A Genetic Algorithm Approach for Inter and Intra Homogeneous Grouping Considering Multi-student Characteristics

A. M. Aseere

King Khaled University
College of Computer Science
ABHA, The Kingdom of Saudi Arabia

Abstract—**This paper addresses the problem of group formation in collaborative learning by considering the students' characteristics. The proposed solution is based on a Genetic Algorithm (GA), which minimizes an objective function that has two main aims. Indeed, the proposed GA's fitness function helps to achieve two objectives: Fairness in the formation of different groups, resulting in intergroup homogeneity, and a low gap in the levels of students within a group, which corresponds to intragroup homogeneity. Exhaustive experiments were conducted using three different sizes of randomly generated data sets and several crossover operators. Indeed, the order crossover and the crossovers based on random keys representation are experimented. The reported results show that the proposed approach guarantees the efficient grouping of students. In addition, comparisons with existing approaches based on GA confirm the ability of the proposed approach to provide greater intergroup and intragroup homogeneity. In addition, the uniform crossover based on random keys representation ensures better grouping quality than do the other experimented crossover operators.**

Keywords—*Genetic algorithm; group formation; intragroup homogeneity; intergroup homogeneity; fitness; permutation; random keys representation*

I. INTRODUCTION

Group formation is a crucial issue in collaborative learning. At present, group work is becoming increasingly recommended in various disciplines. Group work is mandatory for the realization of projects that have a high workload, and which generally require different skills. It is well established that group work is extremely beneficial for the training of students, and allows them to improve the group-work skills that are highly sought by recruiters.

The task of forming groups is a very delicate one due to the disparity of students in terms of skills. The success of student groups relies on several factors, such as personality, expertise in performance, and the collaboration of the students within the group. In collaborative learning [1], experts recommend forming a group consisting of students with complementary skills while ensuring the narrowest possible gap between the levels of students. Very large differences among group members can hinder cooperation. In addition, when forming groups from a set of students, the grouping technique must ensure fairness between groups. In fact, the

success of each student depends on the success of the group to which the student belongs.

Different approaches are presented in the literature to answer research questions related to the problem of group formation. The systemic literature review of the subject presented in [2] cites several works that answered the four important research questions related to the problem, namely the homogeneity/heterogeneity of groups, the learner characteristics considered, the ideal group size, and the techniques used for automated group formation. The works cited in [2] present different answers and points of view regarding each research question. The size of the groups may depend on the total number of students, the tutor's choice, or the supervisory capacity of the institution/faculty. In [3], it is mentioned that members become less productive when the group size is large. Thus, groups consisting of three to seven members may be reasonable. Whether the type of group is heterogeneous or homogeneous has been studied in several works. According to [2], most works have addressed heterogeneous grouping, and fewer were interested in homogeneous groups.

The heterogeneity of a group can assist students to achieve good results and can help them to learn how to collaborate and interact with different types of classmates. However, excessive differences among students in a group can hinder cooperation and collaborative learning [1]. In [4], authors present a literature review concerning the problem of group formation. Indeed, based on the studied literature, the authors provided taxonomies of the characteristics of group formation and surveyed the techniques for forming groups. Automatic heterogeneous or homogeneous grouping can be performed based on students' characteristics [5]. These characteristics could be academic (grades, tests, self-evaluations, and so on), cognitive (learning styles intelligence types, and so forth), personality traits (such as leadership skills), or other considerations. Various types of characteristics are used for automatic group formation based on different optimization techniques [2], [4]. In this paper, a specific objective function based on different types of characteristics is proposed for achieving the desired type of grouping.

GA based approaches have been used in several studies to accomplish automatic group formation [2], [4]. In [1], the authors proposed a two-step approach, namely a

categorization step that aims to maintain intragroup diversity and intergroup balance, followed by an optimal formation step that uses a GA to provide an approximate solution for group formation after determining each student's category. In [6], an adaptive GA is employed to seek the best combinations of groups. The maximized fitness function is the sum of two terms: the balanced score based on score category assigned to the average scores of the students, and the inverse of the absolute distance between the average scores of groups. In [7], the maximally diverse grouping problem is addressed. The absolute distance between different attributes of various groups is maximized. A hybrid technique combining a local search algorithm and a GA was used to solve the posed problem. An approach based on GA was proposed to form heterogeneous groups [8]. The index of learning styles and the academic attributes of students were used. The proposed algorithm aims to minimize the distances among separate groups. Using GA approach to group formation considering multiple student characteristics was proposed [5]. In the aim to form inter and intra homogeneous groups, the considered computed fitness function of the GA corresponds to the squared differences with regard to all the characteristics considered for each group and the entire sample. However, the proposed fitness function focuses on intergroup homogeneity rather than on intragroup homogeneity. Since the average values of the characteristics of each group are considered, it may be difficult to avoid having a combination of students with high skills and others with very low ones within a group. In fact, the success of an unbalanced group is not guaranteed. In general, the students that are at higher levels take on most of the workload.

