A Genetic Algorithm-Based Multiple Characteristics Grouping Strategy for Collaborative Learning

Hui-Wen Tien[1], Yu-Shih Lin[1], Yi-Chun Chang[2(✉)], and Chih-Ping Chu[1]

[1] Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan (R.O.C)
[2] Department of Computer Science and Information Engineering, HungKuang University, Taichung, Taiwan (R.O.C)
changyc@sunrise.hk.edu.tw

Abstract. Literatures have indicated that well-balanced groups facilitate students' learning performance in collaborative learning environments. For instructors, to construct well-balanced groups needs to take efforts and time to consider large number of students and characteristics. Hence, how to automatically construct well-balanced collaborative learning groups has been a popular issue for collaborative learning. This paper proposes a genetic algorithm (GA)-based grouping strategy to assist instructors in constructing inter-homogeneous and intra-heterogeneous collaborative learning groups considering multiple student characteristics. Several data sets with different problem sizes, such as number of students and characteristics, are employed as experimental materials. Experimental results have demonstrated that the proposed grouping method is effective, efficient and robust.

Keywords: Collaborative learning · Well-balanced groups · Genetic algorithm (GA) · Multi-objective optimization

1 Introduction

Collaborative learning is defined as learners learning together in order to solve problems and accomplish common goals [1], which has validated as one useful technique to enhance students' learning performance in education context [2–4]. Further, adequate groups can assist learners to promote their learning performance through the interactions of group members [5–7]. Numerous researches have indicated the factors that influence collaborative learning success, including inter-group homogeneity, intra-group heterogeneity, grouping criteria and so on [8–10]. The grouping criterion contains different grouping characteristics of students, such as knowledge level, leadership, gender, and so on [10–13]. That is, how to construct adequate groups is one of the important issues in collaborative learning.

Since the number of students and characteristics become larger which results in that grouping problem is harder to solve by instructors, the consideration for all the student characteristics becomes a multi-objective problem. Hence, to cope the aforementioned

problems, this paper proposes a grouping strategy which adopts Genetic algorithm (GA) [11, 14, 15] with Technique for Order Performance by Similarity to Ideal Solution (TOPSIS) [7, 16] to achieve inter-homogeneous and intra-heterogeneous collaborative learning groups. The reasons adopting GA with TOPSIS in this paper is that GA with TOPSIS is one suitable method to solve grouping problems for multiple objectives problems. Besides, the proposed grouping strategy can facilitate instructors to obtain a better grouping result within a reasonable computation time, even for large scale problems. To evaluate the performance and robustness of the proposed strategy, several experiments with different problem sizes have also been conducted to compare the proposed method with other competing strategies.

The rest of this paper is organized as follows. Section 2 introduces the basic concepts of GA and TOPSIS. Section 3 describes the mathematical and algorithmic formulation of the proposed grouping strategy. Section 4 analyzes the results with different problem sizes. Section 5 presents the conclusions of the proposed strategy.

2   Related Work

This section describes the basic concepts of Genetic algorithm (GA), and Technique for Order Performance by Similarity to Ideal Solution (TOPSIS).

GA is proposed by Holland in 1975 [17]. The spirit of GA is "natural selection and survival of the fittest", and it has been widespread used in many complex optimization problems. GA comprises several components processing including chromosome encode, initial population, fitness evaluation and three genetic operator including selection operator, crossover operator and mutation operator. Each operation is detailed in the following section.

TOPSIS is proposed by Hwang and Yoon in 1981 [18]. TOPSIS has solved many decision making multi-objective problems. The procedure of TOPSIS can be described in the following four steps [19, 20]:

Step 1.   Constructs a $I \times Q$ decision matrix with respect to I alternatives and Q criteria

