J. Comput. Educ. (2014) 1(1):99–111
DOI 10.1007/s40692-014-0003-0

# Smart e-course recommender based on learning styles

**Moushir M. El-Bishouty · Ting-Wen Chang · Sabine Graf · Kinshuk · Nian-Shing Chen**

Received: 28 October 2013 / Revised: 15 January 2014 / Accepted: 24 February 2014 /
Published online: 21 March 2014


**Abstract** A student's learning style is the approach for learning that best allows the student to gather and to understand knowledge in a specific manner. Providing students with learning materials and activities that fit to their learning styles seems to have high potential to make learning easier for them. This research aims at providing teachers with recommendations on how to best extend their existing e-courses in learning management systems to accommodate more students with different learning styles. A smart e-course recommender tool has been developed for this purpose, which analyzes the e-courses with respect to their support levels for different students' learning styles, recommends learning objects to be added to the courses, and visualizes the recommendations and the improvement in the course support level for students' with different learning styles. The experimental results indicate that the tool has the ability to recommend suitable learning objects that,

M. M. El-Bishouty (✉) · T.-W. Chang · S. Graf · Kinshuk
Athabasca University, Athabasca, Canada
e-mail: moushir.elbishouty@gmail.com

T.-W. Chang
e-mail: tingwen.dodochang@gmail.com

S. Graf
e-mail: sabineg@athabascau.ca

Kinshuk
e-mail: kinshuk@athabascau.ca

M. M. El-Bishouty
City for Scientific Research and Technological Applications, Alexandria, Egypt

N.-S. Chen
National Sun Yat-sen University, Kaohsiung, Taiwan
e-mail: nianshing@gmail.com

once being added, significantly improve the course support level for accommodating more students with different learning styles.

**Keywords** Smart e-course recommender · Learning styles · Genetic algorithm · Learning management systems · Adaptive learning

## Introduction

Learning styles have been an area of considerable interest in the literature (Bostrom et al. 1990). Many educational theorists and researchers consider learning styles as an important factor in the learning process and agree that incorporating them in education has potential to facilitate learning for students (Graf et al. 2009). A student's learning style is defined as a unique collection of individual skills and preferences that affects how a person perceives, gathers, and processes information (Clay and Orwig 1999). Learning styles affect how a person learns, including also the aspects of how a person acts in a learning group, participates in learning activities, relates to others, and solves problems. Basically, a person's learning style is the method that best allows the person to gather and to understand knowledge in a specific manner.

In the literature, many learning-style models exist, such as Kolb (1984), Honey and Mumford (1982), Pask (1976), and Felder and Silverman (1988a, b). Among them, our research utilizes Felder and Silverman Learning Style Model (FSLSM), as it is one of the models that have been most often used in the recent times, and some researchers even argue that it is the most appropriate model for use in e-learning systems (Carver et al. 1999; Kuljis and Liu 2005). FSLSM combines major learning-style models such as Kolb (1984) and Pask (1976). By combining these models, FSLSM describes learning styles in more detail. Felder and Silverman proposed four dimensions of learning styles (active/reflective, sensing/intuitive, visual/verbal, and sequential/global). FSLSM describes student's learning style by his/her preference on each of the four dimensions, measured on values between $+11$ to $-11$, in steps of $\pm 2$. This enables a quite detailed description of the students' learning styles.

According to Felder and Silverman (1988a, b), providing students with learning material and activities that fit their preferred ways of learning seems to have high potential to make learning easier for them. In this paper, we present a smart tool for providing teachers with recommendations on how to best extend their existing e-courses in learning management systems (LMSs) to accommodate more students with different learning styles. The concept of the genetic algorithm (GA) (Holland 1975) is utilized to generate the recommendations considering the structure of the courses and students' learning styles. In the next section, related work is illustrated. In "Smart e-course recommender" section, the course analysis mechanism, the recommendation algorithm and the system implementation are presented. The experimental results are discussed in "Experiments and results" section, followed by the conclusions in "Conclusions" section.

