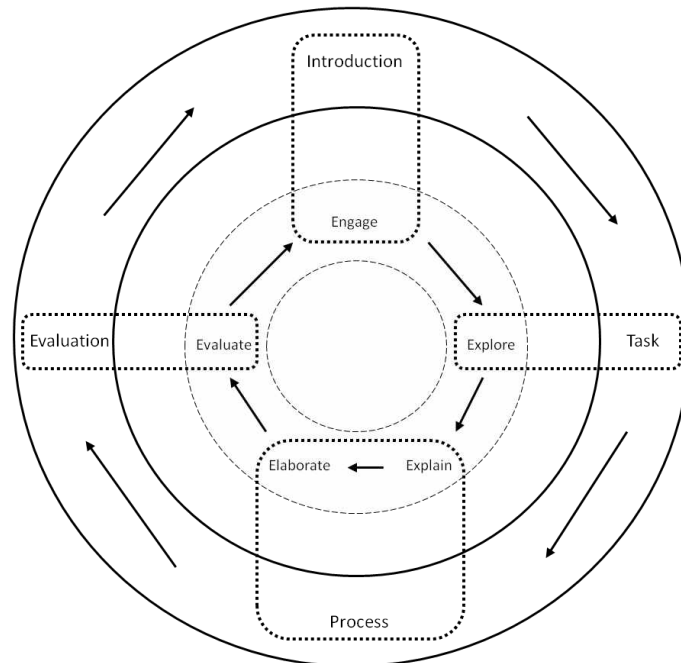


Figure 3. Integrating webquest scenario and process of scientific inquiry

In this paper, the computer-based cognitive tools that have been used, in the context of the proposed didactic framework, are the following: (a) Microsoft Office Word (algorithms/ pseudo-codes), (b) CMAP software (concept maps), (b) MATLAB software (computer programming) and (d) zunal.com (webquests). The figure below illustrates the allocation of cognitive tools to the main actions of the process of scientific inquiry.



Figure 4. Cognitive tools in the context of the process of scientific inquiry

As it can be observed, an emphasis should be placed on the actions of engagement and evaluation. Designing online educational environments through WebQuest scenarios requires special attention to interactivity. As a result, students are engaged in activities that encourage and facilitate learning. Java applets can provide a number of advantages for online use, including interfaces with various media formats (e.g., text, graphics, animation and sound). In combination with multimedia (i.e. video and images) and, probably, some lecture slides can provide interactivity to engage students in active learning. An important aspect of effective teaching is the careful assessment. The CMAP software can be used as a tool to assess students' understanding of a given concept. In addition, algorithms written in pseudo-codes and source codes in MATLAB can be evaluated with respect to their structure, convergence criterion, rate of convergence, computing error and computational results. Finally, rubrics have been implemented as the main grading criteria for each WebQuest assignment (Table 1). Total score is calculated by taking into account (a) the overall quality of the team project (70%) and (b) the cooperation among the members of the student groups (30%).

Table 1. Grading criteria

Criteria to assess the overall quality of the team project	Criteria to assess cooperation among the members of the student groups
<ul style="list-style-type: none"> ▪ Learning outcomes ▪ Completeness ▪ Structure-organization ▪ Fluency ▪ Connectivity ▪ Images and Tables ▪ MATLAB files ▪ Computational results 	<ul style="list-style-type: none"> ▪ Participation ▪ Responsibility ▪ Behavior

3. Applications of Inquiry-based Learning

The proposed framework reflects the integration of technology, pedagogy and content in the teaching and learning Applied Mathematics. It is not suggested to replace the conventional methods of teaching in higher education (lectures, tutorials, seminars, etc.). However, the quality of conventional university teaching can be considerably improved by its use. The proposed framework had been applied in the Department of Management Science and Technology of Athens University of Economics and Business (Greece), during the academic year 2015-2016. In order to show some of the potential applications of inquiry-based learning, this section presents several examples of using numerical methods and optimization techniques for solving (a) nonlinear equations, (b) systems of linear equations and (c) constrained/unconstrained optimization problems.

3.1 The action of “Engagement”

In this study, several WebQuest scenarios have been implemented for the modules of solving (a) nonlinear equations, (b) systems of linear equations and (c) optimization problems.

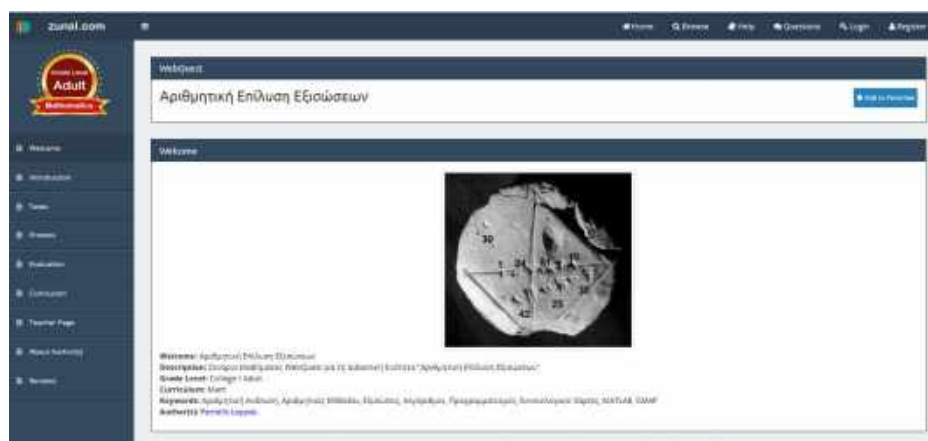


Figure 5. Webquest for solving nonlinear equations (available at: <http://zunal.com/webquest.php?w=225581>)

