Chapter 5
Evaluation of Student Knowledge Using an e-Learning Framework

Margarita Favorskaya, Yulya Kozlova, Jeffrey W. Tweedale and Lakhmi C. Jain

Abstract This chapter introduces the concept of adding a fuzzy logic classifier to e-Learning framework. This conceptual model uses Fuzzy Logic Evaluation Sub-systems (FLESs) to implement "Theory of probability" curriculum. The customized sub-system is used to dynamically evaluate student knowledge. It is essential that the FLES-PRobabilty (FLES-PR) capture's the students' interest to maintain their motivation and increase the effectiveness of the learning experience. Given that interactive systems increase the education efficiency and the individual abilities of student, the routine actions of teacher are must be delegated to e-Learning system. In this chapter, artificial intelligence concepts, techniques, and technologies are used to deliver the e-Learning requirements. For instance, a fuzzy logic scheme is created to evaluate student knowledge when using the FLES-PR. The curriculum is delivered using two FLES modules instantiated using the "Matlab" 6.5 fuzzy toolbox environment. Each sub-system provides structured lessons, representing topics, content, and additional contextual parameters. The FLES is designed to gain the students attention, highlights the lesson objective(s),

M. Favorskaya (✉) · Y. Kozlova
Institute of Informatics and Telecommunications, Siberian State Aerospace University,
31 Krasnoyarsky Rabochy, Krasnoyarsk 660014, Russian Federation
e-mail: favorskaya@sibsau.ru

Y. Kozlova
e-mail: ivt_yulya@sibsau.ru

J.W. Tweedale
University of South Australia, Mawson Lakes, SA 5095, Australia
e-mail: Jeffrey.Tweedale@unisa.edu.au

J.W. Tweedale
Defence Science and Technology Organisation, PO Box 1500, Edinburgh,
SA 5111, Australia

L.C. Jain
Faculty of Education, Science, Technology and Mathematics, University of Canberra,
Canberra, ACT 2601, Australia
e-mail: lakhmi.jain@unisa.edu.au

© Springer International Publishing Switzerland 2015

D. Sharma et al. (eds.), *Fusion of Smart, Multimedia and Computer Gaming Technologies*,
Intelligent Systems Reference Library 84, DOI 10.1007/978-3-319-14645-4_5

stimulates recall of prior knowledge, and progressively elicits new material to guide increased performance by providing feedback using benign assessment to enhance retention. The proposed evaluation system is designed as a functioning plug-into the universities "Moodle" server to leverage from the existing course management, learning management and virtual learning environment. It also recommends the pace and complexity of learning as the student progresses through the curriculum. This chapter case study discusses the success of the FLES-PR software tool and explains how it has been validated against the manual results of three human experts.

Keywords Concept mapping · e-Learning · Fuzzy logic · Knowledge evaluation · Interactive system · Higher education

5.1 Introduction

Social media and virtual community applications continue to enhance computer literacy. The growing awareness of information streaming is also being accelerating with the evolution of the Bring Your Own Device (BYOD) products. Similarly, this new generation of tech-savvy students are now demanding better access to education remotely and at their convenience. The rapid growth of web resources and services has had a major influence on the education community. The current generation of students have the daily skills and knowledge to access many software tools, web resources, and social networks [1–3]. The Computer-Based Training (CBT) has evolved to include richer sources of multi-media and interactive learning material. More recently, the Intelligent Tutoring System (ITS), the CBT systems, and adaptive hypermedia have been widely used to development web-based courses [4]. These systems and individual motivational drivers are central to providing distance; a remote and external learning Serrano-Cámara et al. classify motivation as follows [5]:

- **Intrinsic** motivation is an important phenomenon and is used to achieve high-quality learning results.
- **Extrinsic** motivation is promoted through specifically identified internal regulation. The course goals are aimed at promoting increase engagement.
- **Psychology** motivation is focused on proving rewards that separate the tendency to avoid regulation for fear of punishment.
- **Amotive** motivation results in a fluctuating level of participation based on episodes of apathy or socially derived distractions.

The e-Learning systems provide easy access to learning material in a safe and flexible environment. Given on-line access, these knowledge warehouses now provide prospective students with learning opportunities any time and anywhere. This approach to learning attracts new students and aids in retaining current students. The on-going development of the BYOD products and wireless communication technology continues to provide more flexible or mobile learning (m-Learning). The same technology also facilitates group technology enabling, face-to-face or collaborative learning in a virtual environment [6]. This technology is also spreading to support industry and virtual conferencing [7]. These trends will continue as technology facilitates increased capability, therefore researchers must endeavour to provide more robust means of scalable assessment using methods like FLES.

This chapter is structured as follows. Section 5.2 provides a brief literature review in e-Learning systems while Sect. 5.3 includes a discussion about the conceptual model. Section 5.4 presents a the fuzzy logic classification scheme used to evaluate student knowledge. Section 5.5 describes FLES-PR system. Section 5.6 discusses the results of the test and validation process using three subject matter experts. Finally, Sect. 5.7 provides a summary and possibilities for future development.

5.2 Background

The e-Learning is not new, however it has shaped the way teachers deliver learning opportunities to students. Typically teachers try to demonstrate respect for their students, while building on their strengths by using their preferred learning styles in a classroom [8]. Achieving this remotely or on-line is often problematic and often isolates or delays feedback about the students learning style. Computers provide a one-to-one interface that provides feedback based on the software that operates the application. Pre-defined computer-based interactions must be resourcefully encoded using innovative applications that need to simulate the ability to work cooperatively and negotiate individualised learning. Pedagogical frameworks have evolved to support the ITS and the CBT using computer software [9]. These systems focused on specific learning goals by providing a digital representation (taxonomy) of the syllabus. The core activities were designed to facilitate [10]:

- Student-centred learning.
- Fostered competency-based expectations.
- Aligned curriculum objects and automated assessment.
- Invokes evidence-based decision-making.
- Provides targeted and structured instruction.
- Ultimately provides a safe, supportive, inclusive learning environment.

Regardless, a curriculum must be ordered (using a watershed design) and contain a full set of lessons relating to a specific topic. Stockinger and De Pablo believe that the Computer Assisted Learning (CAL) will contain components to implement a variety of remote training methods. These could include: structured warm-up activities, alternative presentation skills, reading material, virtual demonstrations, multi-media aids, note-taking, discussion, case studies and questionnaires [11].

They suggest using tools like; Authorware (Macromedia), IconAuthor (Asymetrix), and CourseBuilder (Discovery Systems) to assist in optimising the CBT content.

Conati and Manske reported the performance of the tools and methods used within the CBT ITS rely on the Artificial Intelligence (AI) techniques. They concluded that knowledge representation and programming techniques would be used to focus on delivering instructional content and support that is tailored to the individual [12]. Goldberg et al. recently identified a pedagogical design that included five adaptive training elements. These include: curriculum, instructional strategy, performance measures, pedagogical interventions, and a student data model [9]. Their Generalized Intelligent Framework for Tutoring (GIFT) system was based on the traditional sense, decide and action cycle. It also included a learning management system to maintain the curriculum and on-going student competency.

