ACO-driven Personalized E-learning with Perspectives and Learning Aims

S. Chakraverty[1], Sushma Hans[2], and Vaibhav Mittal[2]
[1]Division of Computer Engineering, Netaji Subhas Institute of Technology, Delhi, Delhi, India

Abstract – *A proposal scheme to personalize students' learning based on her learning aim in an e-learning environment. The system acknowledges a distinct set of students' aims that prioritize various Learning Objects (LOs) such as theory, case studies etc. according to student's Learning Aim (LA). We also propose a new way to model the course as a directed acyclic graph in which nodes at each level represent a collection of different perspectives for the same concept. All concept perspectives encapsulate a set of LOs. This allows students to strengthen their learning along both dimensions; depth of coverage as well as perspective breadth by suitably adding to her repertoire of multiple viewpoints contributed by different authors on the same concept. This paper illustrates an Ant Colony Optimization (ACO) based personalized e-learning scheme to generate an initially optimized path taking into account the priorities of learning objects and the contribution of concept perspectives for different learning aims.*

Keywords: Personalized E-learning, Learning Objects, Learning Aims, Ant Colony Optimization, Directed Acyclic Graph, Concept Perspectives

1 Introduction

The Internet has become a centerpiece of a learner's educative environment, allowing her to learn at any time and any location conveniently [1]. Acknowledging the difference between diverse users' learning aims and their characteristics necessitate the need to create personalized environment for learning. A Personalized e-learning system tailors the courses according to individual users' requirements and preferences. Enormous work has been done in the field of personalized e-learning based on user profiles [2]. The profiles determine student behaviors that guide the e-learning system in generating a personalized path for each student.

An immense variety of learning material as well as different student's learning preferences drives the need for more and more sophisticated personalization techniques and ideas. Studies mainly emphasize on adaptation of course material based on user's learning style, background, qualification and experience [3, 4, 5]. Existing systems do not recognize the relevance of ordering various LOs according to users' Learning Aim (LA).

As an example to explain the motivation for the work in this paper, let us suppose that a learner's aim for taking up a course is to carry out project work. Then the priority assigned to the various LOs may be ordered as: detailed theoretical concept, case study, and simulation. Another user who aims at gaining cursory knowledge on the subject can choose basic concepts followed by an example. Therefore, we need to bring in the primary aim of learning as a steering factor for personalizing a course.

Further, for each concept that is part of the syllabus for a course, it can be explained by different authors from their own individual perspectives. Moreover, new perspectives that emerge when progress is made in the field must be added so that the course is regularly updated. On one hand, this allows the course to be enriched greatly by incorporating diverse ideas and viewpoints from different authors honed by individual experience. On the other hand, this also benefits learners by gaining maximally from the repository of perspectives according to their own needs, interests and time constraints. An optimized selection of these perspectives allows the learner to broaden her vision by studying different interpretations, thereby broadening her vision and augmenting her learning achievement. This paper proposes a framework that emphasizes the significance of user's LA while selecting the LOs as well as adding to the kitty of perspectives for grasping a concept.

Section 2 presents a discussion of related work. Section 3 introduces the proposed framework for an e-course and implementation of Ant Colony Optimization (ACO) based personalized learning path generation. In Sections 4 formalizes the experimental results. Section 5 discusses the simulation results. In Section 6, we conclude the paper and give directions for future work.

2 Related Work

The domain of personalized e-learning has benefitted from significant contributions made by different authors. Marwah et al. (2009) represented the content structure as a Directed Acyclic Graph (DAG) of LOs [6]. They proposed an algorithm that works on Elimination and Optimized Selection (EOS) to generate an adaptive learning path. The authors assess the concept relevance in the specific domain to recommend appropriate concepts to the user.

Wong et al. (2009) introduced a course sequencing technique that exploits rule-based prescriptive and ACO based inductive planning [7]. They developed DYnamic Learning Path Advisor (DYLPA) to provide recommendations for most appropriate learning pathway for the learner. Inductive

mechanism of DYLPA identifies similar learners from user logs and induces a suitable path for the new learner.

Marquez et al. (2008) structured the course in the form of sequencing graph in which the pedagogical team decides the sequencing of children according to course requirements [8]. This paper describes a way to generate learning path using ACO keeping the assistance provided by a pedagogical team through Bayesian Networks (BN).

