THE SHORTEST ADAPTIVE LEARNING PATH IN E-LEARNING SYSTEMS

By

Marwah Mustafa Alian Abdel-Rahim

Supervisor

Dr. Riad Jabri

This Thesis was Submitted in Partial Fulfillment of the Requirements for the Master's Degree in Computer Science.

Faculty of Graduate Studies

The University of Jordan

July, 2007

This Thesis/Dissertation (The Shortest Adaptive Learning Path in E-learning Systems) was successfully defended and approved on 30 July, 2007.

Examination Committee	**Signature**
Dr. Riad Jabri, Chairman Assoc. Prof. of Programming Languages and Compilers	----------------------------
Dr. Ahmad Sharieh, Member Assoc. Prof. of Parallel Processing	----------------------------
Dr. Mohamad Belal Al-Zoubi, Member Assoc. Prof. of Computer Graphics	----------------------------
Dr. Amjad Hudaib, Member Assist. Prof. of Software Engineering	----------------------------
Dr. Taleb Al-Sarie, Member Prof. of Computer Science (Arab Open University)	----------------------------

ACKNOWLEDGEMENT

I would like to thank Dr. Riad Jabri, to whom I am most grateful, and the examination committee for their guidance and support. Special thanks to my parents, family, and friends for their encouragement and care.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATIONS

Expression	Abbreviation
Learning Object	LO
Learning Object Metadata	LOM
Instructor Led Training	ILT
Computer Based Training	CBT
Canadian Core Learning Object	CanCore
Learning Objects Graph	LOG
Information Management Systems	IMS
Relevant Knowledge First	RKF
Knowledge Unit	KU
Adaptive Course Generation	ACG
Adaptive Educational Hypermedia Systems	AEHS
Eliminating and Optimized Selection	EOS

THE SHORTEST ADAPTIVE LEARNING PATH IN E-LEARNING SYSTEMS

By

Marwah Mustafa Alian Abdel-Rahim

Supervisor

Dr. Riad Jabri

ABSTRACT

The main challenge of e-learning systems is to provide courses tailored to different students with different learning rate and knowledge degree. Such systems must be also efficient, as well as adaptive. However, the most recent research can be classified into two major groups. The first group emphasizes the need for E-learning to be adaptive. While the second group, emphasizes the efficiency of such systems. In this research, we set an objective to achieve both efficiency and adaptivity. This can be accomplished by selecting a representative algorithm for the first group and a representative algorithm for the second one, and attempting to combine them. This is justified by the fact that the first one aimed at improving the ability to select dynamically an appropriate learning object for a specific learner, while the second one aimed at selecting a learning path that costs least time and effort.

In order to decide how these two approaches can be combined, the representative approaches were further analyzed, implemented and then experimented. As a result, a

formalization and some modifications to these algorithms were suggested and a new approach is proposed.

The computational results of the proposed approach have been compared to the computational results of the selected algorithms as well as to the selection results performed by experts. The comparisons have shown its superiority in terms of producing more tailored and more optimized selection of learning objects. Furthermore, the proposed approach has demonstrated its competitiveness to assumed experts in terms of the selected sequences of learning objects for different learners with different needs.

CHAPTER1: INTRODUCTION

The term e-Learning refers to online learning delivered over the world wide web via public internet or private intranet (Yu *et al*., 2006). It is concerned with the computer based implementation of an educational system, thus it is a result of a computer oriented analysis and design of such system. Furthermore, web based education and training is a hot research area. Most of the progress made in this field has been influenced by the evolving technological infrastructure.

However, the main challenge of the most recent research is to provide efficient and adaptive e-Learning systems. To achieve efficiency, the e-Learning systems are modeled as a directed graph where each node represents a Learning Object (LO) (Viet and Si, 2006). Each LO may contain one concept, one object, an image, or an audio session. Two nodes are connected if there exist a dependency relation, such that one node is a prerequisite to the other. Given a target node, the resulting graph can be used to determine the shortest path leading to such node. One of the most important features which has not been fully explored in this approach is the ability of the learning system to adapt to the learner’s profile (Yanwen and Zhonghong, 2004).

The e-Learning systems act as an adaptive system if they select the path of learning that meet the student's requirements and needs and discard those paths, which are not in accordance with these needs. Furthermore, such an adaptive learning must be as efficient as possible (Andreev and Troyanova, 2006). To achieve such efficiency and adaptivity, two groups of solutions do exist. The first group emphasizes the need for e-Learning to be adaptive (Atif *et al*., 2003; Karampiperis and Sampson, 2004; Liu and Greer, 2004; Viet and Si, 2006). The second group emphasizes the efficiency by selecting learning path which costs the least time and effort (Zhao and Wan, 2006).

Based on these solutions the aim of our proposed research is to select a representative algorithm from each group. Then to combine these algorithms, in order to create a shortest path that is tailored for the learner's needs. Hence, the benefits of both groups are to be obtained.

This thesis is organized as follows; Chapter 2 gives literature review and basic definitions and concepts of e-Learning and graph theory. Chapter 3 and Chapter 4 constitute an analysis and implementation of the selected e-Learning algorithms. These algorithms are Eliminating and Optimized Selection (EOS) (Liu and Greer, 2004) and the shortest learning path (Zhao and Wan, 2006). Chapter 5 introduces a new approach that combines these algorithms with a respective modification of their different phases (elimination, selection and optimization). A conclusion and discussion are given in Chapter 6.

CHAPTER 2: BACKGROUND AND LITERATURE REVIEW

e-Learning can break through the limit of space and time, reduce learning cost and improve learning efficiency. Therefore, many community websites based on e-Learning have been constructed; community residents can get some information or some study courses (Yanwen and Zhonghong, 2004).

e-Learning is one of the most innovative applications, since it can radically change the learning process of many people. e-Learning was initially developed inside specific environments, where homogeneous contents were developed to homogeneous people communities (Soine, 2001). A major current focus in designing modern e-Learning systems is the actual concentration on efficient production of instructional components or objects, which are interoperable and reusable (Najjar , 1996).

There were interests to represent the knowledge of the world with a methodology that identifies classes of objects with common properties in a hierarchical structure where some classes are specializations of others. This way was called *Ontology*. It was defined as: "An ontology may take a variety of forms, but necessarily it will include a vocabulary of terms and some specification of their meaning. An ontology is virtually always the manifestation of a shared understanding of a domain that is agreed between a number of parties. Such agreement facilitates accurate and effective communication of meaning, which in turn leads to other benefits such as inter-operability, reuse and sharing."(Studer *et al.*, 1998).

Ontologies have been used in computer science in many areas such as: Configuration Systems, Software Engineering, Information Retrieval, Conceptual

Modeling, Interoperability, Enterprise Modeling, Electronic Commerce, and many other fields in the research and production areas (Muñoz, 2004).

Pythagoras and Demetrios (2004) addressed adaptive navigation support in educational hypermedia systems by proposing a framework based on the use of ontologies and learning objects metadata.

Burgos and Specht (2006) show how several methods in adaptive learning can be addressed using IMS learning design, also they introduced a definition of four questions to classify adaptive educational methods, and those questions are:

- What parts or components of the learning process are adapted?
- What information does the system use for adaptation?
- How does the system gather the information to adapt to?
- Why does the system adapt?

According to the previous four questions, the application of adaptive methods to Educational Hypermedia Application can be structured.

Henze and Nejdl (2003) developed a logical characterization of adaptive educational hypermedia and web-based systems (AEHS), and discussed the applicability of this approach.

Brusilovsky and Vassileva (2003) introduced three approaches for the use of course sequencing and mentioned the benefits and lack of each of them. The idea of the first approach depends on the concept that since a sequencing mechanism can evaluate many options for the next step then it can check if the next step presented by the author is a good one. A more progressive way is courseware reuse that depends on growing in popularity and it supports course reusability. It assumes that courses are developed from reusable content objects. The second one is adaptive courseware generation, which has the idea of generating a course suited to the needs of the students before they encountered it. The third approach is dynamic courseware generation, it generates an individualized course depending on a specific learning goal and the learner's knowledge. Then they discussed two models of course sequencing techniques, the dynamic course generation system and the concept-based course maintenance system.

Kreuz and Roller (2001) proposed the heuristic Relevant Knowledge First (RKF) for making decisions in configuration processes based on the relevance of objects in a knowledge base. The relevance function that used has two components, the first one based on reinforcement learning and the second component depend on forgetting. The proposed RKF speed up the configuration process and improve the quality of the solutions relative to the reward value that users given when using the object.

For the assessment of the relevance of objects, they considered two factors which correspond to the antagonism between conservatism and innovation:

- Objects are very likely to be relevant if they have already been useful for similar tasks, and objects that did not help to find solutions will probably not help in the future.
- New objects should be taken into consideration in order to avoid conservatism, because objects can be forgotten.

They proposed a formula to calculate relevance of an object from the time since it was last used (during forgetting phase), and the rewards obtained by users (during learning phase). Every time a decision has to be made, the relevance of all objects in question at the actual time t is calculated, and a random generator selects one object.

Kreuz and Roller (2001) suggested their own definition of relevance as follows: "The relevance at the time of t of an object o in the context of a task class c is calculated as a function of time since a last access (forget if o is not part of the solution) and the rewards given by a user (train, if o is part of the solution)".

$$\text{rel}(o,t,c) = \begin{cases} \text{train}(o,t,c) & \text{if o is part of the solution} \\ \text{forget}(o,t,c) & \text{if o is not part of the solution} \end{cases}$$

Using this formula an object seems to be important if it is used frequently. The relevance should be increased.

An application of corpus-based terminology extraction in interactive information retrieval presented in (Peñas *et al.,* 2001). Using this approach, the terminology obtained in an automatic extraction procedure without any manual revision, to provide retrieval indexes and a "browsing by phrases" facility for document accessing in an interactive retrieval search interface. In addition, they suggested that the combination of automatic terminology and interactive search provides an optimal balance between controlled-vocabulary document retrieval and free text retrieval.
The method used in this work based on the comparison of two corpora extracted from the web: the first one, an appropriate corpus in the domain and, the second, a corpus in a different and more general domain. The comparison of terms in both corpora facilitates the detection of specific terms of the determined domain.

They also proposed a formula for term weighting that gives a relevance value to every detected term, in order to select the most relevant terms in the domain. This formula satisfies two constraints:

- Less frequent terms in the domain corpus should have less relevance.
- Highly frequent terms in the domain corpus should have higher relevance, unless they are also very frequent in the comparison corpus or they appear in a very small fraction of the documents in the domain corpus.

The formula considers term frequency in the collection ($F_{t,sc}$), document frequency of terms in the collection($D_{t,sc}$), and term frequency in a more general domain ($F_{t,gc}$). The relevance formula: Relevance (t,sc,gc)= 1-

$$\frac{1}{\log_2\left(2+\frac{F_{t,sc} \times D_{t,sc}}{F_{t,gc}}\right)}$$

Atif *et al.* (2003) expanded the learning object metadata to accommodate individual learner's needs, and to enable dynamic generation of personalized learning routes. In this work, they suggested learning objects construct to be used as building blocks to root out individual learning deficiencies, and then they proposed an algorithm to give learning routes suited for individual learners, adjusted to learner's profile.

Carchiolo *et al.* (2002) proposed an adaptive system for e-learning, which provides students with all paths from an initial knowledge to a desired one. The paths are retrieved and optimized based on student profile and teacher profile. Thus discarding those paths, which are not in accordance with the student's needs; the remaining paths are presented to the student to select one path and learn its course units.

Based on this system, Zhao and Wan (2006) proposed an algorithm to select the shortest learning paths to learn the target knowledge. They assumed that a course is modeled as a graph, in which each node represents a knowledge unit (KU), and two nodes in the graph are connected if the first node is a perquisite to the later node. In addition, they considered the weight of the course graph to be managed by teachers. Then they defined the best learning path as the learning process that will cost the least time and effort. Thus, they introduced the shortest learning paths algorithm.

Benlamri *et al*. (2003) represented the content structure of the course by learning object graph (LOG), and classified the peaks of LOG into two categories: Mandatory learning object, and secondary learning object. Based on this structure, Viet and Si (2006) built an adaptive course generation (ACG) system to create adaptive courses for each learner based on evaluating demand, ability, background and learning style of them. In the course content there is a test in each section, an algorithm is proposed to select the learning objects (LO) from the learning object graph, which are suitable for the requirements of learner.

Karampiperis and Sampson (2004) addressed the learning object selection problem in intelligent learning systems and they introduced a decision model that mimics the way the instructional designer decides. They proposed a function that estimates the suitability of a learning object for a specific learner. The same methodology they proposed in educational hypermedia systems (Sampson and Karampiperis, 2004).

Karampiperis and Sampson (2005) suggested some changes on the previous methodology, such that they construct a similar function with several assumptions; the first one is that the elements of the user model defined by the designer and remain the same during the life cycle of the system. The second assumption is the learners characteristics and preferences stored in user model and the structure of the educational resource description model have been defined by the instructional designer. Then they used this suitability function for weighting the connections of the learning paths graph in adaptive educational hypermedia systems (AEHS). They assumed that using this function make the most suitable path is the shortest between two nodes, and they used simulation to compare the learning paths generated by the proposed methodology with ideal ones produced by a simulated perfect rule-based AEHS.

In Liu and Greer (2004) a framework for individualized learning object selection is proposed. This framework gives a suggestion to select a group of suitable learning objects for the learner, also it evaluates the suitability of a learning object using information about the learning object, information about learner, and historical information about the learner and the learning context. This framework was divided into three steps: eliminating irrelevant learning objects depending on some features of the learning object, the second step was to select learning object depending mainly on educational information and pedagogical principles, finally optimization for the selected learning objects had to be performed.

The analysis of the above-mentioned work reveals the fact that they can be classified in two major groups; the first group emphasizes the need for e-Learning to be adaptive (Atif *et al*., 2003; Viet and Si, 2006; Karampiperis and Sampson, 2004; Liu and Greer, 2004). While the second group, emphasizes the efficiency (Zhao and Wan, 2006; Pythagoras and Demetrios, 2004).

As a representative for the first group, we select the work suggested by Liu and Greer (2004); while a representative for the second one is the work suggested by Zhao and Wan (2006). This is justified by the fact that Eliminating and Optimized Selection (EOS) suggested by Liu and Greer (2004) aimed at improving the ability to select dynamically an appropriate learning object for a specific learner, while the shortest learning path suggested by Zhao and Wan (2006) aimed at selecting a learning path that costs least time and effort.

Our research aims at obtaining the benefits of both groups this can be achieved by an attempt to combine the above-mentioned representative algorithms. In order to decide how these two approaches can be combined, the above mentioned representative approaches were further analyzed, implemented and then experimented. As a result, a formalization and some modifications to the above algorithms were suggested. Finally, a new approach is proposed to combine these representative algorithms.

In the analysis of these representative algorithms, some concepts related to e-Learning and Graph Theory are used. Thus, to clarify these concepts, the following subsections give basic definitions and concepts of e-Learning and graph theory.

2.1. Definitions and Concepts

Online learning is the use of network technology to design, deliver, select, administer, and extend learning (Fournier *et al.*, 2006). Thus, e-Learning can be defined as any type of learning delivered electronically (Codone, 2001). The "e" in e-Learning stands for education, it is not about bandwidth, servers, and cables (Masie, 2004). The term e-Learning was originally coined by Jay Cross in 1998, who has also suggested since then that: “It has become trite to point out that the ‘e’ doesn’t matter and that it’s the learning that counts” (Fournier *et al.*, 2006).

Brandon Hall (1997) defines e-Learning as “instruction delivered electronically wholly by a web browser, through the Internet or an intranet, or through CD-ROM or DVD multimedia platforms”. While according to Yu *et al.* (2006) The term "e-learning" refers to on-line learning delivered over the World Wide Web via the Internet.

Thus, the common understanding of e-Learning includes web-based training or learning products delivered via a web browser over a network. e-Learning is just a media. Hence, everything fundamental about learning applies as well.
e-Learning products can be acquired and used in two primary ways: by purchase of commercial off the shelf products or through customized builds of content produced for specific purposes (Codone, 2001).

Examples of the areas in which e-Learning products may be used:

- Introductory training to employees, customers, or other personnel.
- Refresher or remedial training.
- Training for credentialing, certification, licensing, or advancement.
- Academic/educational accreditation via college and university online learning.
- Promotion of products, policies, and services.
- Support organizational initiatives by increasing motivation through easily accessible learning.
- Orientation to geographically disparate personnel.

- Essential and nonessential learning opportunities for different users and in different subjects.
- Coaching and mentoring through online instruction and collaboration.
- Distributed online training and communication to build communities of practice.
- Standard and common training through fixed content accessible to all users.

However, for users that are interested in e-Learning, the following skills are needed: Self Advocacy: "I need to learn", Self Sufficiency: "I am responsible for my learning", Self Confidence: "I can Learn", Learning Process: "I know how I learn", and Self Evaluation: "I know whether I am learning". Without those skills, e-Learning is acknowledged as difficult (Masie, 2004).

The use of e-Learning offers benefits not realized in traditional training. e-Learning is beneficial to education, corporations and to all types of learners, some of these benefits (Horton, 2000; Afaneh *et al.*, 2006):

1. Users can learn at their own computer, without leaving the work site
2. Training may be done in bite-size chunks, when and where it is needed
3. e-Learning satisfies the training needs of a geographically dispersed workforce, without large investments in Travel and Living expenses.
4. Training can be completed at the learner's own pace e-Learning has been demonstrated to increase learning retention rates.
5. Multimedia presentation and interactivity reinforce understanding and application, Interactive training activities allow users to develop and practice skills easily.

6. e-Learning allows students to select learning materials that meet their level of knowledge, interest and what they need to know to perform more effectively in an activity.
7. Real-world business examples help users understand the context of each lesson and apply what they're learning
8. Learners receive a consistent message regardless of when or where they access training.
9. e-Learning can rapidly reach and make productive large numbers of learners.
10. Cost saving: e-Learning is more cost effective than traditional learning because less time and money is spent.
11. e-Learning encourages students to take personal responsibility for their own learning. When learners succeed, it builds self-knowledge and self-confidence in them.

However, beside the benefits lists before there are some shortcomings of e-Learning that can be summarized as follows (Afaneh *et al.*, 2006):

1. Learners need to have access to a computer as well as the Internet.
2. Learners need to have computer skills with programs such as word processing, Internet browsers, and e-mail.
3. Slow Internet connections or older computers may make accessing course materials difficult.
4. Managing computer files and online learning software for learners of beginner-level may seem complex. Some of the students also may have trouble installing software that is required for the class.

5. Instructions are not always available, so learners need to have discipline to work independently and they will feel isolated from others.
6. e-Learning requires as much time for attending class and completing assignments as any traditional classroom course. This means that learners with low motivation or bad study habits may fall behind.

2.2. Brief History of e-Learning

e-Learning (pre-1983): Before computers were widely available, Instructor Led Training (ILT) was the main training method. In ILT students had to get away from their office to focus on their studies and to interact with their teacher. This way meant high costs and down time during office hours, so training providers try to search for a better way to train (Thomson, 2007). An important milestone for the development of eLearning was the building of Interactive Satellite Television Network by the IBM in 1983 (Cross and Berkeley, 2004).

e-Learning (1984-1993): The technological advancements of the Multimedia era were Windows 3.1, Macintosh, CD-ROMs, and PowerPoint. Training providers try to make training more transportable and visually engaging by making computer based training (CBT) courses delivered via CD-ROMS. This provided time and cost savings that instructor-led training couldn't achieve, as well as reshaped the training industry. Despite these benefits, CD-ROM courses lacked instructor interaction and dynamic presentations, making the learning experience slower and less engaging for learners (Thomson, 2007). The development of information and communication technology (ICT) introduced more tools for use of distant education. For example, courses were courses transmitted by radio. This initiated vast telephone and radio based distant education projects, such as Pennsylvania State College in 1922 (Cross and Berkeley, 2004).

e-Learning (1994-1999): As the web evolved, training providers began thinking how this new technology could improve training. The advent of email, web browsers, HTML, media players and simple JAVA began to change multimedia training face. Basic mentoring via email, intranet computer based training (CBT) with text and simple graphics emerged which provided low quality delivery (Friel, 2007).

e-Learning (2000-now): Technological advances including JAVA/IP network applications, rich streaming media, high broadband access and advanced web site design are the real causes of the training industry development. In 2005 Live instructor-led training (ILT) via the Web can be combined with real time mentoring, improved learner services and up to date engaging content, to create a highly effective learning environment. These solutions provide cost savings, higher quality learning experiences, and are setting the standard for the next wave of e-Learning (Friel, 2007).

2.3. e-Learning Strategies and Concepts

There are two learning strategies: the instructive model and constructive model; the instructive model simulate the teacher task in a class room, the user goes step by step towards the course objectives. This model does not take the differences between students in account. The system based on this model will be less interactive with users or learners.

The constructive model allows students to build their own knowledge following different learning paths based on background of each one of them. Hence, in constructive model learning is tailored to the learner needs.

In e-Learning systems, the term network is used to reflect the way by which the learners access learning objects. Thus, there are two modes to access learning objects: synchronous and asynchronous learning. In synchronous mode the learner has to synchronize his schedule with another person or event. The asynchronous mode delivers learning without regard to distance of time constraints (Benlamri *et al.*, 2003). For example, the event in a live training - like a class- is synchronous, because the event and the learning occur at the same time.

