Neuro–fuzzy synergism for planning the content in a web–based course

G.D. Magoulas
Department of Information Systems and Computing,
Brunel University, UB8 3PH, United Kingdom
E–mail: George.Magoulas@brunel.ac.uk

K.A. Papanikolaou and M. Grigoriadou
Department of Informatics, University of Athens, T.Y.P.A. Buildings, GR–157 84 Athens, Greece.
E–mail:{spap, gregor}@di.uoa.gr

Keywords: adaptive learning environments, web–based course, adaptive lesson presentation, neural networks, fuzzy logic, learner evaluation, reasoning under uncertainty

Received: January 25, 2000

In this paper, neuro–fuzzy synergism is suggested as a means to implement intelligent decision making for planning the content in a web–based course. In this context, the content of the lesson is dynamically adapted to the learner's knowledge goals and level of expertise on the domain concepts s/he has already studied. Several issues that affect the effectiveness of the lesson adaptation scheme are investigated: the development of the educational material, the structure of the domain knowledge and the assessment of the learner under uncertainty. A connectionist–based structure is proposed for representing the domain knowledge and inferring the planning strategy for generating the lesson presentation from pieces of educational material. The learner's assessment is based on relating learner's behavior to appropriate knowledge and cognitive characterisations and on embedding the knowledge of the tutors on the learning and assessment processes into the system by defining appropriate fuzzy sets. The proposed neuro–fuzzy adaptation scheme is applied to a web–based learning environment to evaluate its behavior and performance.

1 Introduction

Distance Learning through the Web offers an instructional delivery system that connects learners with educational resources. Its main features are the separation of instructor and learner in space and/or time, the use of the educational media/technology to unite instructor and learner and transmit the course content, and the change of the teaching–learning environment from *tutor–centered* to *learner–centered*. The design of a Web–based learning environment includes informed decisions about what comprises the educational content and how this should be sequenced and synthesised, taught and learned. This process is essential in distance learning, where the instructor and the learners typically have minimal face–to–face contact.

The vision of a new generation of web–based learning environments, which possess the ability to make intelligent decisions about the interactions that take place during learning, encourages researchers to look at novel forms of co–operation and communication between tutors, learners, developers and computers and to investigate the technical possibilities for their realisation. Towards this direction, Adaptive Learning Environments (ALE) have instantiated a relatively recent research area that integrates two distinct technologies in computer assisted instruction: Intelligent Tutoring Systems (ITS) and Educational Hypermedia (EH) systems (Brusilovsky 1996). This is in effect a combination of two opposed approaches to computer assisted learning systems: the more directive tutor–centered style of traditional Artificial Intelligence (AI) based systems and the flexible learner–centered browsing approach of an EH system (Davidson 1999).

In this context, the notion of *adaptation* is defined as the concept of making adjustments in the educational environment to accommodate diversity in the learner needs and abilities, in order to maintain the appropriate context for interaction. To this end, in an ALE, the selection, sequencing, and synthesis of educational content takes into account the nature of the content, or task, that is to be taught and also the knowledge level of the learner. The whole procedure is based on understanding the learning and instructional process, as well as the learner characteristics and educational needs (Mc Cormack & Jones 1997).

In general, two methods are proposed in the literature for implementing adaptation in an educational environment: *adaptive presentation*, or *content sequencing*, and *adaptive navigation*, or *link–level adaptation* (Brusilovsky 1996). In the first case, the content of a hypermedia page is generated or assembled from pieces of educational material according to the knowledge state of the learner (Papanikolaou et al. 1999, Vassileva 1997). In the second case, altering visible links to support hyperspace navigation is suggested (Stephanidis et al. 1997, Weber & Specht 1997). Both methods generate a new form of co–operation and communication between learner and system, which is based on the ability of the educational environment to be adapted to the behavior of the learner and make intelligent decisions about the interactions that go on during tutoring. The purpose of adaptation is to avoid information disorientation and overload by presenting the educational material according to the learner's knowledge background and abilities, provide individualised tutoring and, finally, reduce the cognitive effort of learning (Kuhme, 1993).

Many questions are still open in this context. For example, questions related to the role of the tutor as well as of the learner in future ALEs. Furthermore, questions related to the requirements of these systems, the kind of interaction that the learner should have with the system, and the development of appropriate methods of assessing information about the behavior of the learner in the course of learner–system communication. Finally, the organisational and social problems that may arise from the application of these systems have not been investigated thoroughly.

This paper investigates the use of methods from computational intelligence, such as fuzzy logic and artificial neural networks, to handle inexact information about the learner, incorporate tutor's viewpoint into the educational environment and perform lesson adaptation. To this end, a neuro–fuzzy approach is proposed in order to adapt the content of the hypermedia page accessed by a particular learner to current knowledge level, goals, and characteristics of the learner. In this way the educational environment performs adjustments appropriate to each learner by restricting the navigation space in order to protect learners from information overflow. The proposed neuro–fuzzy adaptation scheme incorporates ideas from cognitive science to evaluate learner's knowledge under uncertainty and structure the domain knowledge of the course.

The paper is organised as follows. In Section 2, we present the procedure we have adopted for the development of the educational material. Sections 3, 4 and 5 suggest some novel alternatives to the reasoning and knowledge representation mechanisms in the context of ALE systems, which are based on the use of neuro–fuzzy methods. The connectionist knowledge representation model is presented in Section 3. Section 4 proposes an approach to instructional design that exploits the connectionist-based structure of the domain knowledge. Section 5 suggests neuro–fuzzy synergism to evaluate the knowledge of the learner on already studied concepts of the lesson. In Section 6, applications of these methods in the context of a Web–based learning environment for distance learning are presented. The paper ends, in Section 7, with a discussion and concluding remarks.