Step 2.   Determines the positive-ideal solution and negative-ideal solution

Step 3.   Uses the m-dimensional Euclidean distance to compute two measures

Step 4.   Computes the relative closeness to the ideal solution

3   The Proposed Strategy

The details of different components and main features of the proposed method will be shown in this section. Of them one special component focuses on pre-categorizing all students in order to ensure the quality of intra-heterogeneous groups. The other components include the encoding of solutions, initial population, fitness function, the selection, crossover, and mutation. The final subsection describes the elitism. Figure 1 presents the flow of the proposed method.
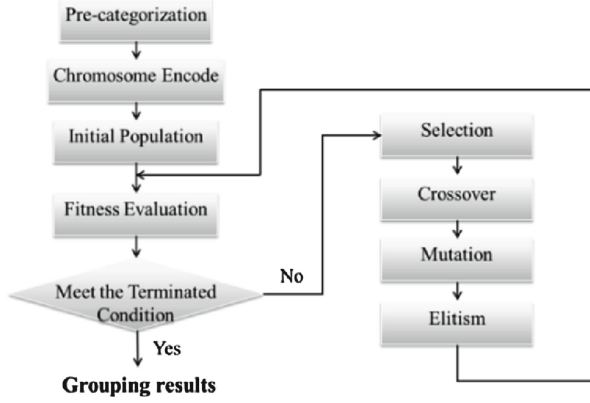
Fig. 1. The main flow of the proposed method

3.1 Pre-categorizing

To evaluate all features, each characteristic of a student in a class must be a numerical value. However, each characteristic may have different values spreading in a range, which includes following steps:

Step 1. Normalizing every considered characteristic should be taken. After normalization, every characteristic is fitted between 0 and 1. It is calculated as following:

$$Z_{sq} = \frac{X_{sq} - X_q^{min}}{X_q^{max} - X_q^{min}} \tag{1}$$

Step 2. The characteristics of each student are calculated to obtain their represented value P_s which represents their overall performance in the class with S students

Step 3. The mean value $\bar{P}$ and standard deviation P^σ of the class are derived from all students' P_s value in the class

Step 4. Three categories L_i, where $1 \leq i \leq 3$, which represent low, medium and high performance of the students were divided based on the mean $\bar{P}$ and the standard deviation P^σ. The concept is expressed as Eq. (2) and N_s is denoted as a student, where $1 \leq s \leq S$

$$\begin{aligned} L_1 &= \{N_s | P_s < \bar{P} - P^\sigma\} \\ L_2 &= \{N_s | \bar{P} - P^\sigma \leq P_s \leq \bar{P} + P^\sigma\} \\ L_3 &= \{N_s | \bar{P} + P^\sigma < P_s\} \end{aligned} \tag{2}$$

Step 5. The number of students in three categories are calculated to obtain their respective ratio R_{L_i}, where $1 \leq i \leq 3$, for the summation fitting the instructor's assignment in which each group has M members. The operations for

calculating ratio of three categories is expressed in the following *Ratio-calculating* Algorithm, where N_{L_i} denotes number of members in category L_i. Let *TE* denotes total error value of the ratio, CP_i denotes the carrying probability of the corresponding ratio R_{L_i}, and ε_i demotes the carry value of the corresponding ratio R_{L_i}, where the ε_i must be 1 or 0

Ratio-calculating Algorithm

Input   : $N_{L_1}, N_{L_2}, N_{L_3}, M$

1 : $R_{L_i} = \left\lfloor \dfrac{N_{L_i}}{\sum_{j=1}^{3} N_{L_j}} M \right\rfloor$, where $1 \le i \le 3$

2 : $TE = \sum_{i=1}^{3} \left(\dfrac{N_{L_i}}{\sum_{j=1}^{3} N_{L_j}} M - \left\lfloor \dfrac{N_{L_i}}{\sum_{j=1}^{3} N_{L_j}} M \right\rfloor \right), CP_i = \dfrac{N_{L_i}}{\sum_{j=1}^{3} N_{L_j}} M - R_{L_i}$

3 : Let $f: \{1, 2, 3\} \to \{1, 2, 3\}$ be a function, where $N_{L_{f(1)}} \le N_{L_{f(2)}} \le N_{L_{f(3)}}$