## Related work

Providing students with courses that fit to their learning styles is a two-step process. The first step is to identify students' learning styles, and the second step is then to adapt the courses to fit with students' learning styles. For identifying learning styles based on FSLSM, Felder and Soloman developed the Index of Learning Styles (ILS) (Felder and Soloman 1997), a 44-item questionnaire. Each question in the questionnaire relates to one of the four dimensions of learning styles (active/reflective, sensing/intuitive, visual/verbal, and sequential/global), and with 11 questions for each dimension, the learning styles preferences are expressed with values between $+11$ to $-11$ per dimension, with steps $\pm 2$.

Using questionnaires for identifying learning styles has underlying assumption that the learning styles are stable for a long period of time. However, the stability of learning styles is still a controversial issue. As soon as learning styles change, the results of the questionnaires are not valid anymore, and students would have to complete the questionnaire again in order to identify their ever-changing learning styles. Therefore, depending on learning-style models, researchers have presented different approaches for dynamically detecting students' learning styles from the behaviors and actions of learners in learning systems (Chang et al. 2009; García et al. 2007; Schiaffino et al. 2008; Xenos 2004). For example, based on FSLSM, Ozpolat and Akar (2009) used NBTree classification algorithm in conjunction with Binary Relevance classifier to classify students according to their interests and then detect learners' learning styles. Graf and Kinshuk (2006) calculated students' learning styles based on patterns of behavior, which are relevant for each learning-style dimension according to FSLSM.

As for the second step of adapting the courses to fit with students' learning styles, many studies have indicated that adaptive learning is a critical requirement for promoting students' learning performance (Chang et al. 2009; Brusilovsky and Maybury 2002; Sessink et al. 2003). Researchers have presented different approaches to provide adaptive learning materials, learning strategies, and courses according to individual students' learning styles (Carver et al. 1999; Pena et al. 2002; Trantafillou et al. 2003). Paredes and Rodríguez (2004) presented a framework that adapts the course structure and sequencing to the student's profile and uses implicit information about students' behavior gathered by the system during the learning process in order to dynamically modify the course structure and sequencing. Graf and Kinshuk (2007) introduced a concept for LMSs with adaptivity based on FSLSM. They developed a prototype system with an add-on feature that enables the open source LMS Moodle to automatically provide adaptive courses that fit to the learning styles of students.

Experimental results have indicated that providing adaptive courses based on students' learning styles plays an effective role in enhancing the learning outcomes by reducing learning time and increasing learners' satisfaction (Graf and Kinshuk 2007; Popescu 2010; Tseng et al. 2008). However, the ability of a course to be adapted for a particular learning style is limited to the suitability of the course contents for that learning style. If the existing course contents do not support particular (one or more) learning style, then the adaptive system will not be able to provide efficient adaptive course that fits to that learning style, unless teacher is willing to add additional content to suit that particular learning style. For this purpose, a mechanism is proposed in this paper to analyze existing course structure and contents in LMS to measure the course support level for diverse learning styles of students. Based on the analysis results and different students' learning styles, a GA is employed in the proposed mechanism for providing teachers with recommendations of learning objects (LOs) that could be added to the existing course. These recommendations aim at improving the course support level for students' with different learning styles.

## Smart e-course recommender

Smart e-course recommender is an interactive tool that analyzes and visualizes the suitability of existing courses in learning management systems with respect to students' learning styles, and generates recommendations on how to best extend those courses to accommodate more students with different learning styles. Teachers can first define various types of LOs that can be added to the course structure, and then the tool recommends the most suitable types of LOs and where these additional LOs should be added in the course structure. In addition, it visualizes the course support level for different learning styles of students before and after adding the recommended LOs, so as to clearly visualize the benefits the course will have due to the additions. In the next subsections, the structure of e-courses, the analysis mechanism and the recommendation algorithm are described.