The e-Learning systems are more expensive to create than traditional curriculum; however, they are gain significant traction because they service people, who have issues attending conventional classroom training. They support learners, who are geographically dispersed, constrained by family commitments, limited by transportation, mobility, geopolitical conflict and even personality. “e-Learning can be defined as the use of computer to deliver a broad array of solutions to enable learning and improve performance” [13]. It can be tailored to deliver cognitive, interpersonal, and psychomotor skills across many domains. The delivery can also be configured to support self-paced, instructor, facilitator, and even “blended learning” [14]. Each application will typically provide components with content, tutoring, coaching, mentoring, and collaborative functionality to support the virtual classroom. A number of open source systems have evolved e-Learning Management System (e-LMS)[1] such as eFront,[2] Moodle,[3] Dokeo,[4] Claroline,[5] Ilias,[6] Sakai,[7] and Olat.[8] Moodle is being adopted by many universities including the Siberian State Aerospace University and the University of South Australia. These tools are distributed with a repository of tools to add functionality. There is also a growing community of practitioners, who are developing add-on components to extend in-built functionality. This should be modularised objects, learner-centred, provide a granular approach with engaging content that is interactive and tailored to the individual (personalised).

Wiley describes the content of e-learning in terms of atoms or manageable “chunks” called learning objects [15]. He describes how, until recently, a Lego block was mistakenly used as a metaphor to explain the simplicity of learning

[1] www.openelms.org/.

[2] http://www.efrontlearning.net.

[3] https://moodle.org.

[4] http://www.dokeos.com.

[5] http://www.claroline.net.

[6] http://www.ilias.de.

[7] http://www.sakaiproject.org.

[8] http://www.olat.org.

objects. Govindasamy also suggests this approach is to simplistic and should be avoided [16]. Both state that Lego blocks are distinctly different from learning objects because [15]:

- Any Lego block can be joined with any other Lego block.
- Lego blocks can be assembled in any manner.
- Assembling Lego blocks are simple for children to assemble.

Unfortunately atoms are more complex because they don't all fit together, they are only attracted to atoms with mating structures and special skills are required to achieve the outcome. The e-Learning systems simplify this process but providing a collection of tools in component form that free course to focus on delivery. Moodle was originally made for education, training, and development environments to help educators create on-line courses that would focus on interaction and collaboration. There are over 50,000 registered users and the framework has recently been extended to support business. The environment is categorized based on its intended use. The key elements include the:

- Course Management System (CMS).
- Learning Management System (LMS).
- Virtual Learning Environment (VLE).

Moodle runs without modification on Unix, Windows, MacOS, and many other systems that support PHP scripting language and a database compliant with the Sharable Content Object Reference Model (SCORM)[9] and the Aviation Industry Computer-Based Training Committee (AICC)[10] standards. However, its installation requires certain technical proficiency of PHP technology [13]. It provides a suitable platform for researchers to improve components that enhance delivery. This chapter uses the Matlab fuzzy library[11] to design and test the Fuzzy Logic Evaluation Sub-systems—PRobability (FLES-PR) to implement a curriculum focused on "Theory of probability".

5.3 Existing Research

The volume of literature pertaining to the development of the CBT and e-Learning systems is substantial; therefore the authors focus on the key topics relating to developing the concept model and evaluation using fuzzy logic. Chrysafiadi and Virvou conducted an excellent review on this topic. It spans the past decade and catalogues the mental models used to represent students digitally [17]. These

---

[9] http://scorm.com/scorm-explained/.

[10] http://www.aicc.org/joomla/dev/.

[11] For Mathworks fuzzy logic toolbox documentation, see http://www.mathworks.com.au/help/fuzzy/functionlist.html.

commonly include: overlay, stereotype, perturbation, machine-based, cognitive, constraint-based, Bayesian, predictive, fuzzy and ontological student models. The most popular approaches uses the stereotypes, overlay, fuzzy, Bayesian networks, predictive model, and cognitive theory. For instance:

- The Adaptive Hypermedia System that integrates a Thinking Style (AHS-TS) using the overlay student model demonstrated by Mahnane et al. [18].
- A web-based adaptive learning system called PDinamet designed by Gaudioso et al. was used to teach physics in a secondary education [19].
- The ITS called DEPTHS combining a stereotype and an overlay model with the fuzzy rules [20] was designed to help students learn software design patterns.
- The INQPRO system, which predicts the acquisition of scientific inquiry using Bayesian networks [21].
- Finally Peña-Ayala et al. discuss a predictive model that can dynamically build the cognitive maps to set the fuzzy-causal relationships among the lecture's option properties and the student's attributes in web-based educational systems [22].

Cline et al. [23] proposed the ITS using a Concept Mapping Tool (CMT). This system automatically evaluates performance based on the maps provided by the instructor of a course. A typically the CMT includes the following components:

- An Applet written in Java language to assist the designer to draw the concept maps.[12]
- HTML Web pages used to dynamically generated content by using Java ServerPages.[13]
- A DBMS—in this case, using "MySQL".[14]
- An Expert System to implement the rule-based evaluation system such as JESS.[15]

The e-Learning system must be capable of monitoring and sharing knowledge successfully to be effective. Desmarais and Baker discuss a student's meta-cognitive model [24], and Fiorella et al. [25] show how planning, monitoring, and evaluation can be used to highlight how high levels of meta-cognitive function effect an individual learning (including e-Learning environment). Peña-Ayala also used activity theory to design the Adaptive e-Learning System (AeLS) [26]. The proposed framework for the AeLS includes: principles, architectures, and dynamic perspectives associated with activity theory within e-Learning system. The architecture considers three types of activities at various levels such as individual, collective, and network. To learn more, Desmarais and Baker provide a useful review of intelligent learning environments [27]. These systems all interact with students to

[12] See docs.oracle.com/javase/tutorial/deployment/applet/.

[13] See www.oracle.com/technetwork/java/javaee/jsp/.

[14] See www.mysql.com.

[15] See www-lium.univ-lemans.fr/lehuen/master1/clips/jess/jess61RC1man.pdf.

provide a customized (individual) learning experience. The first challenge is to transform the curriculum into a digital format. Most rely on existing ontologies, rules, and structures similar to those systems listed above. Lesson or topic information is initially populated by subject matter expert knowledge (typically experienced lecturers). When deployed, the delivery is customized based on individual performance parameters that evolve over time. In this chapter, the authors show how the FLES framework can be used to analyze, model, and study the human praxis and how the "Moodle" AeLS is used to deliver the "Theory of probability" curriculum.

Most Artificial Intelligence (AI) applications use classifiers to either divide (segregate data) or control the flow of learning parameters. They can also be used to pre-process data into categories, prior to inferring any further action(s) [28]. The most popular techniques include: random forest, Neural Networks (NNs), Support Vector Machines (SVMs), k-nearest neighbour, Bayes, Decision Trees, and Fuzzy Logic. The system designer will select, which technique is appropriate based on the desired performance, efficiency, and the type of data being prosecuted [29].