Fung et al. (2011) employed a concept clustering technique to group similar concepts and determine precedence between various course modules [9]. The precedence relation information is then passed to a GA to optimize the learning sequence of relevant course modules.

Bhaskar and Chithralekha (2010) identified various LOs such as concept, example and case study of the course based on user's intentions and preferences [10]. The system sequences these LOs according to psychology of the learner. The authors make use of GA to generate a learning path accommodating learner's content, presentation and media level contexts.

Acampora et al. (2008) defined the course in an ontological form in which various concepts are connected through three types of relations: the *Has- Part*, the *Is-Required-By* and the *Suggested-Order* [11]. The proposed system exploits memetic algorithm to determine an optimized path from the defined ontology.

Azough et al. (2010) implement Genetic Algorithms (GA) to establish an optimal path adapted according to learner's profile [12]. The generated path satisfies the pedagogical objectives set by the experts for the course.

Hong et al. (2005) utilized the cosine similarity measure to determine the degree of relatedness between the concepts [13].The cosine similarity measure takes into account the term frequencies as well as the Importance weights assigned to concepts. These relations are then passed to a GA to determine the optimal learning path for an individual learner. Liu and Yang (2005) proposed an Adaptive & Personalized e-Learning System (APeLS) that represents the course in the form of dual weighted directed graph [14]. Each node represents a learning object and its weight signifies the required learning time as defined by the experts. Directed edges express the precedence relationship and their weights imply the difficulty level from one node to the next node. The system establishes the best learning path from the course graph using Dijkstra's Algorithm. It also evaluates the quality of the determined path on the basis of user's goals and learning achievement. Zhao and Wan utilized the same course structure [15] with a new algorithm to find out shortest learning paths to gain the required knowledge.

Jiuxin et al. (2008) put forward a self adaptive framework of learning objects in accordance to the learning context of user [16]. Learning context includes the learner's internal and digital environments such as access devices, network and personal information. The framework generates different versions of learning objects dynamically according to learner's learning context.

A scrutiny of the above mentioned works reveals two lacunae: Firstly, a concept is considered closed once it is part of a course. In reality a concept is very much a dynamically changeable artifact. It can be enriched by contributions from different authors, each adding his/her own expertise. Besides, as time progresses, new perspectives on the same concept emerge. These fresh viewpoints must get incorporated for the course to remain relevant. Examining different perspective provide additional knowledge and leads to more satisfaction towards the user's LA. We recognize that there is a need to allow adding different perspectives for a concept in a flexible and modular manner. Depending upon the current state-of-art of a domain, the relative importance of different perspectives may change. Therefore, the learner and the recommendation system that guides the learner in choosing an optimal path must be assisted by attributing contributory weights to the various perspectives.

Secondly, past work does not identify the significance of prioritizing various type of LOs according to user's learning aims. LA is a vital parameter that influences the user's selection of learning material as well as the learning process.

3   Proposed Framework

We explain below the proposed framework for modeling the course structure in a personalized e-learning system. We provide an algorithm for the system to recommend an initial learning path that is fine-tuned to the learner's own aim for taking up the course. This algorithm is named PAL(.) (Perspectives and Aims for Learning).

3.1   Course Organization

The course is organized into a sequence of nC concepts $C_1, C_2 \ldots C_{nC}$. The learner has to proceed from one concept to the next to go through the course and cannot entirely miss any concept. Each concept is presented from different authors' points of view, called *Perspectives*. The course is divided into levels in which each level contains various perspectives of a concept. The number of levels is equal to the number of concepts prescribed for the course. This provision allows more than one expert to contribute to the course material and subjects the learner to various angles of a concept as interpreted by different subject experts. The learner has the choice to grasp the same concept from one or a combination of these various perspectives to strengthen her knowledge by broadening her vision. This aspect of learning is assessed by a metric called *Concept-Depth CD*.

Further, each concept is broken down into a set of pre-defined Learning Objects *LO* including artifacts such as basic theory, detailed theory, case study, audio-visual demonstration, simulation, point-wise summary and quiz. For a given level L_j, there are $nLO(j)$ learning objects prescribed for the corresponding concept attached with that level. Each author contributing to a concept must cover all *LOs* necessary for that concept. At least one *LO*, the basic theory, must be chosen by the learner to concretize the underlying concept. However, it is not essential for a learner to go through all these learning objects.