2 Developing the educational material

The learning process requires motivation, planning, and the ability to analyse and apply the educational material being taught. In a traditional lecture, the teacher relies on a number of visual cues from the students to enhance and adapt the instructional process. A quick glance, for example, can reveal who is attentively taking notes, pondering a difficult concept, or preparing to make a comment or a question. This type of feedback is missing from a distance learning course and the educational material has to accommodate, in a way, this entire interaction; this can be done, for example, by embedding all the possible questions and common learners' misunderstandings. Therefore, the selection, sequencing, and synthesis of the educational material of a Web–based course must be based on understanding the context of learning, the nature of the content, or task that is to be taught, the instructional objectives, the learners' characteristics, preferences and educational needs, the processes of learning and the constraints of the medium. Consequently, in a Web–based learning environment, the educational material has to incorporate different types of information and levels of explanation, address different learning styles and educational needs.

The following procedure for the development of the educational material has been proposed by Grigoriadou (Grigoriadou et. al. 1999b) and applied for the development of a web-based module named "Introduction to Computer Science and Telecommunications" (DIUA 1999):

- *Create the content outline* based on analysing the audience, defining instructional goals and objectives.
- *Review educational material* that has been proven effective in the traditional lectures.
- *Develop and organise* the educational material following a predefined structure. It is divided into manageable segments: chapters, units, sub-units, and pages. A chapter is a collection of units, while a unit is a collection of pages, tests and (optionally) sub-units. The educational material includes definitions of domain concepts, texts written in a user–friendly way incorporating various levels of explanation, diagrams–images, examples, exercises and simulations and adopts a hypermedia way of presentation.

The presentation of the domain knowledge follows principles that lead to the "deep approach" of learning, that is to relate new ideas to previous knowledge and new concepts to every day experience. Furthermore, this approach aims to organise and structure the content, supplement the theory with a variety of practical tasks and activities and, finally, provide learners with self–assessments and assessments to test their knowledge (Vosniadou et. al. 1996).

However, the greatest challenge in the presentation of the educational material is to build an environment in which the learners are motivated to assess their personal knowledge goals and objectives, and become active participants in the overall learning process. The research literature suggests that the appropriate match of the students to the learning experience has a significant impact on their achievement (Bennett et. al. 1984). Furthermore, instructors need to provide opportunities for students to learn in a way that suits their preferred style of learning (Ellis 1994). Adaptive lesson presentation is a promising research area towards the development of a learning environment in which the learners are motivated to assess their personal knowledge goals and objectives in a way that suits to their learning preferences and knowledge level.

3 Modeling the knowledge of the domain

A key point in producing a learner–adapted system, i.e. a system that meets the individual educational needs and objectives of each particular learner, is to structure the domain knowledge in such a way that it will be possible to do adaptations.

The structure of the domain knowledge is based on symbolic methods and is usually represented as a semantic network of domain concepts, or generally elementary pieces of knowledge for the given domain, related with different kinds of links (Brusilovsky 1996). Alternatively, the use of a concept level hierarchy (Anjaneyuly 1997), or a graph of concepts (Vassileva 1997) has been suggested.

A subsymbolic approach is proposed in this paper: a connectionist–based structure of the domain knowledge is presented that allows the adaptation of the educational material to the individual learner's level of understanding. The subsymbolic approach provides an attractive alternative to traditional symbolic AI methods since it exploits the well known generalisation capabilities of the artificial neural networks to handle the uncertainty in modeling the knowledge of the learner and introduce human–like reasoning (Kasabov 1996). Human–like reasoning aims to support adaptivity and provide ALE with the ability to enhance and adapt the instructional process according to the learner in the human–tutors way. The main characteristic of the proposed approach is that the decomposition of the domain knowledge in modules (see Figure 1), such as knowledge goals, concepts, educational material is incorporated in the connectionist architecture.

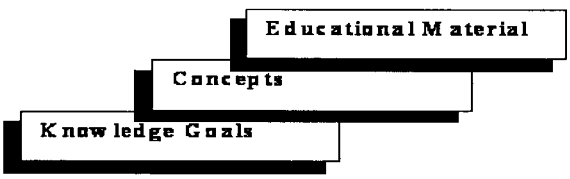

Figure 1: The domain knowledge of the lesson.

An important issue in the development of an environment that will support pedagogical decisions is to provide various types of educational material on the same knowledge (Grigoriadou et al. 1999b). As a first step towards this direction we have mapped the domain knowledge in the three layers of a connectionist model, as shown in Figure 2, with each layer providing a different type of information.

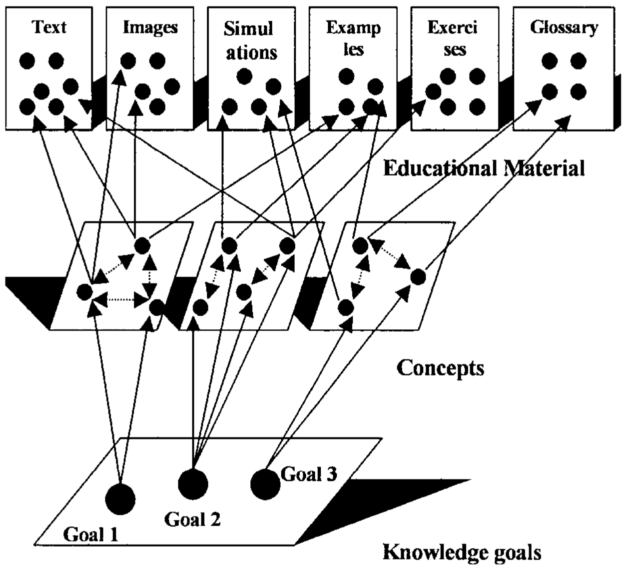

Figure 2: Connectionist–based structure for the domain knowledge.