In 1965 Zadeh defines the fuzzy logic as a form of logic based on graded or qualified statements in lieu of crisp values that are strictly true or false [30, 31]. In 1972, Zadeh introduced conventional techniques to analysis behaviour that is strongly influenced by human judgment, perception, and emotions [32]. Fuzzy logic enables membership to partial sets in lieu of existing crisp variables or non-membership. Zadeh believed that people operate using linguistic variables to manage groups of crisp sets to represent infinite valued logic [33]. For instance, when driving a car, they go faster or slower. They accept noise or imprecise inputs using membership functions to approximate a range of crisp sets. Sugeno subsequently implemented a fuzzy inference system capable of goal reaching functionality that was reviewed by error data at each range interval [34]. Fuzzy logic has been used to solve a variety of problems [35] such as classification [36], modelling [37] industrial controllers [38], and e-learning, for instance:

- Bai and Chen [39] used fuzzy logic to create an automated method for grading.
- Jia et al. [40] created adaptive learning system based on fuzzy set theory
- Taylan and Karagözoglu [41] developed the Adaptive Neuro-Fuzzy Inference System (ANFIS) to evaluate the Student Academic Performance (SAP) using a Sugeno fuzzy model to achieve clustering.
- Dias and Diniz [42] developed the Fuzzy Quality of Interaction (FuzzyQoI) with three cascading sets of five fuzzy inference systems.
- Lo et al. [43] proposed an adaptive web-based learning system based on students' cognitive styles.
- Millán et al. [44] created a Bayesian network to student model engineering.
- Tourtoglou and Virvou [45] used Simulated Annealing to promote collaborative learning between both for trainers and trainees using UML.

The FLES application uses fuzzy logic to evaluate student knowledge based on the linguistic variables, fuzzy membership functions, and a set of fuzzy rules.

5.4 Knowledge Evaluation Based on Fuzzy Logic

Membership functions are used for representation of linguistic groups of crisp sets. Fuzzy modelling is considered as a process of applying to specific systems or problems. In recent years, fuzzy modelling has increasingly been used to solve control and decision-making problems. A traditional fuzzy logic system differs from a traditional control system, when it uses knowledge provided by experts to build rules that express a logic using linguistic variables. A fuzzy variable is described by a set of three parameters (α, X, A), where α is a name of fuzzy variable, X is a set of α definition, and A is a fuzzy set on X set describing a restriction $\mu_A(X)$ on value of fuzzy α variable [46]. Here, a linguistic variable (β) is a set comprising of four tuples (β, T, X, G, M), where T is a defined set of fuzzy linguistic variables (term-set) with X as a definition area, G is a syntactic procedure used to work with elements of the T term-set (for example, a generation of new terms) [47], and M is also considered a semantic procedure, which is used to transform each crisp value to a linguistic variable as a result of G.

A fuzzy control system includes three main stages. These stages include fuzzification, inference (decision making unit and its contextual rules) and defuzzificaiton as shown in Fig. 5.1:

- **Fuzzification** involves a transformation of crisp input values into linguistic variables. It is possible that a value could be owned by several sets although it will have a degree of membership, based on rules, and, like the tipping problem, the maximum value is chosen.
- **Fuzzy Inference** is realized using a pre-conceived set of rules to derive a linguistic output based on the degree of membership. The concept is shown in Fig. 5.2. Given two inputs are applied to three deciding tuples. In parallel, a single crisp output can be generated.
- **Defuzzification** transforms the fuzzy derivatives to a crisp output based on the membership assigned during the inference process. The simplest transformation

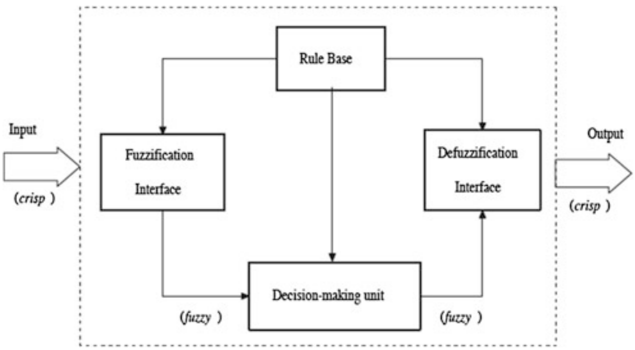

Fig. 5.1 Basic configuration of fuzzy logic system [48]

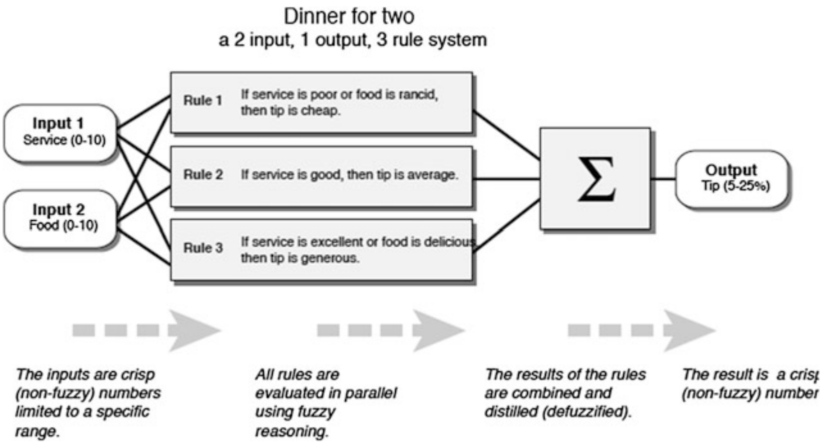

Fig. 5.2 Basic scheme of fuzzy inference [48]

is typically assigned the "first maximum". Common alternate variants could use the "center of gravity", "middle maximum", or a "de-fuzzified height".

The design of fuzzy logic system involves creating one or more sets containing linguistic variables and their associated rules. The direct approach employs expert estimations although computational techniques can be used to determine the relative frequency of data elements and possible membership values. An expert will typically structure rules based on own understanding and how other might perceive them. The FLES provides a knowledge evaluation system based on fuzzy logic to control learning tasks. A feedback system is included to direct the flow and provide estimates of the student success while learning. The conceptual scheme used to develop this knowledge evaluation system is presented in Fig. 5.3.

This system implements three interfacing strategies based on fuzzy logic such as task selection (Sect. 5.4.1), evaluation (Sect. 5.4.2), and prompts to help students acquire knowledge (Sect. 5.4.3).

5.4.1 Strategy for Task Selection

The goal of the strategy for task selection is to maximise the system efficiency, when determining the next element of the curriculum the current student needs to complete based on the subject set. This efficiency can be defined using two factors. They include the students' progression and total size of the topic sub-set. Therefore, the quantity of topic sub-set needs to be balanced by both size and difficulty. The level of tasks should progressively escalate in order for the student to save time and maximise their effort. Similarly the set should be structured to enable the system to estimate a students' knowledge with minimal repetition. Therefore, a smaller set of

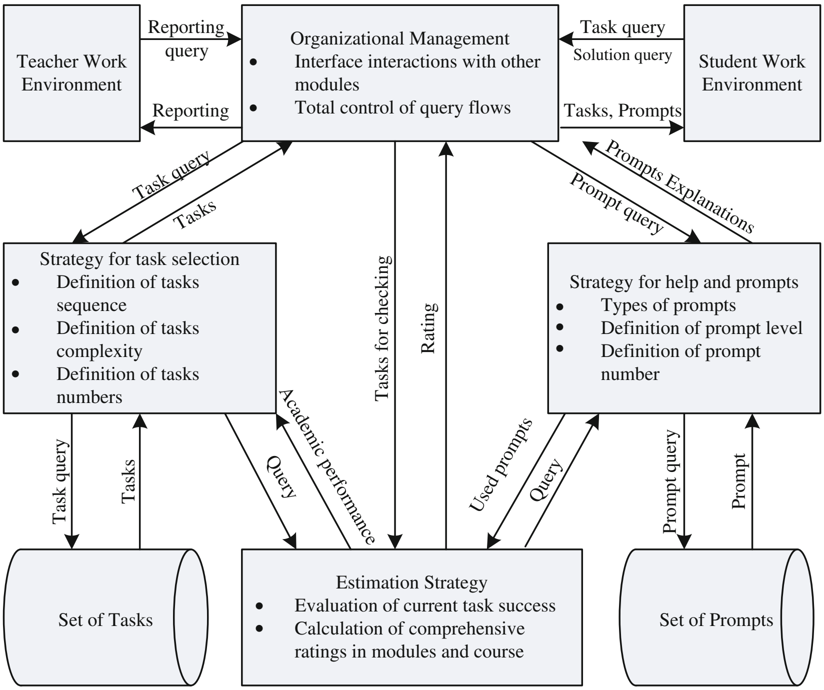

Fig. 5.3 The conceptual scheme of knowledge evaluation system

complex tasks can be presented to successful students, while a more extensive set of tasks that gradually increases in complexity is offered to less successful students.