As an initial recommendation, the learner is prescribed a learning path that includes an optimized combination of *LOs* for each concept as guided by her learning objective and time constraints. As she progresses along the course, the dynamic recommendation system can dynamically adjust the learning path based on her performance and choices.

3.2 Inputs

1. *Course Graph:* A weighted Directed Acyclic Precedence Graph (DAPG) for the course: *CG(V,E)*. If node v_1 precedes v_2, it is denoted as $v_1 \rightarrow v_2$.

1.1 *Levels, concepts and Perspectives: CG(.)* is partitioned into discrete levels: $L_1, L_2, \dots L_i \dots, L_{nL}$, each level representing one specific concept of the course. Each level comprises a subset of vertices: $L_{j=} \cup \{v_{j,k}\}$, each vertex representing a definite perspective $P_{j,k}$ for the same concept. For a given level L_j associated with one concept, there are $nP(j)$ number of perspectives. For convenience, the course starts with a single concept from a unique introductory perspective. This node is designated as v_1. Each of the vertices belonging to a specific level leads to all vertices of the next level. Hence we have:

$$\forall k, m: k \ \varepsilon \{1..nP(j)\} \ and \ m \ \varepsilon \{1..nP(j+1)\}, \ v_{j,k} \rightarrow v_{j+1,m}$$

This precedence rule ensures that whatever perspective(s) a learner chooses to learn a concept, she has to make a transition from the concept encapsulated at that level to the next concept in the prescribed sequence.

1.2 *CG(.) Node Weights:* Each vertex has a weight $T(v_{j,k})$ assigned by experts that denotes the total time taken by a learner visiting that node if she were to learn all the learning objects contained within that node.

1.3 *CG(.) Edge Weights:* Edge $e_{j,k1,k2}$ represents a transition from a perspective at one level (concept) $P_{j,k1}$, to the a perspective at the next level (another concept) $P_{j+1,k2}$. The edge weight $DL(e_{j\,k1,k2})$ represents the Difficulty Level (DL) in making the corresponding transition. DL is determined by an expert. We normalize the DL at a particular level by dividing it with maximum DL at that level. A learner transits from one level to next via exactly one of these possible paths. Once having reached the next level, she can add to the number of perspectives provided at that level without additional difficulty.

2. *Learning Aims:* User chooses from among a list of primary Learning Aims *{LA}*, indicating the purpose of her taking the course. For example, the learning aims possible for a course can be 1. Gaining in-depth knowledge 2. Preparing for an interview 3. Apply associated practical skills 4. Satisfy cursory interest 5. Professional training 6. Preparing for examination. Let there be *nLA* such learning aims. LA_a is the learning aim associated with user U_a.

3. *Learning Objects***:** Each vertex $v_{j,k}$ can be exploded into sub-graph $LOG_{j,k}(V,E)$, whose vertices represent Learning Objects $\{LO_j\}$ and whose edges represent possible transitions from one learning object to another. Vertex $v_{j,k,l}$ represents the lth LO of the kth perspective at the jth conceptual level. The starting node is fixed and represents a mandatory *LO* that the learner must cover to have a basic

understanding of the concept. The learner can close the concept by exiting from any of the visited nodes. Unlike the main course graph, there is no difficulty in moving from one LO to another within a concept. As an independent entity, this sub-graph does not have edge weights. However, in order to enable the agents to move freely between LOs, they are all assigned zero weights. Thus, the ants can collect LOs to enhance learning as long as the maximum duration of the course is not exceeded.

LOG(.) Node weights: The total time $T(v_{j,k})$ allocated to a vertex $v_{j,k}$ of the graph $C(.)$ is divided among its various *LOs* as per the priorities pertaining to the learning aim LA_a of a user U_a. The time taken at the vertex $v_{j,k,l}$ of the sub graph $S_{j,k}(.)$ is its weight:

$$T_{j,k,l}(a) = \frac{T(v_{j,k}) \times LALOPT(a,l)}{\sum_{l=1}^{nLO(j)} LALOPT_j(a,l)}$$

4. *Priority Table:* Experts prescribe a Priority Table LALOPT$_j(.)$ for each level *i.e.* concept. This table shows the relative priorities assigned to each of the Learning Objects *LO* for each of the possible Learning Aims *{LA}*. This tables has *nLA* rows and *nLO(j)* columns.