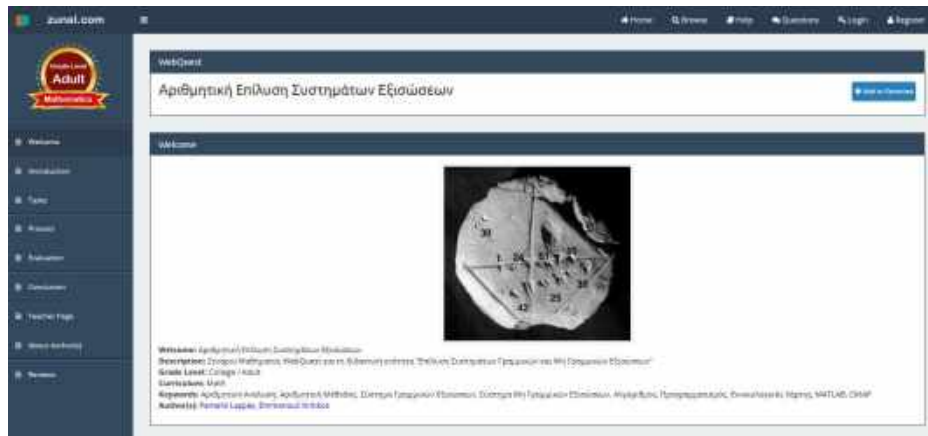


Figure 6. Webquest for solving systems of linear equations (available at: <http://zunal.com/webquest.php?w=210833>)



Figure 7. Webquest for solving constrained and unconstrained optimization problems (available at: <http://zunal.com/webquest.php?w=287970>)

Java applets, videos and web links are available in WebQuest scenarios so as to engage students in active learning.

3.2 The Action of “Exploration”

3.2.1 Solving NonLinear Equations

Students should recognize that all numerical methods of finding roots are iterative procedures that require a starting point (i.e., an estimate of the root). Therefore, this estimate can be crucial since a bad starting value of the root may fail to converge. The basic numerical methods for solving nonlinear equations are the following: (1) Bisection Method, (2) False Position Method, (3) Newton-Raphson Method and (4) Secant Method.

3.2.2 Solving Systems of Linear Equations

Students should recognize that there are two classes of numerical methods for solving systems of linear equations: direct and iterative methods. Direct methods are characterized by the fact that they transform the original equations into equivalent equations that can be solved more easily. The most popular direct methods are: (1) Gauss Elimination, (2) LU Decomposition and (3) Gauss-Jordan Elimination. In addition, iterative methods start with a guess of the solution and then repeatedly refine the solution until a certain convergence criterion is reached. The most popular iterative methods are: (1) Jacobi Iteration and (2) Gauss-Seidel Iteration.

3.2.3 Solving Constrained and Unconstrained Optimization Problems

Students should recognize that optimization is an important tool in decision science and the analysis of the physical systems. In order to make use of this tool, they must first identify an “objective function” which is a quantitative measure of the performance of the system under study. In addition, the objective depends on

certain characteristics of the system, known as “decision variables”. It should be noted that the crucial point here is to find values of the decision variables that “optimize” (maximize or minimize) the objective function. Mention should also be made of the fact that often decision variables are “constrained”. On closer examination, students should recognize that the process of identifying objective function, decision variables and constraints for a given optimization problem, known as mathematical modeling, is the first and the most important step in optimization process. Once the model has been formulated, an optimization algorithm can be used to find a solution. In reality, there is a collection of algorithms depends on the nature of the optimization problem (constrained or unconstrained). Students should study optimization algorithms for unconstrained optimization problems with one variable function (Golden Search Method) and multivariable functions (Newton Method, Steepest Descent Method), as well as, constrained optimization problems (Penalty Function Method).

3.2.4 Representing Algorithms

Students, after recognizing numerical methods for solving (a) nonlinear equations, (b) systems of linear equations and (c) constrained/unconstrained optimization problems, are going to use pseudo-code in order to represent an algorithm for each of the numerical methods. Figures 8 and 9 depict the algorithms for the Bisection Method and the Jacobi Iteration, respectively.

Algorithm Bisection Method

Inputs: $f(x)$, a , b , ε

Begin

$$k = \left\lceil \frac{\log(b - a) - \log \varepsilon}{\log 2} \right\rceil + 1$$

For $n = 1:1:k$

$$x_n \leftarrow \frac{a+b}{2}$$

If $f(x_n) = 0$ **then**

Exit

End-If

If $f(a)f(x_n) > 0$ **then**

$$a \leftarrow x_n$$

else

$$b \leftarrow x_n$$

End-If

End-For

End

Output: x_n

End Bisection Method

Algorithm Jacobi Iteration

Inputs: n , a_{ij} , ε

Begin

$$\text{norm} \leftarrow \varepsilon + 1$$

Repeat until $\text{norm} \leq \varepsilon$

$$\text{norm} \leftarrow 0$$

For $i = 1:1:n$

$$x_i \leftarrow \frac{-\left(\sum_{\substack{j=1 \\ j \neq i}}^n a_{ij} z_j\right) + a_{i,n+1}}{a_{ii}}$$

If $\text{norm} < |x_i - z_i|$ **then**

$$\text{norm} \leftarrow |x_i - z_i|$$

End-If

End-For

For $i = 1:1:n$

$$z_j \leftarrow x_i$$

End-For

End-Repeat

End

Output: x_i

End Jacobi Iteration

Figure 8. Pseudocode for the bisection method

Figure 9. Pseudocode for the jacobi iteration

3.3 The Action of “Explanation”

Students are able to use CMAP software so as to construct their concept maps. Below, several examples of concept maps are presented related to constrained/unconstrained optimization. Figure 10 illustrates the first concept map where a student starts to recognize that the field of Mathematics is divided into Applied Mathematics and Pure Mathematics. Typically, Applied Mathematics is combined with Numerical Analysis and Operations Research. Operations Research is also known as “Decision Making” or “System Analysis” or “Quantitative Analysis” or “Management Science”. This concept map can be further extended by adding new concepts related to numerical optimization such as optimization algorithm (Figure 11), optimization problem (Figure 12) and mathematical software package (Figure 13).