### Course structure

In order to use the proposed tool for existing e-courses, it is assumed that a course consists of several units and a unit can (but does not have to) consist of several sections. A section normally starts with a commentary which provides learners with a brief overview of the section. Subsequently, there is an area before content objects (ABC) that may include a few LOs that aim at motivating the learners and making the section interesting for them, for example, animations or short video clips. After this area, the content object is presented. In the next area, namely, area after content objects (AAC), other types of LOs may be presented like self-assessment quiz, discussion forum, real-life applications, etc. The conclusions of the section can exist either right after the last content object or at the end of the section.

### Course analyzing mechanism

The course analyzing mechanism is based on FSLSM (Felder and Silverman 1988a, b). It measures how well each section of an existing e-course fits to each of the eight poles of FSLSM. It presents the support level (percentage) for diverse learning styles by calculating the average of three factors: the availability, the frequency, and the sequence of the LOs (El-Bishouty et al. 2012). The availability factor measures the existence of LO types that support each learning style. The frequency factor measures the number of LOs that support each learning style. The sequence factor measures how well this object type in that place fits with each of the eight learning styles of FSLSM.

The analyzing mechanism currently considers the following eleven types of LOs:

- *Commentaries* provide learners with a brief overview of the section.
- *Content Objects* are used to present the learning material.
- *Reflection Quizzes* include one or more open-ended questions about the content of a section.
- *Self-Assessment Tests* include several close-ended questions about the content of a section.
- *Discussion Forum Activities* provide learners with the possibility to ask questions and discuss topics with their peers and instructor.
- *Additional Reading Materials* provide learners with additional sources for reading about the content of the section.
- *Animations* demonstrate the concepts of the course in an animated multimedia format.
- *Exercises* provide learners with an area where they can practice the learned knowledge.
- *Examples* illustrate the theoretical concepts in a concrete way.
- *Real-Life Applications* demonstrate how the learned material can be related to and applied in real-life situations.
- *Conclusions* summarize the content learned in a section.

The availability of types of LOs is considered as a factor to infer how well a section can fit each of the eight different learning styles. The mechanism measures the existence of those LO types in a section that can support each learning style (ls) in relation to all types of LOs that can support that specific learning style (11 LO types are considered in this study). The availability factor is calculated using the formula 1, its value range from 0 to 1, where 1 indicates a strong suitability of support to the learning style, and 0 means no support:

$$\mathrm{Ava}_{\mathrm{ls}} = \frac{(\#\,\text{of existing LO types that support ls}\ )}{(\#\,\text{of LO types that support ls})} \tag{1}$$

On the other hand, the frequency factor measures the number of those LOs in a section that support each learning style with respect to the frequency threshold. The frequency threshold represents number of LOs in a section that are sufficient to fully support a particular learning style. This threshold is predefined and can be adjusted by the teacher if needed. If the number of LOs that support a particular learning style (ls) in a section is less than the value of the frequency threshold, then the frequency factor is obtained by the formula 2. Otherwise, the frequency factor takes the value 1, which means a full frequency support level for that learning style. The obtained values for both, the availability factor and the frequency factor, range from 0 to 1, where 1 indicates a strong suitability of support to the learning style, and 0 means no support.

$$\mathrm{Freq}_{\mathrm{ls}} = \frac{(\#\ \text{of existing LOs that support ls})}{(\text{frequency threshold})} \tag{2}$$

Actually, not only the types but also the order and the position of the LOs affect the suitability of a course regarding different learning styles. The sequence factor measures the suitability of the sequence of LOs for different learning styles. We calculate the sequence factor for each LO according to its type, location (ABC or AAC), and order. It is determined according to how much this LO type in that place fits for each of the eight learning styles of FSLSM. The sequence factor for each learning style is calculated using the formula 3. In this formula, $f_{\mathrm{ls}}\ (LO_i) = 1$, if $\mathrm{LO_i}$ is suitable for that learning style at that location, and $f_{\mathrm{ls}}\ (LO_{\mathrm{i}}) = 0$ if not. The weight $w$ represents how well the position of a LO fits to the specific learning style, and $n$ is the number of LOs in the section. Formula 3 represents the weighted mean of $f_{\mathrm{ls}}(LO_{\mathrm{i}})$. Its value ranges from 0 to 1, where 1 indicates a strong suitability of support to the learning style, and 0 means no support.