Asynchronous learning occurs in an online course in which you complete events at different times, and communication occurs via time -delayed email or in discussion list postings (Codone, 2001).

Regardless of e-Learning strategies Learning Object is the core of any e-Learning system, in the following subsections we introduced this instructional technology concept that is commonly known as the “learning object”. Followed by a brief discussion of learning objects characteristics, learning objects metadata, and some concepts related to learning objects.

2.3.1. Learning Objects

Learning objects are defined as electronic units of educational information that are flexible, reusable, customizable, interoperable, retrievable, facilitate competency-based learning, and increase the value of content (Berg, 2007).

According to (Doe, 2007), a learning object is defined as a structured, standalone resource that encapsulates high quality information in a manner that facilitates learning and pedagogy.

The definition highlights two aspects of learning objects, “learning” and “object” with the underlying theme being “quality”. Quality relates to the following:

- Concept matter accuracy and authenticity.
- Pedagogical or educational value.
- The information in a resource is relative to its objective.
- LO features that represent usefulness.
- Technical “soundness” of learning objects (Doe, 2007).

Each learning object will be an encapsulation of its metadata and learning content when processed by the content-packaging, and will be packaged and classified so as to facilitate discovery and reuse by instructors and learners

2.3.2. Learning Objects Characteristics

Gerry Paille has defined the characteristics of Learning Objects as follows (Paille,2007):

- Learning objects are digital
- Learning objects can be stored in a database or repository
- Learning objects can be described using a metadata standard or specification
- Learning objects are discoverable through searching a database
- Learning objects are interoperable in that they are independent of hardware, operating system and browser type
- Learning objects tend to be, but are not necessarily, small or granular in nature
- Learning objects tend to be, but are not necessarily, disassociated from context

- Learning objects are reusable
- Learning objects can be repurposed for different educational contexts
- Learning objects have an explicit educational purpose

The common characteristics of learning objects are accessibility, interoperability, adaptability, reusability, and granularity (LOAZ, 2007).

Interoperability describes the capacity of items of software or hardware to work together. Also Interoperability can be described as “a condition that exists when the distinctions between information systems are not a barrier to accomplishing a task that spans multiple systems” (Aroyo *et al.*, 2006). While in e-Learning interoperability is associated with the design of Web-based resources, that can operate across various forms of hardware platform, browser type and courseware delivery system (Oliver, 2001).

Adaptability ensures that the learning object is tailored perfectly for individual learners needs.

Reusability is the most important characteristic, because Entire courses may not be appropriate for re-use in different institutions, but individual learning objects can be selected, and reused as components of a much wider course.

Granularity refers to how accurately we choose to break down and store our learning objects. The unit of a learning object can be a program, a course a module, a lesson, a segment, or a raw object.(LOAZ, 2007).

2.3.3. Learning Objects Metadata (LOM)

Metadata is structured data about data. Learning object metadata is data about learning objects and resources. Metadata describes how and when and by whom a particular set of data was collected, and how the data is formatted. Metadata is essential for understanding information stored in data warehouses and has become increasingly important in XML-based Web applications.

The purpose of the LOM Standard is to facilitate search, evaluation, acquisition, and use of learning objects, for instance by learners or instructors or automated software processes. This standard also facilitates the sharing and exchange of learning objects, by enabling the development of catalogs and inventories while taking into account the diversity of cultural and lingual contexts in which the learning objects and their metadata are reused (Godby,2007).

Examples of LOM standards are **IEEE** Learning Object Metadata, The Canadian Core Learning Object (**CanCore)** Metadata, and the Information Management System (**IMS)** Learning Resource Meta-data.

The Canadian Core Learning Object (CanCore) Metadata Application Profile was intended to facilitate the interchange of records describing educational resources and the discovery of these resources both in Canada and beyond its borders. CanCore is based on and fully compatible with the IEEE Learning Object Metadata standard and the IMS Learning Resource Meta-data specification.

2.3.4. Learning Object Attributes

It is important to understand the functional requirements of learning objects in terms of courseware authoring, interaction and media selection. When developing courseware content, the instructor may breabk down the subject matter into a network of concepts representing several layers of different details to achieve the instructional goals. In the same manner learning objects represent small and reusable chunks of instructional media. This object-based segmentation of knowledge has been adopted in (Atif *et al.,* 2003) to provide a constructive approach to e-learning

In standard LOM groups such as general, rights, lifecycle, classification and annotation to describe the static features of the learning object are included. However, additional features were added by (Atif *et al.*, 2003) such as educational, technical and relation to dynamically adapt the learning object to learners' needs. The structure of learning object attributes is shown in Figure 1. Where:

- The General category: groups the general information that describes the learning object as a whole.
- The Lifecycle category: groups the features related to the history and current state of this learning object and those who have affected this learning object during its evolution.
- The Meta-Metadata category: groups information about the metadata instance itself.

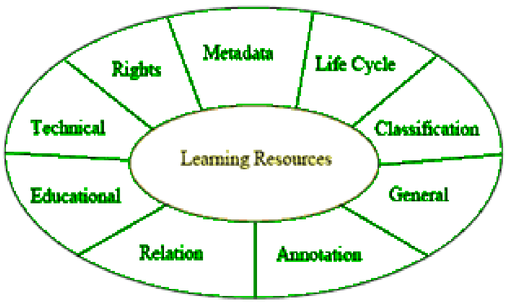


Figure 1: Learning Object Attributes (Atif *et al.*, 2003).

The Technical category groups the technical requirements and technical characteristics of the learning object. While the Educational category groups the educational and pedagogic characteristics of the learning object.

The Rights category groups the intellectual property rights and conditions of use for the learning object (IEEE LOM 2002).

The Relation category groups features that define the relationship between the learning object and other related learning objects. It reflects the self-adaptability nature of the learning object. As a response to a learner state, a new learning sequence of learning objects is generated to control the the presented material to be suitable for the learner state. Different learners follow different learning routes suitable to their background level and learning type (Atif *et al.*, 2003).

The Annotation category provides comments on the educational use of the learning object and provides information on when and by whom the comments were created. And the Educational element presents features related to media selection, analogy, assessment and customization (IEEE LOM 2002).

The technical attributes represent the synchronization and layout features describing respectively the level of synchronization involved in combining multiple media, and the actual time and space distribution of the learning media.

The Classification category describes this learning object in relation to a particular classification system. It may be used also to provide certain types of extensions to the LOM Schema, as any classification system can be referenced. Collectively, these categories form the LOM Schema.

2.3.5. Learning Objects Graph (LOG)

The Learning Objects Graph (LOG) is a directed graph which represents all possible learning paths . All learning objects (LO) belonging to the same course are connected together into a graph structure using oriented arrowheads. Each LO may contain one concept, principle, a definition, worked example, an exercise (Zhao and Wan, 2006), an image, or an audio session.

Two nodes are connected if there exist a dependency relation between them. The relationship between two LO can be divided into three types: precedence relationship, succession relationship and parallel relationship (Zhao and Wan, 2006). As an example a relationship exists between two nodes if one node is a prerequisite to the other.

2.3.6. Learning Objects Suitability

The suitability of a learning object requires evaluation based on its features. Whether a learning object is suitable depends on the context where the learning object is used, and some properties of the learning object, such as:

- Learning object appropriateness with respect to the learning goals.
- Its usefulness for the learners.
- Learning object pedagogical value.
- Learning object popularity among learners.
- Endorsement by instructors.

Learner characteristics play a significant role in learning object selection. The more that is known about a learner the better the selection of learning objects that can be made for him/her.

2.3.7. Adaptive Learning

Adaptive learning addresses the fact that individuals learn differently by adapting the presentation of learning content to meet the varying needs and learning preferences of different individual learners. It is important because it enables learners to select their modular components to customize their learning environments. Secondly, it enables them to get flexible solutions that dynamically adapt content to fit individual learning needs. Experience has shown that the best way to improve learning is to respond to clearly identified needs and clearly articulated solutions (Martinez, 2007).

2.4. Learning Process using Learning Objects

The learning object attributes represented in Figure1 personalized learning by providing five LO functionalities. These are (Atif *et al.*, 2003): LO sequencing, LO structure, LO presentation, LO navigation support, and LO interactivity. The following subsections describe these LO functionalities and their contribution to enable adaptive learning.

2.4.1. Learning Objects sequencing

Adaptive e-Learning systems enable computer agents to automatically and dynamically compose personalized lessons for a specific learner. To achieve this objective, instructional design should not be structured in the traditional sequential format where all learners given the same instruction regardless of their needs and background. But learning objects should be invoked dynamically to form a learning path that is suitable to root out the learning deficiencies of individual learners.

As learners navigate in the e-Learning system, the system will adapt the content based on the learner information. For example, learners might be sent to different learning objects in the content based on user-initiated request for clarification of prerequisite knowledge, or user requests for preferred knowledge presentation, such as examples, case studies or procedural information (Atif *et al.*, 2003).

e-Learning system gives alternative learning styles through the use of additional learning objects such as examples, case studies, and procedural information, in order to provide personalized learning. These options give learners the flexibility to choose a suitable learning path instead of a rigid one.

The learning path-building process, which contains the sequence of objects exposed to a learner performed dynamically based on the learner's needs (Atif *et al.*, 2003).

2.4.2. Learning Objects Structure

Learning objects consist of a sequence of learning tasks to accomplish the objectives set up by the instructor for a good understanding of concepts presented in the LO (Atif *et al.*, 2003). These are combination of learning resources that can be: slides, examples, questions, problems, simulations, case studies, experiments, diagrams, graphs and so forth. These are structured in a way to allow different learning styles (i.e. auditory, visual …etc) at different learner levels (e.g. beginner, trainer …etc) depending on the learner's profile and preferences.

2.4.3. Learning Objects Presentation

This functionality describes the way individualized learning materials contained into the learning object are dynamically presented to the learner. In order to contributes further to learning, instructional designers use the most effective medium to present specific information. There is a need for instructional designer to map a learning content to an appropriate media. Many studies suggest how to select specific media or a combination of media for presenting specific kinds of learning content (Benlamri *et al.*, 2003).

Assembly instructions are best comprehended when an assembly task is presented using a combination of illustrations and text highlighting the major steps. Procedural information for operating a particular device appears to be more helpful for learners to acquire when a combination of animation or video and text is presented to learners. For problem-based learning, an animation with verbal narration is effective. Also solving a mathematical equation may be better illustrated through a graphical illustration. Pictures with text or verbal narration appear to be helpful to drive the learner to focus on specific features of the pictures, because pictures increase recognition accuracy. Sound appears to be an effective way in learning a particular foreign language (Benlamri *et al.*, 2003).

2.4.4. Learning Objects Navigation

Different Learning Objects have different navigation alternatives, depending on their type, role, content and structure. For example, a learner starting a problem solving LO is recommended to go through all problem solving steps, however, it is not recommended to explore all alternatives in a LO consisting of a number of examples/case studies describing the same concept (Atif *et al.*, 2003).

2.4.5. Learning Objects interactivity

Interactivity may differ from one Learning Object to another depending on its type and role. e-Learning systems allow learners to interact with most Learning Objects, and especially with those Learning Objects related to problem solving, questionnaires and self-assessment. Learner's responses are saved into learner profile and may be used for personalization purposes and future guidance.

2.5. Automatic Course Sequencing Techniques

There are two approaches for the automatic use of course sequencing (Brusilovsky and Vassileva, 2003): adaptive courseware generation, and dynamic courseware generation, in this section we will explain the idea of each one of these approaches and discuss its advantages.

2.5.1. Adaptive Courseware Generation:

The idea of this approach is to generate a course suited to the needs of the learners. This approach can deliver adaptivity for small group of students, and it allow learners to communicate through the shared context and learn from each other. Another advantage of this technique is that the static course that it generates can be delivered by a regular course management system (Brusilovsky and Vassileva, 2003).

2.5.2. Dynamic Courseware Generation:

The goal of this approach is to generate an individualized course taking into account a learning goal and the initial knowledge level of learner. If the learner does not meet expectation, the course dynamically re-planned. This approach applies adaptively to an individualized learner. In order to generate an individualized course, this course should take into account the learner's knowledge, goals, and timeframe, and to be adaptive it considers their difficulty, and rate of progress (Brusilovsky and Vassileva, 2003).

2.6. Graph Theory

Graph theory is a branch of mathematics. In mathematics and computer science, graph theory is the study of graphs, mathematical structures used to model pair wise relations between objects from a certain collection (Biggs, 2007).

A graph is a set V of vertices and a set E of edges - pairs of elements of V. This simple definition makes Graph Theory the appropriate language for discussing relations on sets. Among the topics of interest are topological properties such as connectivity, paths, cycles, and distances in graphs.

A course in e-Learning systems is modeled as a directed graph. Thus, in the following subsections we will explain some basic concepts in graph theory that we will use.

2.6.1. Brief history

The paper written by Leonhard Euler on the Seven Bridges of Königsberg and published in 1736 is considered as the first paper in the history of graph theory. This paper, as well as the one written by Vandermonde on the knight problem carried on with the analysis sites initiated by Leibniz. Euler's formula relating the number of edges, vertices, and faces of a convex polyhedron was studied and generalized by Cauchy and L'Huillier, and is at the origin of topology.

More than one century after Euler's paper on the bridges of Königsberg and while Listing introduced topology, Cayley was led by the study of particular analytical forms arising from differential calculus to study a particular class of graphs, the trees.

The involved techniques mainly concerned the enumeration of graphs having particular properties. Enumerative graph theory then rose from the results of Cayley and the fundamental results published by Pólya between 1935 and 1937 and the generalization of these by De Bruijn in 1959. Cayley linked his results on trees with the contemporary studies of chemical composition. The fusion of the ideas coming from mathematics with those coming from chemistry is at the origin of a part of the standard terminology of graph theory. In particular, the term graph was introduced by Sylvester in a paper published in 1878 in Nature.

The work of Ramsey on colorations and more specially the results obtained by Turán in 1941 is at the origin of another branch of graph theory, the extremal graph theory.

The autonomous development of topology from 1860 and 1930 fertilized graph theory back through the works of Jordan, Kuratowski and Whitney. Another important factor of common development of graph theory and topology came from the use of the techniques of modern algebra. The first example of such a use comes from the work of the physicist Gustav Kirchhoff, who published in 1845 his Kirchhoff's circuit laws for calculating the voltage and current in electric circuits (Biggs, 2007).

The introduction of probabilistic methods in graph theory, is at the origin of another branch, that is random graph theory. Research in this branch has enabled mathematicians across the globe to advance the theory of graphs significantly

2.6.2. Graphs and Basic Concepts

Graphs are useful structure in Computer Science. They arise in all sorts of applications, including scheduling, optimization, communications, and the design and analysis of algorithms.

Definition:

Graph: A graph G is a pair G = (V,E) where V is a finite set, called the vertices of G, and the elements of E are called the edges of G. $E = \{(u,v) \mid u, v \in V\}$.

Vertex: A vertex is a terminal point or an intersection point of a graph. It is the abstraction of a location such as a city, an administrative division, a road intersection or a transport terminal (Rodrigue *et al.*, 2006).

Edge: An edge *e* is a link between two nodes. The link (*i* , *j*) is of initial node (vertex) *i* and of terminal node *j*. for example, a link is the abstraction of a transport infrastructure supporting movements between nodes. For an edge e that joins vertices u and v we write *e* =(u,v).

If the edge has a direction that is commonly represented as an arrow, the graph is called directed graph, and when an arrow is not used, the graph is called undirected graph as shown in Figure 2.

A graph that has a weight, or numeric value, associated with each edge, is called weighted graph (Black and Tanenbaum, 2007).

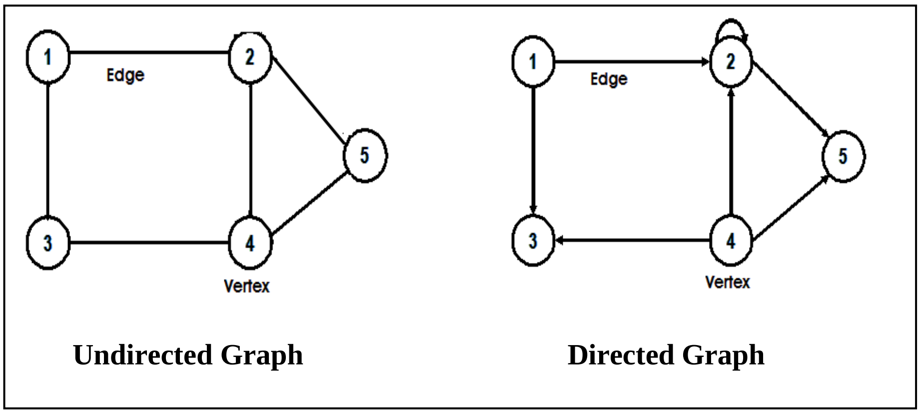


Figure 2: Directed and undirected graphs

2.6.3. The Degree of a Vertex

In an undirected graph, the degree of a vertex is the number of incident edges. If two vertices u and v are joined by an edge, then they are adjacent.

In a directed graph the in-degree is the number of incoming edges, and the out-degree is the number of outgoing edges. If there is an edge from u to v, then v is adjacent to u.

2.6.4. Paths and Cycles

A sequence of links that are traveled in the same direction is called a Path. For a path to exist between two nodes, it must be possible to travel an uninterrupted sequence of links. Finding all the possible paths in a graph is a fundamental attribute in measuring accessibility and traffic flows (Rodrigue *et al.,* 2006). Another definition of a path is a list of vertices of a graph where each vertex has an edge from it to the next vertex.

A cycle refers to a chain where the initial and terminal node is the same and that does not use the same link more than once is a cycle. Also it can be defined as a path in a graph that starts and ends at the same vertex and includes other vertices at most once

(Black and Tanenbaum, 2007). Acyclic graph is a graph with no path that starts and ends at the same vertex.

Acyclic graph is a graph with no path that starts and ends at the same vertex. In the other hand Directed acyclic graph is a directed graph with no path that starts and ends at the same vertex. Also known as DAG, acyclic directed graph, oriented acyclic graph.

CHAPTER 3: ANALYSIS AND IMPLEMENTATION OF ELIMINATING AND OPTIMIZED SELECTION APPROACH

Learners in an online virtual course may have different backgrounds than those in a traditional course. The expected benefits of a learning object and the learning effect gained from it are usually different from learner to learner. So, the traditional approach that presents one content selection to all learners becomes inadequate in an online learning environment.

Different learners have their distinctive characteristics and learning styles. Also they may use different software, and hardware. Thus, a framework for individualized learning object selection, called Eliminating and Optimized Selection (EOS) was proposed in (Liu and Greer, 2004). This framework represents an approach to select a short list of suitable learning objects appropriate for the learner and the learning context. An outline of e-Learning system using EOS approach is proposed in Figure 3.

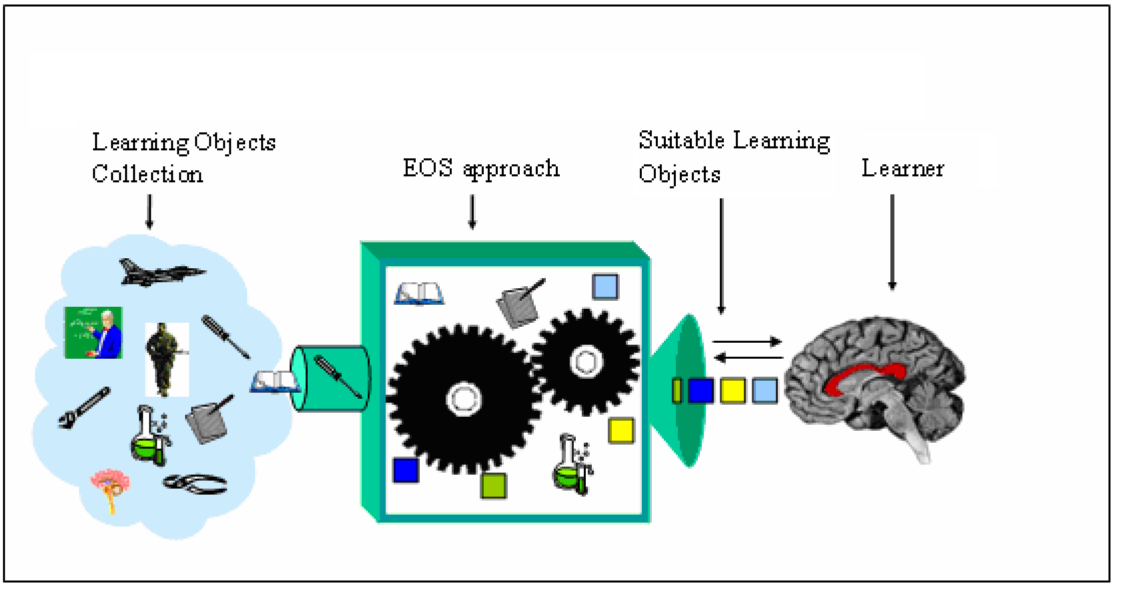


Figure 3: selecting suitable learning objects for learners.