As mentioned in Section 2, Applied Mathematics practice is mostly through digital computation. Therefore, concept maps related to source code, computer programming and specific numerical methods for solving constrained/unconstrained optimization problems are provided. Specifically, Figure 14 illustrates a concept map that focuses on the basic components of a programming language (e.g., GNU Octave) such as operators, selection and loop statements. Figure 15 analyzes concepts related to algorithms and compilation. Finally, Figure 16 is associated with a concept map that presents all executable computer programs. These programs, in MATLAB/GNU Octave software, are called “.m files” and can be divided into functions and scripts. Functions implement the numerical methods for solving numerical problems, while scripts, usually, call function/s to solve specific numerical problems.

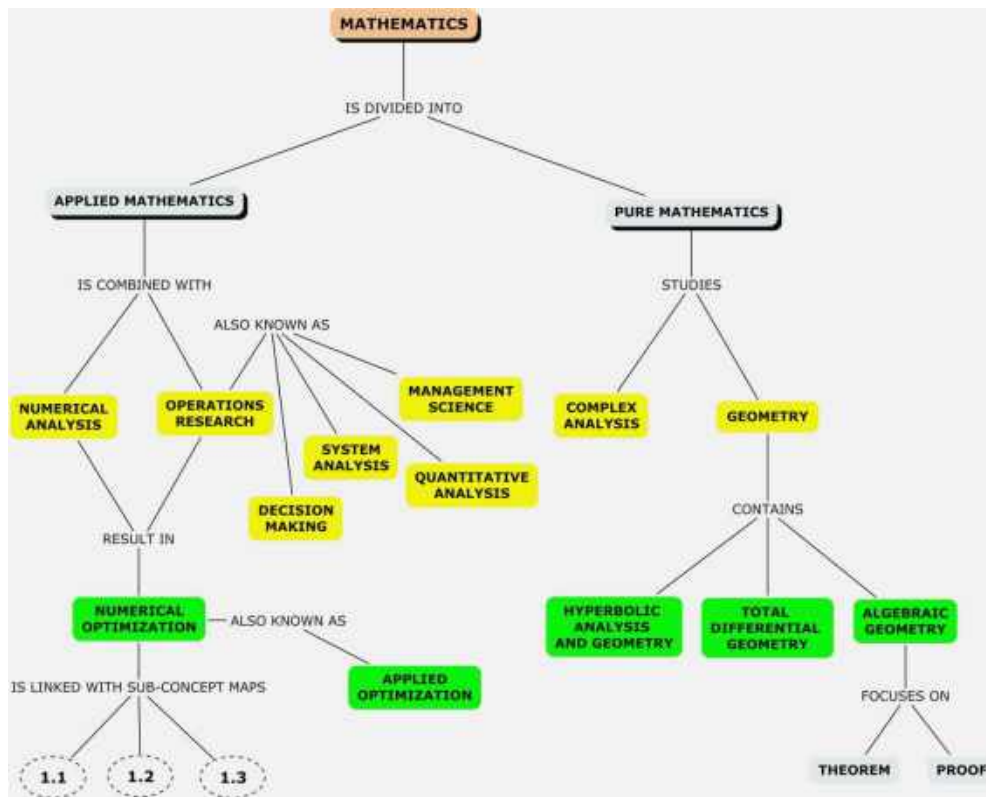


Figure 10. The concept of “Mathematics”

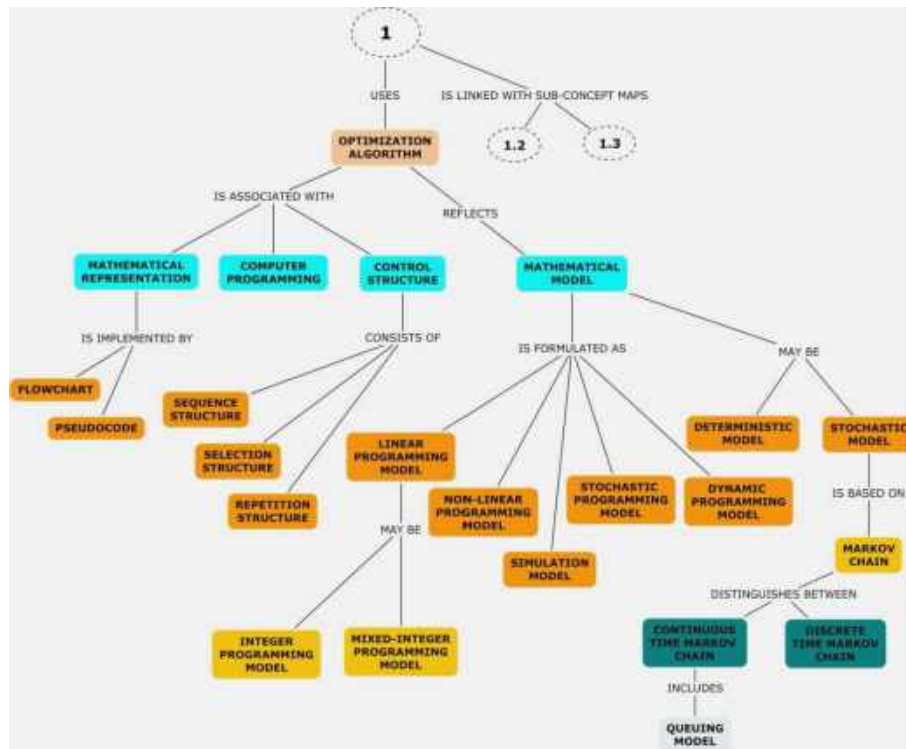


Figure 11. The concept of “Optimization Algorithm”