The strategy of tasks selection involves creation of tasks sequence, assignment of complexity level of current task, and definition of numbers of tasks. The teacher makes a decision on relevance of task selection in current topic, course module, or whole course. This strategy is implemented by a set of rules reflected the methodology, which is used by the teacher.

As an example set of rules to define the strategy that defines the level of complexity for a specific task is explained using the following five rules:

IF (number_successful_scores = “small”) THEN (student_status = “unknown”).
IF (number_non-successful_scores = “large”) THEN (student_status = “non-successful”).
IF (number_successful_scores = “large”) OR (number_non-successful_scores = “small”) THEN (student_status = “successful”).
IF (student_status = “non-successful”) OR (student_status = “unknown”)THEN (level_current_task = “simple”).
IF (student_status = “successful”) THEN (level_current_task = “complex”).

During this definition, the tasks’ sequence is also controlled to ensure the student builds concrete knowledge. This is achieved using the following fuzzy rules:

IF (student_solves_current_tasks = "poor") THEN (current_task = "back_to_ previous_topic").
IF (student_solves_current_tasks = "good") THEN (current_task = "go_to-following_topic").

Two linguistic variables (good and poor) are used to test the students' ability to solve the current task. Membership for the current task is sequentially assigned based on the values "poor" and "good". These values are used to guide the system to proceed with the next topic or return to the previous topic to re-enforced the concept being taught, for instance:

IF (student_solves_current_tasks = "poor") THEN (number_tasks = "increase").
IF (student_solves_current_tasks = "good") THEN (number_tasks = "decrease").

5.4.2 Evaluation Strategy

The evaluation strategy involves two sub-strategies: the evaluation of current task and the comprehensive evaluation of course module. In the first strategy, each type of task requires the own evaluation procedure. Where the membership is a degree of the students answer to the correct answer as a result of current task evaluation. Therefore, the following evaluations may be applied to different types of elements mentioned in Sect. 5.3:

- Menu—a simple enumeration of menu items.
- Calculation—an estimation in any metric, for example, Euclidean.
- Word—an enumeration of answer variants or a semantic evaluation by usage any metric.
- Sentence—a semantic evaluation by use of any metric.
- Equation—a number of operations required for correct equation transform.
- Correspondence—a number of correspondences.
- "Hot" points—an estimation in any metric, for example, Euclidean.
- Sequence—the result can be presented as hierarchical oriented graph (tree); therefore, a nodes and edges superposition of graphs.
- Hypertext—a text semantic evaluation.
- Sound—can be applied previous evaluation types according of task context.

Each of estimations mentioned above includes a number of task and fuzzy membership degree to correct answer.

In the second strategy, it is required to build a comprehensive evaluation of course module. However, the use of fuzzy evaluations for current tasks is not suitable for receiving the clear teacher score. Let us introduce the following set of fuzzy rules:

IF (number of correct answers = "large") THEN (score = "high").
IF (number of correct answers = "small") THEN (score = "low").
IF (number of prompts = "large") THEN (score = "low").
IF (number of prompts = "small") THEN (score = "high").

As a result, a fuzzy set will be received instead of fuzzy numbers. Therefore, a correspondence between a fuzzy set and a fuzzy number of correct answers ought to be defined. It may be a simple sum of memberships or weighted sum of memberships for all tasks provided by Eq. 5.1, where x_i is a ith task in a module, $\mu(x_i)$ is a membership of ith task answer to correct answer, k_i is a weighted coefficient of ith task, N is a task number in a module, μ is a membership of fuzzy set $\mu(x)$ to large number of solved correctly tasks.

$$\mu = \frac{\sum_{i=1}^{N} k_i \, \mu(x_i)}{N} \tag{5.1}$$

A membership of fuzzy set $\mu(x)$ to a small number of solved correctly tasks $\bar{\mu}$ is determined by Eq. 5.2.

$$\overline{\mu} = 1 - \mu \tag{5.2}$$

In complex cases, the multi-dimension fuzzy set can be introduced. The dimension of such set will be a number of tasks, and a membership function will define its membership to large number of solved correctly tasks. Based on the received estimations, fuzzy rules, and linguistic variables, a fuzzy module evaluation can be executed.

5.4.3 Strategy of Help and Prompts

The strategy of help and prompts includes the definition of prompt types, levels, and numbers. On the one hand, such prompts ought to be understandable for a student and help to solve the task. On the other hand, they could not include an explicit answer. Such prompts are generated by experienced teaches, for example:

IF (number_prompts = "small") THEN (prompt = "non-deep").
IF (number_prompts = "large") THEN (prompt = "deep").

Also non-fuzzy rules can be introduced as a partial case of fuzzy rules, for example:

(number_prompts < 6).
IF (number_prompts = 6) THEN (show_solution).

Based on fuzzy rules and linguistic variables, the prompt types, levels, and numbers can be determined.

5.5 Conceptual Model of Course "Theory of Probability"

Let us consider the practical realization of course "Theory of probability" reducing it by three topics for better representation of test results processing such as module 1 "Classical definition of probability", module 2 "The basis of probability theory", and module 3 "Theorems for probability calculus". Each topic is a separate module with own five test tasks. The tasks are presented for a student with increased complexity. A student can ask a prompt.

In module 1 "Classical definition of probability", the basis of combinatorics is learned, and a definition of probability is introduced. The skills provided by this module are used in other modules. Five following tasks are represented into it:

Task 1. Any two-digit number is thought. It is required to find a probability that a random two-digit number is a pre-determined number.
Task 2. Two dice are broken. It is required to find a probability that a digit sum in the shown sides is even, and a digit "six" will be at least on one of two dice sides.
Task 3. Let 5 cards are randomly extracted from a 36 card batch. It is required to find a probability that all cards have red color.
Task 4. Let 5 cards are randomly extracted from a 36 card batch. It is required to find a probability that two aces are among them.
Task 5. Let 5 cards are randomly extracted from a 36 card batch. It is required to find a probability that one jack is among them.

In module 2 "The basis of probability theory", the main terms are introduced such as a space of elemental events, an elemental outcome, an event of probability, an impossible event, etc. The main combinations from elemental events are studied. In tasks 1–5, it is required to introduce the elemental events and state event A through them. Let a rifleman had three shots.

Task 1 A = {"A rifleman had one successful shot"}.
Task 2 A = {"A rifleman had two successful shots"}.
Task 3 A = {"A rifleman had at least one successful shot"}.
Task 4 A = {"A rifleman had at least one unsuccessful shot"}.
Task 5 A = {"A rifleman had at least two successful shots"}.