The key features of the EOS approach are to evaluate the suitability of a learning object in its situated context and to optimize the evaluation by using historical information about the learner, the learning object, and the learning context. The suitability of a learning object requires an evaluation based on its features. Whether a learning object is suitable depends on its own features and the context where it is used (Liu and Greer, 2004).

The analysis of this framework reveals the fact that the attributes of a learning object can be classified into two groups: eliminating attributes and selecting attributes as shown in Figure 4. These attributes are used in different phases of EOS. The eliminating attributes are used in the filtering phase where certain Learning objects are eliminated if they do not match the learner's needs. The selecting attributes are used in the selection phase where each learning object assigned a value according to the comparison between the selecting attributes and learner's characteristics. The resulted set of learning objects will be candidate to enter the optimization phase, in which a value assigned to these learning objects according to the history of using learning objects by previous learners.

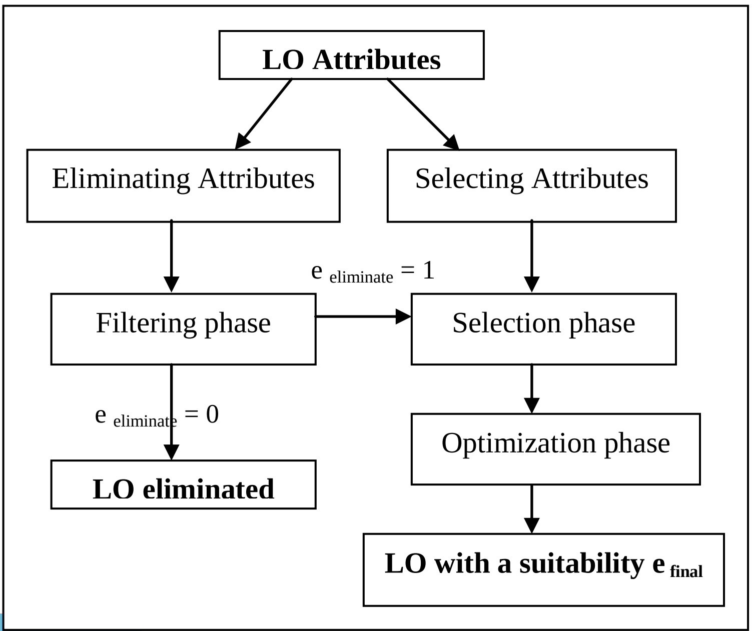


Figure 4: Learning object attributes classification

Thus, the Eliminating and Optimized Selection (EOS) approach consists of three main phases:

Phase 1: Eliminating irrelevant objects.

Phase 2: Selecting a candidate learning object.

Phase 3: Optimization for the selected learning objects.

In order to implement these steps, information about learning objects and learners are required.

3.1. Required Information for Individualized Selection

The learning object metadata have a defined set of attributes that describes learning object. These attributes used to decide when the learning object is suited in a certain context. Some information about the learner is necessary in addition to the information about learning object. Historical usage of learning objects can also help in optimizing selection.

For the purpose of our implementation, we organize such required information in five tables that are described bellow: Table1 Learning Object, Table 2 Learner, Table3 Language, Table 4 Environment, and Table 5 History of using Learning Objects, along with attributes that link these tables together. These tables are based on Cancore standard, as defined in (Liu and Greer, 2004). Some categories were expanded based on the educational literature (Honey and Mumford, 1992) for each table.

Table 1 contains the Learning object attributes. Some of the categories used in this table are described bellow:

Pedagogical Objective: describes the concept that the learning object presents and what is expected to achieve by the learner after presenting this learning object. Pedagogical objective is a critical attribute for determining the suitability of a learning object. Pedagogical objectives might be indirectly achieved from attributes such as keyword and description. An ontology-based representation of pedagogical objectives may serve much better.

Table1: Learning Object attributes.

Attribute Name	Explanation
Learning Object ID	An Identifier of the learning object
Language ID	The language in which the content is presented
Environment ID	The technical requirements needed for presenting the learning object
Current learner ID	Current learner using the leaning object
Pedagogical Objective	The concept represented in the learning object
Cost	The price of the learning object
Expected Reading Level	The reading capability required by the learning object.
Prerequisite	The knowledge needed by the learning object
Typical Learning Time	Time needed for working with the learning object
Presentation Type	The way of presenting the content of the learning object

Expected Reading Level: indicates the reading capability, which the learning object requires the learner to have. Learners in the same level of education or in the same age may have different reading ability. Reading level actually an important role.

Prerequisite: specifies the knowledge needed by the learning object. but it is a very important factor for deciding the suitability of a learning object for a specific learner.

Learner characteristics play a significant role in learning object selection. The information about the learner can be used to decide the degree of the match between learning object features and the learner's preferences (Liu and Greer, 2004). For instance, Financial Situation attribute means how much money will the learner be able to pay for accessing a learning object. While the Time attribute provides information about the time that the learner is going to spend on a learning object. A lengthy learning object may be not a good choice for a learner who can devote only very limited time.

Table 2 contains attributes of the Learner table, which are self-explanatory. Theoretically, the more that is known about a learner, the better the selection that can be made for this learner.

Table 2: Learner attributes.

Attribute Name	Explanation
Learner ID	An Identifier for the learner
Learner Name	First Name and last name of the learner
Learning objective	The subject or topic the current learner is going to learn
Learner Type	Learner's category
Background	Information about related knowledge or experiences of the learner
Knowledge in Related Area	Learner's level of domain related knowledge
Preferred Language	Language that the learner prefers
Reading Level	Learner's capability of understanding written materials
Listening Level	Learner's capability of understanding vocal materials
Reading Speed	Learner's speed of reading
Preferred Presentation Type	Learner's preference about the way in which the content is presented
Learning Style	Learner's way of learning new concepts or knowledge
General Academic Achievement	Information about the learner's academic performance
Environment ID	Computer environment (hardware, and software)
Financial Situation	Financial restriction
Time	Time the learner wishes to spend

Table 3 indicate attributes used for designing the table of Learning object history, which includes features relating to quality and appropriateness of a learning object, these features provide very useful information for optimized selection. Information defined in Learning object history are related to:

Previous Learners*:* models or records of learners who accessed the learning object in the past as well as their actions, evaluation, cognitive state, and achievement related to the learning object (Liu and Greer, 2004).

Previous Instructors*:* teachers who have accessed the learning object and their endorsements of the learning object.

Table 3: Learning Object History attributes.

Attribute Name	Explanation
Learner ID	Learner identifier
Learning Object ID	Learning object identifier
Accessing Time	The time when the learning object is accessed by the learner
Learner status	The learner status after using the learning object
Learning Style	Learner's way of learning new concepts or knowledge
Learner Type	Learner's category
General Academic Achievement	Information about the learner's academic performance
Interactions	Actions the learner makes while accessing the learning object
Evaluation	The learner's opinion about the learning object
Achievement	The assessment result of the learner after working with the object
Previous instructor ID	Teachers who have accessed the learning object
General Popularity	How often the learning object is selected for all type of learners
Specialized Popularity	How often the learning object is selected for certain type of learners

For the purpose of implementation, Table 4 and Table 5 were introduced to define the Language, and the Environment characteristics respectively.

Table 4: Language attributes.

Attribute Name	Explanation
Language ID	The identifier of the language
Language Name	Human language name

Table 5: Environment attributes.

Attribute Name	Explanation
Environment ID	The identifier of the environment
Software	Operating system type in the environment
RAM	Memory exist in the environment
CPU	CPU type used in the environment

3.2. Formalization of EOS

Based on learning object attributes a general framework to evaluate the suitability of a learning object is given in Figure 5. Where Eliminate (S) is a function that calculate the value $e_{eliminate}$ (0 or 1) for each LO_j in S, and then constructs the set S' as composing of learning objects with $e_{eliminate}$ equal (1). Select (S') is a function that assign a value e_{select} - considering selecting attributes- for each learning object in S', after that the function Optimize (S') is applied, in order to assign a value $e_{optimize}$ for each learning object in S'. Finally, the function Suitability (S') is applied to assign e_{final} for each LO in S', where e_{final} is the final evaluation result of the learning object and it is calculated as in Equation (1).

$$e_{final} = e_{eliminate} \times (e_{select} + e_{optimize}) \ldots\ldots\ldots\ldots\ldots (1)$$

where $e_{eliminate}$, e_{select} , and $e_{optimize}$ are calculated by Equation (2),(3), and (5) respectively.

The learning object that has the highest e_{final} value is the most suitable learning object. In the following sections we will discuss how to calculate each value of $e_{eliminate}$, e_{select} , and $e_{optimize}$.

Let $S = \{LO_1, \ldots, LO_j\}$ be the set of the learning objects from which an E-learning system is composed

$S_{eliminate} = Eliminate(S)$

where: Eliminate (S) constructs the sets $S_{eliminate}$ and S' such that:

- $S_{eliminate} = \{ e_{eliminate1}, \ldots, e_{eliminate\ j} \}$
- $e_{eliminate}$ is a value assigned for each $LO_j \in S$ as :

$$e_{eliminate\ j} = \prod^{i} a_{eliminate\ i} \quad , a_{eliminate} \in \{0,1\}$$

- $S' = \{ LO_j \in S \mid e_{eliminate\ j} = 1 \}$

$S_{select} = Select(S')$

where: Select (S') constructs the set S_{select} such that:

- $S_{select} = \{ e_{select\ 1}, \ldots, e_{select\ j} \}$
- e_{select} is a value assigned for each $LO_j \in S'$ as :

$$e_{select\ j} = \sum_{i} W_i \times a_{select\ i} \quad ; \ W, a_{select} \in [0,1]$$

- W_i is calculated by Equation (4)

$S_{optimize} = Optimize(S')$

where: Optimize(S') constructs the set $S_{optimize}$ such that:

- $S_{optimize} = \{ e_{optimize\ 1}, \ldots, e_{optimize\ j} \}$
- $e_{optimize}$ is a value assigned for each $LO_j \in S'$ as:

$$e_{optimize\ j} = \sum_{i} W_i \times a_{optimize\ i} \quad ; \ W, a_{optimizei} \in [0,1]$$

- W_i integer values to be given

$S_{suitability} = Suitability(S')$

where: Suitability (S') constructs the set $S_{suitability}$

- $S_{suitability} = \{ e_{final\ 1}, \ldots, e_{final\ j} \}$
- $e_{final\ j}$ is a value assigned for each For each $LO_j \in S'$ as:

$$e_{final\ j} = e_{select\ j} + e_{optimize\ j}$$

Figure 5: Evaluation of the suitability of Learning Objects

3.2.1. Eliminating Irrelevant Objects

The first phase in EOS approach is eliminating irrelevant objects, in other words, filtering process. This step depends on some attributes such as the following attributes:

- Pedagogical objective (Keyword, or Description)
- language
- Environment condition (software, hardware)

- Financial cost

The eliminating attributes are constraints so they are binary variables (1 or 0). If any attribute of the eliminating attributes did not match the requirements of the learner, the learning object will be omitted. In this step if an attribute satisfies the requirements, it has a value (1), and if the attribute does not fit in the current context, it has a value (0). Hence, the eliminating phase is based on applying Equation (2) for each learning object.

$$e_{\text{eliminate}} = \prod^{i} a_{\text{eliminate } i} \qquad \text{where } a_{\text{eliminate}} \in \{0,1\}\ldots\ldots\ldots\ldots\ldots (2)$$

In Figure 6, we formalize a function that is used to calculate $e_{\text{eliminate}}$ for each learning object. This function is called Eliminate(S).

Let $a_1, a_2, \ldots\ldots, a_8$ be the following attributes respectively:

- a_1: Objective in Learner table
- a_2: Concept in Learning Object table.
- a_3: Financial Situation in Learner table
- a_4: Cost in Learning Object table
- a_5: Environment ID in Learner table
- a_6: Environment ID in Learning Object table
- a_7: Language ID in Learner table
- a_8: Language ID in Learning Object table

four eliminating criteria are computed as follows:

$a_{\text{eliminate1}} = (a_1 = a_2)$
$a_{\text{eliminate2}} = (a_3 \geq a_4)$
$a_{\text{eliminate3}} = (a_5 = a_6)$
$a_{\text{eliminate4}} = (a_7 = a_8)$

We define a function F1 that returns 1 or 0 as follows:

F1 ($a_{\text{eliminate1}}$, $a_{\text{eliminate2}}$, $a_{\text{eliminate3}}$, $a_{\text{eliminate4}}$)
If $a_{\text{eliminate1}} \cap a_{\text{eliminate2}} \cap a_{\text{eliminate3}} \cap a_{\text{eliminate4}}$ then
return 1
Else
return 0

Figure 6: Calculating of eliminating criteria $e_{\text{eliminate}}$

3.2.2. Selecting Candidate Learning Object

To select the candidate learning objects. A suitability evaluation for each learning object is performed. This proceeds as follows:

- An importance analysis of the features surrounding each LO or context is performed. This analysis is reflected by assigning weight (W) for each attribute (feature) of the learning objects in a given context.
- A degree of match between these attributes and the requirement is performed. This degree is represented by a value between 0 and 1, and it is denoted by a_{select}

Thus, the selecting criteria for each LO is based on Equation (3).

$$e_{select} = \sum W_i \times a_{select\,i} \quad \text{where} \quad W, a_{select} \in [0,1] \quad \ldots\ldots\ldots\ldots\ldots \; (3)$$

where W is calculated by Equation (4) and $a_{select\,i}$ is calculated as shown in Figure 7.

For the purpose of our implementation, we will use time, presentation type, and reading level as selecting attributes for the learning object. We will use the learner style as a context to determine the importance of these selecting attributes. For example, if the learner style was visual then the most importance LO attribute will be the time then the presentation type, and finally the expected reading level. But if the learner style was auditory, then the attributes will be arranged according to their importance as follows: presentation style, time, and finally expected reading level. If the learner style was tactile and kinesthetic (i.e. learn by doing), then the most importance feature of the LO will be expected reading level, time, finally presentation style.

Hence, the importance of each attribute is presented by a weight W_i. According to the context, since in different context a learning object attribute affects the suitability in various ways. For the purpose of our implementation, the weight is calculated as in Equation (4).

$$W_i = P_i / N \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4)$$

Where:

P_i : the preference degree of the selecting attribute (i) according to the learner.

N : the number of selecting attributes.

For instance, if the learner style was auditory, then the weight for presentation style =1, weight for time =2/3, and finally weight for expected reading level =1/3. The degree of match for each attribute is a value in the interval [0, 1]. Figure 7 shows a formal definition for calculating the degree of match for each selecting attribute.

Let the properties of a Learning Object defined as $a_1 \ldots. a_i$
Let the properties of a Learner defined as $a_1 \ldots. a_j$
Where $a_1 \ldots. a_i$, $a_1 \ldots. a_j$ are integer values

Let n be the number of selecting attributes

Then

$a_{select\ i}$ is defined as the degree of match for each selecting attribute for a learning object, where:

$$a_{select\ i} = \begin{cases} 0 & , \text{if } a_j < a_i \quad i,j=1 \ldots n \\ 1 & , \text{if } a_j = a_i \quad i,j=1 \ldots n \\ (a_i / a_j) & , \text{if } a_j > a_i \quad i,j=1 \ldots n \end{cases}$$

Figure 7: Calculating the degree of match a_{select}

3.2.3 Optimization Phase

In some situations, a learning object which match a learner's preferences might not be the best for the learner, so the selection of the most suitable learning object can be optimized based on:

- Previous usage of the learning object.
- Expert's evaluation.
- Similar learner's experience.

- Popularities of the learning object.

In our implementation of optimization phase, we consider the following:

- General popularity of the learning object.
- Specialized popularity of the learning object.
- Previous similar learner's evaluation for the learning object.

Furthermore, the similarity between learners is based on learner style, learner level (e.g. beginner, expert… etc), and learner academic achievement. In order to select the learning objects that are suited for individualized learner, optimization phase is based on optimization criteria $e_{optimize}$ that can be calculated using Equation (5).

$$e_{\,optimize} = \sum W_i \text{ x } a_{\,optimize\ i} \quad \text{..................} \quad (5)$$

Figure 8 shows the calculation of $e_{optimize}$ for each learning object.

Let S'= { LO_1,LO_2,…LO_i } be the set of selected learning objects
Let Av be the average of similar previous learners evaluation
Let L_{given} be the current learner using the system
Let L_{cls} be Learning Style for current learner
Let L_{pls} be Learning Style for previous learner
Let L_{ctl} be Learner Type for current learner
Let L_{ptl} be Learner Type for previous learner
Let L_{pev} be the previous learner evaluation for $LO_i \in S'$
Let G_p be General Popularity of $LO_i \in S'$
Let S_p be Specialized Popularity of $LO_i \in S'$
Let w_1,w_2 ,and w_3 be weights assigned for Av, G_p , and S_p , respectively.

For each $LO_i \in S'$
$a_{optimize1} = G_p$
$a_{optimize2} = S_p$
$a_{optimize3} = Av(L_{pev})$

$$Av = \begin{cases} \text{average } (L_{pev}) & , \text{if } (L_{cls} = L_{pls}) \cap (L_{ctl} = L_{ptl}) \\ 0 & , \text{otherwise} \end{cases}$$

$$e_{optimize\ i} = (w_1 \times a_{optimize\ 1}) + (w_2 \times a_{optimize\ 2}) + (w_3 \times a_{optimize\ 3})$$

Figure 8: Calculating optimization criteria $e_{optimize}$

3.3. Implementation of EOS

Based on the previous analysis and formalization, EOS was implemented using Visual Basic.Net. This is because it was not implemented in (Liu and Greer, 2004). EOS was implemented in terms of its three phases and based on the following assumptions:

- Three Types for learners: Beginner, Trainer, and Expert.
- Languages: English, Arabic, French, etc.
- Three learning styles: Visual, Auditory, and Tactile & kinesthetic (Learn by doing).
- The weight for general popularity considered as: 0.5, while for previous learners evaluation it was 0.25, and finally 0.25 for specialized popularity.
- Nine presentation styles: Text, exercise, table, diagram, simulation, audio, slide, problem statement, and video.
- Minimum requirements for the environment (hardware,Software): P3 CPU 1300 MHz/ 128 Ram /16 VGA/Win98 , P4 CPU 200 MHz / 264 RAm /32 VGA/ WinXP ,P4 CPU 2300MHz / 512 RAM / 64 VGA/ WinXP, Centrino PM CPU 1.3 /1G Ram / 128 VGA / Win98, BM CPU 1.6/ 2G Ram /128 VGA/ Mac,… etc.
- Four values: weak, good, very good, and excellent were considered for Reading Level, Listening Level, and General Academic Achievement.
- An hour was the time measure unit.
- For the purpose of comparison process, numerated values were given for learning styles, presentation styles, learners types, reading level, listening level, and general academic achievement.

- Concepts were extracted from the ACM Computing Curricula 2001 for Computer Science (ACM, 2001), which defines 950 topics organized in 132 units and 14 areas.
- Six Objectives were considered (ACM, 2001): Knowledge and Understanding, Design and Implementation, Modeling, Method and Tools, Information Management, and Critical Evaluation and Testing.

EOS was experimented on three different learners and three learning objects (LO_1, LO_2, LO_3), the characteristics of these learning objects are given in Table 6.

Table 6: The characteristics of the three LOs that were used for EOS experiment

Learning Object	Characteristics		
	Presentation Type	Time	Required reading Level
LO_1	Exercise	1 hour	Excellent
LO_2	Table	3 hours	Very Good
LO_3	Diagram	1 hour	Good

The first learner was a beginner with a very good reading level and had 12 hour for learning, his learning style was Visual and his preferred presentation type was videos. The second learner was a trainer with a good reading level, 5 hours to learn, his learning style is Auditory and his preferred presentation type was audios. Finally, the third learner was an expert with an excellent reading level, his learning style was Tactile & Kinesthetic (learn by doing), 20 hours for learning and he preferred slides as a presentation type.

The results are given in Figure 9. These results show how the suitability of the three learning objects varies from one learner to another. For example, the suitability for LO_1 is 0.66, 0.74, 0.78 for Learner 1, Learner 2 and learner 3 respectively.

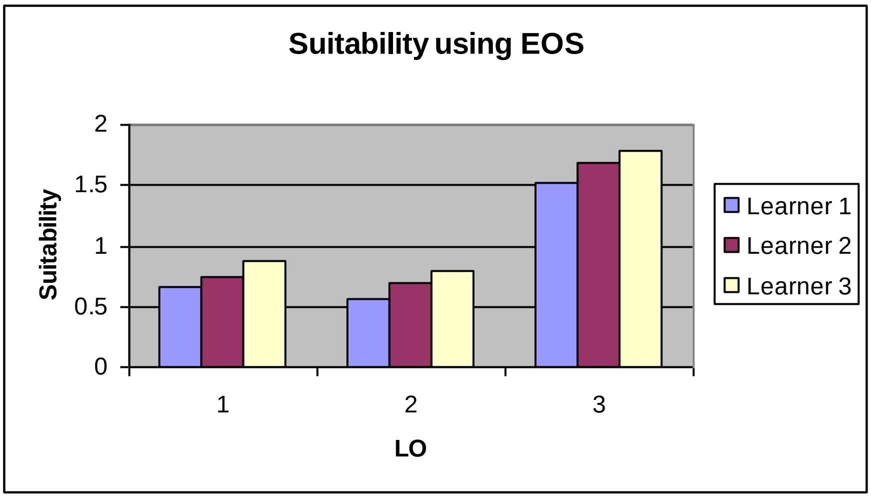


Figure 9: The suitability of three LO for three different learners

CHAPTER 4: ANALYSIS AND IMPLEMENTATION OF THE SHORTEST LEARNING PATH ALGORITHM

In e-Learning systems a course is modeled as a graph, in which each node represents a Knowledge Unit (KU), and two nodes in the graph are connected if the first node is a perquisite to the later node. An algorithm for selecting the shortest learning path is proposed in (Zhao and Wan, 2006). In the following sections, we will analyze the concepts that they depend on and the algorithm that they proposed.