$$\mathrm{Seq}_{\mathrm{ls}} = \frac{\sum_{i=1}^{n} f_{\mathrm{ls}}(\mathrm{LO}_i) \times w_i}{\sum_{i=1}^{n} w_i}, \quad 0 < w \leq 1 \tag{3}$$

After calculating the average of the three factors for each section, the results can be aggregated for the whole course. Consequently, the course support level is calculated, which ranges between 0 % (means no support for any learning style) and 100 % (indicates full support for all learning styles).

## Recommendation algorithm

The smart e-course recommender enables teachers to decide how many additional LOs they could possibly add to the course structure with the aim to improve the course support level for better accommodating students' different learning styles. Based on this specific number, the recommendation algorithm identifies which types of LOs should be added and where in the course structure these LOs should be placed in order to achieve the best improvement in the course support level. The recommendation algorithm then utilizes the concept of the GA to solve an optimization problem aiming at maximizing the course support level.

GAs are heuristic search techniques, based on the mechanism of natural selection and genetics, invented by Holland (1975). GAs have been used to solve very wide range of practical problems, for example, the problem of course timetabling (Ross and Come 1994; Yu and Sung 2002). A typical GA starts with an initial set of random solutions, called a population, where each individual in the population is called a chromosome. A chromosome, a string of genes, represents a solution to the problem, and genes represent individual elements of a solution. Each chromosome is evaluated by its fitness score as computed by the objective function of the problem. Chromosomes evolve through successive iterations, called generations. New chromosomes (offspring) are formed using three primary genetic operators: selection, crossover (mating), and mutation, which form a new generation of population. This process continues to achieve the optimal solution.

Defining an appropriate representation of chromosome depends on the problem, and it is part of the art of using GAs. In our algorithm, the chromosome is represented by a two-dimensional array of integers. The width of the array is determined by the number of LOs to be added into the course. One dimension of the array represents the value of the LO type; it ranges from 1 to 11, where each value is associated with one of the 11 types of LOs (listed earlier in "Course analyzing mechanism" section). On the other hand, the sequence of an LO is represented in the other dimension of the array. The sequence variable represents the order of the LO in the sequence of LOs that the course consists of, which ranges from 1 to m (where, m is the total number of LOs in the course).

In the proposed algorithm, the individuals (chromosomes) of the initial population are randomly generated using Mersenne Twister (Matsumoto and Nishimura 1998). Mersenne Twister (MT) is a uniform pseudorandom number generator (PRNG). MT is widely used as a fast pseudorandom number generator. After generating the initial population, each chromosome is evaluated and associated with its fitness score. The chromosome evaluation is done by modifying the course structure according to the chromosome data and then re-analyzing the course structure using the analysis mechanism described in "Course analyzing mechanism" section. The higher the fitness score, the better the chromosome. Consequently, selection, crossover, and mutation operators are used to generate successive generations till reaching to the best available chromosome. The fitness score here is actually representing the course support level, which is ranging from 0 to 100 %.

Selection operator chooses individuals (chromosomes) from the population for mating. In the proposed algorithm, rank selection operator is utilized (Kumar and Jyotishree 2012). In the rank selection, the individuals are sorted by fitness score. The probability that an individual is selected is then inversely proportional to its position in this sorted list, i.e., the individual at the head of the list is more likely to be selected than the next individual, and so on through the sorted list. Crossover operator randomly selects a crossover point within a chromosome and then interchanges the two selected chromosomes (parent) at this point to produce two new chromosomes (offspring). The idea behind crossover is that the new chromosome may be better than both the parents if it takes the best characteristics from each of the parents. Besides, mutation operator randomly alters a gene of a chromosome that produces spontaneous changes in various chromosomes.