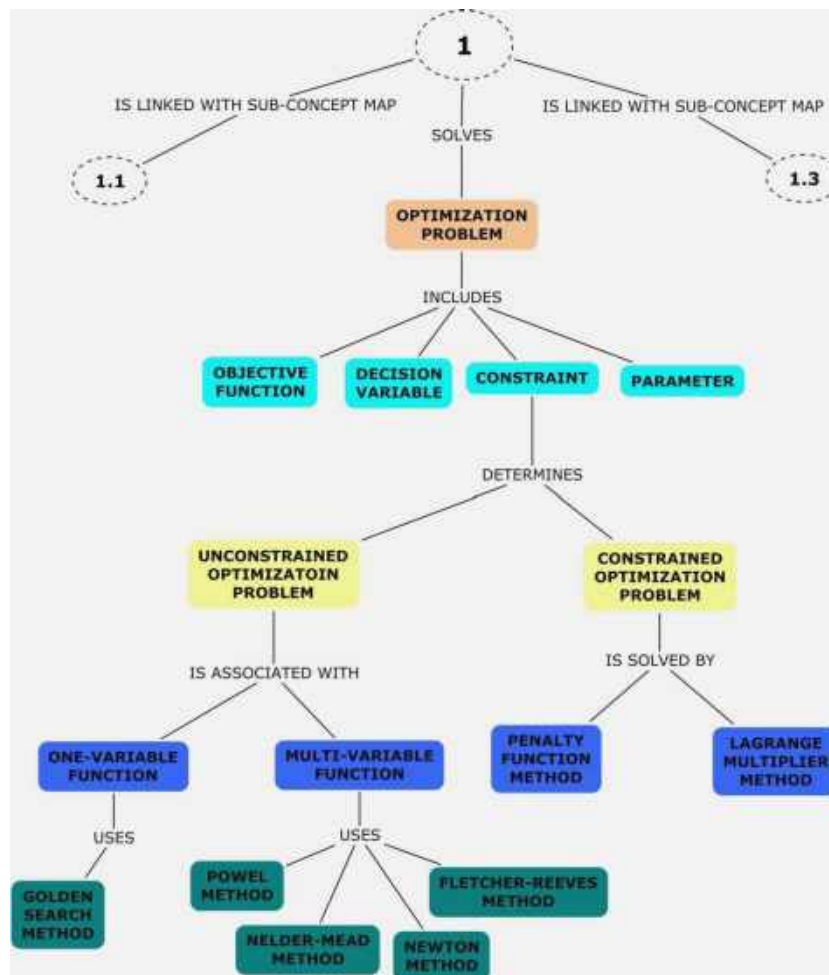


Figure 12. The concept of “Optimization Problem”

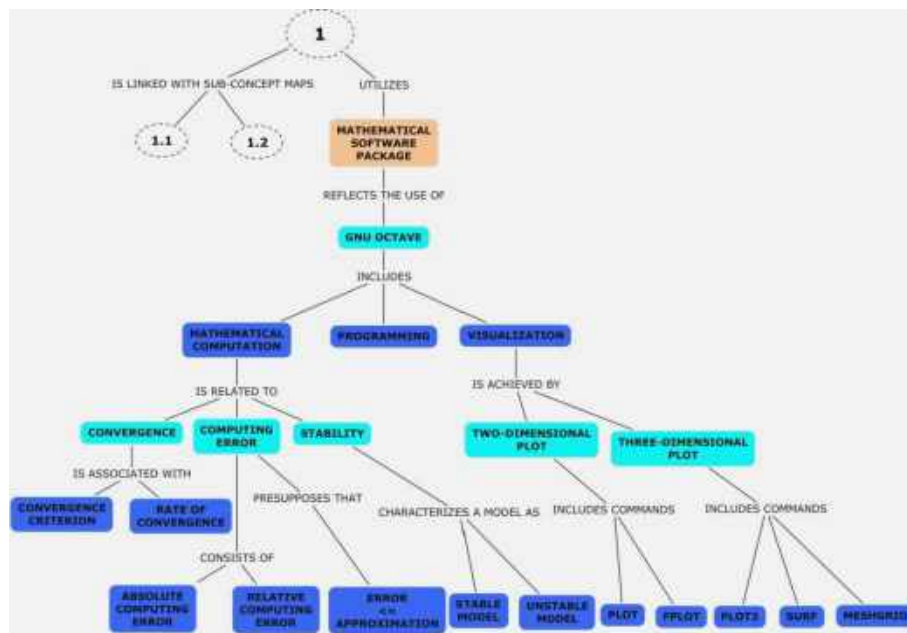


Figure 13. The concept of “Mathematical Software Package”