In module 3 "Theorems for probability calculus", the theorems of probabilities adding and multiplying, a term of conditional probability, the dependent and independent events are introduced. It is required to study modules 1 and 2 for successful learning of module 3. In tasks 1–5, it is necessary to express an event probability A through probabilities of elemental outcomes. Let a rifleman have three shots. The probability by one successful shot is equal 0.7. Other shots are independent events.

Task 1 A = {"A rifleman had one successful shot"}.
Task 2 A = {"A rifleman had two successful shots"}.
Task 3 A = {"A rifleman had at least one successful shot"}.

Task 4 A = {“A rifleman had at least one unsuccessful shot”}.
Task 5 A = {“A rifleman had at least two successful shots”}.

A student introduces a symbolic expression, which is a main formula for a task. Then a calculation is started. The summarized answer is a probability value or the used formula, if a calculation is not required.

For all represented tasks, three levels of prompts are supported:

- Level 1. The topic, for which a current task is situated, is pointed for a student. In most cases, this is enough. A student learns the topic and examples and solves a task.
- Level 2. The main formulas for current task solving are prompted for a student. A student learns them and solves a task.
- Level 3. A full solving of task with comments is represented for a student.

For tasks evaluation, the following test scheme was suggested in order to compare the results provided by Fuzzy Logic Evaluation System (FLES) with the experts’ evaluation. Three FLESs were introduced for increasing of results reliability:

- The first level FLES (FLES1). A single task is estimated by a pre-determined teacher score, if a task was solved correct. A score is decreased, if errors took place or prompts were used.
- The second level FLES (FLES2). A module is estimated by the own scaled score. A module evaluation is not a simple addition of tasks scores. A module is estimated as a comprehensive and finite set of knowledge.
- The third level FLES (FLES3). A whole course is estimated as a finite set of knowledge, and such evaluation also is not a simple addition of all modules scores.

The proposed approach for knowledge evaluation is differed from the used systems in order to evaluate the weak formalized fuzzy factors as an attempt for simulation of expert evaluations.

5.6 Fuzzy Logic Evaluation System-PRobability (FLES-PR)

Our efforts were directed on FLES creation into “Matlab” 6.5 environment [49]. As an example, the course “Theory of Probability” was used. The FLES-PR was designed as a hierarchical system of fuzzy derivation, which is presented in Fig. 5.4.

The FLES-PR system includes three levels. The fuzzy controller of first level FLES1 estimates a quality of the solved tasks and analyzes the errors and the used prompts. The fuzzy controller of second level FLES2 provides a quality evaluation of module learning. The fuzzy controller of third level FLES3 evaluates a quality of whole course learning. Each level of fuzzy controller includes two global tasks. The first task is a creation of linguistic parameters by the expert’s help. The second task is a design of fuzzy rules base, which contains the expert’s reasoning and a structure

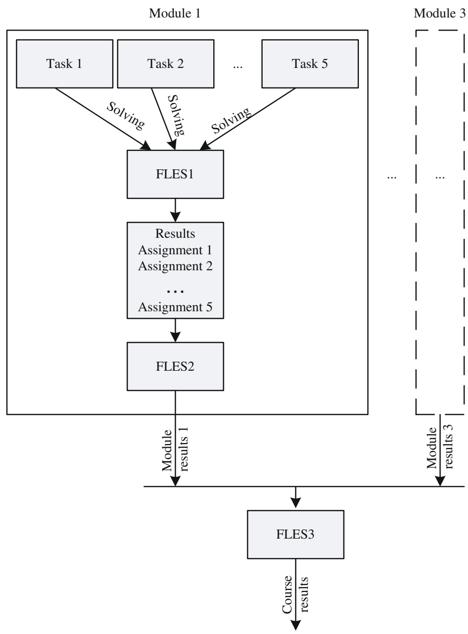

Fig. 5.4 The hierarchical system of fuzzy derivation FLES-PR

of knowledge domain. The work of fuzzy controllers FLES1, FLES2, and FLES3 are described in Sects. 5.6.1, 5.6.2, and 5.6.3, respectively. Section 5.6.4 indicates some test results of FLES-PR application.

5.6.1 Definition of Linguistic Variables and Fuzzy Rules in FLES1

Let us consider a set of linguistic variables and then a fuzzy rules base, which are used into a fuzzy controller FLES1. Three linguistic variables were introduced such as “Type_error”, “Prompt_level”, and “Task_rating”:

- Linguistic variable “Type_error” is based on the typical errors. According to three types of errors including a calculation error, an error in a formula, and error in a formula choice, three values of linguistic variable “Type_Error” can be represented as “minor”, “major”, and “rough”.
- Linguistic variable “Prompt_level” uses three levels of prompts (Sect. 5.5). Therefore, “level 1”, “level 2”, and “level 3” are values of this linguistic variable.
- Linguistic variable “Task_rating” is the output of fuzzy controller FLES1, which has values “low”, “middle”, and “high”.

Semantics of the FLES1 linguistic variables are situated in Fig. 5.5.

For the fuzzy controller FLES1, the following fuzzy rules were designed:

IF (Type_error = "minor") AND (Prompt_level = "level 1") THEN (Task_rating = "high").
IF (Type_error = "minor') AND (Prompt_level = "level 2") THEN (Task_rating = "middle").
IF (Type_error = "minor") AND (Prompt_level = "level 3") THEN (Task_rating = "low").
IF (Type_error = "major') AND (Prompt_level = "level 1") THEN (Task_rating = "middle").
IF (Type_error = "major") AND (Prompt_level = "level 2") THEN (Task_rating = "low").
IF (Type_error = "major") AND (Prompt_level = "level 3") THEN (Task_rating = "low").
IF (Type_error = "rough") AND (Prompt_level = "level 1") THEN (Task_rating = "low").
IF (Type_error = "rough") AND (Prompt_level = "level 2") THEN (Task_rating = "low").
IF (Type_error = "rough") AND (Prompt_level = "level 3") THEN (Task_rating = "low").

A screen of the FLES1 is represented in Fig. 5.6, which illustrates a case, when a student had not any error but used a prompt of level 2. Numbers 1–9 are the numbers of rules, according to which a fuzzy derivation is formed. Task rating is equal 15.5 that correspond to a "satisfactory" score.

5.6.2 Definition of Linguistic Variables and Fuzzy Rules in FLES2

For fuzzy controller FLES2, following linguistic variables were introduced:

- Linguistic variable "Number_correct_answers" has two values "small" and "large".