## System implementation

The smart e-course recommender tool has been implemented as a web-based application with PHP scripting language using MySQL relational database management system as the backend. It is a stand-alone server side application that connects to an LMS database, retrieves the existing course structure, applies the course analyzing mechanism, generates recommendations, visualizes the recommended LOs in the course structure, and shows how well the course fits with students' learning styles before and after considering the recommendations.

As shown in Fig. 1, the user interface of the smart e-course recommender consists of two parts: the settings part (the left side of the interface), and the visualization part (the right side of the interface). In the *Course Structure* section on the left side, the course structure is displayed in terms of units/sections and a list of LOs in each section. The teacher can browse the course and select a particular unit/section by clicking on it. The *Recommendation Settings* section on the left side enables the teacher to decide on the number and the suitable types of LOs to be recommended. Once the teacher has provided these settings, he/she can click on the *Generate a Recommendation* button. The tool then generates the recommendation, shows the new (recommended) LOs in the course structure (displayed in blue), re-analyzes the course, and updates the visualization part on the right side. Furthermore, in the *Advanced Settings* section on the left side, teachers can set the value of the frequency threshold (as described in "Course analyzing mechanism" section) and select the LO types to be considered while analyzing the course support level.

The visualization part on the right side shows how well the course and a particular selected unit/section fit with students' learning styles (El-Bishouty et al. 2013). It consists of four similar visualization components. The upper two components visualize the course support level for the whole course before and after the recommendation. The bottom two components show the selected unit/section support level before and after the recommendation.

Each component in the visualization consists of two parts. The upper part consists of a set of bars to show the strength of the harmony of the course with each of the eight learning-style poles (i.e., active, reflective, visual, verbal, sensing, intuitive, sequential, and global) in terms of percentage (calculated by the average of the three factors described in "Course analyzing mechanism" section). Each learning-style dimension is represented by two horizontal bars, one for each pole, where the two poles show the two different preferences of the dimension. The longer the bar is, the more the course fits with the learning style. The lower part of the component contains only one bar that shows the overall course support level for diverse learning styles (calculated by the average of the support level of the eight poles). Moving the cursor over any bar displays a tooltip with more details about the calculated support level.

## Experiments and results

Experiments were carried out to evaluate the performance and the efficiency of the recommendations provided by the smart e-course recommender tool. The tool was applied on real online courses in Moodle LMS at a university in Western Canada. The original course (before generating recommendations) consisted of 167 LOs. The tool analyzed the course support level for students' with different learning styles. The analysis results indicated that the value of the original course support level was 53 (out of 100).

Recommendations were then generated by the tool using the proposed GA (described in "Recommendation algorithm" section) aiming at improving the


**Fig. 1** Smart e-Course recommender–user interface

course support level. The efficiency of a GA is greatly dependent on tuning its parameters such as population size, crossover rate, and mutation rate. In the literature, there are several recommended settings for GA parameters. Dejong (1975) presented one of the combinations of parameters that work for optimization functions. Dejong offered a population size of 50, 0.6 for crossover rate, 0.001 for mutation rate, and 1000 as the number of generations. Grefenstette (1986), on the other hand, searched for optimal GAs for a given set of numerical optimization problems. Grefenstette presented settings for online performance, which are more suitable for our tool since it provides online recommendations. Grefenstette proposed a population size of 30, 0.95 for crossover rate, and 0.01 for mutation rate.

The evaluation consisted of two phases. The first phase aimed at measuring the improvement in the course support level and the optimal support level that could be reached after recommending (adding) different numbers of LOs. In this phase, Dejong settings were applied, which offered larger search space (population size of 50). The algorithm was run for different number of LOs (5, 10, 15, 20, 25, and 30) for 20 trials each. For each number of LOs, the optimal course support level was recorded. Figure 2 shows the relation between the number of recommended LOs and the course support level. For example, after considering the recommendation of adding 15 LOs, the course support level improved from 53 % (original) to 65 %. These results indicated a significant improvement in the course support level with respect to the recommendations.