Table 1 - Learning objects' priority according to learning aim for a particular concept

Priority Table LALOPT$_j$(R[nLA] ×C[nLO$_j$])						
Learning Aims {LA}	Learning Objects {LO$_j$} for the concept at Level L_j					
	$LO_{j,1}$	$LO_{j,2}$	...	$LO_{j,l}$	...	$LO_{j,nLO(j)}$
LA_1						
LA_2						
.						
LA_a						
.						
LA_{nLA}						

5. *Learning Perspective Contribution Table:* A user adds to her learning when she learns from different perspectives for a given concept. At each level, the contribution of each perspective towards learning varies according to the learning aims of the learner. Experts determine how much a particular perspective on a concept will contribute to the overall learning of a user as per her desired learning aim. This is captured in a Perspective-Aim Contribution Table *PACT$_j$[.]* declaring the contributions of various Perspectives towards the users' Learning Aims. Let $C_{j,z}$ be the relative contribution of the zth perspective at j^{th} level for a specific LA.

Table 2 - Concept perspectives constribution towards various learning aims

Contribution Table PACT$_j$(R[nLA],C[nP$_j$])					
Learning aim {LA}	Perspectives *{P}* for a concept at Level L_j				
	$P_{j,1}$	$P_{j,2}$	...	$P_{j,l}$	...	$P_{j,nP(j)}$
LA_1					
LA_2					
.					
LA_a.					
.					
LA_{NLA}					

5.1 *Perspective sub-graph*: All perspectives within a level are interconnected into a completely connected sub-

graph $PG_j(V,E)$ whose vertices comprise all vertices of a level L_j. There is no difficulty in moving from one perspective to another. Hence the edge weights are not significant but are assigned zero value to enable the learning agents to move smoothly from all perspective to another in an effort to her perspective width.

5.2 *Concept Depth v/s Learning Achievement:* It is assumed that the first vertex visited at a given level is the primary contributor to Learning Achievement α_a of a user. Additional vertices add to the concept depth Ω_a.

6. *Timing Constraints: T_{max}:* The maximum time decided by the user to finish the course.

7. Source node V_s and destination node V_d provided by the user U_a.

3.3 Output

An optimum Initial Learning Path ILP_{opt} comprising at least one and possibly more node(s) from each level and learning objects gathered for each node.

Let $nP\_taken(j)$ be the number of perspectives that the learner has taken at level L_j. Let the nodes of this path be $ILP_{opt}=\{v_{j,k,l}\}$. Then index j varies from 1 to nL, for each of these levels k varies from 1 to $nP(j)$ and each combination of j and k, l varies from 1 to the number of LOs picked up at node $v_{j,k}$ of $CG(.)$.

3.4 Objectives

1. Learning of a user comprises of three factors:

a) Coverage Factor (CF): It measures learning of a user across various concepts selected in the learning path. CF only considers the first concept perspective selected at particular level in the learning path. The learning achievement at each level L_j is given by:

$$LA_{j,1} = M_j(1 - e^{-\propto t_{j,1}}) \qquad (1)$$

Where, M_j is the maximum learning achievement at that level set by the experts. α is the user' Learning Ability determined through pre-test, and $t_{j,1}$ is the time taken by the learner at the first node she visits at that level. CF can now be defined as:

$$F = \sum_{j=1}^{nL} LA_j \qquad (2)$$

b) Depth Factor (DF): It evaluates depth of learning of a specific concept attained by selecting various concept perspectives. The depth of learning at a particular level is:

$$DF_j = \frac{\sum_{k=1}^{nP\_taken(j)} LA_{j,k} \times C_{j,k}}{\sum_{k=1}^{nP(j)} LA_{j,k}} \qquad (3)$$

The overall depth factor is

$$DF = \sum_{j=1}^{nL} DF_j \qquad (4)$$

c) Cumulative Difficulty Level (CDL): CDL signifies the difficulty of the selected path. Only the transition edges from one level to another level of the selected path contribute to CDL as adding perspectives do not increase the difficulty. It is calculated through summation of difficulty level of all these transition edges.

$$CDL = \sum_{j=1}^{nL} DL_j \qquad (5)$$

The main aim of undertaking a course is to

(1) Maximize Coverage Factor CF:

$$\sum_{m \in PT_{op}} CF_m^{op} > \sum_{p \in PT, n \in PT_p} CF_n^p$$

(2) Maximize Depth Factor DF:

$$\sum_{m \in PT_{op}} DF_m^{op} > \sum_{p \in PT, n \in PT_p} DF_n^p$$

Where PT_{op} signifies the optimized path and $PT = \{P_0, P_1, \ldots\ldots, P_n\}$ represents set of all possible paths from source to destination concept.