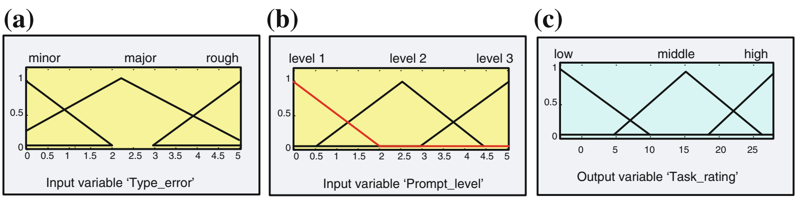

Fig. 5.5 Semantics of the FLES1 linguistic variables: **a** input variable "Type_error", **b** input variable "Prompt_error", **c** output variable "Task_rating"

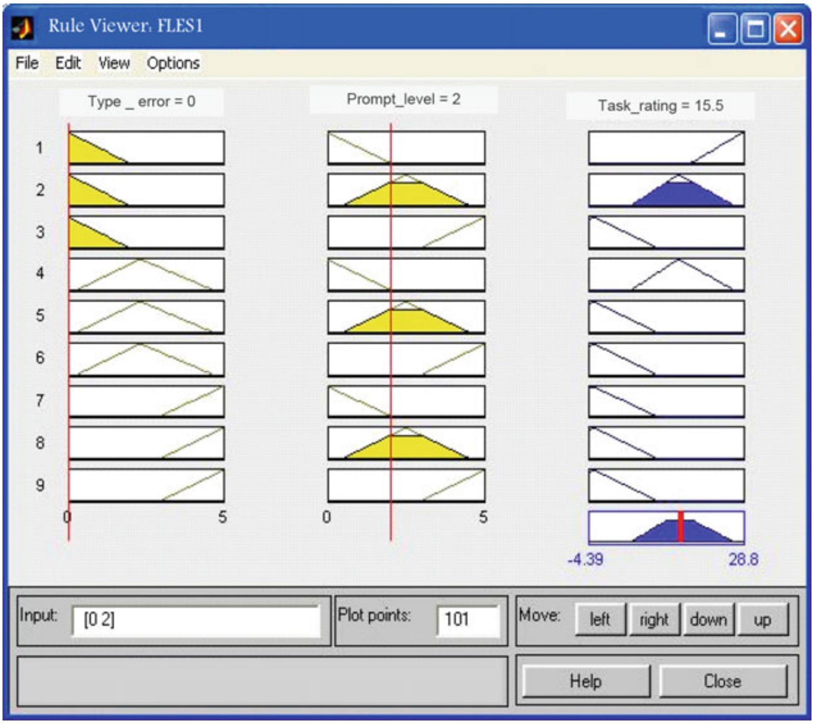

Fig. 5.6 Screen of the FLES1

- Linguistic variable "Number_prompts" involves number and level of the used prompts. Values of this linguistic variable are "small" and "large".
- Linguistic variable "Evaluation_module" is the output of fuzzy controller FLES2 with values "low", "middle", and "high". A value of this linguistic variable is into interval [0…25].

Semantics of the FLES2 linguistic variables are presented in Fig. 5.7.

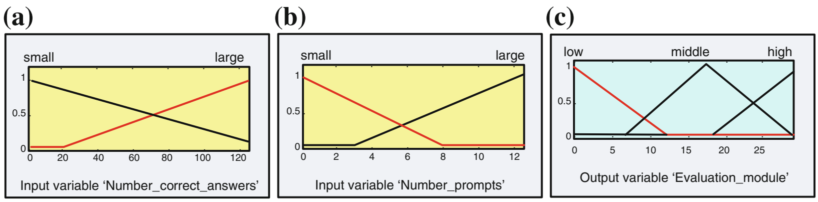

Fig. 5.7 Semantics of the FLES2 linguistic variables: **a** input variable "Number_correct_answers", **b** input variable "Number_prompts", **c** output variable "Evaluation_module"

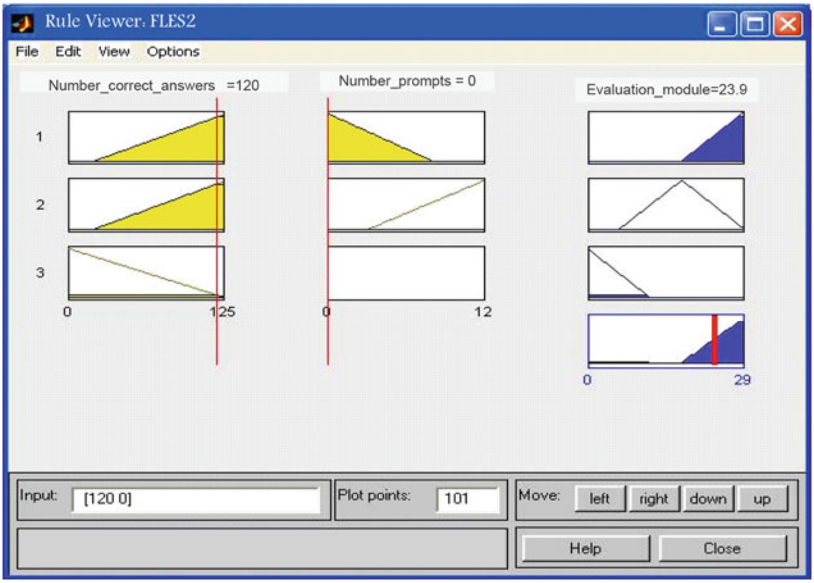

Fig. 5.8 Screen of the FLES2

The following fuzzy rules are used for the FLES2:

IF (Number_correct_answers = “large”) AND (Number_prompts = “small”) THEN (Evaluation_module = “high”).
IF (Number_correct_answers = “large”) AND (Number_prompts = “large”) THEN (Evaluation_module = “middle”).
IF (Number_correct_answers = “small”) THEN (Evaluation_module = “low”).

A screen of the FLES2 is show in Fig. 5.8 for a case, when a student made a calculation error in one task and did not use any prompt. The estimation of module is equal 23.9.

5.6.3 Definition of Linguistic Variables and Fuzzy Rules in FLES3

Three linguistic variables were proposed for fuzzy controller FLES3:

- Linguistic variable “Total_number_correct_answers’ summarizes the numbers of correct answers from all modules and has two values “small” and “large”.
- Linguistic variable “Total_number_prompts” summarizes the numbers of prompts from all modules. Values of this linguistic variable are “small” and “large”.

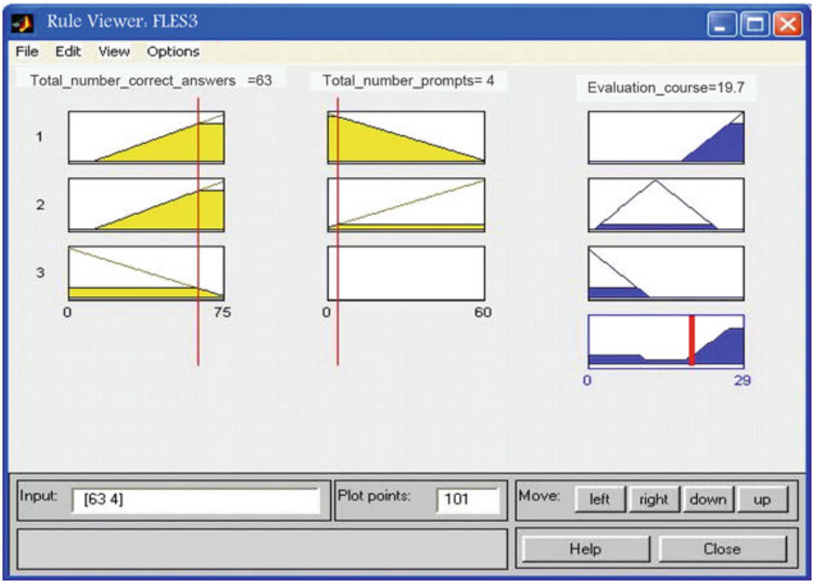

Fig. 5.9 Screen of the FLES3

- Linguistic variable "Evaluation_course" is the output of fuzzy controller FLES3 with values "low", "middle", and "high".

The fuzzy rules for the FLES3 are similar to the fuzzy rules for the FLES2. Figure 5.9 provides a screen of the FLES3.