4 : for $j=1 \to 2$ do

5 :   if $R_{L_{f(j)}} = 0$

6 : $\varepsilon_{f(j)} = 1$

7 : $R_{L_{f(k)}} = \left\lfloor \dfrac{N_{L_{f(k)}}}{\sum_{l=j+1}^{3} N_{L_{f(l)}}} \times (M - \sum_{l=1}^{j} \varepsilon_{f(l)}) \right\rfloor$, where $j < k \le 3$

8 : $TE = \sum_{k=j+1}^{3} (\dfrac{N_{L_{f(k)}}}{\sum_{l=j+1}^{3} N_{L_{f(l)}}} \times (M - \sum_{l=1}^{j} \varepsilon_{f(l)}) - R_{L_{f(k)}})$

9 : $CP_{f(k)} = \dfrac{N_{L_{f(k)}}}{\sum_{l=j+1}^{3} N_{L_{f(l)}}} \times (M - \sum_{l=1}^{j} \varepsilon_{f(l)}) - R_{L_{f(k)}}$, for $j < k \le 3$

10:    else if $TE > 0$ then

11:         $\varepsilon_{f(j)}$ with a probability CP_j is 1, otherwise is 0

12:         $TE = TE - \varepsilon_{f(j)}$

13: $\varepsilon_{f(3)} = TE$

14: $R_{L_i} = R_{L_i} + \varepsilon_i$, for $1 \le i \le 3$

Output : $R_{L_1}, R_{L_2}, R_{L_3}$

Furthermore, the students are sorted into three new categories L_i', where $1 \le i \le 3$, according to the ratio shown in Eq. (3), where N_s' denotes sorted students in ascending order.

$$L_1' = \left\{ N_s' \big| 1 \le s \le \left\lfloor \frac{S}{M} \right\rfloor R_{L_1} \right\}$$

$$L_2' = \left\{ N_s' \big| \left\lfloor \frac{S}{M} \right\rfloor R_{L_1} \le s \le \left\lfloor \frac{S}{M} \right\rfloor (R_{L_1} + R_{L_2}) + S \bmod M \right\}$$

$$L_3' = \left\{ N_s' \big| \left\lfloor \frac{S}{M} \right\rfloor (R_{L_1} + R_{L_2}) + S \bmod M \le s \le \left\lfloor \frac{S}{M} \right\rfloor (R_{L_1} + R_{L_2} + R_{L_3}) + S \bmod M \right\}$$

$$\tag{3}$$

3.2   Encoding of the Solution and Initial Population

In GA, a chromosome, i.e. an individual, represents a feasible solution for problem. An individual with S students is represented by a matrix. The number of columns in the matrix corresponds to M members of each group and the number of rows corresponds to G groups. If there are S mod M remainder students, there will be S mod M groups having M + 1 members.

An individual	Level 1		Level 2			Level 3	
	1	2	3	4	5	6	7
	8	9	10	11		12	13
	14	15	16	17		18	19

Fig. 2. An example of an individual

In initial population stage, the students are randomly assigned to their corresponding levels according to the ratios of three categories. Hence, each group has three categories members with the ratios. An example of an individual is shown in Fig. 2. The Levels 1, 2 and 3 collect the students belonging to categories L_1, L_2 and L_3, respectively.

3.3   Fitness Evaluation

The main goal of the proposed method is to obtain inter-homogeneous and intra-heterogeneous groups. Therefore, it is important to define an adequate evaluation to fit the main goal. In this paper, the fitness evaluation is for evaluating the fitness value of individuals with Q characteristics. The grouping problem is to optimally assign S students, each with Q characteristics, to G groups. To address the problem, a fitness value F_i was shown in Eq. (4), where $\overline{GP}$, GP^σ and ω respectively denote the mean value of all groups of an individual, the standard deviation of all groups of an individual, and the weight of $\overline{GP}$ and GP^σ. In Eq. (4), GP_g denotes the relative closeness to the ideal solution for the gth group of an individual, where $1 \leq g \leq G$. Equation (5) expresses the mathematical formulation of GP_g, where W_q, E_g and $Z_{s,q}^g$ respectively denote the weight of criteria q, the number of members in the gth group, and the value of the qth criterion of the sth learner of the gth group of an individual. The ideal value of V_q^+ and V_q^- must be 1 or 0 according to the criteria $q, 1 \leq q \leq Q$.