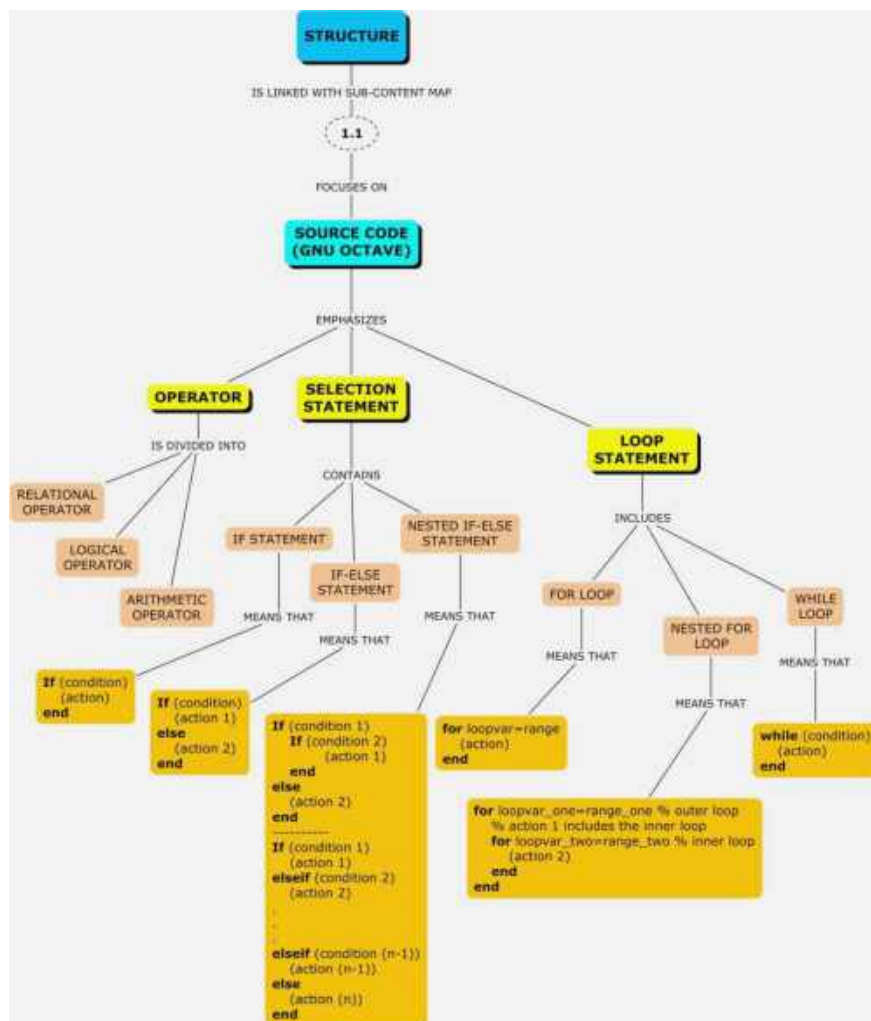


Figure 14. The concept of “Source Code”

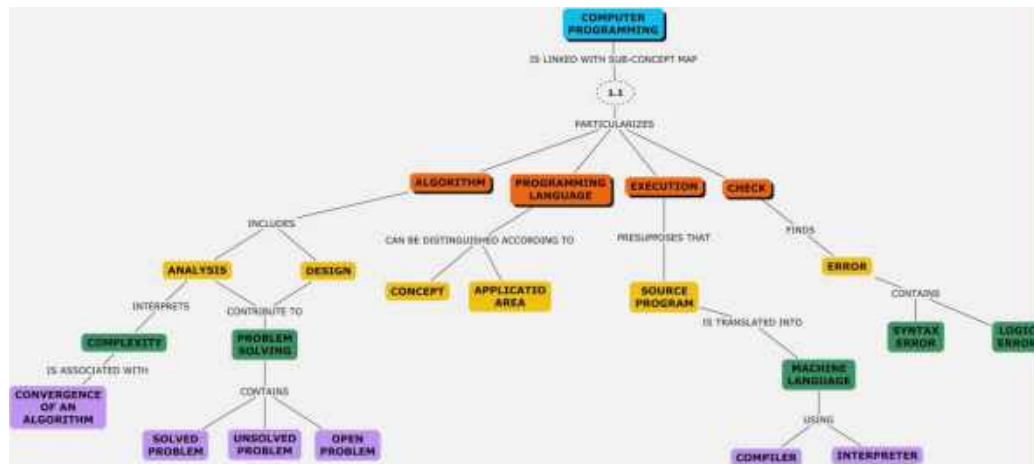


Figure 15. The concept of “Computer Programming”

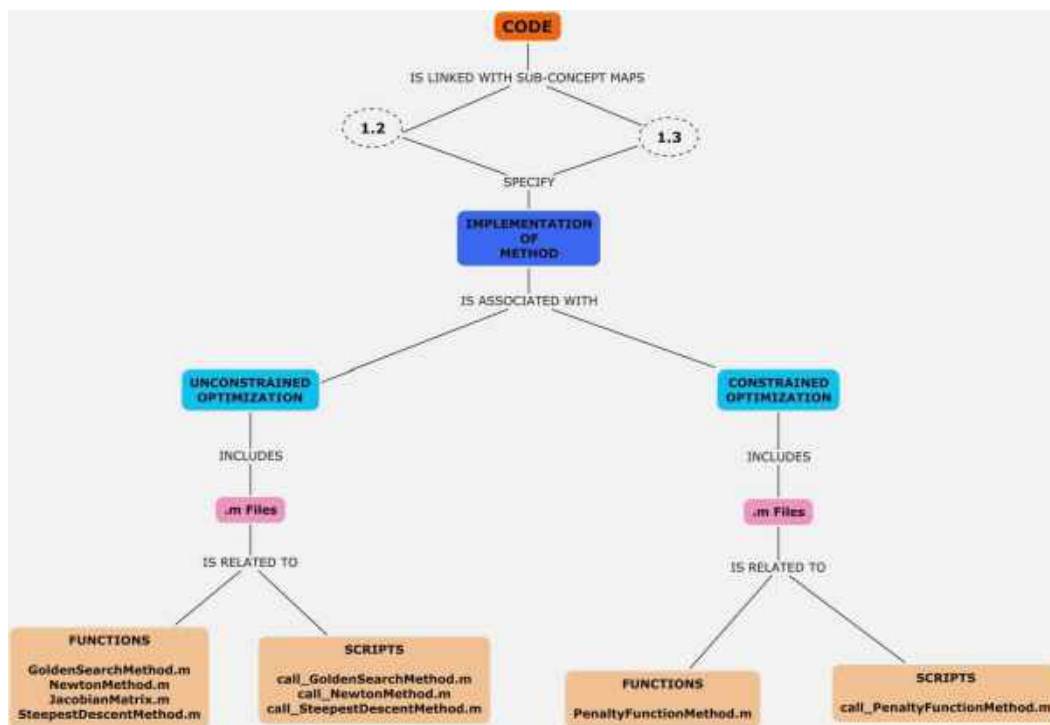


Figure 16. The concept of “Implementation”

3.4 The Action of “Elaboration”

The knowledge of computer programming makes the problem solving process easier at this stage. After an algorithm has been formulated, computer programming leads from an original formulation of a computing problem to executable computer programs. This stage focuses on understanding the fundamental mathematical concepts and mastering problem-solving skills using numerical methods. Table 2 presents all .m files needed to implement for solving numerical problems with the help of MATLAB software.