5.6.4 Test Results of "FLES-PR"

Into experiments, 26 students were involved. For simplifier presentation of receiving results, let us consider a methodology applied for three students' evaluation. Student 1 is a successful student, who makes small number of errors. Student 2 is a good student, who solves the tasks by using prompts and help of the teacher. Student 3 is a non-successful student, who makes large number of errors and uses prompts. The set of tasks represented in Sect. 5.5 was used into this experiment. Short description of test results is situated in Table 5.1.

The results from Table 5.1 were the input data for FLES-PR system, and also they were represented for three university teachers. All evaluations were normalized to score 25 at the level of task, the level of module, and the level of course. The results for all three students were also tabulated using the same layout and then graphed. Figure 5.10 shows the results for students 2 and 3 (where student 1 was similar to student 2).

Table 5.1 Description of test results of three students

Task	Student 1	Student 2	Student 3
Module 1			
Task 1	Right	Right	Right
Task 2	Right	Right	Prompt of level 2
Task 3	Calculation error	Calculation error	Calculation error
Task 4	Right	Prompt of level 1	Prompt of level 1
Task 5	Right	Prompt of level 1	Prompt of level 2
Module 2			
Task 1	Right	Right	Incorrect formula
Task 2	Right	Incorrect formula	Incorrect formula
Task 3	Right	Prompt of level 2	Prompt of level 2
Task 4	Right	Right	Right
Task 5	Right	Right	Right
Module 3			
Task 1	Right	Right	Prompt of level 2
Task 2	Right	Right	Right
Task 3	Right	Right	Prompt of level 1
Task 4	Calculation error	Calculation error	Calculation error
Task 5	Right	Right	Calculation error

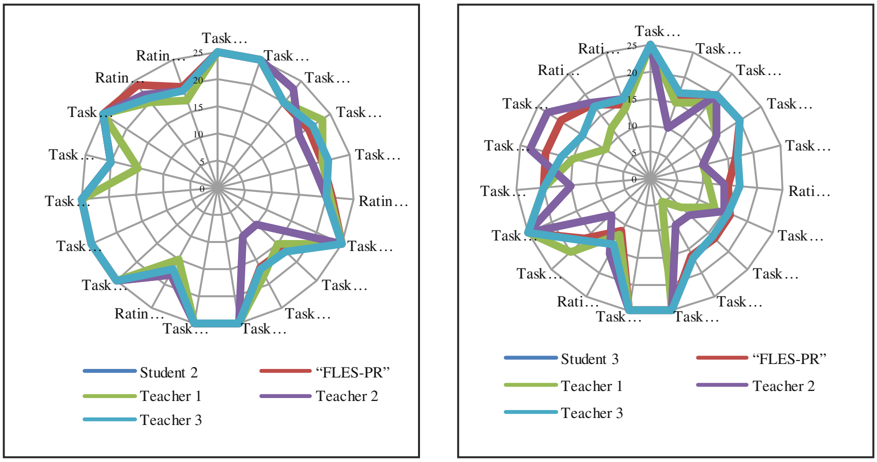

Fig. 5.10 Screen of the FLES3

Teacher 1 has a strong bias on Task 2.2, scoring all three students lower in all instances. Similarly Teacher 2 has a small bias on Task 1.2 and 3.4, however overall FRES-PR provides an excellent correlation for the initial settings (Fig. 5.11). Over time, further validation will enable fine-tuning to improve the overall results.

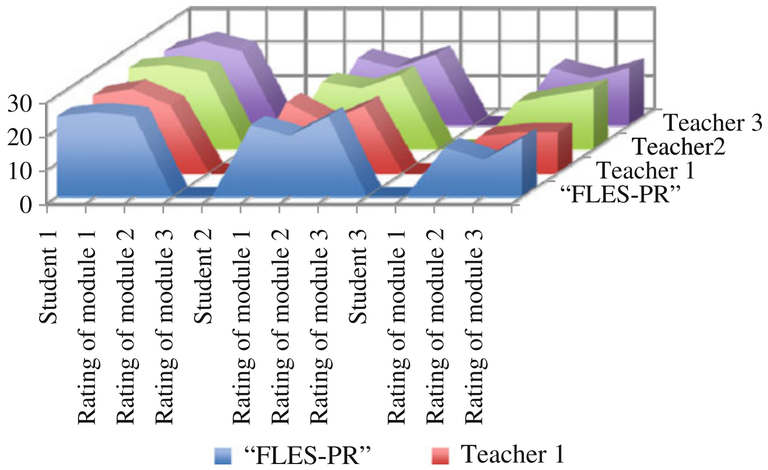

Fig. 5.11 Screen of the FLES3

The data verifies that FLES-PR makes good decisions and this was validated based on the experts' evaluations. The maximum deviation of each expert is within a standard deviation and considered within the variation of individual estimations made by each teachers. However, these deviations are decreased in module evaluations, and a whole course is evaluated with a variety 2 in a 25 score interval. Such variety can be considered as the independent. The received results prove the possibility for the usage of the proposed fuzzy logic knowledge evaluation in e-Learning university practice.

5.7 Conclusion

In this chapter, the conceptual model of e-Learning framework was developed as a set of related preparation, study, evaluation, and adaptation processes. A knowledge evaluation system based on the fuzzy logic controllers was the object of automation, which controls the tasks and the solved tasks flows, and also estimates the student success during the course learning. The conceptual scheme of knowledge evaluation system was represented, which realizes three interfacing strategies based on fuzzy logic such as the strategy for task selection, the evaluation strategy, and the strategy of help and prompts.

Our contribution connects with a fuzzy logic application FLES, which includes three levels. The fuzzy controller of first level FLES1 estimates a quality of the solved task and analyzes the errors and the used prompts. The fuzzy controller of second level FLES2 provides a quality evaluation of module learning. The fuzzy controller of third level FLES3 evaluates a quality of whole course learning. For each level, the linguistic variables by the expert's help and a rules base were created.

The hierarchical system of fuzzy derivation FLES-PR for course “Theory of probability” was design into “Matlab” 6.5 environment. This software tool works under the management of e-Learning university server “Moodle” providing all transmission functions between a student and a teacher. The real tasks from the course “Theory of probability” were represented for the students. The received by FLES-PR scores were compared with the expert’s recommendations, and provided a variety 2 in a 25 score interval. These estimations permit to hope that other university courses based on the FLES core can be automated for the e-Learning distance interactions. Also the management sub-system realization needs in some functional improvements.