The architecture is based on the notion of knowledge goals that learners willingly adopt, in an attempt to provide learners with a means of regulating the environment in which they learn. In addition, it gives them the opportunity to select the next knowledge goal according to their educational needs. To this end, in the first layer the knowledge goals, which are referred to a subset of the domain knowledge, are defined while the second layer consists of the concepts of the domain knowledge. In the third layer the educational material 'related to each concept is represented in different categories, such as text, images, simulations, examples, solved and unsolved–exercises and so on.

A knowledge goal, in the first layer, is associated with its corresponding concepts in the second layer. Each concept corresponds to a single concept node of a specially designed dynamic neural network for each goal, named Relationships Storage Network–RSN (Michos et al. 1995). Each RSN is described by:

$$\mathbf{x}(k+1) = sat(\mathbf{T}\mathbf{x}(k) + \mathbf{I}), \tag{1}$$

where $\mathbf{x}$ is a real n–dimensional–vector with components x_i $(i = 1, ..., n)$, which denotes the state or activity of the i–th concept node; $\mathbf{T}$ is a $n \times n$ symmetric weight matrix with real components $\mathrm{T}(i,j)$; $\mathbf{I}$ is a constant vector with real components I(i) representing external inputs; sat is the saturation activation function ($sat(t) = 1$, if $t \geq 1$; $sat(t) = -1$, if $t \leq -1$; $sat(t) = t$ otherwise).

The training of each RSN is performed off–line using groups of patterns that establish relationships among concepts for a knowledge goal and are defined on $\{-1, 1\}^n$. A storage algorithm that utilises the ***eigenstructure*** method is used for specifying the appropriate $\mathbf{T}$ and $\mathbf{I}$ parameter values (Michel et al. 1991). This algorithm guarantees that patterns of concept combinations are stored as asymptotically stable equilibrium points of the RSN (see Michel et al. 1991 for a description of the algorithm). When two or more concepts are active in a pattern, i.e. the corresponding components of the pattern are $\{1\}$, this indicates that a relationship among these concepts has to be established. Relationships among concepts are represented by the internal connection weights (the matrix $\mathbf{T}$). Note that the groups of patterns are generated in accordance to particular strategies for planning the content of the lesson. The human instructional designer has determined these strategies (more details on the planning strategies will be presented in the next section).

The RSN operates in a saturated mode for all discrete points in time, $k = 0, 1, 2, \ldots$, and its equilibrium points are located on the boundary of the $[-1, 1]^n$ hypercube, i.e. the state space is the set of vertices of $[-1, 1]^n$:

$$\begin{aligned}\{-1,1\}^n &= \{\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n \mid \\ &\quad x_i \in \{-1,1\}, \forall\, 1 \leq i \leq n\}.\end{aligned}$$

Note that during operation, the node inputs (initial states) are supplied after evaluating the learner's knowledge on the concepts of a knowledge goal; details on the evaluation procedure will be presented in Section 5. Depending to the pattern applied to the input, the state vector of the RSN is forced to move to a certain part of the concepts' space. In this way, the RSN performs associate inference depending on the input pattern and, unlike the general use of an associative memory, it operates synchronously: (i) it updates the states of its nodes simultaneously, and (ii) the input pattern is kept unchanged until convergence of the network (see Michel et al. 1991 for details on the operation of this type of dynamic neural network).

In Table 1, an example of knowledge goal, referred to the chapter *Network Architectures*, with its associated concepts (the corresponding RSN has $n = 26$ nodes) is exhibited. Note that the *Number* (No) for the outcome concepts in the table indicates their sequencing in the course. Other examples of knowledge goals are: Data Communication Basics, Transmission Means, Network Topology, Local Area Networks, Wide Area Networks, Internet Administration, etc.

In the third layer of the connectionist model, the educational material related to each concept is organised in categories. Weights connecting the second and the third layer are unique for each concept, and each concept may be associated with several categories of educational material. The educational material is then joined under a predefined form of presentation to generate a lesson. The associated concepts receive different characterisations depending on the knowledge goal: some of them, named *outcome* concepts, are fully explained in the HTML pages constructed for the corresponding goal using text, images, examples, exercises etc.; others, named *related* concepts, are simply mentioned in the HTML pages of the goal. These are related to specific outcome concepts but they are not so important for the selected goal. Finally, there are *prerequisite* concepts,

Table 1: The 26 concepts of the Knowledge goal "ISO Architecture".

No	Outcome	Prerequisite	Related
1	Multi-layer architecture	Layer, Communication protocol	
2	Open Systems Interconnection		I.S.O.
3	Physical layer	Transmission means, Synchronisation	Digital trans-mission
4	Data Link layer	Packet, Error detection and correction	Trans-mission means
5	Network layer	Packet routing	Packet
6	Transport layer	Flow control, Traffic metering	Packet
7	Session layer	Synchronisation, Communication half/full duplex	Packet
8	Presentation layer	Compression, Encryption	Compati-bility
9	Application layer	File transfer protocol, Virtual terminal	

which are necessary for the learner to understand the outcome concepts of a goal. Thus, a generated lesson includes:

- complete presentation, in terms of text, images, examples, and simulations (if any), of the outcome concepts;
- links to the main HTML pages of the prerequisite concepts;
- links to the related concepts in a glossary;
- tasks and questions.

4 The instructional designer

The issue of instructional design is of major importance in the development of an educational environment (Burns & Capps 1988, Reusser 1996). In this context, *instructional planning* is the process of mapping out a global sequence of instructional goals and actions that provides consistency, coherence and continuity in the instructional process (Dijkstra et. al. 1992, Vassileva 1995). In our case this can be applied in two levels:

- by *planning the content*; that is, presenting concepts related to the selected knowledge goal by taking into account learner's background knowledge. In this way, the content of a hypermedia page is generated from pieces of educational material based on a goal–oriented way of teaching which is supposed to be adequate to adults who are motivated to learn a specific knowledge goal.