$$F_i = \omega \cdot \overline{GP} + (1 - \omega) \cdot (1 - GP^\sigma),$$

$$where\ \overline{GP} = \frac{\sum_{g=1}^{G} GP_g}{G}\ \ and\ \ GP^\sigma = \sqrt{\frac{\sum_{g=1}^{G} \left(GP_g - \overline{GP}\right)^2}{G - 1}} \tag{4}$$

$$GP_g = \frac{\sqrt{\sum_{q=1}^{Q}(\frac{\sum_{s=1}^{E_g} Z_{s,q}^g}{E_g} - V_q^-)^2 \cdot W_q}}{\sqrt{\sum_{q=1}^{Q}(\frac{\sum_{s=1}^{E_g} Z_{s,q}^g}{E_g} - V_q^+)^2 \cdot W_q} + \sqrt{\sum_{q=1}^{Q}(\frac{\sum_{s=1}^{E_g} Z_{s,q}^g}{E_g} - V_q^-)^2 \cdot W_q}},$$

$$where \ E_g = \begin{cases} M+1, & g \leq S \bmod M \\ M, & others \end{cases}$$
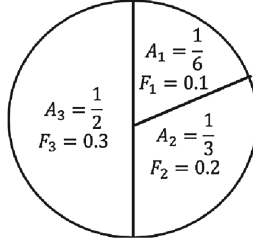
(5)

Fig. 3. Roulette wheel example

3.4   Selection Operator

The objective of selection operator is selecting the better individuals from current population to ensure the more suitable individuals can be preserved in next generation. Roulette wheel selection mechanism [21] is employed in this paper. Figure 3 presents an example of Roulette wheel. An individual i of current population has a portion A_i of all areas in the roulette according to their fitness value F_i and has a random probability to select the corresponding individual in next generation. The larger proportion an individual has in the area, the more probability it can be preserved in next generation.

3.5   Crossover Operator

The goal of crossover operator is interchanging genes of individuals to recombine features. C1 operator is one of the famous crossover operator [22]. One version of C1 operator [10] is modified to fit the chromosome data structure in this proposed strategy. There are two steps in the proposed version. Firstly, chooses two parents with cross-over probability P_c to do crossover and randomly generates crossover points for each group. Next, offspring preserves the corresponding parent's left part of genes according to the crossover point of each row while the right part of genes are derived from the order of genes in different levels of the other parent.

3.6   Mutation Operator

The goal of mutation operator is altering genes to reproduce new individuals [23]. The mutation operator includes two steps in the proposed strategy. In the first step, each

level has a probability P_l to do mutation. In the second steps, if a level is selected then two genes are randomly chosen to swap in the selected level.

3.7 Elitism

In this stage, the proposed strategy uses the elitist schema to select the best individual to replace the worst one from the population in generation k. Therefore, it can be sure that the population is better in generation $k + 1$ than in generation k and the best individual will be kept in $k + 1$ generation.

4 Experiments and Discussions

This paper employs several data sets, such as number of students and characteristics, with different problem sizes as experimental materials to analyze the solution quality, executing time, search speed and stability. Subsect. 4.1 describes the experimental setting. Subsect. 4.2 compares the performance including execution time and search speed of the proposed strategy with other competing methods. Subsect. 4.3 analyzes the solution quality of the proposed strategy. Subsect. 4.4 presents the analyses of robustness.

4.1 Experimental Setting

All experiments run on a laptop computer with an Intel Core i7, 2.10 GHz processor, 4 GB RAM using C programming language in CodeBlocks environments. There are five students in each group. The probability of crossover is set to 0.85, the probability P_l is set to 0.15, the population size is set to 20 and the number of generation is set to 100. Each scenario is run 100 times for calculating the average fitness value of the 100 runs in order to get the objective representation of fitness value and execution time of each data set. There are different problem sizes of simulated (uniformly distribution) data sets with four characteristics to compare among random, GA [10] and the proposed method. The first and the second characteristics are set to positive features while the third and the fourth are set to negative features. Each weight of characteristic is set to 0.5.