4.1. Learning Path

In a graph that represents a course in e-Learning system, each node views a knowledge unit (KU) or a Learning object (LO) and each learner requests the target knowledge while accessing the system.

Learners have to navigate through the knowledge unit graph, and he/she may have a number of learning paths to reach the target knowledge unit (Zhao and Wan, 2006).

In (Zhao and Wan, 2006), the relationship between knowledge units in the graph is classified as: precedence, succession, and parallel relationship. Figure 10 shows these relationships according to (Zhao and Wan, 2006) classification.

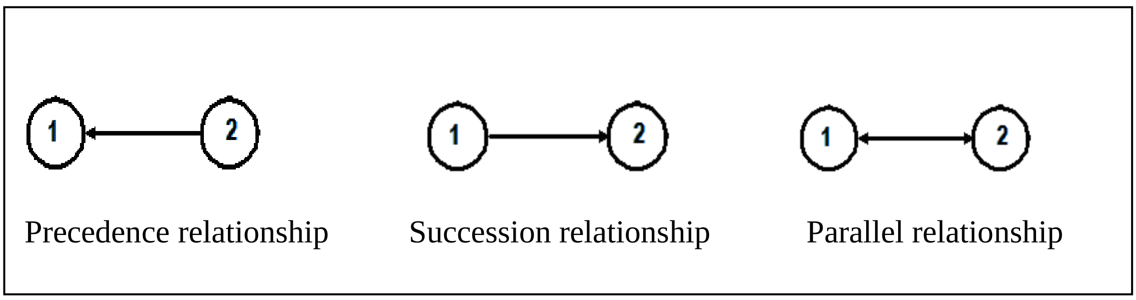


Figure10: Relationship between knowledge units

This classification does not represent the actual relation between course units in e-Learning system, since it must be based on the use of ontologies and course unit (learning object) metadata, where ontologies are specifications of the conceptualization and corresponding vocabulary used to describe a course.

4.3. Arrowhead Weight

The weighted directed graph that represents the course is a graph that has a weight, or numeric value, associated with each edge. Zhao and Wan (2006) assumed that the weight assigned initially by teachers as the difficulty to reach the next knowledge unit then assigned new values through a statistical analysis of previous learner's sessions.

According to Zhao and Wan (2006) the best learning path defined as the learning process that cost the least time and effort. Thus, an algorithm for selecting the shortest learning path is proposed.

4.3. The Shortest Learning Path Algorithm

The shortest learning path algorithm is based on a structure that consists of n nodes representing the course graph and three main steps (Zhao and Wan, 2006):

- Constructing an initial adjacency matrix ($D = p_{ij}$).
- Constructing a medial node matrix ($V = v_{ij}$), this changes according to conditions in a loop.
- Searching for the shortest path from node i to node j.

The proposed algorithm by Zhao and Wan (2006) for selecting the shortest learning path is given in Figure 11.

(1) Construct an initial adjacency matrix D=pij , pij is the arrowhead weight from node i to node j. If there is no arrowhead exists between the two nodes or the orientation of the arrowhead is reverse or i=j, pij=M.

(2) Construct a medial node matrix V=vij , the value of vij is the ID of node i.

(3) Start iterative operation (the times of iterative operation equal to the number of nodes). we first compare pij (i, j=2,3…, n)with pi1+p1j, there are three cases:

a) If pij> pi1+p1j, replace pi1+p1j with pij in the next adjacency matrix, and vij=1;

b) If pij<pi1+p1j, there is no change in the next adjacency matrix and node matrix;

c) If pij=pi1+p1j, there are two results, one is as a) and the other is as b)

So we get the next adjacency matrix D1 and node matrix V1, then begin the next iterative operation until we get Dn and Vn .

(4) Search for the shortest path from vi to vj,

a) If vij=i , the shortest path is vi to vj;

b) If vij ≠ i , we suppose vij=k, the shortest path is vi –vk- vj ,then

if vik=i, it means that there is no medial node between vi and vk; if vkj=k , it means that there is no medial node between vk and vj , else we can find a medial node k.

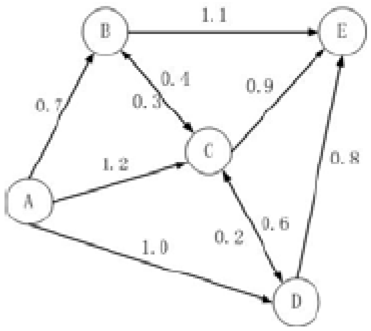
c) Repeat a) and b) until we find out all of the medial nodes.

d) Connect vi and vj with these medial nodes, we can get the concrete shortest learning paths (one path or more than one).

Figure11: The shortest learning path algorithm (Zhao and Wan, 2006).

According to the first step, it is mentioned that if there is no arrowhead between two nodes or the orientation of the arrowhead is reverse, then P_{ij} =M. but no definition for M was given. Thus, for the purpose of our implementation we consider M as infinity.

When we investigated the iterative operation in step2, we discarded some mistakes in declaring the replacement process between P_{ij} and $P_{i1} + P_{1j}$. Based on this replacement ($P_{i1+}P_{1j}$ with P_{ij}) and following the example that was introduced by Zhao and Wan (2006) and shown in Figure12, we discovered that P_{ij} will never changed.



Suppose the IDs of A,B,C, D, and E are:1,2,3,4 and 5. Then the initial matrices are:

$$D=\begin{pmatrix} M & 0.7 & 1.2 & 1.0 & M \\ M & M & 0.3 & M & 1.1 \\ M & 0.4 & M & 0.2 & 0.9 \\ M & M & 0.6 & M & 0.8 \\ M & M & M & M & M \end{pmatrix} \quad V=\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 & 4 \\ 5 & 5 & 5 & 5 & 5 \end{pmatrix}$$

After 5 times of iterative operation, the resulted adjacency matrix and medial node matrix will be as follows:

$$D_5=\begin{pmatrix} M & 0.7 & 1.0 & 1.0 & 1.8 \\ M & 0.7 & 0.3 & 0.5 & 1.1 \\ M & 0.4 & 0.7 & 0.2 & 0.9 \\ M & 1.0 & 0.6 & 0.8 & 0.8 \\ M & M & M & M & M \end{pmatrix} \quad V_5=\begin{pmatrix} 1 & 1 & 2 & 1 & 2 \\ 2 & 3 & 2 & 3 & 2 \\ 3 & 3 & 2 & 3 & 3 \\ 4 & 3 & 4 & 3 & 4 \\ 5 & 5 & 5 & 5 & 5 \end{pmatrix}$$

$$D_5=\begin{pmatrix} M & 0.7 & 1.0 & 1.0 & 1.8 \\ M & 0.7 & 0.3 & 0.5 & 1.1 \\ M & 0.4 & 0.7 & 0.2 & 0.9 \\ M & 1.0 & 0.6 & 0.8 & 0.8 \\ M & M & M & M & M \end{pmatrix} \quad V_5=\begin{pmatrix} 1 & 1 & 2 & 1 & 4 \\ 2 & 3 & 2 & 3 & 2 \\ 3 & 3 & 2 & 3 & 3 \\ 4 & 3 & 4 & 3 & 4 \\ 5 & 5 & 5 & 5 & 5 \end{pmatrix}$$

Figure 12: An example used to apply the algorithm of shortest learning path (Zhao and Wan, 2006).

Furthermore, there was inaccuracy in representing the iteration as shown in Figure 11 (i,j=2,3,…n) which means that the iterative process will not deal with the values P_{11}, P_{1j}, or P_{i1} but when we examine the example in Figure 12 we discovered that i=1…n but j=2..n. Also, in the inner steps of the loop, the author used ($P_{i1} + P_{1j}$) all the time. By testing the example, we recognized the fact that the true is to change the value (1) with a separated iteration (e.g. k, n,…etc) and not to be 1 all the time.

So, we have implemented this algorithm with an appropriate correctness as mentioned above. The algorithm was written using MATLAB 7.0, and experimented on the example given in (Zhao and Wan, 2006) and shown in Figure 12.

Finally, we found that this algorithm is a version of All-pairs shortest path (Floyd-Warshall algorithm). Also, this algorithm was implemented using MATLAB 7.0 and compared with the shortest learning path algorithm as in section 4.4.

4.4. The Floyd-Warshall Algorithm

Floyd-Warshall algorithm computes the values $d_{ij}^{(k)}$ in order of increasing values of k. Its input is an $n \times n$ matrix W representing the edge weights of an n-vertex directed graph $G = (V, E)$. That is, $W = (wij)$, where $wij = 0$ if $i=j$ and wij = the weight of the directed edge (i,j) if $i \neq j$ and $(i,j) \in E$ and $wij = \infty$ if $i \neq j$ and $(i,j) \notin E$. The bottom-up procedure in Figure 13.a can be used to implement the Floyd-Warshall algorithm, it returns the matrix $D^{(n)}$ of shortest-path weights. (Cormen *et al.*, 2001)

We need to compute not only the shortest-path weights but also a predecessor matrix $\Pi = (\pi_{ij})$, where π_{ij} is NIL if either $i = j$ or there is no path from i to j, and otherwise π_{ij} is the predecessor of j on some shortest path from i. This can be implemented by the procedure in Figure 13.b (Cormen *et al.*, 2001).

The incorporation of the $\Pi^{(k)}$ matrix computations into the Floyd-Warshall procedure is shown in Figure 14.

FLOYD-WARSHALL(W)
$n \leftarrow rows[W]$
$D^{(0)} \leftarrow W$
for $k \leftarrow 1$ to n
    do for $i \leftarrow 1$ to n
        do for $j \leftarrow 1$ to n
            do $d_{ij}^{(k)} \leftarrow \min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$
return $D^{(n)}$

a. The procedure of Floyd-Warshall algorithm (shortest-path weights)

PRINT-ALL-PAIRS-SHORTEST-PATH(Π, i, j)
if $i = j$
  then print i
  else if π_{ij} = NIL
    then print "no path from" i "to" j "exists"
    else PRINT-ALL-PAIRS-SHORTEST-PATH(Π, i, π_{ij})
  print j

b. The procedure of Print-All-Shortest-Path from i to j.

Figure13: Floyd-Warshall and print all shortest path algorithms (Cormen *et al.*, 2001).

FLOYD-WARSHALL(W)
$n \leftarrow rows[W]$
$D^{(0)} \leftarrow W$
$\Pi^{(0)}$ = NIL
for $k \leftarrow 1$ to n
    do for $i \leftarrow 1$ to n
        do for $j \leftarrow 1$ to n
            do if $(d_{ik}^{(k-1)} + d_{kj}^{(k-1)}) < d_{ij}^{(k-1)}$
                then
                    $d_{ij}^{(k)} \leftarrow \min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$
                    $\pi_{ij}^{(k)} = k$
return $D^{(n)}$

Figure 14: The Floyd-Warshall procedure with $\Pi^{(k)}$ matrix computation.

When we apply Floyd-Warshall algorithm to the graph in Figure 12, through 5 times of iterative operation, we get the following adjacency matrix D^5 and predecessor matrix Π^5, which is equal to D_5,V_5 respectively:

$$D^5 = D_5 = \begin{pmatrix} \infty & 0.7 & 1.0 & 1.0 & 1.8 \\ \infty & 0.7 & 0.3 & 0.5 & 1.1 \\ \infty & 0.4 & 0.7 & 0.2 & 0.9 \\ \infty & 1.0 & 0.6 & 0.8 & 0.8 \\ \infty & \infty & \infty & \infty & \infty \end{pmatrix} \qquad \Pi^5 = V_5 = \begin{pmatrix} 1 & 1 & 2 & 1 & 2 \\ 2 & 3 & 2 & 3 & 2 \\ 3 & 3 & 2 & 3 & 3 \\ 4 & 3 & 4 & 3 & 4 \\ 5 & 5 & 5 & 5 & 5 \end{pmatrix}$$

We can see that weight d_{15} =1.8 is the weight of the shortest path from node 1 to node 5. Then, we search for the medial nodes. We get π_{15}=2, that means that the medial node is 2 then we found π_{25} =5. Thus, the shortest path from node 1 to node 5 is 1-2-5. This means the result of Floyd-Warshall is the same as one of the two results that were obtained by the shortest learning path algorithm. Thus, the proposed algorithm is a version of Floyd-Warshall algorithm.

For further implementation, we have used corrected version of the shortest path algorithm. However, the above mentioned implementation of Floyd-Warshall algorithm was to ensure the correctness of the shortest learning path algorithm and its ability to be adapted in our suggested approach.

CHAPTER 5: ADAPTIVE SHORTEST PATH

5.1. Adaptive Navigation

Adaptive Content Selection is the first step to adaptive navigation, which is a goal of e-Learning systems. The instructional plan of an adaptive system can be considered as two interconnected spaces: the knowledge space and the media space (Brusilovsky *et al.*, 2003).

The knowledge space is a set of small domain knowledge elements. Each domain knowledge element represents an elementary fragment of knowledge for the given domain. Concepts of domain knowledge can be named in different ways, such as: concepts, knowledge items, topics, knowledge elements, but in all cases they denote elementary fragments of domain knowledge (Pythagoras and Demetrios, 2004). In addition, ontologies consist of definitions of concepts relevant for the domain, their relations, and axioms about these concepts and relationships.

The content space structuring can be based on the use of learning object metadata, where some attributes represents the relation between a learning object and another, and the type of the relationship.

5.2. Discovering Suitable Learning Path

The result of merging the knowledge space (ontology plane) and the media space(content space) is a directed acyclic graph (DAG) of learning objects inheriting relations from both spaces. This graph contains all possible navigation paths that a learner can follow to reach his learning goal (Pythagoras and Demetrios, 2004). Thus, there is a need to optimize such navigation paths as well as to select the path that is most suitable for the learner. In order to achieve this, we suggest the following approach:

1. Given a DAG that represents all possible navigation paths, a sub graph that is relevant to a learner is constructed.
2. The sub graph is augmented with weights that represent the suitability of learning objects for the learner.
3. A shortest path algorithm is then applied to select an adaptive path that is as suitable and as shortest as possible for the learner.

Figure 15 represents the proposed approach in e-Learning system, and a flowchart for the proposed approach is given in Figure 16.

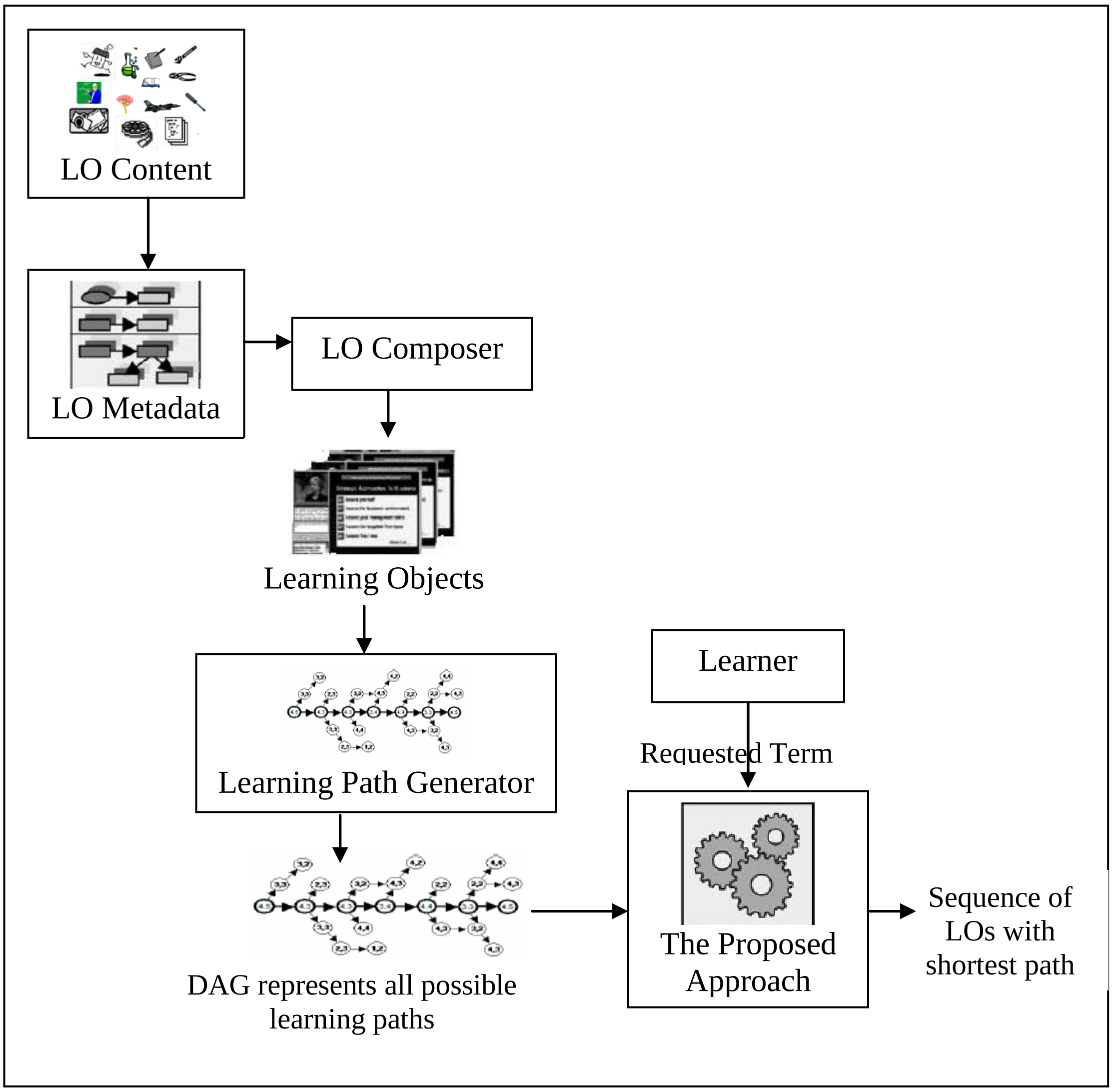


Figure 15: The proposed approach in e-Learning system.

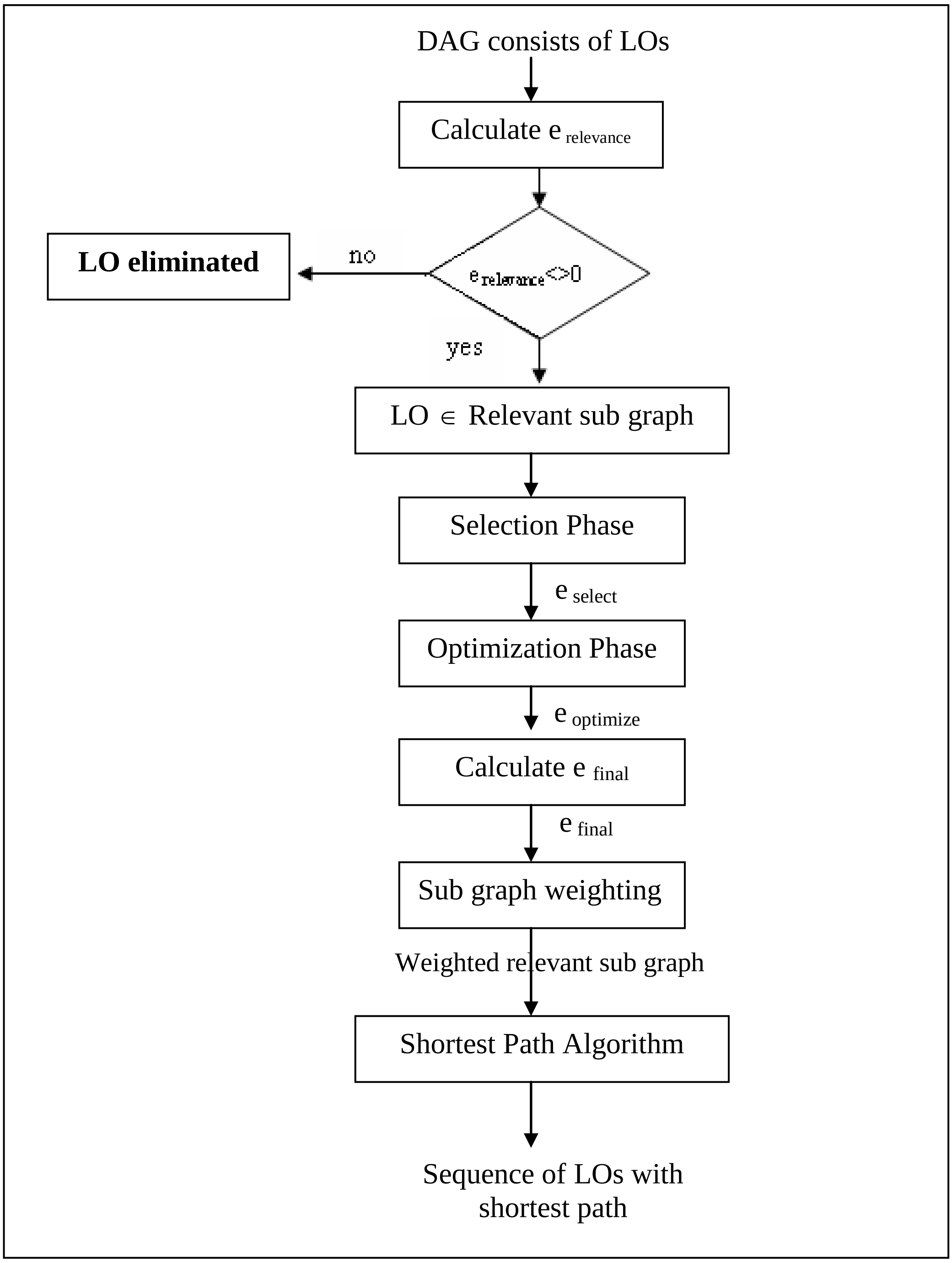


Figure 16: A flowchart for the proposed approach.