- by *planning the delivery* of the educational material. The planning of delivery is responsible for the optimal selection of the educational material, tutorial activity and presentation style, i.e. the appropriate teaching method. The use of multiple approaches in teaching methods increases the possibilities to meet the needs of a wide range of learners who have different learning styles, time constraints and abilities.

Below, we propose the formulation of the planning strategy retrieval for selecting the content of a knowledge goal in the context of the dynamics of the connectionist network.

To this end, different strategies for planning the content are implemented by means of the stored patterns of the RSN: a *strategy*, in the form of a collection of m patterns defined on $\{-1, 1\}^n$, is stored in the RSN using the eigenstructure method (Michel et. al. 1991); some examples are:

- *Strategy A*: learner achieves a knowledge goal when s/he studies successfully all the outcome concepts of this goal.
- *Strategy B*: learner has successfully studied all the prerequisite concepts of a knowledge goal. Then, in order to achieve this goal, s/he has to study only the outcome and the related concepts.
- *Strategy C*: learner has successfully studied several prerequisite or related concepts of a knowledge goal. Then, in order to achieve this goal, s/he has to study the entire outcome concepts and the rest of the prerequisite and related concepts.
- *Strategy D*: learner has failed in a number of outcome concepts. Then in order to achieve this goal, s/he has to study only these outcome concepts and their prerequisite and related ones.

As mentioned in the previous section, the patterns of relationships are stored as asymptotically stable equilibrium points of the RSN and the network is capable of organising its internal states in accordance to the underlying structure of the stored patterns. During tutoring, the evaluation of the learner's knowledge on the concepts of a goal (this will be described in the next section) formulates the input pattern (initial state vector) of the RSN. The input pattern, during the recall operation of the dynamic network, converges to an equilibrium point, i.e. to one of the m stored patterns that implement a planning strategy. This equilibrium point (i.e. the final state vector) defines the output response of the RSN and is used for deciding the content of the lesson, i.e. present the concepts of the knowledge goal that the learner has to learn next.

The planning of the delivery is also based on the results of the real–time recall operation and establishes instructional objectives (tutorial activities), which are specific steps leading to the knowledge goal attainment.

However, the optimal selection of the educational material should also take into account the relevant importance of each concept for achieving a knowledge goal, as well as the progress of the learner concluding from the learner–evaluating module. Currently, the selection of the educational material is based on weighted priorities, as these are defined by the weight values connecting the second and the third layer of the connectionist network.

5 Evaluating the learner

The assessment of learner's knowledge is based on an *overlay model*. The idea of the overlay model is to represent an individual learner's knowledge of the subject as an "overlay" of the domain knowledge. For each domain concept, an overlay model stores some value (binary/ qualitative measure/probability), which is an estimation of the learner knowledge level for this concept. Overlay models are domain independent and flexible and were originally developed in the area of ITSs and learner modeling (see the work of Wenger (Wenger 1987) for a review on ITSs). In our approach, each concept is associated with linguistic rating values characterising the learner's knowledge, i.e. *{EI, I, RI, RS, AS, S}* = *{ Extremely Insufficient, Insufficient, Rather Insufficient, Rather Sufficient, Almost Sufficient, Sufficient }*. This scale has been experimentally found to provide evaluation results closer to human–tutors evaluation performance, when compared with previous work in the area (Panagiotou & Grigoriadou 1995, Stathacopoulou et al. 1999). The exact rating value is calculated by means of the three–stage evaluation procedure that will be described below.

Learner's knowledge assessment is based on two types of information: answers to questions that evaluate the cognitive part of the learner's knowledge (Nkambou 1999), and measurements that evaluate the behavior part, which is related to awareness, interest, attention, concern, and responsibility factors (Embenson, 1990; Krathwohl et al. 1964). In both cases, several factors contribute to uncertainty in the evaluation procedure, such as careless errors and lucky guesses in the learner's responses, changes in the learner knowledge due to learning and forgetting, and patterns of learner responses unanticipated by the designer of the learner model. Thus, the development of an accurate model for evaluating the learner's knowledge is based on uncertain information.

Various types of questions organised in categories can be used, e.g. multiple choice, fill–in–the–blanks, multiple correct answers. Each question is related to a subset of the domain knowledge that the learner should acquire and should have a weight representing its importance or complexity as regards the evaluation of a knowledge characteristic and ability of the learner (Bertles 1994):

- *Fact oriented questions*: Questions regarding memorising that are used to test the knowledge of concept definitions and topic aspects directly related to the content of the lesson.

- *Higher order comprehension questions*: They are used to test the understanding of the conceptual and semantic units of the lesson. Their aim is to test the conceptual model constructed by the learners and evaluate their misconceptions.
- *Generalisation questions*: Questions examining the ability of comparison, differentiation, abstraction and generalisation.
- *Questions related to the recognition of functional interrelations*: They test the ability of linking newly acquired with already existing knowledge on the concepts.

For example, in the web–based module "Introduction to Computer Science and Telecommunications" offered by the Department of Informatics of the University of Athens (DIUA 1999, Grigoriadou et al. 1999b), identifying comprehension regarding the concepts *data link layer*, *network layer*, and *session layer* is performed using questions like:

- *Which of the OSI layers handles each one of the following functionalities:*
  1. Breaking and transmitting bit streams into frames.
  2. Determining which route to use through the subnet.
  3. Providing synchronisation.

In order to answer the above question the learner has to recall his/her knowledge regarding the *multi–layer structure of the OSI model*. These questions help to test learner's comprehension of the functions undertaken by each layer of the OSI model.

In addition, several measurements are recorded from the learner–educational program interaction and used for evaluating the learner's behavior. For example, the number of questions and exercises that the learner tried to answer or solve, the points scored, the number of learner attempts before giving the correct answer, the frequency of the encountered misconceptions, the number of repetitions of a topic by the learner, the time s/he spends for self–assessment, the type of information the learner prefers (text, pictures, sound, video, simulations, URLs) and how often s/he navigates through the HTML pages of the educational material supplied for a knowledge goal. Thus, by analysing the learner's answers and by processing the various measurements conducted by the system, it is possible to trace gaps in the knowledge of the learner.