The second phase aimed at defining the termination criteria of the GA to achieve the best solution in a reasonable time. In this phase, Grefenstette settings were used, as it was designed for online performance. Similar to phase one, the algorithm was

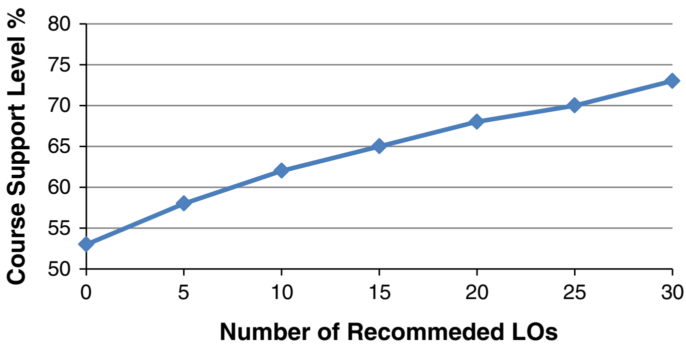


**Fig. 2** The improvement in the course support level per the number of recommended LOs

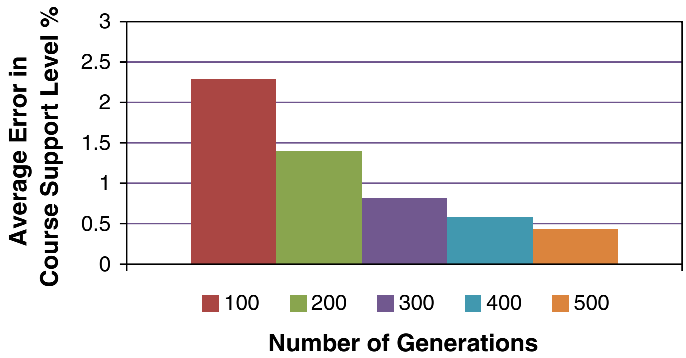


**Fig. 3** The average error in the course support level per the number of generations

run for different number of LOs (5, 10, 15, 20, 25, and 30) for 20 trials each. The number of generations for each trial was set to 500. Considering the optimal values recorded in the first phase, the average errors in the support level for all trials were calculated after 100, 200, 300, 400, and 500 generations. As shown in Fig. 3, the average error was less than 1 %, when the GA was run for 300 generations. Therefore, Grefenstette settings and a number of 300 generations were considered as standard settings for the proposed GA.

## Conclusions

In this paper, a smart e-course recommender tool is presented that analyzes and visualizes the suitability of existing courses in learning management systems with respect to students' learning styles, and generates recommendations on how to best extend those courses to accommodate more students with different learning styles. Experiments were carried out to evaluate the performance and the efficiency of the recommendations provided by the tool. The results indicated a significant improvement in the course support level with respect to the recommendations. This tool supports a teacher, who is willing to add additional content, to decide the most suitable types of LOs and where those LOs should be added into the course structure. The added LOs can help students with more suitable learning materials and activities that fit their preferred ways of learning so as to make learning easier for them. The number of LOs to be added can be decided based on the desired course support level. The tool can be applied for different course contents and structures. Identifying more LO types helps the recommendation algorithm to provide more efficient recommendations. The tool currently considers eleven types of LOs. However, the tool is flexible so that new types of LOs can be added if necessary. The future plan of the research includes investigating the impact of extending courses for students with different learning styles through conducting pilot studies with teachers and students.

## References


Bostrom, R. P., Olfman, L., & Sein, M. K. (1990). The importance of learning style in end-user training. *MIS Quarterly, 14*(1), 101–119.

Brusilovsky, P., & Maybury, M. T. (2002). From adaptive hypermedia to the adaptive web. *Communications of the ACM, 45*(5), 30–33.