(3) Minimize Cumulative Difficulty level CDL for the optimized path:

$$\sum_{m \in PT_{op}} CDL_m^{op} > \sum_{p \in PT, n \in PT_p} CDL_n^p$$

All these factors contribute towards the overall fitness *f(.)* of the solution for the particular user with specific *LA* according to the function

$$f = w_1 \times CDL - (w_2 \times CF + w3 \times DF) \qquad (6)$$

where w_1, w_2 and w_3 are the constant weights assigned according to these factors significance. Minimum the value of *f*, most appropriate is the path for the user.

3.5 Constraints

The system discards all those paths in which *time taken > T_{max}*. This constraint signifies that user must have to keep in mind the time deadline set by her.

3.6 An ACO Scheme for personalized learning

ACO is meta-heuristic algorithm employed to solve many optimization problems [17, 18]. ACO initializes many artificial ants that walk through the input graph to find optimized path. The movements of each ant depend on the pheromones laid by the ants on the path. Ants prefer the path with comparatively high amount of pheromones on it. The pheromones value of the selected path is updated simultaneously as well as after an ant reached the destination on the basis of fitness of the obtained solution. The pheromones also evaporate from the paths to avoid local optima solution. Selection of next node is based on probability state transition rule that works on two factors: heuristic factor $\vartheta_{u \to v}$ as well as pheromone factor $\tau_{u \to v}$. Next node will be selected according to probability distribution determined by

$$p_{u \to v} = \frac{\tau_{u \to v}^{\alpha} \times \vartheta_{u \to v}^{\beta}}{\sum_{w \in v} \tau_{u \to w}^{\alpha} \times \vartheta_{u \to v}^{\beta}} \qquad (7)$$

where $\alpha \geq 0$ and $\beta \geq 1$ determine the relative impact of pheromone and heuristic factor on decision of the ant. Heuristic factor $\vartheta_{u \to v}$ is calculated according to following rule:

$$\vartheta_{u \to v} = LA_v \qquad (8)$$

The local update of pheromone value on the selected path is determined by

$$\tau_{u \to v} = \begin{cases} \tau_{u \to v} + \epsilon \times \delta & DL_{u \to v} = 0 \\ \tau_{u \to v} + \frac{\epsilon}{DL_{u \to v}} & otherwise \end{cases} \qquad (9)$$

where ϵ denotes the initial pheromone value set on the edge $u \to v$ and δ denotes a large positive constant. The global update of pheromones value on the selected path is done by

$$\tau_{u \to v} = \tau_{u \to v} + \frac{\psi}{pathValue(ILP_{current})} \qquad (10)$$

where ψ controls the pheromone value deposited on the best path, set as T_{max}. The evaporation of the pheromones value is computed by

$$\tau_{u \to v} = \begin{cases} \epsilon & \epsilon > \tau_{u \to v} \times (1 - \mu) \\ \tau_{u \to v} \times (1 - \mu) & otherwise \end{cases} \qquad (11)$$

Here μ signifies the evaporation rate of pheromones on each edge.

The pseudo code in Figure 1 describes the implementation of ACO algorithm to find optimal path according to user's aim LA_a

```
PAL (
Input:    CG = (V,E), LALOPT, PACT, LA, LA_a , T_max
Output: ILP_opt , bestPathValue )
Begin
1. Initialization phase
   – Initialize an initial population of ants X_i , i = 1,2...,m
     and bestPathValue = ∞
   – Initialize all constant parameters and pheromone trails
     value for each edge in CG.
2. Starting tour
   – Place all ants on source node Vs
   – Maintain list time_spent for each ant to store time
     spent on each added edge.
3. Search ant path
   For counter i=1 to number_of_iterations do
     For ant X_k counter k= 1 to m do
     – Select_Next_Node (u, v, CG)  /* Select_Next_Node
       select node v to move to from u in CG */
     – Add v to ant's path and store time spent on vertex v
       to ant time_spent list
       If ( next vertex v  = destination node V_d)
4. Compute pathValue for each ant's path according to
user's aim LA_a
        – Compute pathValue using formula (6) and total
          timeSpent on the selected path
        If (timeSpent > T_max)
           – Set pathValue = ∞
        Else
           If (bestPathValue > pathValue)
              – Set bestValue ← pathValue
              – ILP_opt ← ant.path
           End  If
        End If
        – Re-initialize ant so that it can start to search for
          path again.
     End If
     End For
5. Update the pheromone factor
   – Update pheromones locally on selected  path by  ant
     using (9)
   – Update pheromones globally on path selected by  ant
     using (10)
   – Evaporate pheromones value on the path with formula
     (11)
   End For
   – Return ILP_opt , bestValue
End
```