To this end, a three stages neuro–fuzzy procedure (see Figure 3), originally proposed in (Panagiotou & Grigoriadou 1995) and extended in (Grigoriadou et. al. 1999a), is applied. The first stage fuzzifies inputs that contribute to the evaluation of the level of understanding based on the estimations of experts to the degree of association between an observed input value and the learner's knowledge on the concepts. Note that a 9–level discretisation of the universe of discourse is applied to the inputs. Depending on the input, a fuzzy subset is generated for each measurement or answer contributing to the evaluation. The next

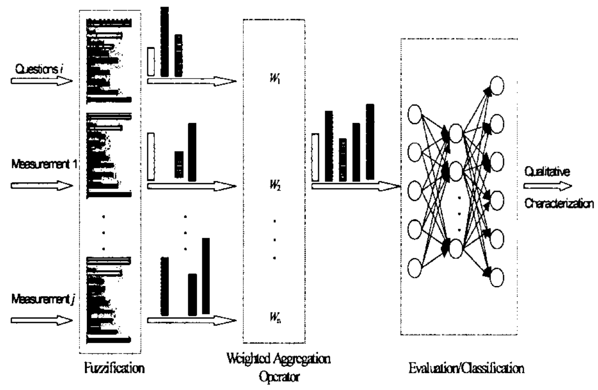

Figure 3: The three stages of the knowledge evaluation procedure.

stage realises a weighted aggregation operation utilising the intersection operator that processes these fuzzy subsets. The weights are evaluated using the Saaty's method (Saaty 1978) and determine the importance of each preliminary decision, expressed by a fuzzy subset, in evaluating the learner's knowledge. The last stage consists of a Multi–Layer Perceptron (MLP) that evaluates the knowledge of the learner with regard to a concept by classifying him to one of the categories $\{EI, I, RI, RS, AS, S\}$. To this end, the max rule is applied to the output vector of the MLP. The MLP is trained using the BPVS algorithm (Magoulas et al. 1997) to imitate the tutors evaluation procedure and can be adapted to a tutor's subjective evaluation procedure.

Depending on the concept, the qualitative characterisation of the learner's knowledge is converted to a numeric value in order to feed the RSN. In this way, an n–dimensional input pattern (initial state vector) is formulated, where n depends on the number of concepts of the knowledge goal and determines the number of the corresponding RSN nodes. This form of feedback from learner evaluation procedure guides the instructional method adopted though the RSN recall operation. Note that, when the learner's knowledge with regard to a concept is characterised as *Extremely Insufficient*, a value of approximately 1 is assigned to the corresponding component of the input pattern. This means that the learner certainly has to study this concept. On the other hand, a small value of approximately 0.1 is assigned when the learner's knowledge on a concept is evaluated as *Sufficient*. The exact magnitude for each value is heuristically chosen and depends on the importance of the concept in achieving a knowledge goal and on its characterisation (outcome, related, prerequisite). For example, the set:

$$\begin{aligned}\mu(x) &= \{1|1 + 0.9|2 + 0.75|3 + 0.65|4 \\ &\quad +0.59|5 + 0.1|6\}\end{aligned}$$

is used for the outcome concept *Multi–layer Architecture*, where an integer value $1, 2, \ldots, 6$ is mapped to a linguistic term $\{EI, I, RI, RS, AS, S\}$ and the symbols " | " and " + " are used only as syntactical constructors. This set can be interpreted as the degree of membership of the learner's knowledge on the concept *Multi–layer Architecture* in each of the fuzzy sets associated with the 6 linguistic terms. Some other examples are the sets:

$$\mu(x) = \{0.98|1 + 0.88|2 + 0.58|3 + 0.38|4 + 0.28|5 + 0.1|6\},$$

for the prerequisite concept *Communication protocol*, and

$$\mu(x) = \{0.95|1 + 0.74|2 + 0.55|3 + 0.24|4 + 0.14|5 + 0.1|6\},$$

for the related concept *Compatibility*.

The three stages neuro–fuzzy procedure allows us to incorporate both general and subjective knowledge in the evaluation procedure. General knowledge is incorporated in the definition of the fuzzy sets and in assigning weights to the different performance parameters, which assess the cognitive and behavior parts of the learner's knowledge. This general knowledge is based on the expertise of the tutor in determining the characteristics of the learner and is kept fixed during the learner's evaluation procedure. On the other hand, the MLP, which constitutes the final evaluation stage, represents the experience of the human tutor in evaluating learners and can be adapted by training to a tutor's personal way of evaluation. Furthermore, this hybrid approach permits the representation and processing of incomplete, imprecise and vague information about the learner, i.e. controversial answers and unstable behavior and the exploitation of the MLP generalisation capabilities.

6 Experiments

The Web offers a new way to receive and disseminate information for educational purposes. The chapter on *Network Architectures* of the Web–based module "Introduction to Computer Science and Telecommunications", (DIUA 1999, Grigoriadou et al. 1999b), offered by the Department of Informatics to first year undergraduate students, has been used for testing the proposed approach. In this chapter adult learners have to study 25 knowledge goals, each one containing 10–30 concepts. Experiments have been conducted to evaluate the behavior of the proposed model in adapting the content of a lesson.

Next, the performance of the neuro–fuzzy approach is illustrated in three cases. This low–level test of the system helps to show how the connectionist model and the knowledge evaluation procedure function together to create an operational system.