Table 2. MATLAB files for numerical analysis and optimization

Functions	Scripts
Solving nonlinear equations	
BisectionMethodOfBolzano.m / <i>Bisection Method</i>	call_Bisection.m
MethodOfFalsePosition.m / <i>False Position Method</i>	call_FalsePosition.m
NewtonRaphson.m / <i>Newton-Raphson Method</i>	call_NewtonRaphson.m
SecantMethod.m / <i>Secant Method</i>	call_SecantMethod.m
Solving systems of linear equations	
gauss.m / <i>Gauss Elimination Method</i>	call_gauss.m
LUdec.m / <i>LU Decomposition</i>	call_LUDecomposition.m
gaussJordan.m / <i>Gauss-Jordan Elimination</i>	call_GaussJordanElimination.m
jacobiALG.m / <i>Jacobi Iteration</i>	call_JacobiALG.m
gauseidALG.m / <i>Gauss-Seidel Iteration</i>	call_gauseidALG.m
Solving unconstrained and constrained optimization problems	
GoldenSearchMethod.m / <i>Golden Search Method (Unconstrained Optimization – One Variable Function)</i>	call_GoldenSearchMethod.m
NewtonMethod.m / <i>Newton Method (Unconstrained Optimization – Multivariable Function)</i>	call_NewtonMethod.m
SteepestDescentMethod.m / <i>Steepest Descent Method (Unconstrained Optimization – Multivariable Function)</i>	call_SteepestDescentMethod.m
PenaltyFunctionMethod.m / <i>Penalty Function Method (Constrained Optimization)</i>	call_PenaltyFunctionMethod.m

Figures 17 and 18 illustrate an example of using a function and a script file in MATLAB, respectively. MATLAB files are related to the Bisection Method.

```

function [x,err,xx,Lft,Rgt] = BisectionMethodOfBolzano(f,a,b,TolX,MaxIter)
% BisectionMethodOfBolzano.m solve f(x) = 0 with the Bisection Method.
TolFun=eps;
fa = feval(f,a);
fb = feval(f,b);
if fa*fb > 0
    error('Should be f(a)f(b)<0!');
end
tt=0;
for k = 1: MaxIter
    xx(k) = (a + b)/2;
    tt=tt+1;
    Lft(tt)=a;
    Rgt(tt)=b;
    fx = feval(f,xx(k));
    err = (b-a)/2;
    if abs(fx) < TolFun | abs(err)<TolX
        break;
    elseif fx*fa > 0
        a = xx(k);
        fa = fx;
    else
        b = xx(k);
    end
end
x = xx(k);
xx=xx';
Lft=Lft';
Rgt=Rgt';
if k == MaxIter
    fprintf('Optimal solution within %d iterations\n',MaxIter)
end
end

```

Figure 17. Function file for the bisection method

```

clear ,clc,close
f=inline('exp(x)-2*x-1','x');
a=1;
b=2;
TolX=10^(-5);
MaxIter=50;
[x,err,xx,Lft,Rgt]=BisectionMethodOfBolzano(f,a,b,TolX,MaxIter)

```

Figure 18. Script file – Solving $f(x) = e^x - 2x - 1$

Finally, students provide various graphical presentation formats to highlight the insights that are gained. In particular, the graphical presentations help to simplify complicated issues and convey meaningful insights into the problem. Figure 19 depicts graphical solutions for several problems associated with the application of the following methods: Bisection Method, Secant Method, Golden Search Method and Newton Method.