The implementation of our approach is based on:

- EOS approach to calculate the suitability of learning object (Liu and Greer, 2004).

- A shortest path algorithm on weighted graph suggested by Zhao and Wan (2006).

However, since our approach is based on constructing a sub graph that is relative to the learner, the EOS approach has to be modified to take this into consideration. This is because the initial construction of the DAG will affect the subsequent phases and improve the overall optimization and adaptation.

5.2.1. Modifications on EOS approach

Our modification to the EOS approach is based on introducing relevance calculation. Such relevance calculation is needed to obtain the relevant sub graph. Thus the first phase of EOS is divided into two sub phases:

- Relevance calculation for the requested concept or objective. As a result, the most relevant learning objects will be candidate for the next sub phase.
- Eliminating irrelevant learning object according to the eliminating attributes (the language, the cost, and the environment condition)

Such a modification requires a corpus for the concepts and objectives that presented in the domain ontology. This facilitates the representation of the requested objectives, or concepts as terms of keywords within a domain. For example, a specific concept in a specific domain, or an objective. Based on such terms a relevance value can be computed. For example, terms not frequent in the corpus have a low probability of being representative in the domain. Peñas *et al.* (2001) have defined a formula that gives such a relevance value for the requested terms and we are going to use this formula with some adaptation.

Within the framework of our approach, the following information structures are added:

- Two tables to represent corpus are needed; the first one consists of attributes that represent the concept and related information as shown in Table 7. While the other consists of the attributes that represent the concept objective corpus as shown in Table8.

Table 7: Concepts Domain Corpus attributes.

Attribute Name	Explanation
Concept ID	The identifier of the concept
Concept Name	Description of the concept
Domain	The domain in which the concept frequent
Frequency in Domain	Relative frequency of the concept in the specified domain

Table 8: Concept Objective Corpus attributes.

Attribute Name	Explanation
Concept ID	The identifier of the concept
Concept Name	Description of the concept
Objective	The objective in which the concept frequent
Frequency in Objective	Relative frequency of the concept in the specified objective.

- Some attributes are added to the learning object table, such as Main Domain, Objective, and the attribute specialized popularity is separated into three attributes, Beginners Specialized Popularity, Trainers Specialized Popularity, and Experts Specialized Popularity as shown in Table 9.

Table 9: New Learning object attributes.

Attribute Name	Explanation
Learning Object ID	An Identifier of the learning object
Language ID	The language in which the content is presented
Environment ID	The technical requirements needed for presenting the learning object
Current learner ID	Current learner using the leaning object
Pedagogical Objective	The concept presented in the learning object
Cost	The price of the learning object
Expected Reading Level	The reading capability required by the learning object.
Prerequisite	The knowledge needed by the learning object
Typical Learning Time	Time needed for working with the learning object
Presentation Type	The way of representing the content of the learning object
Objective	The objective of the learning object
Main Domain	The domain to which the concept of this learning object belongs.
General Popularity	How often the learning object is selected for all types of learners
Beginners Specialized Popularity	How often the learning object is selected for beginners
Trainers Specialized Popularity	How often the learning object is selected for trainers
Experts Specialized Popularity	How often the learning object is selected for experts

- A relationship table is constructed to represents the relations between learning objects in the DAG as shown in Table 10.

Table 10: Relationship attributes.

Attribute Name	Explanation
Learning Object ID	An Identifier of the learning object
Related Learning Object ID	An Identifier of the related learning object
Relation Type	The relationship type between the connected learning objects

5.2.2. Constructing Relevant Sub Graph

Based on the DAG that represents all possible navigation paths and the above mentioned modifications as well as the newly introduced information (Table 6, 7, 8, and 9), constructing the sub graph that is relevant to a learner proceeds as follows:

Firstly, a set of learning objects with relevance value denoted by $e_{relevane}$ for each learning object is constructed, where $0 \le e_{relevane} \le 1$. Then, the learning objects with zero value are eliminated. This can be formalized as follows:

Let $S = \{ LO_1, \ldots, LO_j \}$

$S' = Relevance(S)$

where: Relevance (S) is a function that constructs the sets $S_{relevance}$ and S', such that:

- $S_{relevance} = \{ e_{relevance\ 1}, \ldots, e_{relevance\ j} \}$
- $S' = \{ LO_j \in S \mid e_{relevane\ j} \neq 0 \}$

where: $e_{relevance\ j}$ is a value assigned for each $LO_j \in S$

- $e_{relevance\ j} \in \{0, a_{relevance}\}$

where $a_{relevance}$ is calculated by Equation (6).

$$a_{relevance}(c,dom,col) = 1 - \frac{1}{\log_2\left(\frac{2 + F_{c,dom} \times N}{F_{c,col}}\right)} \qquad \ldots\ldots\ldots\ldots\ldots\ldots \quad (6)$$

where:

$F_{c,dom}$: frequency of the requested concept in the specified domain or objective (dom)

$F_{c,col}$: frequency of the requested concept in the all collections .

N: the number of learning objects.

$e_{relevane}$ for a given LO is calculated by the function shown in Figure 17. This function is called by Relevance(S) for each LO $\in$ S.

Let a_1, a_2, ……, a_{10} be the following attributes respectively:

a_1: Requested Concept by the learner
a_2: Concept in Learning Object table.
a_3: Requested Objective or Domain by the learner
a_4: Objective or Domain in Learning Object table
a_5: Financial Situation in Learner table
$a_{6:}$ Cost in Learning Object table
$a_{7:}$ Environment ID in Learner table
a_8: Environment ID in Learning Object table
a_9: Language ID in Learner table
a_{10}: Language ID in Learning Object table

Let a_{11} be the frequency of the requested concept in the specified domain or objective.
Let a_{12} be the frequency of the requested concept in all collection.
Let a_{13} be the number of learning objects in the system.

Five eliminating criteria are computed as follows:

$a_{\text{eliminate1}} = (a_1 = a_2)$
$a_{\text{eliminate2}} = (a_3 = a_4)$
$a_{\text{eliminate3}} = (a_5 \geq a_6)$
$a_{\text{eliminate4}} = (a_7 = a_8)$
$a_{\text{eliminate5}} = (a_9 = a_{10})$

Let $a_{\text{relevance}}$ be a relevance value of the requested term calculated as:
$a_{\text{relevance}} = 1- (1 / \log_2 ((2 + (a_{11} \times a_{13})) / a_{12}))$

If $a_{\text{eliminate1}} \cap a_{\text{eliminate2}} \cap a_{\text{eliminate3}} \cap a_{\text{eliminate4}} \cap a_{\text{eliminate5}}$ then
    return $a_{\text{relevance}}$
Else
    return 0

Figure 17: A function to calculate e_{relevane} .

5.2.3 Sub Graph Weighting

DAG weighting is need to find the shortest path by any shortest path algorithm. Hence, the result of applying the shortest path algorithm is the learning path that covers the desired concepts objects, and reaches the learning goal by providing all information about cognitive characteristics and preferences for the learner. Such a weighting for the DAG is calculated by Equation (7).

$$W = 1- e_{\text{final j}} \qquad \ldots\ldots\ldots\ldots\ldots\ldots (7)$$

e_{finalj} is calculated by a suitability function as shown in Figure 18, where:

- Select (S') is a function that assigns a value e_{select} -considering selecting attributes- for each LO in S', where e_{select} is calculated by Equation (3).
- Optimize (S') is a function to assign a value $e_{optimize}$ for each learning object in S', where $e_{optimize}$ is calculated by Equation (5).
- Suitability(S') is a function to assign e_{final} for each LO in S', where e_{final} is the final evaluation result of the learning object.

S_{select} = Select (S')
where: Select (S') is a function that constructs the set S_{select}, such that:
- $S_{select} = \{ e_{select\ 1}, \ldots, e_{select\ j} \}$
- e_{select} is a value assigned for each $LO_j \in S'$ and calculated by the Equation :

$$e_{select\ j} = \sum_i W_i \times a_{select\ i} \quad ; \; W, a_{select} \in [0,1]$$

- W_i is calculated by Equation (4)

$S_{optimize}$ = Optimize(S')
where: Optimize(S') constructs the set $S_{optimize}$, sucn that:
- $S_{optimize} = \{ e_{optimize\ 1}, \ldots, e_{optimize\ j} \}$
- $e_{optimize}$ is a value assigned for each $LO_j \in S'$ as:

$$e_{optimize\ j} = \sum_i W_i \times a_{optimize\ i} \quad ; \; W, \; a_{optimizei} \in [0,1]$$

W_i integer values to be given

$S_{suitability}$ = Suitability(S')
where: Suitability (S') constructs the set $S_{suitability}$, such that:
- $S_{suitability} = \{ e_{final\ 1}, \ldots, e_{final\ j} \}$
- $e_{final\ j}$ is a value assigned for each $LO_j \in S'$ as:

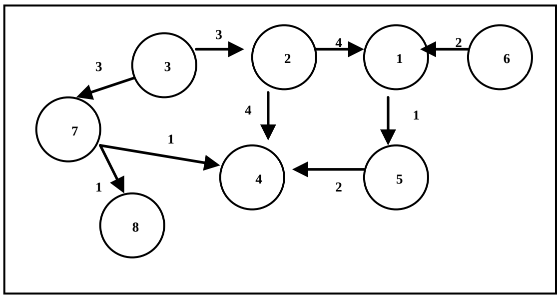
$$e_{final\ j} = e_{relevane\ j} \times (e_{select\ j} + e_{optimize\ j})$$

Figure 18: A function to calculate the suitability of a learning object

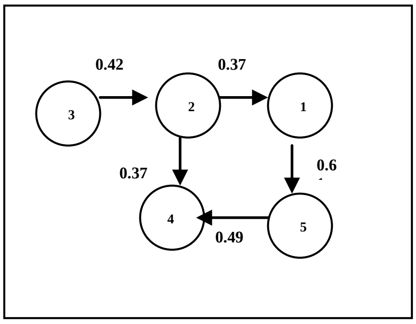
5.2.4. Selecting Adaptive Path Using Shortest Path Algorithm

Based on the previous formalization, and calculation of e_{final}, as well as the fact that the learning object that has the highest e_{final} value is the most suitable learning object for a learner. The weights of the learning objects that are represented in the sub graph are calculated in away that is inversely proportional to their suitability value. Hence, the lower the weight they have the more suitable they are.

For example, Figure 19.a shows a DAG that represents all possible navigation paths between the set of learning objects S= {LO_1,…, LO_8} corresponding to a specific concept. The numbers on the arrowheads represents the relationship between two connected nodes.



a. DAG representing a concept



b. Weighted sub graph

$$\begin{bmatrix} \infty & \infty & \infty & \infty & 0.61 \\ 0.37 & \infty & \infty & 0.37 & \infty \\ \infty & 0.42 & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 0.49 & \infty \end{bmatrix}$$

c. Initial matrix of weights

Figure 19: An example for constructing a weighted sub graph

Applying the function Relevance(S) will produce the set S'= {LO_1,…,LO_5}with $e_{relevane} \neq 0$. Applying the suitability function Suitability (S') and weight calculation as in Equation (7) will produce the weighted sub graph as shown in Figure 19.b. Imposing the weight matrix for this sub graph we get the matrix shown in Figure 19.c. We can see that the weights in the sub graph make the lower value the node has the more suitable the learning object is, which means a sub graph with adaptive paths. In order to select the shortest adaptive learning path, we will apply a shortest path algorithm to the resulted weighted sub graph. This can be achieved by the algorithm that was discussed in Chapter 4.

5.3. Implementation Assumptions for the Proposed Approach

Based on the previous analysis and formalization, our proposed approach was implemented using Visual Basic.Net, and based on the following assumptions:

- Three Types for learners: Beginner, Trainer, and Expert.
- Languages: English, Arabic, French, etc.
- Three learning styles: Visual, Auditory, and Tactile & kinesthetic (Learn by doing).
- Specialized popularity retrieved according to learner type.
- The weight for general popularity considered as: 0.5, while for previous learners evaluation it was 0.25, and finally 0.25 for specialized popularity.
- Nine presentation styles: Text, exercise, table, diagram, simulation, audio, slide, problem statement, and video.
- Minimum requirements for the environment (hardware,Software): P3 CPU 1300 MHz/ 128 Ram /16 VGA/Win98, P4 CPU 200 MHz / 264 RAm /32 VGA/ WinXP , P4 CPU 2300MHz / 512 RAM / 64 VGA/ WinXP, Centrino PM CPU 1.3 /1G Ram / 128 VGA / Win98, BM CPU 1.6/ 2G Ram /128 VGA/ Mac,…etc.
- Four values: weak, good, very good, and excellent were considered for Reading Level, Listening Level, and General Academic Achievement.
- An hour was the time measure unit.

- For the purpose of comparison process, numerical values were given for learning styles, presentation styles, learners types, reading level, listening level, general academic achievement, and environment.
- Concepts, domains, and objectives were extracted from the ACM Computing Curricula 2001 for Computer Science (ACM, 2001), which defines 950 topics organized in 132 units and 14 areas.
- Six Objectives were considered (ACM, 2001): Knowledge and Understanding, Design and Implementation, Modeling, Method and Tools, Information Management, and Critical Evaluation and Testing.
- Four relationships between learning objects were considered as in (Pythagoras and Demetrios, 2004): is part of / has part, references / is referenced by, is based on / is basis for , requires / is required by.

5.4. Implementation Example

In this section, we will demonstrate the proposed approach by considering 19 learning objects that represent the concept "Algorithmic Computation" as shown in Figure 20. The characteristics of these learning objects are given in Table 11.

Each number inside a node in the graph represents the Learning Object ID. While each number on the arrowhead represents the relationship between the connected learning objects, since we have four relationships thus the numbers on the arrowhead are given as follows:

(1) if the relation between the connected learning objects “is part of” / “has part”.

(2) if the relation between the connected learning objects “references”/“is referenced by”

(3) if the relation between the connected learning objects "is based on" / "is basis for"

(4) if the relation between the connected learning objects "requires" / "is required by"

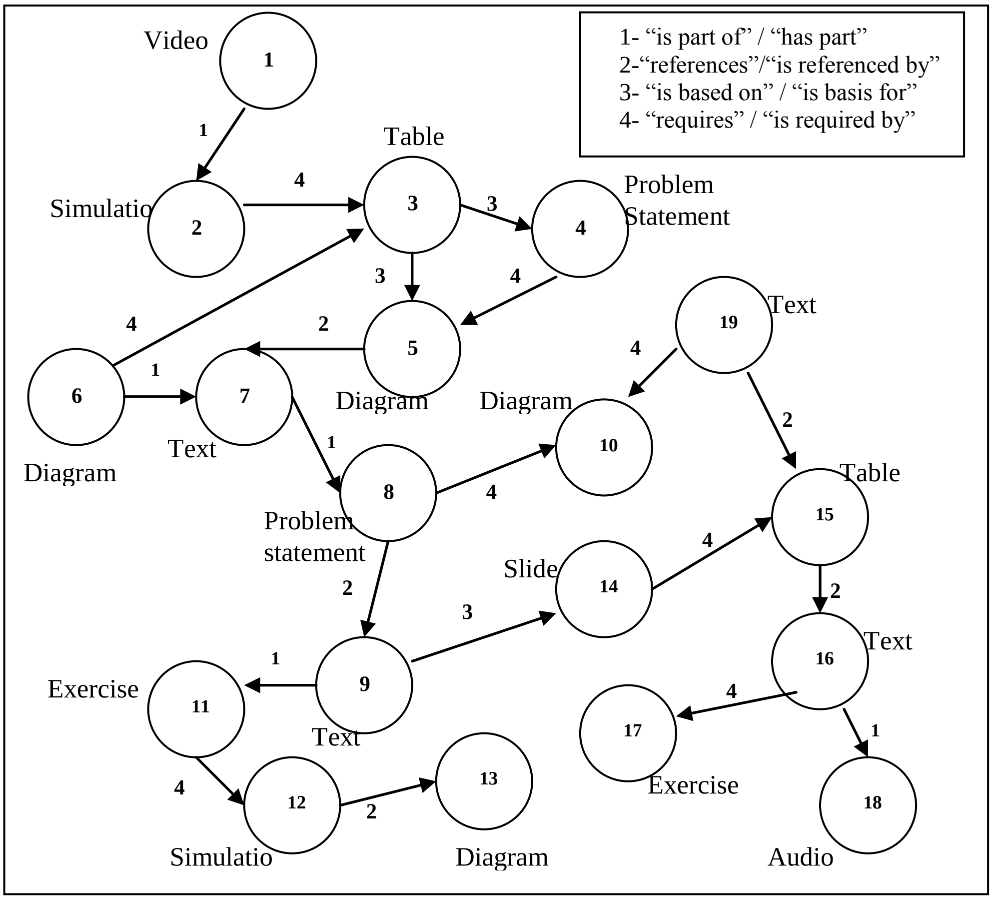


Figure20: A graph representing learning objects for a concept.

We implement the proposed approach for a learner with the following characteristics: a beginner with a good reading level, 5 hours for learning, by doing learning style, and his preferred presentation type is slides.

Table 11: Learning Objects characteristics

LO ID	language	Presentation Style	Time	Cost	Required language Level	Expected Reading Level	Experts Popularity %	Trainers Popularity %	Beginners Popularity %	Environment	General Popularity %
1	English	Video	2	3	Very good	Very good	12	10	5	P3/128Ram/16VGA/Win98	35
2	English	Simulation	3	1	Very good	Very good	10	5	3	P3/128Ram/16VGA/Win98	18
3	English	Table	3	4	Very good	Excellent	13	8	4	P3/128Ram/16VGA/Win98	25
4	English	Problem Statement	3	12	Very good	Very good	12	5	14	P3/128Ram/16VGA/Win98	48
5	English	Diagram	1	1	Good	Very good	10	12	12	P3/128Ram/16VGA/Win98	48
6	English	Diagram	1	1	Good	Very good	15	12	5	P3/128Ram/16VGA/Win98	35
7	English	Text	2	3	Good	Very good	10	21	22	P3/128Ram/16VGA/Win98	50
8	Arabic	Problem Statement	4	5	Good	Good	11	11	13	P3/128Ram/16VGA/Win98	33
9	Arabic	Text	2	1	Good	Very good	30	14	10	P3/128Ram/16VGA/Win98	54
10	Arabic	Diagram	1	15	Good	Good	12	12	12	P3/128Ram/16VGA/Win98	36
11	French	Exercise	2	20	Very good	Very good	9	4	5	P3/128Ram/16VGA/Win98	18
12	French	Simulation	4	16	Excellent	Very good	14	10	8	P4/264RAm/32VGA/WinXP	28
13	French	Diagram	1	2	Good	Excellent	6	15	10	P3/128Ram/16VGA/Win98	10
14	Arabic	Slide	1	3	Very good	Very good	10	20	12	P3/128Ram/16VGA/Win98	32
15	English	Table	2	16	Very good	Very good	15	15	15	P3/128Ram/16VGA/Win98	45
16	English	Text	2	4	Very good	Good	10	5	3	BM/ 2GRam/128VGA/Mac	18
17	English	Exercise	1	5	Excellent	Excellent	12	10	5	P4/512RAM/64VGA/WinXP	27
18	English	Audio	2	10	Excellent	Very good	15	15	5	P4/264RAm/32VGA/WinXP	30
19	English	Text	2	3	Excellent	Excellent	10	20	15	P4/512RAM/64VGA/WinXP	45

The results of applying our approach are as follows:

- A sub graph of seven learning objects that are the most relevant to the learner and as shown in Figure 21. The
- The calculated suitability of these learning objects and their weights as shown in Table 12.