Figure 1 - Pseudocode to find optimized learning path through ACO

The pseudocode in Figure 2 choose next node by applying fitness proportionate selection[19] to cumulative probability list of all adjacent nodes to u.

```
Select_Next_Node (
Input: Current vertex u, Course Graph CG
Output: Next vertex v)
Begin
   – Calculate probability p_{u→w}  using (7) for each of the
     n vertices adjacent to u and store it in probability list
     P where each index i corresponds to the specific
     adjacent vertex w.
   – Create list cp of cumulative probabilities from P
     with size n+1 and cp_i = cp_{i-1} + p_i  taking cp_0 = 0.0
   – Take a random variable rp with value between
     0.0 to 1.0 and set v ← w where w is denoted by
     index i and satisfy the condition cp_i ≤ rp < cp_{i+1}
   – Select random number of LOs' for selected vertex
     including the compulsory LO.
End
```

Figure 2 - Pseudocode to select next node in ant's path

4   Simulation Results

The section presents the experimental results of the proposed algorithm. The proposed algorithm is implemented with JAVA SE 6 platform on Intel Corei3 2.13 Ghz computer with 4GB RAM. The values of parameters in ACO are number of iterations $I = N * 10000$, where N = total number of nodes, ant population $m = 10, \alpha = 0.8, \beta = 1.0, \mu = 0.9, \epsilon = 0.1, \delta = 100$. The values of weights for fitness f are $w1 = 0.5, w2 = w3 = 1.0$

This work utilizes the course graph presented in Fig. 3 and input Table 3 to Table 15 and the time constraint $T_{max} = 20$.
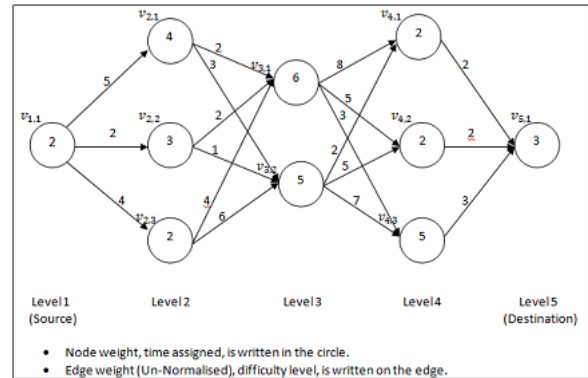

Figure 3 - Course Graph

- Node weight, time assigned, is written in the circle.
- Edge weight (Un-Normalised), difficulty level, is written on the edge.

Table 3 - Number of LO's at Level

Level	1	2	3	4	5
Number of LOs	2	3	2	2	3

Table 4 - Number of Perspectives at Level

Level	1	2	3	4	5
Number of Perspectives	1	3	2	3	1

Table 5 - Maximum Learning Achievement at various Levels

Level	1	2	3	4	5
Un-Normalized Maximum Learning Achievement	10	15	20	20	13

Table 6 - Priority Table Level 1

Learning Aim / LO	1	2
1	0.62	0.38
2	0.52	0.48
3	0.26	0.74

Table 7 - Priority Table Level 2

Learning Aim / LO	1	2	3
1	0.50	0.34	0.16
2	0.26	0.26	0.48
3	0.15	0.21	0.64

Table 8 - Priority Table Level 3

Learning Aim / LO	1	2
1	0.18	0.82
2	0.26	0.74
3	0.89	0.11

Table 9 - Priority Table Level 4

Learning Aim / LO	1	2
1	0.39	0.61
2	0.16	0.84
3	0.61	0.39

Table 10 - Priority Table Level 5

Learning Aim / LO	1	2	3
1	0.33	0.33	0.34
2	0.40	0.25	0.35
3	0.10	0.86	0.04

Table 11 – Contribution Table Level 1

Learning Aim / Perspective	1
1	1.00
2	1.00
3	1.00

Table 12 – Contribution Table Level 2

Learning Aim / Perspective	1	2	3
1	0.45	0.32	0.23
2	0.40	0.17	0.43
3	0.36	0.34	0.30