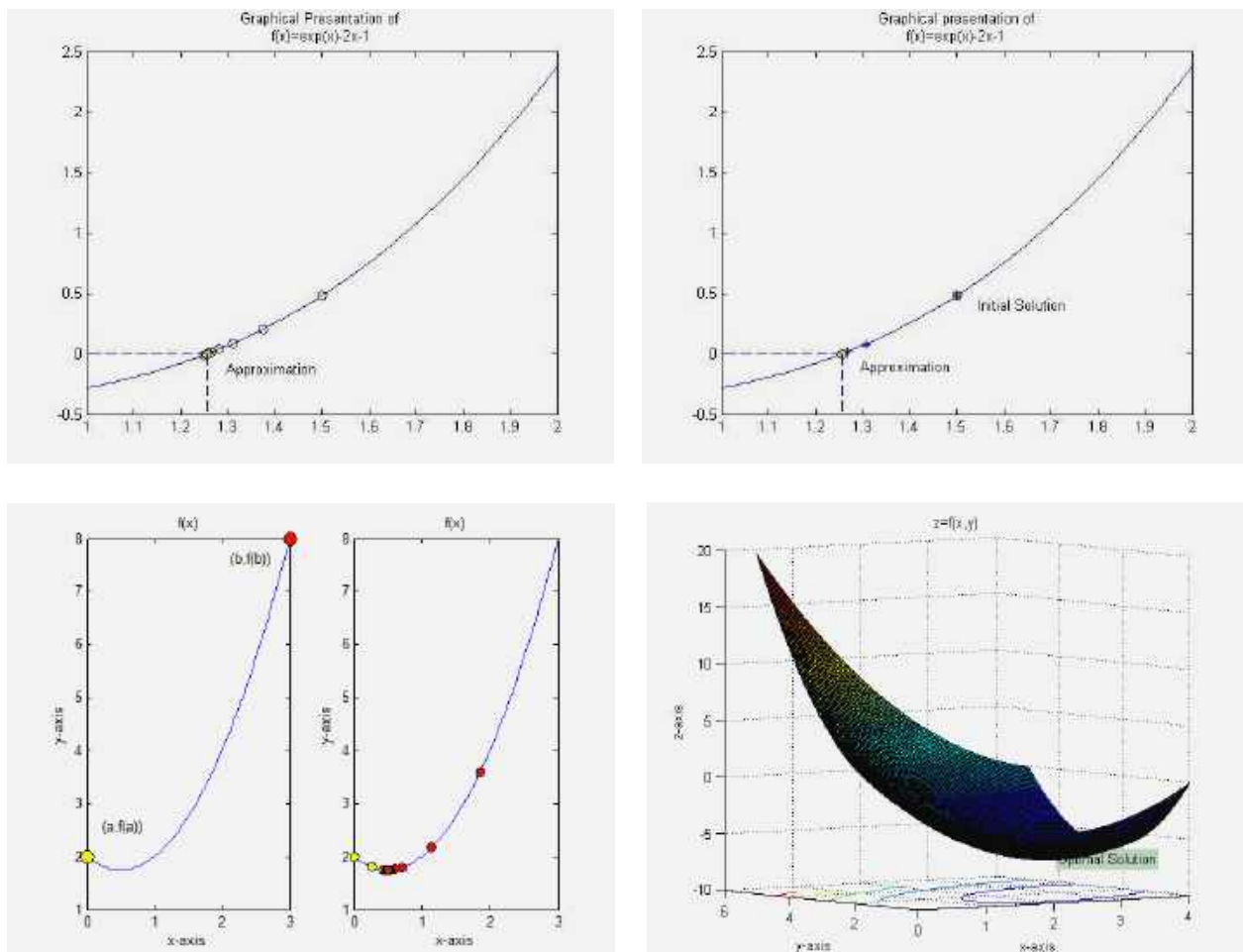


Figure 19. Graphical solutions applying bisection method (top left), secant method (top right), golden search method (bottom left) and newton method (bottom right)

3.5 The Action of “Evaluation”

As described in Section 2, the concept maps, the algorithms and the source codes can be evaluated with respect to several criteria so as to assess students’ understanding. Moreover, rubrics have been implemented as the main grading criteria for each WebQuest assignment (see Table 1).

4. Discussion and Conclusion

The inquiry-based learning signifies a challenging stream of research that has drawn the increasing attention of the research community during the last decade. This study set out with the aim of proposing a didactic framework for teaching Applied Mathematics in higher education. Very little was found in the literature on the combination of technology with educational concepts so as to incorporate inquiry-based learning. An initial objective of the study was to present a didactic framework, based on the Process of Scientific Inquiry, consisting of three pillars, each characterized by the use of a particular cognitive tool: Algorithm for presenting a list of steps to follow in order to solve a problem, CMAP software for constructing concept maps and MATLAB software for computer programming. In addition to this, a WebQuest Scenario was used as an “auxiliary” cognitive tool for incorporating inquiry-based learning. A second objective was to provide several applications related to Applied Mathematics so as to illustrate the potential of the proposed framework. Therefore, several

practice examples of using algorithms, concept maps, webquests and computer programming for solving nonlinear equations, systems of linear equations and constrained/unconstrained optimization problems were provided, indicating how the proposed framework can be applicable in practice.

The proposed framework is not suggested to replace the conventional methods of teaching in higher education. Nonetheless, the quality of conventional university teaching can be considerably improved by its use. The framework had been applied in the Department of Management Science and Technology of Athens University of Economics and Business (Greece), during the academic year 2015-2016. Nine students were separated in three groups of three. The students of the first group were assigned problems from the module of solving nonlinear equations. The students of the second group were assigned problems from the module of solving systems of linear equations, while the students of the third group were assigned problems from the module of solving constrained/unconstrained optimization problems. Three WebQuest scenarios (one for each group) developed for presenting students with challenging problems to solve. All groups followed all the steps of the proposed didactic framework by providing (a) pseudo-codes for representing an algorithm for each of the numerical methods, (b) concept maps for representing their knowledge, (c) executable computer programs for solving numerical problems and (d) graphical presentation formats to highlight the insights that were gained.

Results indicated that there is a strong shift on students' conceptual understanding. Following the framework, students were able to: (a) understand the purpose of scientific research, (b) develop managerial skills (i.e., strategic and critical thinking, communication and conflict resolution by participating in group-based (team) projects), presentation skills and computer programming skills, as well as, (c) solve complex numerical problems (available in WebQuest scenarios) by developing efficient algorithms. Due to the small sample (nine students), there are no specific empirical results. Practice examples were presented to illustrate the potential of the proposed framework. In terms of future work, the goal is to apply the methodology to a sufficiently large sample to provide statistically significant empirical results.