In this paper, a two-objective fitness function is proposed to achieve group formation with intragroup and intergroup homogeneity. The average levels of the groups must be very close, and the gaps in the levels of students within a group must be low. Thus, the proposed fitness function of the GA is the sum of two terms; the first term corresponds to the fitness function proposed in [5], and the second term maximizes the sum of the minimum of the characteristics across all the groups. As the algorithm computes the sum of the minimum characteristics for each group, and then attempts to maximize the minimum value of these sums, the aim is to improve the intragroup homogeneity.

The successful use of a GA relies on appropriate coding of the genotypes (chromosomes) and appropriate operators to generate a new population from the current one. Since each student can only be in one group, permutation can be any valuable solution for genotype coding. Nevertheless, specific crossover and mutation operators must be used for coding based on permutation. In [9], the authors showed that order crossover and interchanging mutation were best choices for permutation. However, in [10], random keys are presented as a robust representation technique that guarantees maintaining feasibility from parents to offspring. Thus, random keys representation makes it possible to use conventional crossover operators, such as a single-point crossover, a two-point crossover, and the like [11]. The random keys representation

technique has been used successfully in job scheduling problem [12]. Permutation is used to code chromosomes, and the grouping corresponding to each permutation is done in a very similar way to the coding proposed in [13]. In [14] some recommendations based on exhaustive simulation results are provided for setting the values for the different parameters of a GA.

The remainder of the paper is organized as follows: Section 2 presents the mathematical formulation of the proposed group formation problem. The proposed fitness function is also defined. Section 3 describes the use of a GA, and presents the different operators used. Section 4 evaluates the performance of the proposed approach. The experimental protocol is presented, and the results obtained are reported and discussed. Finally, Section 5 provides some concluding remarks and recommendations for future work.

II. PROBLEM FORMULATION

This paper proposes a method for combining a number of students into groups in the context of collaborative learning. The group formation is based on various characteristics of the students. The characteristics considered here correspond to the grades of the students in different modules such as Math, Programming Skills, Communications Skills, their GPAs, and so on. The characteristics considered for the groupings can be set by the administration of the faculty according to the requirements of the collaborative learning program.

This paper proposes a method for combining a number of students into groups in the context of collaborative learning. The group formation is based on various characteristics of the students. The characteristics considered here correspond to the grades of the students in different modules such as Math, Programming Skills, Communications Skills, their GPAs, and so on. The characteristics considered for the groupings can be set by the administration of the faculty according to the requirements of the collaborative learning program.

Let us assume that we have to group N students, and that $\mathcal{S} = \{s^1, s^2, \cdots, s^N\}$ is the set of students. Each student $(s^i)_{i=1,\cdots,N}$ is characterized by an array of L characteristics, denoted as $C^i = (c_1^i, c_2^i, \cdots, c_L^i)$. These characteristics could be of different forms, such as academic (grades, tests, self-evaluations, and so on), cognitive (learning styles, intelligence types, and so forth), personality (leadership, personality traits, and the like), or others. The numerical values of these characteristics may vary in different ranges. In such a case, the data must be normalized prior to the optimization process. Normalization aims to reduce the variations of all the characteristics' values to same interval.

Let us assume that $N_G = \frac{N}{P}$ is the number of groups, where P is the number of students per group. For each group $g = 1, \dots, N_G$, we associate a matrix M_g made of the arrays of the P students' characteristics. Thus, M_g is a $P \times L$ matrix defined as follows: $M_g = (C^{1\{g\}}, \cdots, C^{P\{g\}})^T$, where $(C^{i\{g\}})_{i=1,\cdots,P} \in \{C^1, C^2, \cdots, C^N\}$.

The idea is to distribute students within the groups in order to achieve two objectives:

1) Homogeneity between groups, and

2) A low gap in the levels of students within the same group.