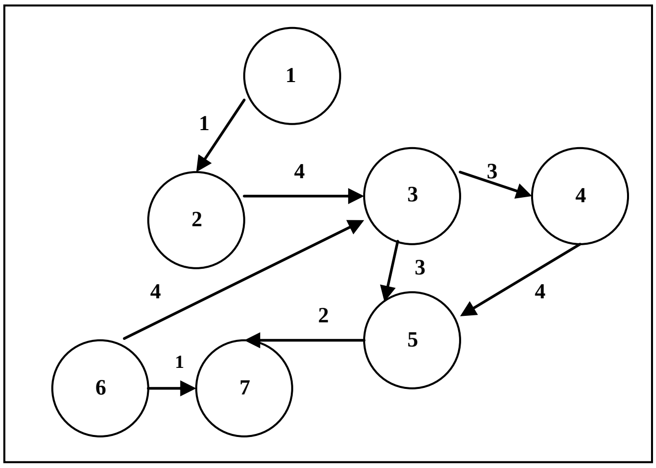


Figure 21: A sub graph obtained after applying the proposed approach.

Table 12: Suitability and weights of learning objects:

LO	Suitability	Weight
1	**0.46**	**0.54**
2	**0.63**	**0.37**
3	**0.58**	**0.42**
4	**0.58**	**0.42**
5	**0.51**	**0.49**
6	**0.43**	**0.57**
7	**0.53**	**0.47**

- The initial adjacency matrix D_1 and the medial node matrix V_1 are:

$$D_1=\begin{pmatrix} M & 0.54 & M & M & M & M & M \\ M & M & 0.37 & M & M & M & M \\ M & M & M & 0.42 & 0.42 & M & M \\ M & M & M & M & 0.42 & M & M \\ M & M & M & M & M & M & 0.49 \\ M & M & 0.57 & M & M & M & 0.57 \\ M & M & M & M & M & M & M \end{pmatrix} \quad V_1=\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 & 4 & 4 & 4 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 6 & 6 & 6 & 6 & 6 & 6 & 6 \\ 7 & 7 & 7 & 7 & 7 & 7 & 7 \end{pmatrix}$$

The shortest adaptive path is represented by the following matrices:

$$D_7 = \begin{pmatrix} M & 0.54 & 0.91 & 1.33 & 1.33 & M & 1.82 \\ M & M & 0.37 & 0.79 & 0.79 & M & 1.28 \\ M & M & M & 0.42 & 0.42 & M & 0.91 \\ M & M & M & M & 0.42 & M & 0.91 \\ M & M & M & M & M & M & 0.49 \\ M & M & 0.57 & 0.99 & 0.99 & M & 0.57 \\ M & M & M & M & M & M & M \end{pmatrix} \quad V_7 = \begin{pmatrix} 1 & 1 & 2 & 3 & 3 & 1 & 5 \\ 2 & 2 & 2 & 3 & 3 & 2 & 5 \\ 3 & 3 & 3 & 3 & 3 & 3 & 5 \\ 4 & 4 & 4 & 4 & 4 & 4 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 6 & 6 & 6 & 3 & 3 & 6 & 6 \\ 7 & 7 & 7 & 7 & 7 & 7 & 7 \end{pmatrix}$$

We can see that the weight d_{27} =1.28 is the weight of the shortest path from node 2 to node 7 then we search for the medial nodes, we get v_{27}= 5, It means that the medial node is 5 then we found v_{25} =3, there is another intermediate node between 2 and 5. Thus, the shortest path from node 2 to node 7 is 2-3-5-7.

The following steps show the calculation of LO1 suitability for the specified learner in the previous example.

Step 1: $e_{relevance}$ calculation

LO1

Language : English
Cost : 3
Environment: P3/128Ram/16VGA/Win98
Concept: Algorithmic Computation
Domain: Algorithms

Learner

Preferred Language : English
Financial situation: 11
Environment: P3/128Ram/16VGA/Win98
Requested Concept: Algorithmic Computation
Requested Domain: Algorithms

Filtering Phase

$e_{relevance}$= 0.862

Step 2: Weights calculation for Selecting Criteria

Learner Style	Preferences	W
	Reading Level	3/3
By doing	Time	2/3
	Presentation type	1/3

Step 3: The degree of match (a_{select})

LO1 Learner

$a_{select} = 0$

$a_{select} = 2/5$

$a_{select} = 0$

Time	Expected Reading Level	Presentation Type
2	Very good = 2	Video = 9

Time	Expected Reading Level	Presentation Type
5	Good = 1	Slides = 7

$e_{select} = 0.268$

Step 4: $e_{optimize}$ calculation

W	$a_{optimize}$
0.5	General Popularity = 35 → $G_p = 0.35$
0.25	Specialized Popularity (for Beginners) = 5 → S_P =0.05
0.25	From LO History similar learners evaluation L_{pev} = 11 → $Av(L_{pev})$ =0.314

$e_{optimize} = 0.266$

Step 5: e_{final} calculation for LO1

$e_{final} = 0.862 \times (0.268 + 0.266) = 0.64$

5.5. Experimental Results

Within the framework of this research, we have conducted several experiments as follows:

- Implementing EOS as discussed in Section 3.3
- Implementing shortest learning path as discussed in Section 4.3.
- Implementing the proposed approach as discussed in Section 5.3

Further experiments were conducted for testing and comparing EOS, and the proposed approach based on a number of created instances of learning object metadata, a number of learners, and simulated usage history of the learning objects. The first experiment was conducted based on different learning objects that represent a concept that may appear in one domain or many domains. The results of applying EOS and the proposed approach are given in Figure 22. The obtained results show that the number of selected learning objects using the proposed approach is less than the number of selected learning objects using EOS. Also the number of selected learning objects using objective is not always greater than the number of the learning objects selected using main domain. This is because when a concept appears in one domain the objective will have less representative learning objects.

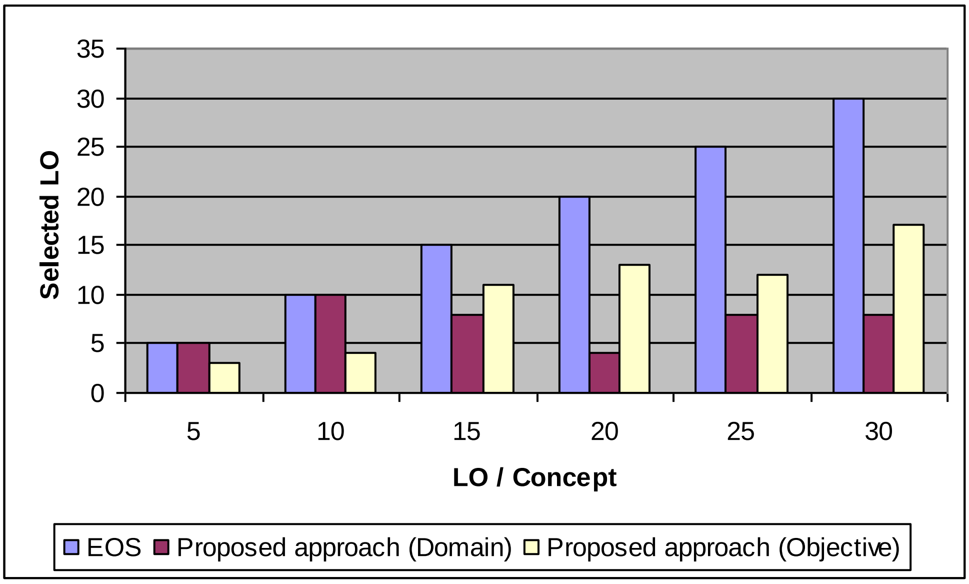


Figure 22: Selection results when a concept appears in one domain or more.

The second experiment was conducted based on concepts that appear in more than one domain and has more than one objective. The results are given in Figure 23. These results show that the number of selected learning objects using objective always greater than the number of selected learning objects using a domain. This is because when a concept appears in more than one domain, each time it has the same objective but in different domains. Thus, when the selection depends on objective all learning objects that represents the specified objective for the requested concept will be retrieved, but in different domains.

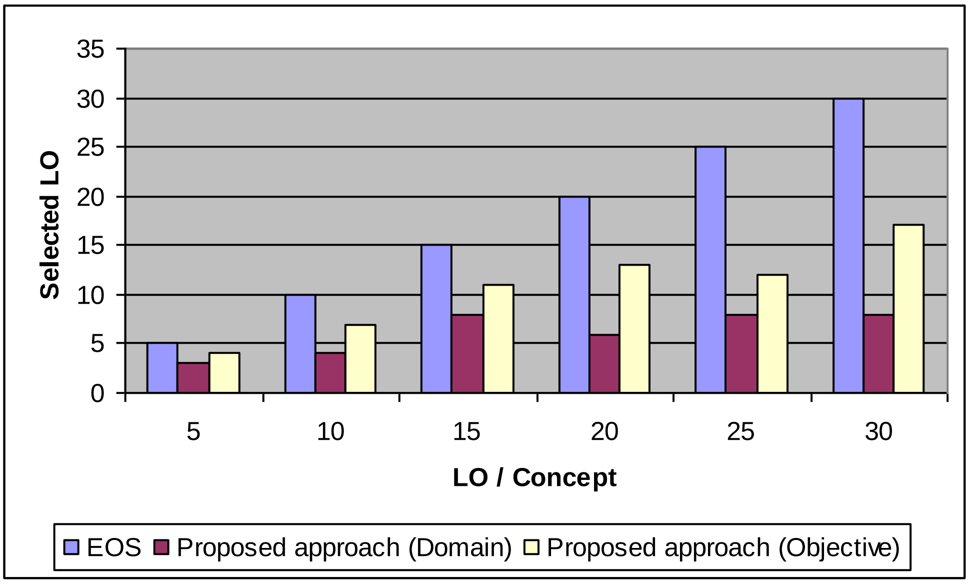


Figure 23: Selection results when concept appears in more than one domain.

The third experiment was conducted by applying the proposed approach to the same learners and learning objects that are used in the implementation of EOS as discussed in section 3.3. The results show that the suitability variation using the proposed approach is more than in EOS approach, as shown in Figure 24. This is because within the framework of the proposed approach the relevance calculation of a concept is added to the calculation of the suitability.

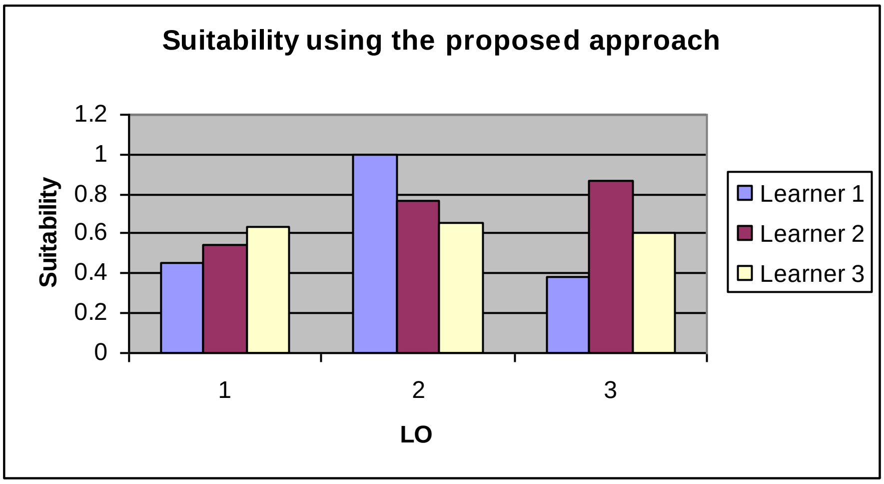


Figure 24: Suitability using the proposed approach.

To evaluate the overall performance of the proposed approach, its selection results were matched with the selection results that obtained by assumed experts. The selection performed by both was on the same simulated data set. This set includes a number of created instances of learning object metadata, a number of learners, and simulated usage history of the learning objects. The matching is based on the formula that was proposed by Karampiperis and Sampson (2005). This formula express the matching success.

$$\text{Matching success (\%)} = 100 * \left[\frac{\text{Correct learning object selected}}{m} \right] \quad \ldots\ldots\ldots\ldots \text{(9)}$$

Where m is the number of requested learning objects from the media space per concept node.

Thus, the evaluation is based on matching the sequence of learning objects selected by the proposed approach and the corresponding sequence selected by three experts with different points of view for preferences. The preferences and the characteristics of the three experts are shown in Table 13. Figure 25 shows the success of such matching, while Figure 26 shows the average matching success.

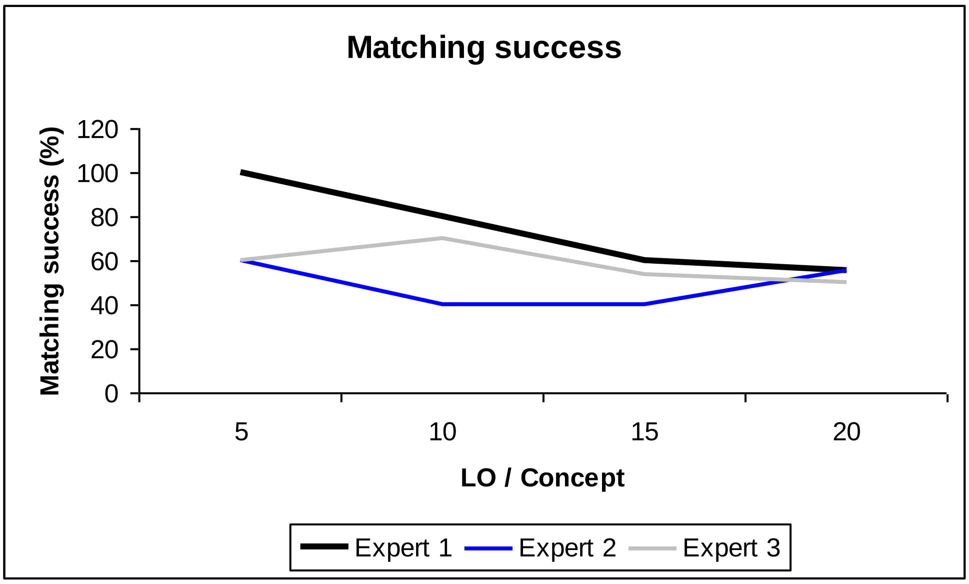


Figure 25: Matching success using the proposed approach

Table 13: The characteristics and preferences of the three experts

Expert	Expert Characteristics	Preferences
Expert 1	-Specialization: Software Engineering -Experience: 3 years -Certification: Microsoft Certified Professional, IT Cambridge. - Master Degree in Computer Science. - Learning Courses: C#, VB.Net.	1. Time 2. Reading Level 3. Presentation Type 4. Learner Type 5. Learning Style
Expert 2	Specialization: English Experience: 3 years Certification: Master Degree , Toefl. Learning Courses: English	1. Learner Type 2. Presentation Type 3. Reading Level 4. Time 5. Learning Style
Expert 3	Specialization: Computer Science Experience: 2 years Certification: IT Cambridge, ICDL. Learning Courses: ICDL, Programming Langues like C++.	1. Learning Style 2. Presentation Type 3. Learner type 4. Reading Level 5. Time

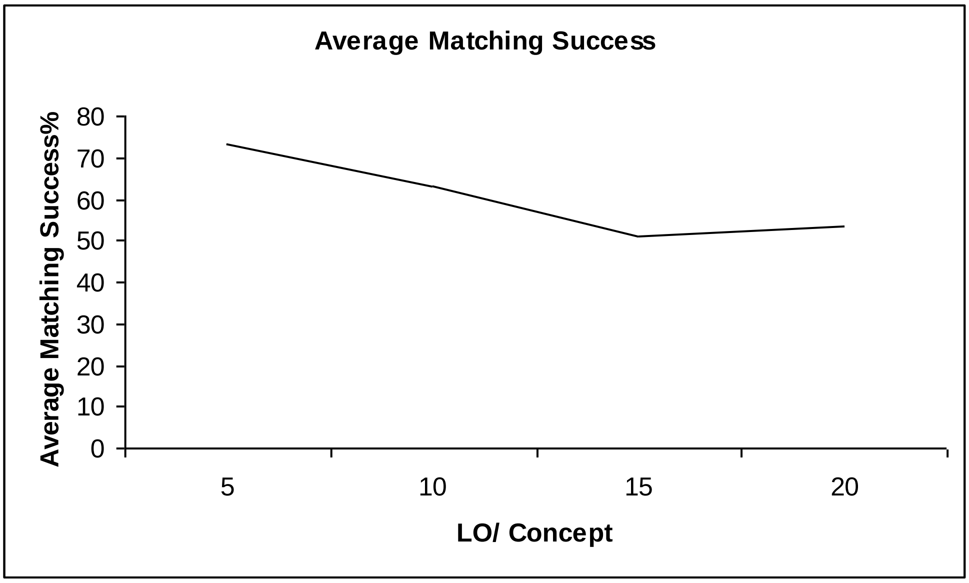


Figure 26: Average matching success using the proposed approach

Both Figures show that this matching is affected by the number of the desired learning objects (m). The average matching is 73%, 63%, 51%, and 53% for 5 LO/Concept, 10, 15, and 20 respectively. Hence, the smaller the number of learning objects is the higher the matching is. However, the selection results of the proposed approach are competitive to the results obtained by the three experts despite the variations in their point of views. Moreover, representing a concept by a small number of learning objects is more efficient than representing it by a large number.

CHAPTER 6: CONCLUSION AND FUTURE WORK

In e-Learning courses, learners may have more diverse backgrounds than those in traditional courses. Thus, selecting a learning path that is suitable for individual learner is recognized as an interesting research area in e- learning systems.

This research aims at improving the ability of selecting appropriate learning objects for a specific learner, as well as to select the shortest learning path for that learner. In order to achieve this, we select two representative algorithms; Eliminating and Optimized Selection and the shortest learning path algorithm in order to obtain the benefits of both.

Based on the DAG that represents all possible navigation paths between learning objects in an e-Learning system, the first step of our approach is to construct a sub graph that is relevant to a learner. The second step is to augment the sub graph with weights that represent the suitability of learning objects for the learner. The third step is to apply a shortest path algorithm to select an adaptive path that is as suitable and as shortest as possible for the learner.

The augmented weights represent the suitability of learning objects. In order to calculate the suitability of a learning object, we have added some modifications to the EOS approach by a proposed framework that contains a suggestion on extending the learning object metadata specifications and selecting a short list of appropriate and relevant learning objects for the learner and the learning context. This selection is based on terms that represent objectives and concepts within a domain or more than one domain. This constitutes an improvements on EOS approach. This is because we have used an ontology based representation for LOs. This representation serves the learning objects selection and comparison much better. Furthermore, the use of such terms instead of keywords and full description is also a better approach. This motivated by the fact that the description is difficult to used for automatic learning objects comparison.

Our experiment showed that the improvement on EOS approach gives more specific and more optimized selection of learning objects that are suitable for the learner.

In addition, we have compared the produced LOs sequences selected by our proposed approach with that selected by different experts. Experiment results showed that the success in learning objects sequencing is affected by the number of learning objects that represents the desired concept and our approach is competitive with the results obtained by these experts. Finally, we have seen that the initial DAG construction affects the subsequent phases and improves the overall performance and adaptation.

Future work

Based on the presentation style (i.e. Text, exercise, table, diagram, simulation, audio, slide, problem statement, and video) of learning objects that represents a specific concept, we define manually the relationships (“is part of” / “has part”, “references”/ “is referenced by”, “is based on” / “is basis for”, “requires” / “is required by") between representative learning objects in the DAG. A suggestion or future work is to automate such a process by automatic generation of a DAG using structured description of the metadata such XML. Further, we have seen that the choice of the filtering algorithms have significant impact on the overall performance. Hence, more research is needed to enhance such a choice.

REFERENCES

(ACM 2001): ACM Computing Curricula 2001 for Computer Science, retrieved June 10, 2007, from: **www.intag.org/pages/WP/ACM_2001_Curricula.doc**.

(Afaneh *et al*., 2006): Afaneh M., Basile V., and Bennett J., (2006), "**e-Learning Concepts and Techniques**", ebook retrieved May 30, 2007, from: **http://iit.bloomu.edu/Spring2006_eBook_files/chapter1.htm#h1_3**.

(Atif *et al*., 2003): Atif Y. Benlamri, R. and Berri J., (2003), "Learning Objects Based Framework for Self-Adaptive Learning", **Education and Information Technologies, IFIP Journal**, Kluwer Academic Publishers, 8:4, pp 345-368.

(Andreev and Troyanova, 2006): Andreev R. D., and Troyanova N. V., (2006), "e-Learning Design: An Integrated Agent-Grid Service Architecture", **IEEE John Vicent Atanasoff 2006 International Symposium on Modern Copmputing** (JVA'06). pp. 208-213. IEEE Computer Society.

(Aroyo, 2006): Aroyo, L., Dolog, P., Houben, G-J., Kravcik, M., Naeve, A., Nilsson, M. and Wild, F., (2006), "Interoperability in Personalized Adaptive Learning", **Educational Technology & Society**, 9 (2), pp. 4-18.

(Berg, 2007): Berg R. "Instructional desing at ICS" retrieved June 3, 2007 from: **http://www.uwex.edu/ics/design/glossary.htm**.

(Benlamri *et al*., 2003): Benlamri R., Atif Y., and Berri J., (2003), "Dynamic Learning Modeler", **International Journal of Educational Technology and Society, IEEE & IFETS**, 6(4), pp. 60-72.

(Biggs, 2007): Biggs, N. "Graph Theory" retrieved at 16-June-2007 from: **http://en.wikipedia.org/wiki/Graph_theory**.