References

1. Barak M (2010) Motivating self-regulated learning in technology education. Int J Technol Des Educ 20(4):381–401
2. Klasnja-Milicev A, Vesin B, Ivanovic M, Budimac Z (2011) E-Learning personalization based on hybrid recommendation strategy and learning style identification. Comput Educ 56(2):885–899
3. Dias SB, Diniz JA (2012) Blended learning in higher education: different needs, different profiles. Procedia Comput Sci 14:438–446
4. Lin YT, Wen ML, Jou M, Wu DW (2014) A cloud-based learning environment for developing student reflection. Comput Hum Behav 32:244–252
5. Serrano-Cámara LM, Paredes-Velasco M, Alcover CM, Velazquez-Iturbide JÁ (2014) An evaluation of students’ motivation in computer-supported collaborative learning of programming concepts. Comput Hum Behav 31:499–508
6. Anaya AR, Luque M, García-Saiz T (2013) Recommender system in collaborative learning environment using an influence diagram. Expert Syst Appl 40:7193–7202
7. Shaw RS (2012) A study of the relationships among learning styles, participation types, and performance in programming language learning supported by online forums. Comput Educ 58 (1):111–120
8. Nicholson P (2007) A History of E-learning. In: Fernández-Manjón B, Sánchez-Pérez J, Gómez-Pulido J, Vega-Rodríguez M, Bravo-Rodríguez J (eds) Computers and education: E-learning. From Theory to Practice, Springer, Heidelberg, pp 1–11
9. Goldberg BS, Holden HK, Brawner KW, Sottilare RA (2011) Enhancing performance through pedagogy and feedback: domain considerations for ITSs. In: Interservice/Industry training, simulation, and education conference (I/ITSEC’2011), paper 11010, pp 3–12
10. Hattie J (2012) Visible Learning for Teachers: maximising impact on learning, Routledge, London
11. Stockinger P, DePablo E (1999) Pedagogical models, methodologies and tools in web-based distant learning. In: Archimed Knowledge Village, pp 1–41, chapter 4
12. Conati C, Manske M (2009) Evaluating adaptive feedback in an educational computer game. In: Ruttkay Z, Kipp M, Nijholt A, Vilhjálmsson HH (eds) Proceedings of 9th international conference on intelligent virtual agents, LNCS 5773. Springer, Berlin, pp 146–158
13. Ghirardini B (2011) E-learning methodologies: a guide for designing and developing e-learning courses. Food and Agriculture Organization of the United Nations, Rome
14. Bersin J (2004) The blended learning book. Pfeiffer, San Francisco
15. Wiley D (2002) The instructional use of learning objects. Agency for Instructional Technology, Bloomington

16. Govindasamy T (2002) Successful implementation of e-Learning pedagogical considerations. Internet Higher Educ, Pergamon 4:287–299
17. Chrysafiadi K, Virvou M (2013) Student modeling approaches: a literature review for the last decade. Expert Syst Appl 40:4715–4729
18. Mahnane L, Laskri MT, Trigano P (2012) An adaptive hypermedia system integrating thinking style (AHS-TS): model and experiment. Int J Hybrid Info Technol 5(1):11–28
19. Gaudioso E, Montero M, Hernandez-del-Olmo F (2012) Supporting teachers in adaptive educational system through predictive models: a proof of concept. Expert Syst Appl 39 (1):621–625
20. Jeremic Z, Jovanovic J, Gasevic D (2012) Student modeling and assessment in intelligent tutoring of software patterns. Expert Syst Appl 39(1):210–222
21. Ting CY, Phon-Amnuaisuk S (2012) Properties of Bayesian student model for INQPRO. Appl Intell 36(2):391–406
22. Peña-Ayala A, Sossa-Azuela H, Cervantes-Pérez F (2012) Predictive student model supported by fuzzy-causal knowledge and inference. Expert Syst Appl 39(5):4690–4709
23. Cline BE, Brewster CC, Fell RD (2010) A rule-based system for automatically evaluating student concept maps. Expert Syst Appl 37:2282–2291
24. Yıldız-Feyziog E, Akpınar E, Tatar N (2013) Monitoring students' goal setting and metacognitive knowledge in technology-enhanced learning with metacognitive prompts. Comput Hum Behav 29:616–625
25. Fiorella L, Vogel-Walcutt JJ, Fiore S (2012) Differential impact of two types of metacognitive prompting provided during simulation-based training. Comput Hum Behav 28:696–702
26. Peña-Ayala A, Sossa H, Méndez I (2014) Activity theory as a framework for building adaptive e-learning systems: a case to provide empirical evidence. Comput Hum Behav 30:131–145
27. Desmarais MC, Baker RS (2012) A review of recent advances in learner and skill modeling in intelligent learning environments. User Model User-Adap Inter 22(1–2):9–38
28. Tweedale JW, Jain LC (2014) Advances in modern artificial intelligence. In: Tweedale JW, Jain LC (eds) Recent advances in knowledge-based paradigms and applications, vol 234 of Advances in intelligent systems and computing. Springer, Berlin, pp 1–18
29. Hand D (2009) Measuring classifier performance: a coherent alternative to the area under the roc curve. In: Machine learning, vol 77. Springer, Berlin, pp 103–123
30. Zadeh LA (1965) Fuzzy sets. Inf Control 8:338–353
31. Zadeh LA (1972) A rationale for fuzzy control. J Dyn Syst Meas Control 94(1):3–4
32. Zadeh LA (1972) A fuzzy set theoretic interpretation of the linguistic hedges. J Cybern 2:4–34
33. Sugeno M (ed) (1985) Industrial applications of fuzzy control. Technology and Engineering, North-Holland
34. Sugeno M, Yasukawa T (1993) A fuzzy-logic-based approach to qualitative modelling. IEEE Trans Fuzzy Syst 1(1):1–7
35. Sastry K, Goldberg D (2004) Designing competent mutation operators via probabilistic model building of neighbourhoods. Genetic and Evolutionary Computation, Springer, Berlin, LNCS 3103, pp 114–125
36. Gacto M, Alcala R, Herrera F (2012) A multi-objective evolutionary algorithm for an effective tuning of fuzzy logic controllers in heating. ventilating and air conditioning systems. Appl Intell 36(2):330–347
37. Kaya M, Alhajj R (2006) Utilizing genetic algorithms to optimize membership functions for fuzzy weighted association rules mining. Appl Intell 24(1):7–15
38. Precup R, Hellendoorn H (2011) A survey on industrial applications of fuzzy control. Comput Ind 62(3):213–226
39. Bai SM, Chen SM (2008) Automatically constructing grade membership functions of fuzzy rules for students' evaluation. Expert Syst Appl 35(2008):1408–1414
40. Jia B, Zhong S, Zheng T, Liu Z (2010) The study and design of adaptive learning system based on fuzzy set theory. In: Pan Z, Cheok AD, Müller W, Zhang X, Wong K (eds) Transactions on edutainment IV, LNCS, Springer, Berlin, vol 6250, pp 1–11

41. Taylan O, Karagözoglu B (2009) An adaptive neuro-fuzzy model for prediction of student's academic performance. Comput Ind Eng 57(3):732–741
42. Dias SB, Diniz JA (2013) FuzzyQoI model: A fuzzy logic-based modelling of users' quality of interaction with a learning management system under blended learning. Comput Educ 69: 38–59
43. Lo JJ, Chan YC, Yeh SW (2012) Designing an adaptive web-based learning system based on students' cognitive styles identified online. Comput Educ 58:209–222
44. Millán E, Loboda T, Pérez-de-la-Cruz JL (2010) Bayesian networks for student model engineering. Comput Educ 55(4):1663–1683
45. Tourtoglou K, Virvou M (2012) An intelligent recommender system for trainers and trainees in a collaborative learning environment for UML. J Intell Decis Technol 6(2):79–95
46. Celikyilmaz A, Turksen IB (2009) Studies in fuzziness and soft computing, vol 240. Springer, Berlin, pp 11–50
47. Zang R, Phillis Y, Kouikoglou V (2005) Fuzzy logic fuzzy control of queuing systems. Springer, Berlin, pp 15–25
48. Abreu G, Ribeiro J (2003) On-line control of a flexible beam using adaptive fuzzy controller and piezoelectric actuators. Control Autom Mag 14(4):377–383
49. Turevskiy A (2002) Fuzzy logic toolbox user's guide, 2nd edn. Matworks, Massachusetts