With reference to [5], a criterion that focuses on achieving the first objective was proposed. In this paper, a modified objective function that helps to satisfy both objectives is proposed.

1) Intragroup homogeneity: The aim is to reduce the deviation between the average of each characteristic of a group and the average of each characteristic of all the students. As proposed in [5], the best grouping is one that minimizes the objective function f_1, defined as:

$$f_1 = \sum_{g=1}^{N_G}\left[(\overline{C_1} - \overline{M_{g,1}})^2 + (\overline{C_2} - \overline{M_{g,2}})^2 + \cdots + (\overline{C_L} - \overline{M_{g,L}})^2\right] \qquad (1)$$

where $\overline{C_j} = \sum_{i=1}^{N} c_j^i$ and $\overline{M_{g,j}} = \frac{1}{P}\sum_{i=1}^{P} c_j^{i\{g\}}$ correspond respectively to the average of the characteristic j over N students, and the average of the characteristic j over P students in the group g.

2) Intragroup homogeneity: In order to guarantee intragroup homogeneity, maximizing the objective function f_2 is proposed as follows:

$$f_2 = \min_{g=1,\cdots,N_G}(\min(M_{g,1}) + \min(M_{g,2}) + \cdots + \min(M_{g,L})) \qquad (2)$$

where $\min(M_{g,j}) = \min(c_j^{1\{g\}}, c_j^{2\{g\}}, \cdots, c_j^{P\{g\}})$ corresponds to the minimum of the characteristic j over P students in a group g. When attempting to minimize the function f_1, there is a significant chance of finding groups in which there is a large difference in the levels of students within the same group. This effect can be reduced by maximizing the function f_2 at the same time.

3) Fitness function of the GA: This paper uses the GA for the automatic grouping of students. For the posed problem, the GA seeks a set of feasible solutions for the optimal solution (the best grouping of the students) that minimizes a specific fitness function. The latter is defined in the following.

Considering equations (1) and (2), the fitness computed by the GA for each individual, i, in the population is defined as:

$$F^i = \beta f_1^i + \frac{1-\beta}{f_2^i + \varepsilon}, \qquad (3)$$

where, f_1^i, f_2^i correspond respectively to the objective functions, f_1, f_2 are computed for the individual i, ε is a small parameter added to avoid division by 0, and $0 \le \beta \le 1$ is a weighting factor.

The best individual in a population is the one that has the lowest value of the fitness function, as defined in (3). The GA tends to decrease the lowest value of the fitness over a number of generations. In this paper, we propose attaching equal importance to f_1^i and f_2^i, so that β is fixed at 0.5. However,

when β is set to 1.0, we retrieve the fitness function proposed in [5], defined as

$$F^i = f_1^i \qquad (4)$$

III. GENETIC ALGORITHM DEPLOYMENT

For a given population (also called a generation), the classical GA executes the following steps in order:

1) The evaluation of the fitness of each individual in the population;

2) the selection of the best individual for breeding to create the new population; and

3) the application of the elitism principle followed by specific crossover and mutation operators to generate the new population.

In a conventional GA. the third step is constituted by the crossover step followed by the mutation step. However, in [15] an adaptation process is proposed, in order to adapt the individuals in the population to the best solution of the algorithm. In our work, we choose the elitism process to copy a few best solutions from the current generation to the future one.

Starting from an initial population, the algorithm repeats steps 1, 2, and 3 during several iterations. In each iteration, it creates a new population that may contain the best individual corresponding to the best grouping of students. The algorithm can be stopped by limiting the number of generations, by setting a threshold for the lowest fitness level, or by limiting the execution time of the algorithm. In general, the number of generations is used as the criterion for stopping. The coding of the individuals in the population, also called genotypes, depends on the posed problem, and is a key factor in selecting the specific genetic operators (crossover and mutation).

1) Genotype coding: Since each student can only be part of one group, it is a proposed that a permutation of the set of the students' indexes $\{1, 2, \cdots, N\}$ be used as a genotype.

Let us consider a $\sigma = \{\sigma(0), \sigma(1), \cdots, \sigma(N-1)\}$, where $\sigma(i)$ is the integer at the ith position of the permutation, and corresponds to the student $s^{\sigma(i)}$. Thus, each individual in the population is a permutation of integers from 0 to $N-1$, and the grouping of the students into groups of P students is performed as follows: The first group consists of the first P integers of the permutation $\sigma(0), \ldots, \sigma(P-1)$, the second group is formed by $\sigma(P), \ldots, \sigma(2P-1)$, and so on. In general, the ith group is formed by $\sigma((i-1) \times P), \ldots, \sigma(i \times P - 1)$.