4.2 Performance Comparisons

For random, GA and the proposed strategy, Tables 1 and 2 present the performance results with different number of students and characteristics, where the performance contains fitness value and execution time. Table 1 shows that random method can save more execution time than GA and the proposed method, but it has poorer fitness value than the others with different number of students. In addition, the proposed strategy can use less execution time to obtain the nearer optimal solution than GA method. Table 2 shows that no matter what the number of characteristics is, the proposed strategy is more effective

Table 1. Compare the performance among random, GA, and the proposed strategy with increasing number of students

Number of students	Random		GA		The proposed method	
	Fitness value	Execution time	Fitness value	Execution time	Fitness value	Execution time
50	0.777272	0.00024	0.786725	0.02818	0.787620	0.02511
100	0.775255	0.00037	0.780705	0.09021	0.783649	0.08892
150	0.778487	0.00065	0.782882	0.20064	0.786500	0.18539
300	0.776038	0.00124	0.778784	0.72348	0.783061	0.70643
500	0.778968	0.00227	0.780785	2.13675	0.785084	2.02228

Table 2. Compare the performance among random, GA, and the proposed strategy with increasing number of characteristics

Number of characteristics	Random		GA		The proposed method	
	Fitness value	Execution time	Fitness value	Execution time	Fitness value	Execution time
10	0.680890	0.00054	0.702620	0.03272	0.704872	0.02982
20	0.689808	0.00083	0.704013	0.04073	0.705553	0.03615
30	0.696040	0.00100	0.707440	0.04616	0.708749	0.04256
50	0.694084	0.00152	0.702103	0.06355	0.702952	0.05873

than the other two. To sum, the proposed strategy has the most effective than the other two methods.

Furthermore, in order to compare the search speed of the proposed strategy with GA method, Fig. 4 presents the variations of optimal fitness value within 1000 generations. It shows the proposed strategy is more efficient than GA because that when the optimal fitness value is set to 0.787, the proposed strategy needs 54 generations and GA needs 340 generations to evolution. In other words, GA will spend more time and efforts to get the same fitness value than the proposed strategy.
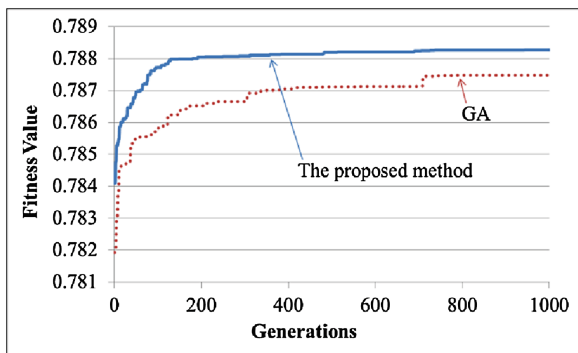

Fig. 4. Evolution of optimal fitness of 50 students with GA and the proposed strategy

4.3 Comparison of Grouping Results

In this subsection, to compare the grouping results of three methods, Tables 3 and 4 present the standard deviation of the inter-groups and intra-groups against different number of students and characteristics. Table 3 shows that despite different number of students, the proposed method always has smaller standard deviation of inter-groups and larger standard deviation of intra-groups than the random and GA methods. Table 4 shows that no matter what number of characteristics is the standard deviation of inter-groups and intra-groups of the proposed method is better than random and GA methods. In other words, the proposed method has better grouping results than random and GA methods against different number of characteristics.

Table 3. Compare standard deviation of inter-groups and intra-group among random, GA, and the proposed strategy with increasing number of students

Number of students	Random		GA		The proposed method	
	Inter-SD	Intra-SD	Inter-SD	Intra-SD	Inter-SD	Intra-SD
50	0.021734	0.314234	0.003427	0.355971	0.001648	0.359978
100	0.018829	0.276894	0.008167	0.304798	0.002329	0.316636
150	0.018885	0.273964	0.010420	0.300004	0.003312	0.315074
300	0.019386	0.284001	0.014084	0.302678	0.005690	0.323426
500	0.017681	0.260933	0.014189	0.273619	0.005806	0.296865

Note: Inter-SD: standard deviation of inter-groups; Intra-SD: standard deviation of intra-groups.