Carver, C. A., Howard, R. A., & Lane, W. D. (1999). Addressing different learning styles through course hypermedia. *IEEE Transactions on Education, 42*(1), 33–38.

Chang, Y., Kao, W., Chu, C., & Chiu, C. (2009). A learning style classification mechanism for e-learning. *Computers & Education, 53*, 273–285.

Clay, J., & Orwig, C. (1999). Your learning style and language learning. Lingual links library, Summer Institute of Linguistic, Inc (SIL) International version 3.5.

Dejong, K. (1975). An analysis of the behavior of a class of genetic adaptive systems. PhD thesis, University of Michigan.

El-Bishouty, M. M., Chang, T.-W., Kinshuk, & Graf, S. (2012). A framework for analyzing course contents in learning management systems with respect to learning styles. In G. Biswas, et al. (Eds.) *20th International Conference on Computers in Education* (ICCE 2012), pp. 91–95.

El-Bishouty, M. M., Saito, K., Chang, T.-W., Kinshuk, & Graf, S. (2013). An interactive course analyzer for improving learning styles support level. *Proceedings of the Third International Workshop on Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data* (HCI-KDD 2013) Lecture Notes in Computer Science (Vol. 7947, pp. 136–147). Springer.

Felder, R. M., & Silverman, L. K. (1988a). Learning and teaching styles in engineering education. *Engineering Education, 78*(7), 674–681.

Felder, R., & Silverman, L. (1988b). Learning and teaching styles in engineering education. *Journal of Engineering Education, 78*(7), 674–681.

Felder, R. M., & Soloman, B. A. (1997). Index of learning styles questionnaire. North Carolina State University. http://www.engr.ncsu.edu/learningstyles/ilsweb.html. Accessed Jan 2014.

García, P., Amandi, A., Schiaffino, S., & Campo, M. (2007). Evaluating Bayesian Networks' precision for detecting students' learning styles. *Computers & Education, 49*(3), 794–808.

Graf, S. & Kinshuk. (2007). Providing adaptive courses in learning management systems with respect to learning styles. P*roceedings of the world conference on e-learning in corporate, government, healthcare, and higher education (e-learn)*, pp. 2576–2583.

Graf, S., & Kinshuk, P. (2006). An approach for detecting learning styles in learning management systems. *Proceedings of the sixth international conference on advanced learning technologies* (ICALT'06), pp. 161–163.

Graf, S., Kinshuk, & Liu, T.-C. (2009). Supporting teachers in identifying students' learning styles in learning management systems: An automatic student modelling approach. *Educational Technology & Society, 12*(4), 3–14.

Grefenstette, J. J. (1986). Optimization of control parameters for genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-16-1, 122–128.

Holland, J. (1975). *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan Press.

Honey, P., & Mumford, A. (1982). *The manual of learning styles*. Maidenhead: Peter Honey.

Kolb, D. A. (1984). *Experiential learning: Experience as the source of learning and development*. Englewood Cliffs, NJ: Prentice-Hall.

Kuljis, J., & Liu, F. (2005). A comparison of learning style theories on the suitability for e-learning. *Proceedings of the conference on web technologies, applications, and services*, ACTA Press, pp. 191–197.

Kumar, R., & Jyotishree, (2012). Blending roulette wheel selection & rank selection in genetic algorithms. *International Journal of Machine Learning and Computing, 2*(4), 365–370.

Matsumoto, M., & Nishimura, T. (1998). Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation, 8*(1), 3–30.

Ozpolat, E., & Akar, G. B. (2009). Automatic detection of learning styles for an e-learning system. *Computers & Education, 53*, 355–367.

Paredes, P., & Rodríguez, P. (2004). A mixed approach to modelling learning styles in adaptive educational hypermedia. *Advanced Technology for Learning, 1*(4), 210–215.

Pask, G. (1976). Styles and strategies of learning. *British Journal of Educational Psychology, 46*, 128–148.