2) Crossover operators: The crossover operators used with permutations are extremely specific. It is not possible to use a conventional single-point crossover, a two-point crossover, and so forth [11] because it is difficult to maintain feasibility from parents to offspring. In [9], the authors showed that the order crossover was the best operator for permutations. In this paper, we experimented with other crossover operators based on the random keys representation presented in [10]. The idea was to use specific mapping to correspond to each permutation in a list of random numbers

(keys). Thus, it was possible to apply a single-point crossover, a two-point crossover, or a uniform crossover to the lists of random numbers. After breeding, inverse mapping was applied to transform the lists of random numbers into permutations.

The use of order crossover and random keys representation for single-point, two-point, and uniform crossovers are presented in this paper.

3) Mutation operator: The interchanging mutation operator is recommended for genotypes coded as permutations [9], [5]. Thus, we propose to use the interchanging mutation with a low mutation rate [14].

4) Selection of individuals for breeding: The strategy based on elitism was proposed to create a new population from a current one, while maintaining the same size [11]. From the current population, α individuals with the best fitness were copied to the new population. The remaining individuals were obtained via the application of crossover operators on the selected parents for breeding. In fact, the selection of parents for breeding was performed via the roulette-wheel method based on the normalized cumulative of the fitness [11].

In this section, we investigate the performance of the proposed grouping approach based on the GA that minimizes the proposed fitness function, as defined in (3), for $\beta = 0.5$.

IV. COMPUTATIONAL EXPERIMENTS AND RESULTS ANALYSIS

A. Experimental Protocol

Computational experience was carried out for three data sets that were generated randomly with three different sizes; small ($N = 20$), medium ($N = 40$), and large ($N = 60$). For all the data sets, the four characteristics that were considered ($L = 4$) were 'Math', 'Programming Skills', 'Communication Skills', and 'GPA'. These characteristics may be required for groups working in a computer science collaborative learning program; for example, on a final graduation project. Since the values for the GPA are different from the values of the other characteristics, the normalization of the data set is mandatory prior to the optimization process.

The normalization approach proposed in [5] was adopted. Accordingly, each characteristic $\left(c_j^i\right)_{j=1,\cdots,L}$ of each student $(s^i)_{i=1,\cdots,N}$, was normalized according to the following procedure:

$$c_j^i = \frac{c_j^i - \left(c_j^i\right)_{min}}{\left(c_j^i\right)_{max} - \left(c_j^i\right)_{min}} \tag{5}$$

where $\left(c_j^i\right)_{min} = min_{i=1,N}(c_j^i)$, and $\left(c_j^i\right)_{max} = max_{i=1,N}(c_j^i)$. Following this normalization, the values of the normalized characteristics varied in the interval of [0,1].

1) Setting the parameters of GA: Based on [14], for N students, the size of the population varied from N to 3N. In all the experiments, we set the size of the population as 2N. We

conducted several experiments on the order crossover and the single-point crossover, the two-point crossover, and the uniform crossover based on random keys representation. The aim is to determine which one is the most appropriate for the proposed genotype coding. The mutation rate was fixed at 0.001 for all the performed experiments and the number of generations was the criterion for stopping the algorithm.

2) Assessment criteria: Obviously, the first criterion used to assess the performance of the proposed approach to group formation based on GA was the best fitness, F_{best}, which is computed as follows:

$$F_{best} = min_{generations}\left(min_{i\in population} F^i\right) \tag{6}$$

To evaluate the importance of the proposed fitness function (3) compared to the one defined by (4), we proposed computing the best grouping corresponding to F_{best} using the following parameters:

$$Avg_{min} = min_{g=1,\cdots N_g}\left(\textstyle\sum_{j=1}^{L} \overline{M_{g,j}}\right) \tag{7}$$

$$Avg_{max} = max_{g=1,\cdots N_g}\left(\textstyle\sum_{j=1}^{L} \overline{M_{g,j}}\right) \tag{8}$$

$$Avg_{median} = median_{g=1,\cdots N_g}\left(\textstyle\sum_{j=1}^{L} \overline{M_{g,j}}\right) \tag{9}$$

The parameters (7), (8), and (9) characterize the degree of homogeneity between different groups, thus corresponding to the first objective. In fact, the higher the values of these parameters the greater the homogeneity between groups.