Table 4. Compare standard deviation of inter-groups and intra-group among random, GA, and the proposed strategy with increasing number of characteristics

Number of characteristics	Random		GA		The proposed method	
	Inter-SD	Intra-SD	Inter-SD	Intra-SD	Inter-SD	Intra-SD
10	0.049219	0.088284	0.007302	0.099160	0.003085	0.102605
20	0.032909	0.059400	0.005383	0.064895	0.002398	0.067735
30	0.026397	0.050038	0.004316	0.055073	0.001742	0.057069
50	0.018552	0.033570	0.003243	0.036931	0.001505	0.038016

Note: Inter-SD: standard deviation of inter-groups; Intra-SD: standard deviation of intra-groups.

4.4 Comparisons of Robustness

To analyze the robustness of optimal fitness values obtained from the random, GA and the proposed strategy, two different data sets, such as different number of students and characteristics, will be conducted to compare the proposed strategy with random and GA methods. Each method would run 100 times with each data set then the standard deviation of these 100 optimal fitness values is computed. Figure 5 shows the variations of standard deviation of the optimal fitness values derived from the proposed strategy is significantly better than the other competing methods. The proposed strategy is more stable than the other two with different number of students. Figure 6 shows that
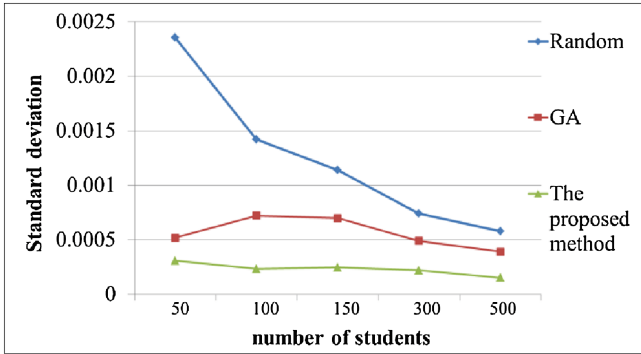
Fig. 5. Variations of standard deviation of the optimal fitness values derived by random, GA and the proposed strategy as the number of students increase
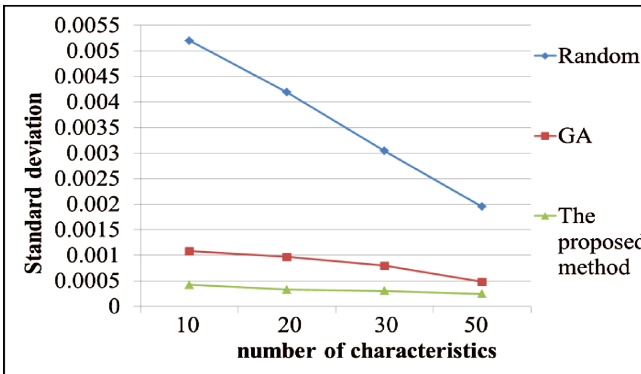

Fig. 6. Variations of standard deviation of the optimal fitness values derived by random, GA and the proposed strategy as the number of students' characteristics increases

the variations of standard deviation of optimal fitness values derived from the proposed strategy is more stable than the other two methods.

5   Conclusion

This paper has proposed the grouping strategy based on GA to address multiple characteristics grouping problems for cooperative learning environments, which constructs better inter-homogeneous and intra-heterogeneous groups. In this paper, the pre-categorization stage and a well-designed chromosome-like data structure has designed to enhance the performance of the proposed strategy. Furthermore, a novel mutation operator is designed for fitting the chromosome, which improves the performance but doesn't lead to non-convergence of the solution.

To evaluate the proposed strategy, several data sets with different problem sizes, such as number of students and characteristics, were conducted to compare the solution quality, executing time, search speed and stability of the proposed strategy with the other methods. The experimental results have demonstrated the proposed grouping strategy (1) possesses better performance, grouping result and robustness than the random and GA methods, and (2) achieves the goal of obtaining better inter-homogeneous and intra-heterogeneous groups.