In the first case, the learner has selected the goal "ISO Architecture" and his performance has been evaluated as *Sufficient* with regard to several prerequisite and related concepts. The knowledge evaluation procedure supplies the corresponding RSN with an input pattern and initialises the RSN operation; the network will eventually come to rest at one of its equilibrium points. To this end, the RSN runs for several cycles and finally settles into a stable state defined by its 26–dimensional state vector. Since, a localist concept coding is used, i.e. each node stands for one domain concept that is related to the selected goal, it is easy to follow the nodes activity changes in response to an input pattern at each recall cycle. For example, 5 out of the 26 node activity levels are exhibited in Figure 4. Following planning strategy C the generated lesson includes all the concepts of Table 1 apart from the successfully studied prerequisite and related ones: *Layer*, *Data compression*, *Encryption*, *Virtual terminal*, *ISO*, *Traffic metering*.

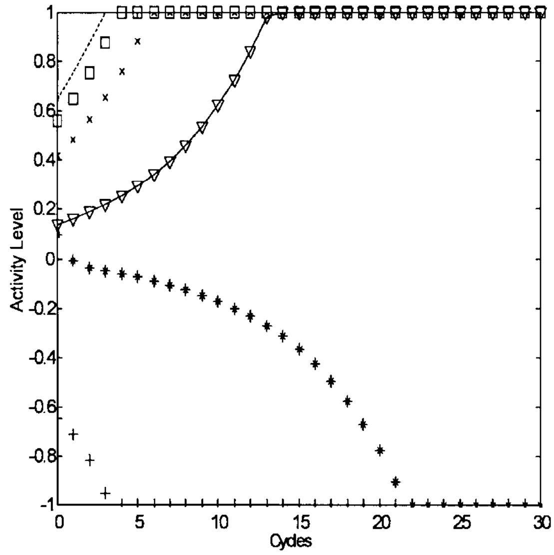

Figure 4: Example of planning strategy C. The concepts: *Communication half/full duplex* ( ), *Compatibility* ( ), *Layer* ($*$), *Network layer* ($\times$), *Packet* ($\triangle$), *Synchronisation* (- - -), *Virtual terminal* ($+$).

□

In Figure 4, it is shown that the activity of the concept node that represents the concept *Virtual terminal* goes to -1, which means that the node is deactivated and the educational material associated with this concept will not be presented. On the other hand, the activity level of the related concept *Compatibility*, in which learner's knowledge has been evaluated as *Insufficient*, goes to $+1$ and the material will be presented. Note that, a node activity level at cycle $= 0$, after transformation to the interval $(0,1)$, is related to the result of the learner's knowledge evaluation procedure for the corresponding concept. Thus, the learner's knowledge has been evaluated as *Insufficient* for the *Compatibility* node, and the value of 0.74 has been assigned to the corresponding component of the input pattern (cf. with this concept's set in Section 5). Similarly, the concept node *Network layer* is activated since the learner has been evaluated as *Rather Sufficient* in this outcome concept.

In the second case, trying to acquire the same knowledge goal, a learner exhibits performance that is characterised as *Extremely Insufficient* with respect to several outcome concepts.

Following planning strategy D, a lesson is generated that presents these outcome concepts, their prerequisite and related ones. The following concepts of Table 1 are included in the lesson: *Multi–layer Architecture, Layer, Communication protocol, Physical layer, Transmission means, Synchronisation.*

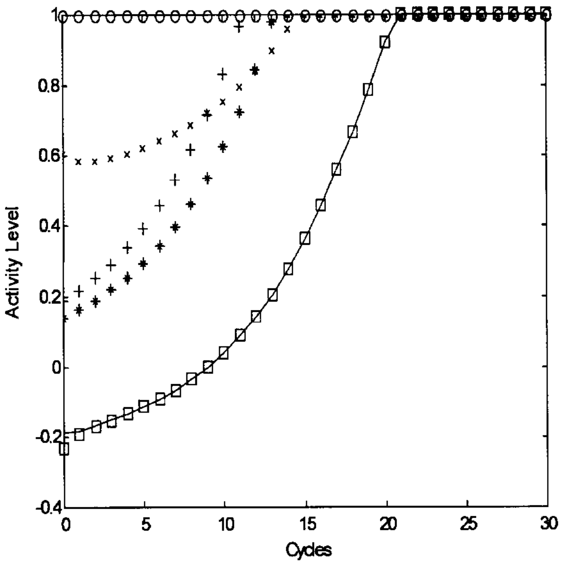

Figure 5: Example of planning strategy D. The concepts: *Communication protocol* ( ), *Layer* ( ), *Multi–layer Architecture* (×), *Physical layer* (○), *Synchronisation* (∗), *Transmission means* (+).

A sample of the RSN's response is shown in Figure 5, in terms of the nodes' activity level. The evaluation of the learner's knowledge with respect to the outcome concept *Physical layer* indicates that s/he has been *Extremely Insufficient*; thus, the corresponding node is rapidly activated. The node of the outcome concept *Multi–layer architecture* is also activated, as well as the nodes corresponding to its prerequisite concepts *Layer* and *Communication protocol*, in which the learner's knowledge has been evaluated as "0.38|*Rather Sufficient*".

The third case is an example of planning strategy B, i.e. a learner with a performance characterised as *Sufficient* regarding all the prerequisite concepts of a goal.

The generated lesson includes, apart from the prerequisite ones, all other concepts of the goal: *Multi–layer architecture, Open Systems Interconnection, International Standards Organisation, Physical layer, Digital transmission, Data Link layer, Network layer, Transport layer, Session layer, Presentation layer, Application layer, Compatibility.* The RSN response in 8 out of the 26 concept nodes is exhibited in Figure 6.

In general, our tests showed that the fuzziness associated with the evaluation of learner's knowledge was handled well by the connectionist model. The performance of the system with 80 learner profiles has been significantly high, almost reaching 100%.