(Black and Tanenbaum, 2007): Black P. E., and Tanenbaum P. J., "Graph" retrieved June 16, 2007 from: **http://www.nist.gov/dads/HTML/graph.html**..

(Brusilovsky, 2003): Brusilovsky, P., (2003), "Developing adaptive educational hypermedia systems: From design models to authoring tools". **Murray, T., Blessing, S., Ainsworth, S. (eds.): Authoring Tools for Advanced Technology Learning Environment.** Dordrecht: Kluwer Academic Publishers 2003, pp.377-409.

(Brusilovsky and Vassileva, 2003): Brusilovsky P., and Vassileva J., (2003), "Course sequencing techniques for large-scale web-based education", **International Journal of Continuing Engineering Education and Life-long Learning**, Vol. 13. pp.75-94.

(Burgos and Specht, 2006): Burgos D., and Specht M., (2006), "Adaptive e-Learning Methods and IMS Learning Design: An integrated approach", **Proceedings of the Sixth International Conference on Advanced Learning Technologies (ICALT'06)**, pp. 1192-1193. IEEE Computer Society.

(Carchiolo *et al.*, 2002): Carchiolo V., Longheu A., and Malgeri M., (2002), "Adaptive Formative Paths in a Web-based Learning Environment", **Educational Technology & Society** 5 (4), 2002.

(Codone, 2001): Codone S., (2001), "**An e-Learning Primer**", Raytheon Interactive, Florida, November 2001.

(Cross and Berkeley, 2004): Cross J., and Berkeley, (2004), "a History of eLearning", pre-publication version. Retrieved May 31, 2007, from: **http://www.internettime.com/Learning/articles/OTH.doc**.

(Doe, 2007): Doe J., "Brief introduction to learning objects in DLNET", retrieved at June 6, 2007, from: **http://www.dlnet.vt.edu/Resources/reports/ARI_LO_Def.pdf**.

(Friel, 2007): Friel M., “e-Learning History”, retrieved June 2, 2007 from: **http://www.newman.ac.uk/Students_Websites/~m.m.friel/hist.htm**.

(Fournier *et al.*, 2006): Fournier H., Dragne C. and Romila D., (2006), "State of the field report E-learning". **National Research Council Canada, Institute for Information Technology**.

(Godby ,2007): Jean Godby, "Metadata standards" retrieved at June 6, 2007 from: **http://www.oclc.org/research/projects/mswitch/1_crosswalks.htm**.

(Hall, 1997). Hall, B. (1997). "**Web-Based Training Cookbook: everything you need to know about online training**". John Wiley & Sons.

(Honey and Mumford, 1992): Honey P., and Mumford A. "**The manual of learning Styles**", Maidenhead: Peter Honey. (1992)

(Horton, 2000): Horton W., “Designing Web-based Training”, (2000) retrieved at May 30, 2007, from: **http://www.ips-inc.com/default.asp?groupname=iLearn&pid=34**.

(IEEE LOM, 2002): Learning Technology Standards Committee of the IEEE, (2002), "Draft Standard for Learning Object Metadata", retrieved April 28, 2007, from **http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf**

(Karampiperis and Sampson, 2004): Karampiperis P., Sampson D., (2004), "Adaptive Learning Object Selection in intelligent learning systems", **Journal of Interactive Learning Research**, vol. 15(4), pp. 389-407. November 2004.

(Karampiperis and Sampson, 2005): Karampiperis P., and Sampson D. (2005), "Adaptive Learning Resources Sequencing in Educational Hypermedia Systems", **IFETS: International Forum of Educational Technology and Society**, pp 128-147.

(Kreuz and Roller, 2001): Kreuz I., and Roller D., (2001), "Relevant Knowledge First, Reinforcement Learning and Forgetting in Knowledge Based Configuration", paper ID: cs.AI/0109034, arXiv 19-Sep-2001.

(Liu and Greer, 2004): Liu J., and Greer J., (2004), "Individualized Selection of Learning Object", **Workshop on Applications of Semantic Web Technologies for e-Learning (SW-EL@ITS’04)**, 30 August-03 September, 2004, Maceió, Brazil.

(LOAZ, 2007): "Learning Object Characteristics" retrieved at June 12, 2007, from: **http://www.loaz.com/learning-objects/learning-object-characteristics.html**.

(Martinez, 2007): Martinez M., "Adaptive learning", retrieved June 15, 2007, from: **http://www.trainingplace.com/source/research/adaptivelearning.htm**.

(Masie, 2004): Masie E., (2004), "701 e-Learning Tips " , a free digital book by The MASIE Center, retrieved May 26, from: **http://www.masie.com/701tips/book/701_e-Learning_Tips.pdf.**

(Muñoz, 2004): Muñoz L. S., (2004), "**Ontology-based Metadata for e-learning Content**", Master’s Thesis. UFRGS, March 10, 2004, Porto Alegre, Brasil..
(Najjar, 1996): Najjar, L. J.,"Multimedia Information and Learning". (1996)._**Journal of Educational Multimedia and Hypermedia**, V. 5, pp. 129-150.

(Henze and Nejdl, 2003): Henze N., Nejdl W., (2003), "Logically Characterizing Adaptive Educational Hypermedia Systems", **Proceeding of the International Workshop of Adaptive Hypermedia and Adaptive Web-based Systems**, pp. 15-28.

(Oliver, 2001): Oliver R., (2001), "Learning objects: supporting flexible delivery of online learning", In G. Kennedy, M. Keppell, C. McNaught & T. Petrovic (Eds.) Meeting at the crossroads: **Proceedings of ASCILITE 2001**, pp 453-460, Melbourne.

(Paille, 2007): Paille G., "e-Learning objects versus any teaching materials", retrieved June 12, 2007, from: **http://edutechwiki.unige.ch/en/Learning_object**.

(Peñas *et al*., 2001): Peñas A., Verdejo F. and Gonzalo J., (2001), "Corpus-based terminology extraction applied to information access", **Proceedings of the Corpus Linguistics, UNED**, Spain.

(Pythagoras and Demetrios, 2004): Pythagoras K., Demetrios S., (2004), "Using Ontologies for Adaptive Navigation Support in Educational Hypermedia Systems", **Proc. of the 3rd International Conference on Adaptive Hypermedia and Adaptive Web-based Systems**, vol. 2, pp. 314-323, Eidhoven, Netherlands, TU/e Pub., August 2004.

(Rodrigue *et al.*, 2006): Rodrigue J. P., Comtois C. and Slack B., (2006), "The Geography of Transport Systems", Electronic edition, retrieved June 16, 2007, from: **http://people.hofstra.edu/geotrans/eng/ch2en/meth2en/ch2m1en.html**.

(Sampson and Karampiperis, 2004): Sampson D., Karampiperis P. (2004), "Knowledge Modeling for Adaptive Content Selection in Educational Hypermedia Systems", **IASTED Conference on Web Based Education WBE**, pp. 408-413.

(Soine, 2001): Soine R., (2001), "Instructional Design in a Technological World: Fitting Learning Activities into the Larger Picture", **Proceedings IEEE International Conference on Advanced learning technologies (ICALT'01)**, pp.49-52. IEEE Computer Society.

(Studer *et al.*, 1998). Studer, R., Benjamins V. and Fensel, D., (1998), "Knowledge engineering, principles and methods". **Data and Knowledge Engineering**, 25(1-2):pp161–197.

(Cormen *et al.*, 2001). Cormen T. H., Leiserson C. E., Rivest R. L., and Stein C., (2001), "**Introduction to Algorithms**", Second Edition, McGraw-Hill Book Company, pp. 629-635.

(Thomson, 2007): Thomson, "History of E-learning" retrieved June 2, 2007, from: **http://www.knowledgenet.com/corporateinformation/ourhistory/history.jsp**.

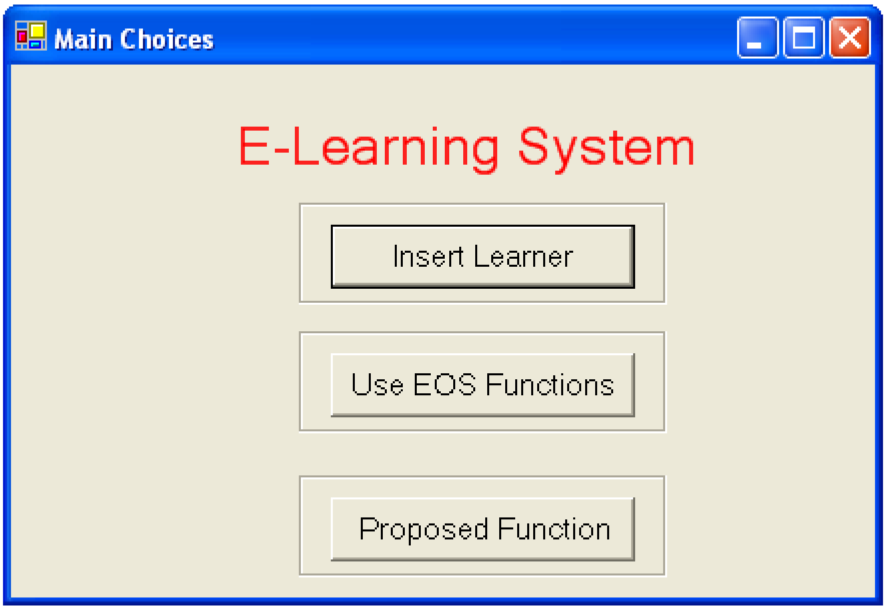
(Viet and Si, 2006): Viet A. N., Si D. H., (2006), "ACGs: Adaptive Course Generation System - An Efficient Approach to Build e-Learning Course", **Proceeding of The Sixth IEEE International Confernce on Computer and Information Technology (CIT'06)**, pp. 259- 265. IEEE Computer Society.

(Yanwen and Zhonghong, 2004): Yanwen W. and Zhonghong W., (2004), "Knowledge Adaptive Presentation Strategy in E-learning", **Knowledge Economy Meets Science and Technology-KEST** 2004.

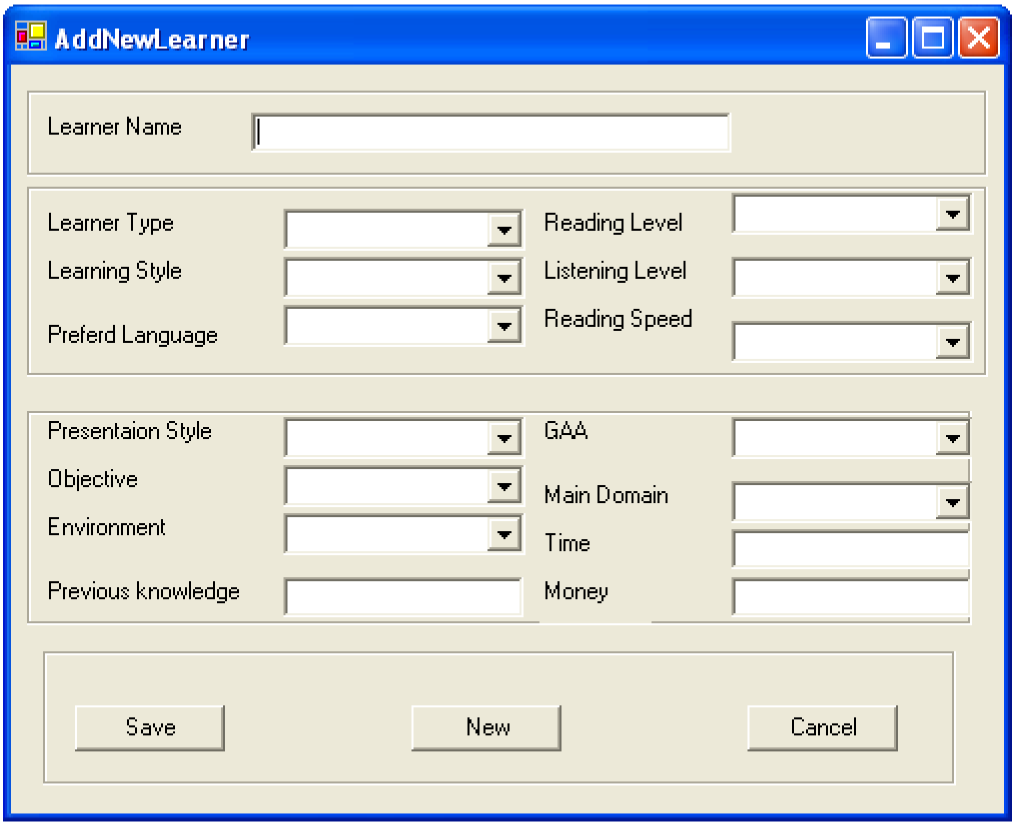
(Yu *et al*., 2006): Yu D., Zhang W., and Chen X., (2006), "New Generation of e-Learning Technologies", **First International Multi-Symposiums on Computer and Computational Sciences, IMSCCS'06.**, pp.455-459. IEEE Computer Society.

(Zhao and Wan, 2006). Zhao C. and Wan L., (2006), "A Shortest Learning Path Selection Algorithm in E-learning", **Proceeding of the Sixth International Conference on Advanced Learning Technologies (ICALT'06)**, pp. 94-95. IEEE Computer Society.

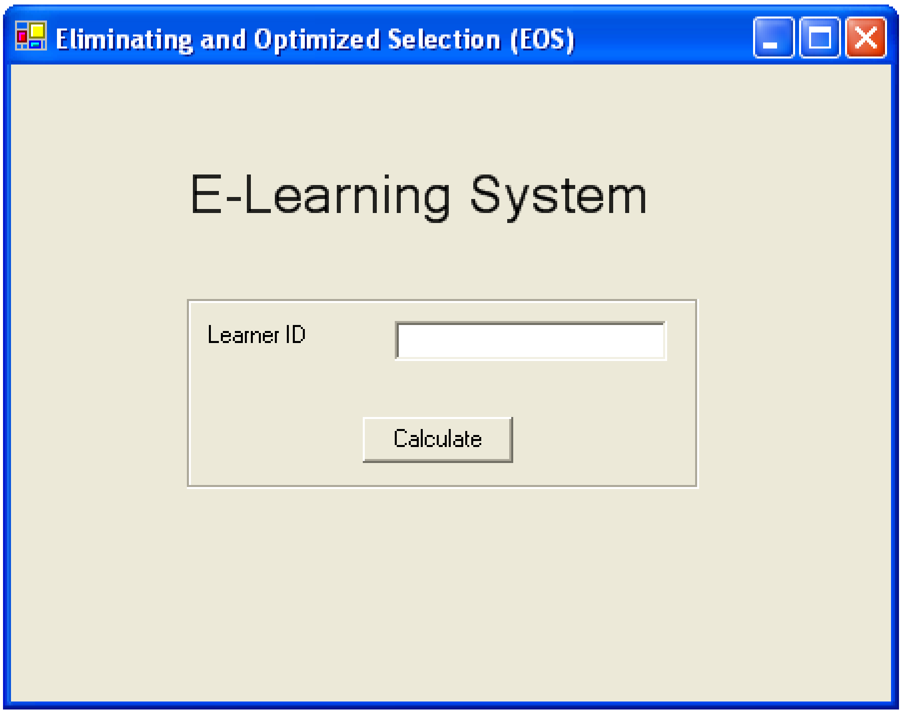
Appendix A
Samples of screens used in the system



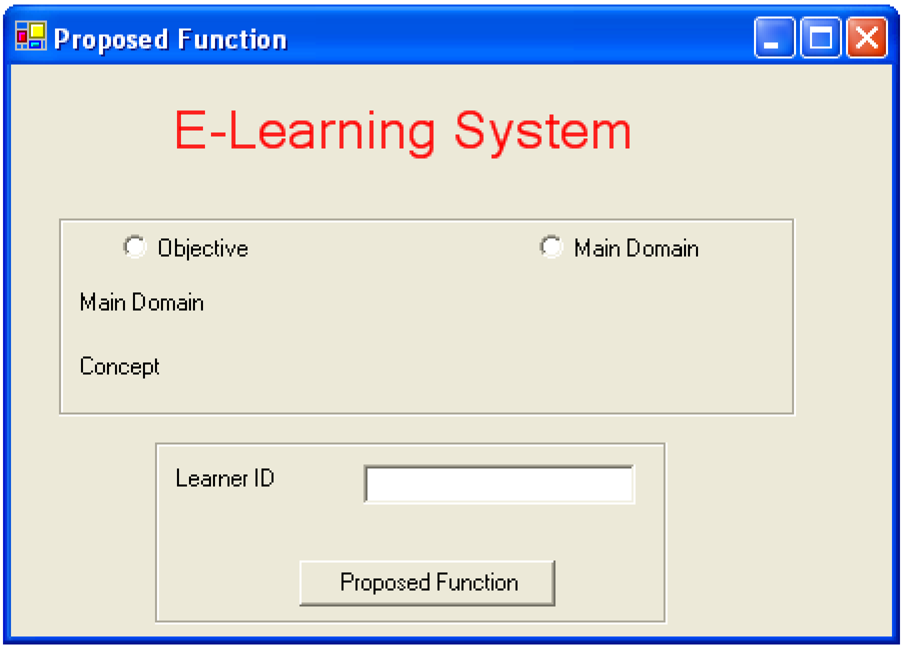
Screen 1: Enables the user to select one of three choices
- Insert new learner information
- Apply EOS functions
- Use the proposed approach for suitability



Screen 2: Enables the user to insert information about new learner



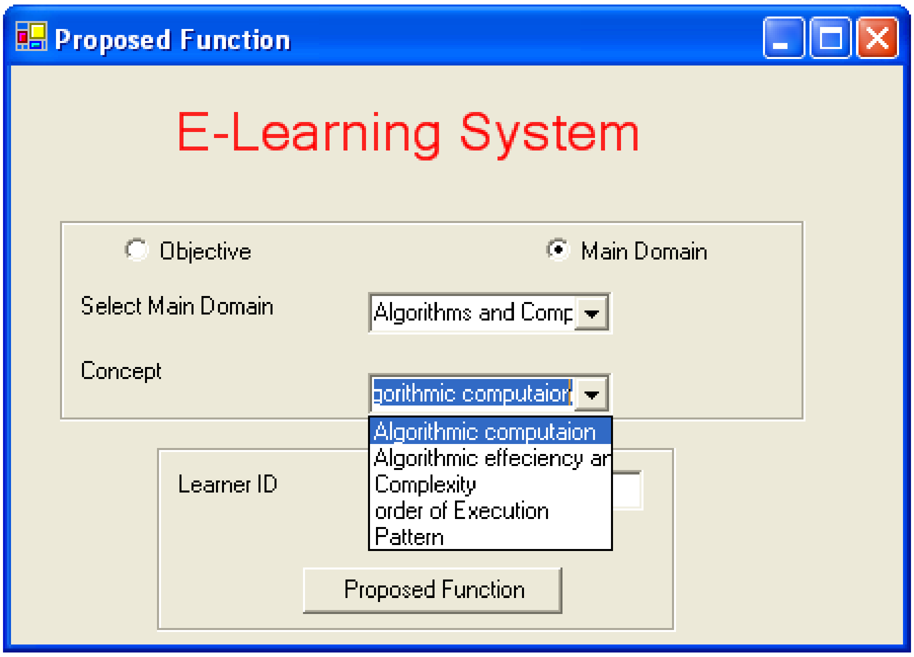
Screen 3: Enables the user to know the List of suitable learning objects obtained by EOS approach and their suitability for a specific learner after the insertion of the learner ID.



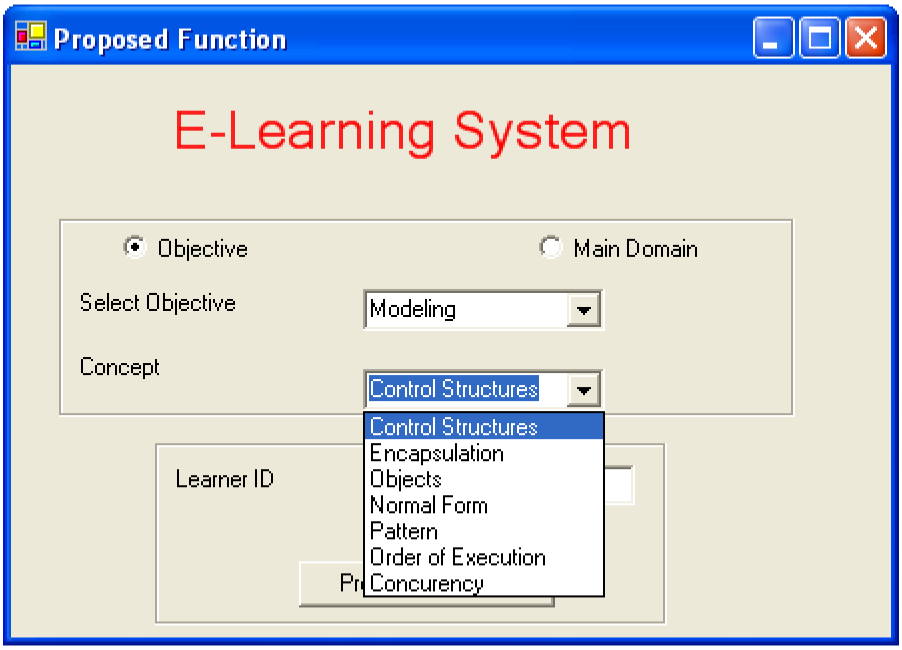
Screen 4: Enables the user to know the List of suitable learning objects obtained by the proposed approach and their suitability for a specific learner after inserting the learner ID and choosing the term (Domain + Concept , or Objective + Concept).