References

1. Dillenbourg, P.: What do you mean by collaborative learning? In: Dillenbourg, P. (ed.) Cognitive and Computational Approaches Collaborative-Learning, pp. 1–19. Elsevier, Oxford (1999)
2. Dillenbourg, P.: Collaborative Learning: Cognitive and Computational Approaches. Advances in Learning and Instruction Series. ERIC, New York (1999)
3. Dewiyanti, S., et al.: Students' experiences with collaborative learning in asynchronous computer-supported collaborative learning environments. Comput. Hum. Behav. **23**(1), 496–514 (2007)
4. Kreijns, K., Kirschner, P.A., Jochems, W.: The sociability of computer-supported collaborative learning environments. Educ. Technol. Soc. **5**(1), 8–22 (2002)
5. Webb, N.M.: Student interaction and learning in small groups. Rev. Educ. Res. **52**(3), 421–445 (1982)
6. Barkley, E.F., Cross, K.P., Major, C.H.: Collaborative Learning Techniques: A handbook for College Faculty. Jossey-Bass, San Francisco (2004)
7. Wang, Y.-M., Elhag, T.: Fuzzy TOPSIS method based on alpha level sets with an application to bridge risk assessment. Expert Syst. Appl. **31**(2), 309–319 (2006)
8. Dalton, D.W., Hannafin, M.J., Hooper, S.: Effects of individual and cooperative computer-assisted instruction on student performance and attitudes. Educ. Tech. Res. Dev. **37**(2), 15–24 (1989)
9. Hooper, S., Hannafin, M.J.: Cooperative CBI: The effects of heterogeneous versus homogeneous grouping on the learning of progressively complex concepts. J. Educ. Comput. Res. **4**(4), 413–424 (1988)
10. Moreno, J., Ovalle, D.A., Vicari, R.M.: A genetic algorithm approach for group formation in collaborative learning considering multiple student characteristics. Comput. Educ. **58**(1), 560–569 (2012)
11. Yin, P.-Y., et al.: A particle swarm optimization approach to composing serial test sheets for multiple assessment criteria. Educ. Technol. Soc. **9**(3), 3–15 (2006)
12. Hwang, G.-J., et al.: An enhanced genetic approach to composing cooperative learning groups for multiple grouping criteria. Educ. Technol. Soc. **11**(1), 148–167 (2008)
13. Yannibelli, V., Amandi, A.: A deterministic crowding evolutionary algorithm to form learning teams in a collaborative learning context. Expert Syst. Appl. **39**(10), 8584–8592 (2012)
14. Falkenauer, E.: Genetic algorithms and grouping problems. Wiley, New York (1998)

15. Wang, D.-Y., Lin, S.S., Sun, C.-T.: DIANA: A computer-supported heterogeneous grouping system for teachers to conduct successful small learning groups. Comput. Hum. Behav. **23**(4), 1997–2010 (2007)
16. Chen, C.-T.: Extensions of the TOPSIS for group decision-making under fuzzy environment. Fuzzy Sets Syst. **114**(1), 1–9 (2000)
17. Holland, J.H.: Adaptation in Natural and Artificial Systems. The University of Michigan Press, Ann Arbor (1975)
18. Hwang, C.-L., Yoon, K.: Multiple attribute decision making. Springer, New York (1981)
19. Shyur, H.-J., Shih, H.-S.: A hybrid MCDM model for strategic vendor selection. Math. Comput. Model. **44**(7), 749–761 (2006)
20. Yu, X., et al.: Rank B2C e-commerce websites in e-alliance based on AHP and fuzzy TOPSIS. Expert Syst. Appl. **38**(4), 3550–3557 (2011)
21. Weise, T., Global optimization algorithms–theory and application. Self-Published (2009)
22. Reeves, C.R.: A genetic algorithm for flowshop sequencing. Comput. Oper. Res. **22**(1), 5–13 (1995)
23. Goldberg, D.E.: Genetic algorithms in search, optimization, and machine learning. Addison-Wesley Longman Publishing, Boston (1989)