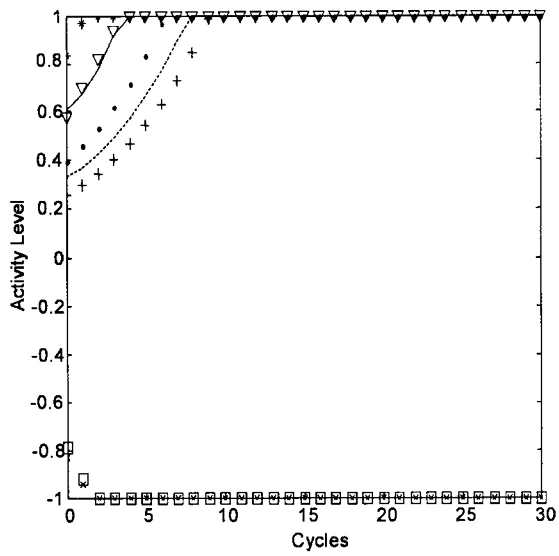

Figure 6: Example of planning strategy B. The concepts: *Application layer* ( ), *Communication half/full duplex* ( ), *Data Link Layer* (∗), *Layer* (×), *Multi–layer Architecture* (△), *Open System Interconnection* (- - -), *Physical layer* (•), *Presentation layer* (+).

7 Discussion and conclusions

Adaptive lesson presentation is a promising method for introducing flexibility into an educational environment. Lesson adaptation aims at minimising the information disorientation and overload of the learner by adapting the educational material to the learner's background knowledge. The design of the educational material supporting such functionality is an important issue contributing to the effectiveness of the adaptation of the overall educational environment.

This paper has described some important parts of the educational environment, which support the adaptivity, how they are designed and how they can be implemented. The paper has focused on two important aspects of developing an adaptive educational environment: the design, i.e. the structure of the domain model and the method of learner's assessment, and the low–level test of the system to show how the two components function together to create an operational system.

An important issue in system adaptation is to adopt a structure of the domain model that will facilitate adaptation of the lesson to the learner's needs. To this end, a specialised connectionist architecture has been developed and a formulation of the planning strategy retrieval for selecting the content of a knowledge goal in the context of the dynamics of the connectionist network has been proposed. This approach may help to accommodate the goal of improving learner's learning process by matching the lesson with a list of stated goals. In each of the goal–based lessons, examples related to the learner's field of experience or explanations in the context with which learner is almost familiar could be provided. For example, learners taking an introductory course might be interested in how it relates to their major course.

In the section about the learner assessment, neuro–fuzzy synergism has been applied to collect information and evaluate what the learner knows about the concepts of the domain. This information usually includes vague, ambiguous, incomplete and some times even contradictory terms; therefore a fuzzy logic–based assessment procedure has been developed. The proposed approach depends on the designer's ability to analyse the cognitive domain suitably, define fuzzy variables and appropriate membership functions for their fuzzy sets by co–operating with experts–tutors, and relate learner's response with appropriate knowledge and cognitive characteristics. The data produced by the learner assessment are only indicative of what the learner may know about the domain concepts. For example, data from the learner assessment might indicate that a learner has little (*Rather Sufficient*) or no knowledge (*Extremely Insufficient*) about concepts of a knowledge goal. These data should be interpreted as meaning that there is only a possibility of this observation being true. Nevertheless, the fuzziness associated with the assessment of the level of understanding seems to be handled well by the connectionist network that makes "decisions" about what concepts should be presented to the learner. Both the output of the network and the results of the assessment procedure can be used by the instructional component to decide when and what kind of educational material to give to the learner. The paper ended with a partial test of the system. A lesson was provided to a learner and the operation of the models was demonstrated. Additional examples, as well as examples of other knowledge goals have been reported in Papanikolaou et al. (2000).

The applicability of the proposed approach can be further extended by exploiting the training and generalisation capabilities of the artificial neural networks to extract information from learner performance parameters, such as the points scored, the time taken etc., to predict trends in performance. These performance patterns could help in deriving instructional strategies to optimally select the appropriate teaching method for each learner.

References

[1] Anjaneyuly K. (1997) Concept level modeling on the WWW. Brusilovsky P., Nakabayashi K., & Ritter S. (eds.) *Proceedings of the Workshop on Intelligent Educational Systems on the World Wide Web*, the 8th International Conference on Artificial Intelligence in Education, Kobe, Japan. Osaka: ISIR.

[2] Bennett N., Desforges C., Cockburn A. & Wilkinson B. (1984) *The Quality of Pupil Learning Experiences*. London: Lawrence Erlbaum Associates.

[3] Bertles K. (1994) A dynamic view on cognitive student modeling in computer programming. *J. of Artificial Intelligence in Education*, 5, 1.

[4] Brulikovsky P.(1996) Methods and techniques of adaptive hypermedia. *User Modeling and User Adapted Interaction*, 6, 2–3, p. 87–129.

[5] Burns H. L. & Capps, C. G. (1988) Foundations of intelligent tutoring systems: an Introduction. Polson M. C. & Richardson J. J. (eds.) *Foundations of Intelligent Tutoring Systems*. London: Lawrence Erlbaum, p. 1–19.

[6] Mc Cormack C., & Jones D. (1997) *Building a Web–based Education System*. New York: Wiley Computer Publishing.

[7] Davidson K. (1999) *Education in the internet – Linking theory to reality*. Available on–line: http://www.oise.on.ca/~ kdavidson/cons.html, accessed December 1999.

[8] Dijkstra S., Krammer H. M. & Van Merrienboer, J. G. (eds.) (1992) *Instructional Models in Computer–based Learning Environments*, NATO ASI Series F, Vol. 104. New York: Springer–Verlag.

[9] DIUA (1999) *Introduction to Computer Science and Telecommunications*, Distance Learning Course, Department of Informatics, University of Athens. URL: http://hermes.di.uoa.gr

[10] Ellis R. (1994) Individual learner differences. Ellis, R. *The Study of Second Language Acquisition*. Oxford: Oxford University Press, p. 471–527.