Table 13 – Contribution Table Level 3

Learning Aim / Perspective	1	2
1	0.50	0.50
2	0.45	0.55
3	0.48	0.52

Table 14 – Contribution Table Level 4

Learning Aim / Perspective	1	2	3
1	0.11	0.50	0.39
2	0.29	0.36	0.35
3	0.46	0.22	0.32

Table 15 – Contribution Table Level 5

Learning Aim / Perspective	1
1	1.00
2	1.00
3	1.00

The learning paths generated for various aims via the proposed ACO algorithm is shown in Fig. 4, Fig. 5 and Fig. 6 highlighted with different colors. The pattern of 0 and 1 associated with each node selected in path represent the LOs selected to maximize learning.
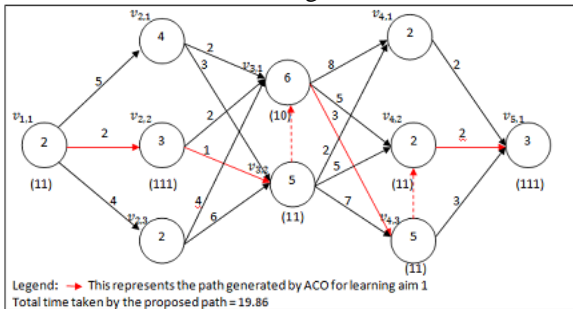

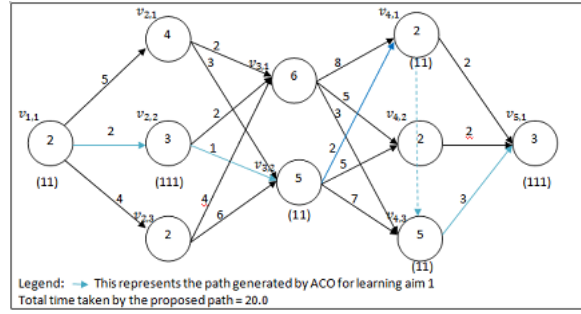Figure 4 – Proposed path for learning aim 1
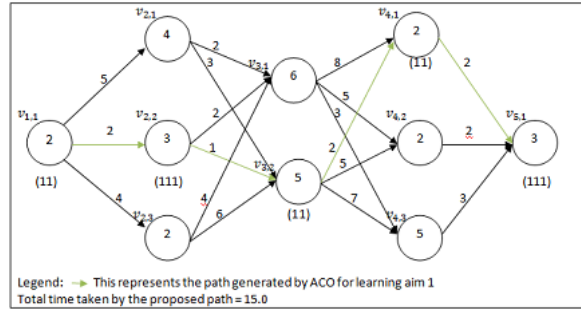

Figure 5 - Proposed path for learning aim 2

Figure 6 - Proposed path for learning aim 3

5 Discussion

Simulation results show that ACO generate optimal path for user with particular learning aim. The generated path capitalizes user's learning by emphasizing the following factors:

- Maximizes depth of learning within the time period T_{max} provided by the user by selecting the relevant concept perspectives that contributes more towards user's learning aim
- Enhances user's learning achievement by selecting appropriate LOs
- Minimizes the difficulty level

6 Conclusion and Future Work

Although optimal learning path generation is not a new idea in e-learning field, this study could still be considered novel as it is the first path generation scheme that broadens the learner's vision by explicitly including various perspectives of a concept to provide in-depth knowledge. It leads to more satisfaction towards user's learning aim. This paper has also presented a positive impact of prioritizing LOs according to user's LA, on user's learning. An ACO based solution scheme has been developed to generate learning path that provides more satisfaction towards user's aim.

For future work, we would like to focus on transforming this static path generation scheme to a dynamic path adaptation. Personalization of evaluation and scoring system according to user's LA could also be an innovative idea for adaptation as it gives more concise knowledge about user's learning that assist in personalization of learning path.

7 References

[1] Chengling Zhao and Liyong Wan, "A shortest Learning Path Selection Algorithm in E-learning", *In. Proc. of the Sixth International Conference on Advanced Learning Technologies (ICALT'06),* IEEE 2006.