References

- Cochran, J. (2009). Pedagogy in operations research: Where has the discipline been, where is it now, and where should it go?. *ORiON: The Journal of the Operations Research Society of South Africa*, 25(2), 161-184. <https://dx.doi.org/10.5784/25-2-80>
- Costică, L. (2015). The importance of computers with MATLAB software in the teaching and learning of geometry in space. *International Journal of Mathematical Research*, 4(1), 16-26. <https://dx.doi.org/10.18488/journal.24/2015.4.1/24.1.16.26>
- Davidovich, N., & Ribakov, Y. (2010). Teaching engineering subjects using MATLAB. *Problems of Education in the 21st Century*, 19, 9-14.
- Dror, I., Schmidt, P., & O'connor, L. (2011). A cognitive perspective on technology enhanced learning in medical training: Great opportunities, pitfalls and challenges. *Medical Teacher: An International Journal of Education in the Health Sciences*, 33(4), 291-296. <https://dx.doi.org/10.3109/0142159X.2011.550970>
- Göktepe, S. (2014). A WebQuest example for mathematics education. *Procedia – Social and Behavioral Sciences*, 116, 2175-2179. <https://dx.doi.org/10.1016/j.sbspro.2014.01.539>
- Gormally, C., Brickman, P., Hallar, B., & Armstrong, N. (2009). Effects of inquiry-based learning on students' science literacy skills and confidence. *International Journal for the Scholarship of Teaching and Learning*, 3(2), 1-22. <https://dx.doi.org/10.20429/ijstl.2009.030216>
- Hanselman, D., & Littlefield, B. (2012). *Mastering MATLAB® 7*. New York, United States of America: Pearson Education.
- Hay, D., Kinchin, I., & Lygo-Baker, S. (2008). Making learning visible: the role of concept mapping in higher education. *Studies in Higher Education*, 33(3), 295-311. <https://dx.doi.org/10.1080/03075070802049251>
- Jain, A., Singhal, M., & Sheel-Gupta, S. (2010). Educational tool for understanding algorithm building and learning programming languages. Paper presented at the Proceedings of the World Congress on Engineering and Computer Science (United States of America), San Francisco 20-22 October.
- Jebamalar-Leavline, E., & Gnana-Singh, D. (2014). On teaching digital image processing with MATLAB. *American Journal of Signal Processing*, 4(1), 7-15.
- Justice, C., Rice, J., Warry, W., Inglis, S., Miller, S., & Sammon, S. (2007). Inquiry in higher education: Reflections and directions on course design and teaching methods. *Innovative Higher Education*, 31(4), 201-214. <https://dx.doi.org/10.1007/s10755-006-9021-9>

- Kinchin, I. (2014). Concept mapping as a learning tool in higher education: A critical analysis of recent reviews. *The Journal of Continuing Higher Education*, 62(1), 39-49. <https://dx.doi.org/10.1080/07377363.2014.872011>
- Kirschner, P., & Erkens, G. (2006). Cognitive tools and mindtools for collaborative learning. *Journal of Educational Computing Research*, 35(2), 199-209. <https://dx.doi.org/10.2190/R783-230M-0052-G843>
- Kong, S. (2008). The development of a cognitive tool for teaching and learning fractions in the mathematics classroom: A design-based study. *Computers and Education*, 51(2), 886-899. <https://dx.doi.org/10.1016/j.compedu.2007.09.007>
- Kyriazis, A., Psycharis, S., & Korres, K. (2009). Discovery learning and the computational experiment in higher mathematics and science education: A combined approach. *International Journal of Emerging Technologies in Learning*, 4(4), 25-34. <https://dx.doi.org/10.3991/ijet.v4i4.1044>
- Lappas, P., & Kritikos, M. (2015). Computer-based cognitive tools in mathematics and operations research: The process of scientific inquiry. Paper presented at the Proceedings of the ERPA International Congresses on Education (Greece), Athens, 4-7 June (pp. 207-212).
- Liang, J., & Pan, W. (2009). A MATLAB-aided method for teaching calculus-based business mathematics. *American Journal of Business Education*, 2(9), 15-40. <https://dx.doi.org/10.19030/ajbe.v2i9.4606>
- Martín-Carballo, A., & Tenorio-Villalón, Á. (2015). Teaching numerical methods for non-linear equations with GeoGebra-based activities. *Mathematics Education*, 10(2), 53-65.
- National Institutes of Health (2005). *Doing science: The process of scientific inquiry* (NIH Publication No. 05-5564). Retrieved from http://www.uwyo.edu/scienceposse/resources/nih_doing-science.pdf
- Nocedal, J., & Wright, S. (2006). *Numerical optimization*. New York, United States of America: Springer.
- Novak, J., & Gowin, B. (1984). *Learning how to learn*. New York, United States of America: Cambridge University Press. <https://doi.org/10.1017/CBO9781139173469>
- Olagide, A. (2014). Teaching numerical analysis to non-mathematics major students. Paper presented at the Proceedings of the International Conference on Science, Education, Arts, Management and Social Sciences (Nigeria), Afe Babalola University, Ado-Ekiti, May (pp. 939-942).
- Ozcelik, E., & Yildirim, S. (2005). Factors influencing the use of cognitive tools in web-based learning environments: A case study. *The Quarterly Review of Distance Education*, 6(4), 295-308.
- Piñá A., Blasco-Tamarit, E., & Muñoz -Portero, M. (2011). Different applications of concept maps in higher education. *Journal of Industrial Engineering and Management*, 4(1), 81-102.
- Psycharis, S. (2011). The computational experiment and its effects on approach to learning and beliefs on physics. *Computers and Education*, 56(3), 547-555. <https://dx.doi.org/10.1016/j.compedu.2010.09.011>
- Richardson, G., & Liang, L. (2008). The use of inquiry in the development of preservice teacher efficacy in mathematics and science. *Journal of Elementary Science Education*, 20(1), 1-16. <https://dx.doi.org/10.1007/BF03174699>
- Vanguri, P., Szymanski-Sunal, C., Wilson, E., & Wright, V. (2004). WebQuests in social studies education. *Journal of Interactive Online Learning*, 3(2), 1-11.
- Vázquez-Cano, E., López-Meneses, E., & Fernández-Márquez, E. (2013). Concept mapping for developing competencies in European higher education area. *International Journal of Humanities and Social Sciences*, 3(17), 7-18.

Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).