This Screen contains two options:

- choosing Main domain + Concept

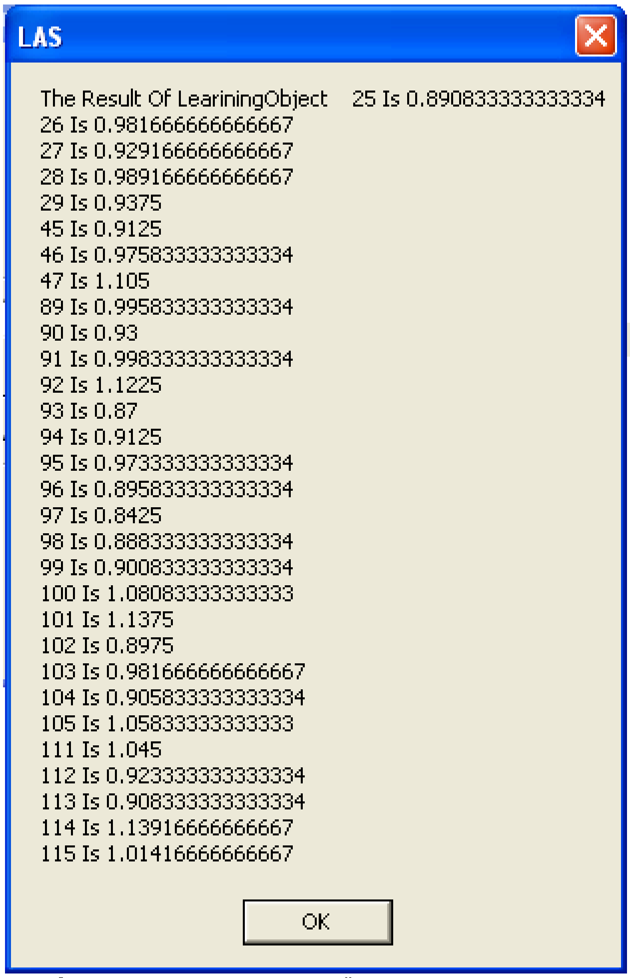


- choosing Objective + Concept

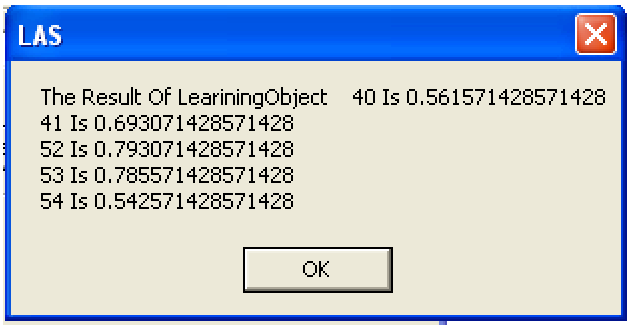


Appendix B
Samples of Implementation results using EOS approach

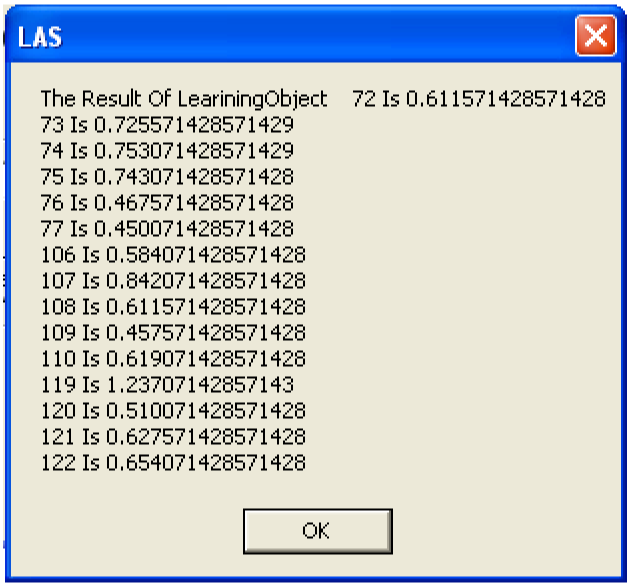
Concept: Normal Form
No. of LO/Concept: 30
Results:



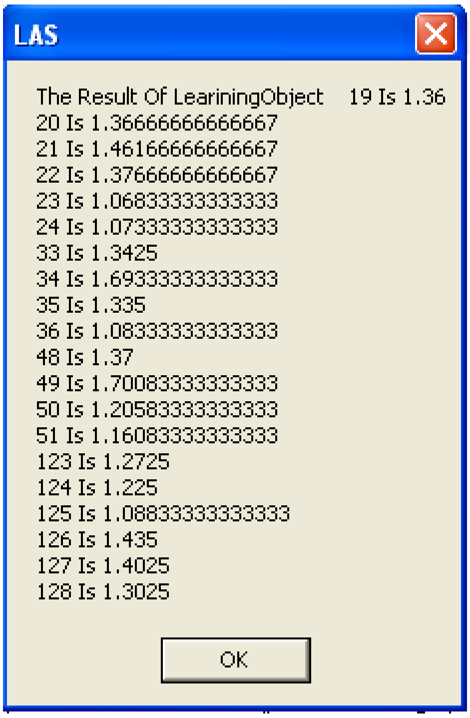
Concept: Function
No. of LO/Concept: 5
Results:



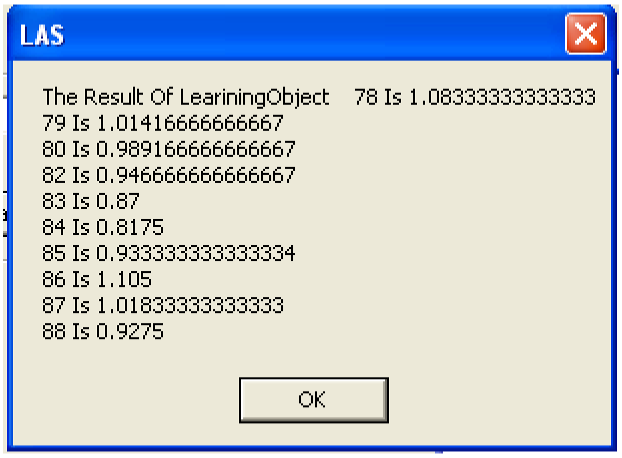
Concept: Pattern
No. of LO/Concept: 15
Results:



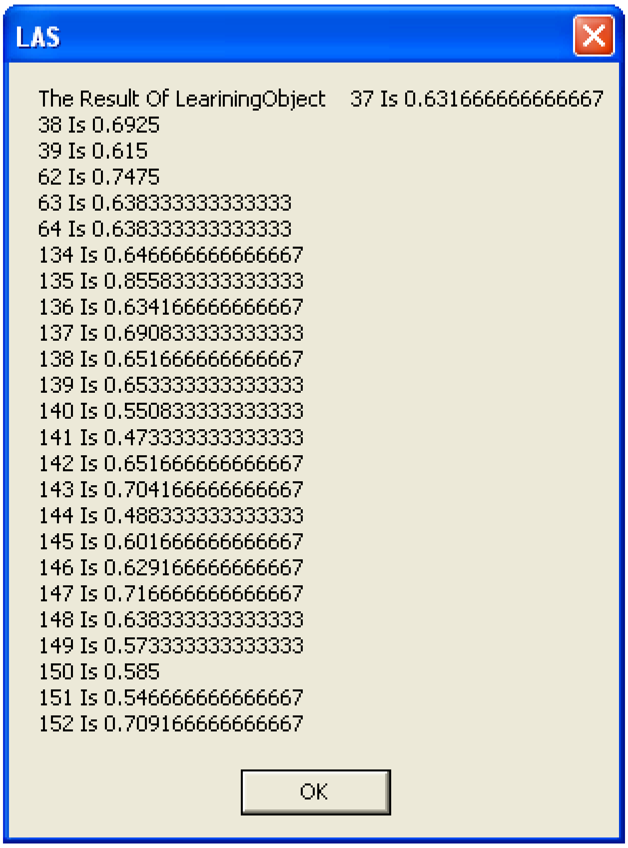
Concept: Order of Execution
No. of LO/Concept: 20
Results:



Concept: Complexity
No. of LO/Concept: 10
Results:

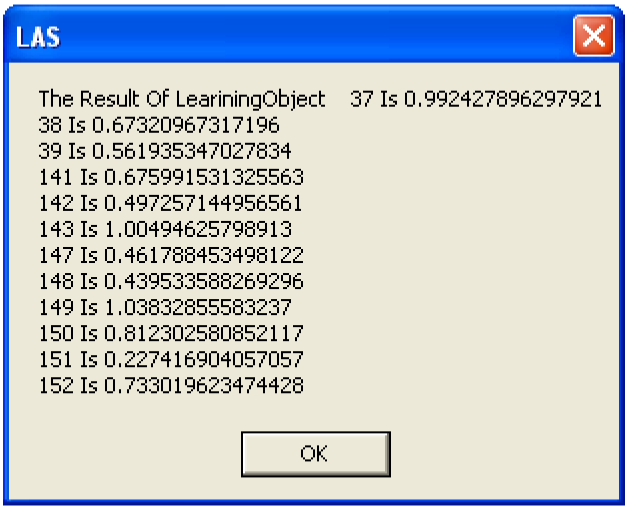


Concept: Encapsulation
No. of LO/Concept: 25
Results:

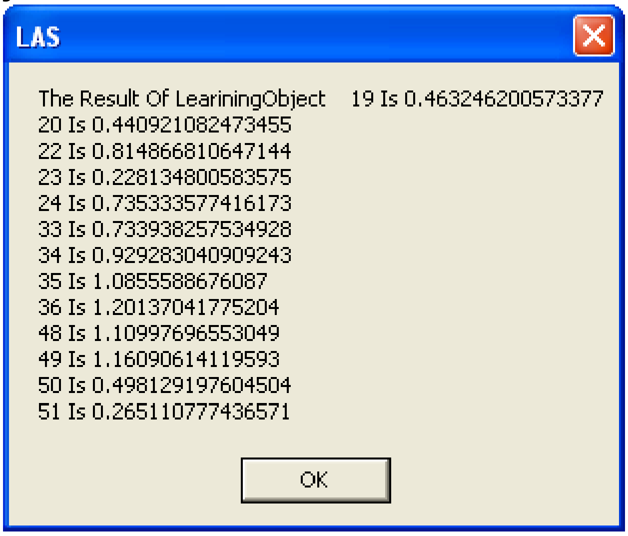


Appendix C
Samples of Implementation results using the proposed approach

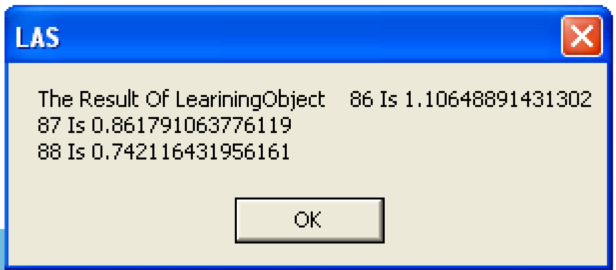
Concept: Encapsulation
No. of LO/Concept: 25
Results using Objective:



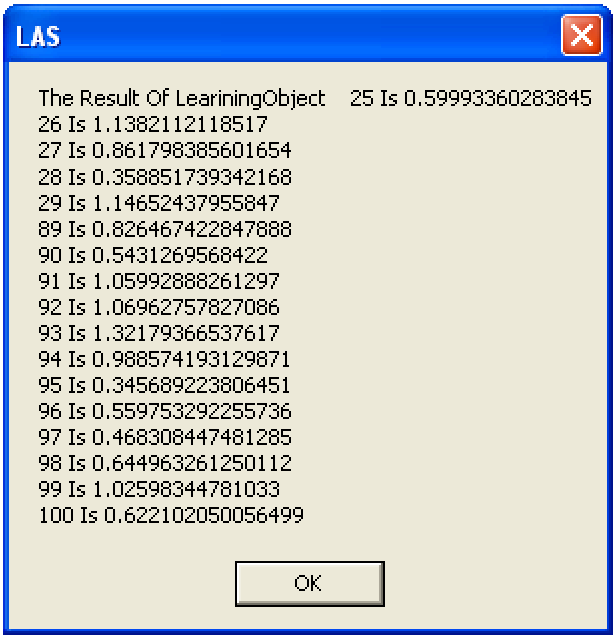
Concept: Order of execution
No. of LO/Concept: 20
Results using Objective:



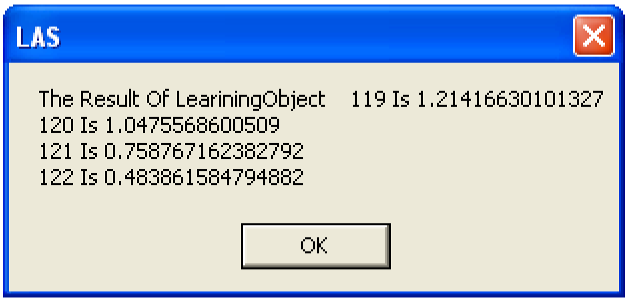
Concept: Complexity
No. of LO/Concept: 10
Results using Objective:



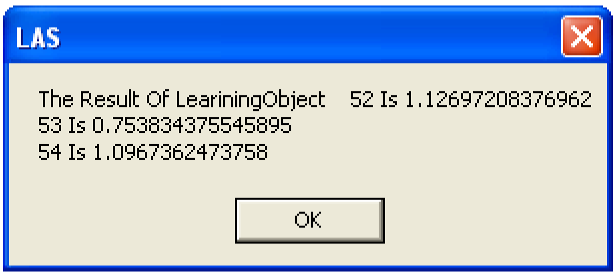
Concept: Normal Form
No. of LO/Concept: 30
Results using Objective:



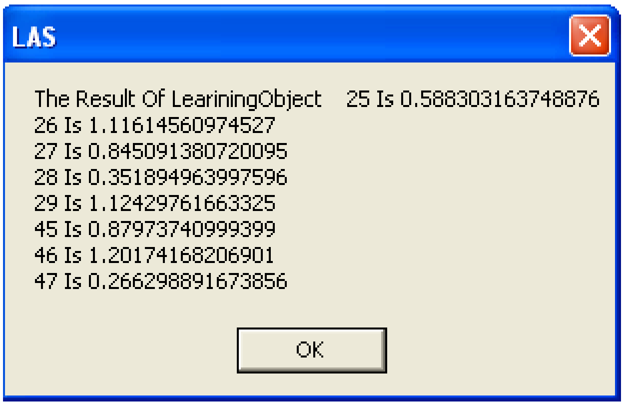
Concept: Pattern
No. of LO/Concept: 15
Results using Objective:



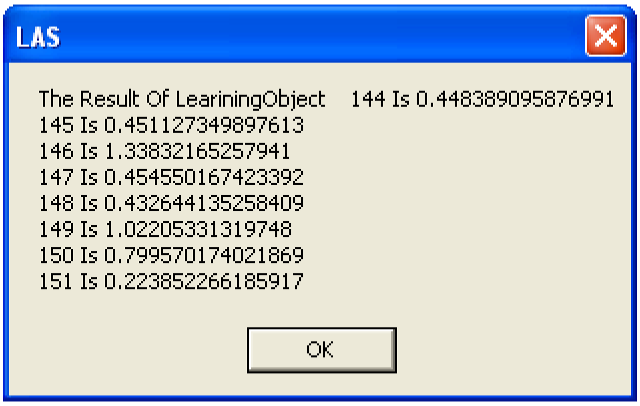
Concept: Function
No. of LO/Concept: 5
Results using Objective:



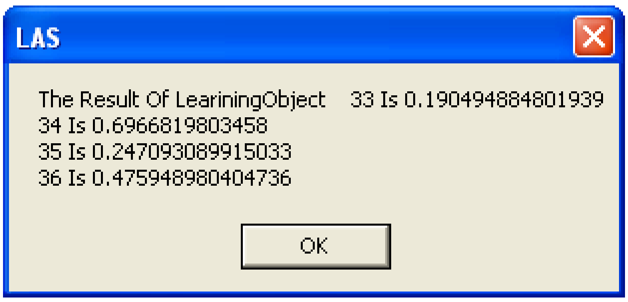
Concept: Normal Form
No. of LO/Concept: 30
Results using Domain:



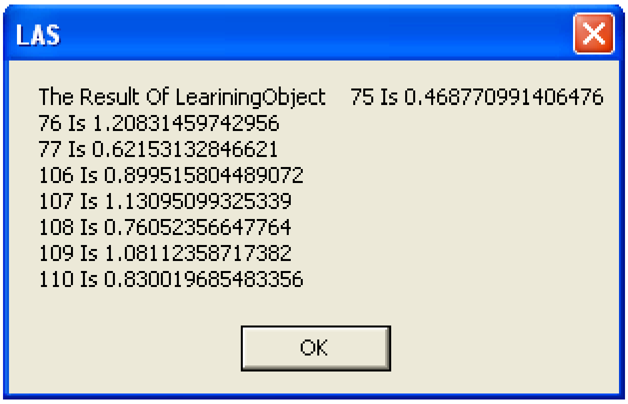
Concept: Encapsulation
No. of LO/Concept: 25
Results using Domain:



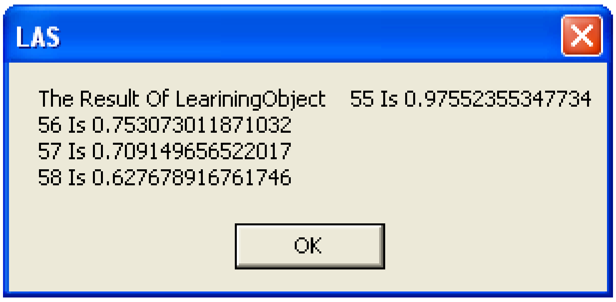
Concept: Order of Execution
No. of LO/Concept: 20
Results using Domain:



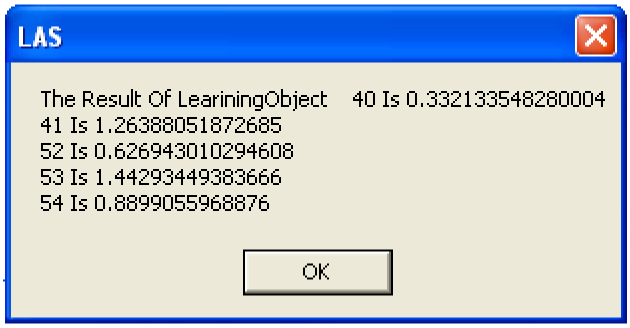
Concept: Pattern
No. of LO/Concept: 15
Results using Domain:



Concept: Concurrency
No. of LO/Concept: 10
Results using Domain:



Concept: Function
No. of LO/Concept: 5
Results using Domain:



الطريق الأقصر والأكثر تكيفاً مع المتعلم في أنظمة التعلم الإلكتروني

اعداد
مروة مصطفى عليان عبد الرحيم

المشرف
د. رياض جبري

الملخص

إن أكثر التحديات التي تواجه أنظمة التعلم الإلكتروني هي عملية توفير المقررات التعليمية الملائمة للدارسين على اختلافهم من حيث مستوى التعلم و درجة المعرفة. إن مثل هذه الأنظمة يجب أن تتصف بالفعالية والتكيفية. أحدث الأبحاث تصنف إلى مجموعتين رئيسيتين؛ المجموعة الأولى تركز على أن يكون نظام التعلم الإلكتروني تكيفياً بالدرجة الأولى، بينما المجموعة الثانية ركزت على الفعالية لمثل هذه الأنظمة. نهدف في هذا البحث الى تحقيق الفعالية والتكيفية ولتحقيق ذلك اخترنا خوارزمية تمثل المجموعة الأولى وخوارزمية تمثل المجموعة الثانية وحاولنا الجمع بينهما، وتم الإختيار بناء على أن الخوارزمية الأولى تحسن قدرة النظام على اختيار جزء المادة التعليمية (Learning Object) الملائمة لمتعلم معين بينما الخوارزمية الثانية تهدف إلى اختيار طريق التعلم الذي يكلف أقل وقت وجهد . لتحديد كيفية الدمج تم تحليل كل خوارزمية و من ثم تمثيلها برمجياً وكذلك تجربتها. نتيجة لذلك اقترحنا عادة تشكيل وبعض التعديلات لهذه الخوارزميات، ثم اقترحنا طريقة جديدة تجمع بين الخوارزميتين.

تمت مقارنة نتائج الطريقة المقترحة مع نتائج الخوارزميات المختارة كما تمت مقارنتها مع نتائج اختيار الخبراء. أظهرت هذه المقارنات تفوق الطريقة المقترحة من حيث تقديمها مواد تعليمة (Learning Objects) أكثر ملاءمة وأكثر تخصيصاً في اختيار المواد التعليمية (Learning Objects). علاوة على ذلك، فإن الطريقة المقترحة تبرهن تنافسها مع الخبراء من حيث تسلسل المواد التعليمية المختارة للمتعلمين على اختلافهم واختلاف حاجاتهم.