[2] S. Schiaffino and A. Almadi, "Intelligent User Profiling", *In Proc. of the Artificial Intelligence: An International Perspective 2009,* pp. 193-216.

[3] M. K. Khribi, M. Jemni and O. Nasraoui, "Automatic Rrcommendations for E-Learning Personalization Based on Web Usage Mining Techniques and Information Retrieval", *In Proc. of the Eighth International Conference of the Advanced Learning Technologies, Santander, July 2008*, pp. 241-245.

[4] A. A. Wabil et al. ,"Exploring the Validity of Learning Styles as Personalization Parameters in eLearning Environments: An Eyetracking Study", *In Proc. of the ninth IEEE International Conference on the Advanced Learning Technologies, Riga, August 2009*, pp. 329-333.

[5] G. Savic and Z. Konjovic, "Learning Style Based Personalization of SCORM E-Learning Courses", *In Proc. of the 7th International Symposium on Intelligent Systems and Informatics, Subotica, October 2009*, pp. 349-353.

[6] Marwah Alian and Riad Jabri, "A Shortest Adaptive Learning Path in eLearning Systems: Mathematical View", *Journal of American Science 2009,* Vol. 5(6), 2009, pp. 32-42.

[7] Lung-Hsiang Wong and Chee-Kit Looi, "Adaptable Learning Pathway Generation with Ant Colony Optimization", *In Proc. of the Educational Technology and Society, National Institute of Education,* 12(3), Singapore, 2009, pp. 309-326.

[8] Jose Manuel Marquez, Juan Antonio Ortega, Luis Gonzalez-Abril and Francisco Velasco, "Creating Adaptive Learning Path using Ant Colony Optimization (ACO) and Bayesian Networks", *IEEE International Joint Conference on Neural Networks, (IJCNN'08), 2008*, pp. 3834-3839.

[9] Fung, Vincent Tam and Edmund Y. Lam, "Enhancing Learning Paths with Concepts Clustering and Rule-Based Optimization", *In. Proc. of the 11th IEEE International Conference on Advanced Learning Technologies* 2011.

[10] Manju Bhaskar, Minu M Das, T. Chithralekha and S. Sivasatya, "Genetic Algorithm Based Adaptive Learning Scheme Generation for Context Aware E-Learning", *International Journal of Computer Science and Engineering, Vol. 02(4),* 2010, pp. 1271-1279.

[11] Giovanni Acampora and Matteo Gaeta, "Optimizing Learning Path Selection through Memetic Algorithms", *In. Proc. of the International Joint Conference on Neural Networks (IJCNN 2008),* IEEE 2008, pp. 3869-3875.

[12] Samia Azough, Mostafa Bellafkih and El Houssine Bouyakhf, "Adaptive E-learning using Genetic Algorithms", *International Journal of Computer Science and Network Security,* Vol. 10, No. 7, July 2010.

[13] Chin-Ming Hong, Chih-Ming Chen and Mei-Hui Chang, "Personalized Learning Path Generation Approach for Web-Based Learning", *4th WSEAS Int. Conf. on E-ACTIVITIES, Miami, Florida, USA,* November 17-19, 2005, pp. 62-68.

[14] Huey-Ing Liu and Min-Num Yang, "QoL Guaranteed Adaptation and Personalization in E-learning Systems", *IEEE Transactions on Education, Vol. 48(4)*, 2005, pp. 676-687.

[15] Chengling Zhao and Liyong Wan, "A shortest Learning Path Selection Algorithm in E-learning", *In. Proc. of the Sixth International Conference on Advanced Learning Technologies (ICALT'06),* IEEE 2006.

[16] C. Jiuxin et al, "The Self-adaptive Framework of Learning Object Based on Context", *In Proc. of the International Conference on Computer Science and Software Engineering, Wuhan, Hubei, December 2008,* pp. 941-944.

[17] Hsiao et al., "Ant Colony Optimization for Best Path Planning", *In Proc. of the International Symposium on Communications and Information Technologies 2004, Japan, October 2004*, pp. 109-113.

[18] Nada M. A. Al Salami, "Ant Colony Optimization Algorithm", *UbiCC Journal, Vol. 4(3), August 2009*, pp. 823-826.

[19] Fitness proportionate selection, Available: http://en.wikipedia.org/wiki/Fitness_proportionate_selection, Last accessed 1st March 2012.