Pena, C., Narzo, J., & Rosa, J. (2002). Intelligent agents in a teaching and learning environment on the Web. *Proceedings of the 2nd IEEE international conference on advanced learning technologies* (ICALT2002), pp. 9–12.

Popescu, E. (2010). Adaptation provisioning with respect to learning styles in a web-based educational system: An experimental study. *Journal of Computer Assisted learning, 26*, 243–257.

Ross, P. & Come, D. (1994). Applications of genetic algorithms. AISB Quarterly 89, Ed. T.C. Fogarty, pp. 23–30.

Schiaffino, S., Garcia, P., & Amandi, A. (2008). ETeacher: Providing personalized assistance to e-learning students. *Computers & Education, 51*(4), 1744–1754.

Sessink, O., Beeftink, R., Tramper, J., & Hartog, R. (2003). Author-Defined storage in the next generation Learning Management Systems. *Proceedings of the 3rd IEEE international conference on advanced learning technologies* (ICALT'03), pp. 57–61.

Trantafillou, E., Poportsis, A., & Demetriadis, S. (2003). The design and the formative evaluation of an adaptive educational system based on cognitive. *Computer and Education, 41*(1), 87–103.

Tseng, J. C. R., Chu, H. C., Hwang, G. J., & Tsai, C. C. (2008). Development of an adaptive learning system with two sources of personalization information. *Computers & Education, 51*, 776–786.

Xenos, M. (2004). Prediction and assessment of student behaviour in open and distance education in computers using Bayesian networks. *Computers & Education, 43*(4), 345–359.

Yu, E., & Sung, K.-S. (2002). A genetic algorithm for a university weekly courses timetabling problem. *International Transactions in Operational Research, 9*(6), 703–717.

**Moushir M. El-Bishouty** received his Ph.D. from the Department of Information Science and Intelligent Systems, the University of Tokushima, Japan. Currently, he is Researcher and Tutor at Athabasca University, Canada. Since April 2011, he works as Assistant Professor at City for Scientific Research and Technological Applications, Egypt. Dr. El-Bishouty's research interests include technology enhanced learning, ubiquitous and mobile learning, adaptivity and personalization, knowledge awareness, and artificial intelligence. He has published more than 40 refereed journal papers, book chapters, and international conference papers. Besides, he received many recognized international awards.

**Ting-Wen Chang** is a postdoctoral fellow at Athabasca University, Canada. His research mainly focuses on technology enhanced learning, adaptivity and personalization, user/student modelling, multimedia Learning instruction, multi-screen learning environment, and computer assisted instruction. He has published more than 20 journal papers and international conference papers, of which three conference papers were awarded with the best paper award.

**Sabine Graf** is an Associate Professor in the School of Computing and Information Systems (SCIS) at Athabasca University, Canada. Her current research is about developing new software tools and systems mainly in the domain of education as well as extending and enhancing open source systems in this domain. Her research deals with adaptivity and personalization of learning technologies in different contexts, ranging from individual adaptation to collaborative technologies and from desktop-based to mobile, pervasive and ubiquitous scenarios. Sabine has published more than 40 refereed journal papers, book chapters, and international conference papers, of which three conference papers were awarded with the best paper award.

**Kinshuk** holds the NSERC/iCORE/Xerox/Markin Research Chair for Adaptivity and Personalization in Informatics. He is also Full Professor in the School of Computing and Information Systems and Associate Dean of Faculty of Science and Technology, at Athabasca University, Canada. Areas of his research interests include learning analytics; learning technologies; mobile, ubiquitous and location aware learning systems; cognitive profiling; and interactive technologies.

**Nian-Shing Chen** is Distinguished Professor of the Department of Information Management, National Sun Yat-sen University, Taiwan. He has published over 250 papers in academic journals, international conferences and book chapters. He is also the author of three books with one text book entitled “Learning Theory & Practice”. His current research interests include assessing e-Learning course performance; online synchronous teaching; mobile learning; knowledge management and information system development methodology.