[11] Embenson S. (1990) Diagnostic testing by measuring learning processes: psychometric considerations for dynamic testing. Frederiksen N. (ed.) *Diagnostic Monitoring of Skill and Knowledge Acquisition*. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

[12] Grigoriadou M., Magoulas G. D. & Panagiotou M. (1999a) A hybrid decision making model for learner evaluation in intelligent tutoring systems. *Proceedings of the 5th International Conference of the Decision Sciences Institute*, Athens, Greece, p. 195–197.

[13] Grigoriadou M., Papanikolaou K., Cotronis Y., Velentzas Ch. & Filokyprou G. (1999b) Designing and implementing a web based course. Chapman G. M. (ed.) *Proceedings of International Conference of Computer based Learning in Science*, Enschede, Netherlands. Czech Republic: Pedagogical Faculty of University of Ostrava, p. H5.

[14] Kasabov N. (1996) *Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering*. Cambridge, MA: MIT Press.

[15] Krathwohl D. R., Bloom B. S. & Masia B. B. (1964) *Taxonomy of Educational Objectives: the Classification of Educational Goals (Handbook II): Affective Domain.* New York: David McKay Co., Inc.

[16] Kuhme T. (1993) User–centered approach to adaptive user–interfaces. *Knowledge-Based Systems*, 6, 4, p. 239–248.

[17] Magoulas G. D., Vrahatis M. N. & Androulakis G. S. (1997) Effective back–propagation training with variable stepsize. *Neural Networks*, 10, p. 69–82.

[18] Magoulas G. D., Papanikolaou K. & Grigoriadou M. (1999) Towards a computationally intelligent lesson adaptation for a distance learning course. *Proceedings of the 11th International Conference on Tools with Artificial Intelligence* , Chicago, USA. Los Alamitos: IEEE Press, p. 5–12.

[19] Michel A., Si Y. & Yen G. (1991) Analysis and synthesis of a class of discrete–time neural networks described on hypercubes. *IEEE Transactions on Neural Networks*, 2, p. 2–46.

[20] Michos S. E., Magoulas G. D. & Fakotakis N. (1995) A hybrid knowledge representation model in a natural language interface to MS–DOS. *Proceedings of the 7th IEEE International Conference on Tools with Artificial Intelligence*, Chicago, USA. Los Alamitos: IEEE Press, p. 480–483.

[21] Nkambou R. (1999) Managing inference process in student modeling for intelligent tutoring systems. *Proceedings of the 11th International Conference on Tools with Artificial Intelligence*, Chicago, USA. Los Alamitos: IEEE Press, p. 19–23.

[22] Panagiotou M. & Grigoriadou M. (1995) An application of fuzzy logic to student modeling. *Proceedings of the IFIP World Conference on Computers in Education*, Birmigham, UK.

[23] Papanikolaou K. A., Magoulas G. D. & Grigoriadou M. (1999) A connectionist approach for adaptive lesson presentation in a distance learning course. *Proceedings of International Joint Conference on Neural Networks*, Washington, USA. CD-ROM Proceedings, IEEE Catalog Number: 99CH36339C, paper # 680.

[24] Papanikolaou K. A., Magoulas G. D. & Grigoriadou M. (2000) Computational intelligence in adaptive educational hypermedia. *Proceedings of International Joint Conference on Neural Networks*, Como, Italy. Los Alamitos: IEEE Press, p. VI-629.

[25] Reusser K. (1996) From cognitive modeling to the design of pedagogical tools. Vosniadou S., De Corte E., Glaser R. & Mandl H. (eds.) *International Perspectives on the Design of Technology–Supported Learning Environments.* New York: Lawrence Erlbaum Associates, p. 81–103.

[26] Saaty T. L. (1978) Exploring the interface between hierarchies, multiple objectives and fuzzy sets. *Fuzzy Sets and Systems*, 1, p. 57–68.

[27] Stathacopoulou R., Magoulas G. D. & Grigoriadou M. (1999) Neural network–based fuzzy modeling of the student in intelligent tutoring systems. *Proceedings of the International Joint Conference on Neural Networks*, Washington USA. CD-ROM Proceedings, IEEE Catalog Number: 99CH36339C, paper # 679.

[28] Stephanidis C., Paramythis A., Karagiannidis C. & Savidis A. (1997) Supporting interface adaptation: the AVANTI Web–browser. Stephanidis C. & Carbonell N. (eds.) *Proceedings 3rd ERCIM Workshop on User Interfaces for All*, Obernai, France.

[29] Vassileva J. (1995) Reactive instructional planning to support interacting teaching strategies. *Proceedings of the 7th World Conference on AI and Education*, Washington, USA. Charlottesville: AACE, p. 334-342.

[30] Vassileva J. (1997) Dynamic course generation on the WWW. Brusilovsky P., Nakabayashi K. & Ritter S. (eds.) *Proceedings of the 8th International Conference Artificial Intelligence in Education*, Kobe, Japan. Osaka: ISIR, p. 498-505.

[31] Vosniadou S., De Corte E., Glaser R. & Mandl H. (Eds.) (1996) *International Perspectives on the Design of Technology–Supported Learning Environments.* New York: Lawrence Erlbaum Associates Publishers.

[32] Weber G. & Specht M. (1997) User modeling and adaptive navigation support in WWW–based tutoring systems. *Proceedings of the 6th International Conference on User Modeling*, Chia Laguna, Sardinia, Italy. Also in Jameson A., Paris C. & Tasso C. (eds) *User Modeling*. Wien: Springer–Verlag, p. 289–300.

[33] Wenger, E. (1987) *AI and Tutoring Systems: Computational and Cognitive Approaches to the Communication Knowledge.* California: M. Kaufmann Publishers, Inc, p. 20.