In addition to the parameters (7), (8), and (9) that evaluate the average values of the characteristics of the groups, and in order to evaluate the gaps in the levels of students within each group, we proposed computing for the best grouping corresponding to F_{best} using the following parameters:

$$Min_{min} = min_{g=1,\cdots N_g}\left(\textstyle\sum_{j=1}^{L} min(M_{g,j})\right) \tag{10}$$

$$Min_{max} = max_{g=1,\cdots N_g}\left(\textstyle\sum_{j=1}^{L} min(M_{g,j})\right) \tag{11}$$

$$Min_{median} = median_{g=1,\cdots N_g}\left(\textstyle\sum_{j=1}^{L} min(M_{g,j})\right) \tag{12}$$

Parameters (10), (11), and (12) correspond to the second objective. In fact, the higher these parameters' values, the greater the intragroup homogeneity.

B. Results Analysis

In the first step in the performance analysis of the proposed approach for group formation, we conducted simulations to assess the importance of the fitness (3) ($\beta = 0.5$) compared to fitness (4) ($\beta = 1.0$). For these simulations, we considered three different sizes (20, 40, and 60) of the randomly generated data sets and the four different crossover operators. The number of generations was set at 200, the mutation rate at 0.001; the Elitism rate, α, was equal to 10% of the population size and finally, for the uniform crossover based random keys, the parameter of the binomial law was set at 0.7 [12].

The results obtained are presented in Table I, Table II, and Tab. III. for $N = 20$, $N = 40$, and $N = 60$, respectively. The reported results are the average over 20 runs. We fixed the

population size at 2N [12] and the initial population is constituted by 2N randomly generated permutations of integer numbers from 0 to $N - 1$.

The results presented in Table I, Table II, and Table III show that the quality of the groupings obtained using the proposed approach was much more appropriate than was the one obtained when using the fitness function defined by (4) and proposed in [5]. This was valid for all the experimental crossover operators. In fact, the values of the parameters (6) - (12) assessing the quality of the groupings obtained via the proposed approach are higher than the values of those obtained via the approach based on the fitness (4). As expected, adding the maximization of the sum of the minimum characteristics of all the groups in the objective function produced a significant improvement in the quality of the groupings.

In addition, the obtained results show that the order crossover was not the most appropriate operator for population breeding for all the data sets in the proposed approach. In fact,

the two-point crossover and the uniform crossover based on random keys representation appeared to be more efficient.

For a further investigation of the performance of the proposed approach, Table IV, Table V and Table VI show the simulation results obtained for the three data sets with 600 generations. For each data set, we considered the crossover operators that provided the best results after 200 generations (Table I, Table II, and Table III). The reported results were the average over 20 runs, and the GA used a population of size 2N for a data set of size N.

The results presented in Table IV, Table V and Table VI confirm the superiority of the proposed GA based on fitness (3) over the GA based on fitness (4). As expected, increasing the number of generations assisted the algorithm to achieve better results than those obtained with 200 generations for all the crossover operators considered. Based on these reported results, we can state that the proposed GA with the uniform crossover based on random keys representation ensures better grouping quality than do the other crossover operators.

TABLE I.   COMPARATIVE ANALYSIS OF THE PERFORMANCE OF GENETIC ALGORITHM BASED ON FITNESS (3) VERSUS FITNESS (4) (POPULATION SIZE = 40, GENERATIONS = 200, MUTATION RATE = 0.001, $\alpha = 4$)

Random Data Set of 20 Students								
Crossover	Fitness	F_{best}	Avg_{min}	Avg_{max}	Avg_{median}	Min_{min}	Min_{max}	Min_{median}
Order	(3)	0,37819	2,31503	2,58601	2,43335	1,57611	2,00153	1,69432
	(4)	0,05711	2,01043	2,5641	2,29812	1,00369	1,86938	1,49700
R. Keys Single-point	(3)	0,36566	2,33045	2,63612	2,46733	1,62994	2,07089	1,72847
	(4)	0,05715	2,06049	2,52151	2,28918	1,04332	1,84752	1,45798
R. Keys Two-point	(3)	0,35583	2,34723	2,62273	2,47493	1,63435	1,93480	1,75328
	(4)	0,04701	2,04859	2,50071	2,27728	1,03092	1,72159	1,43301
R. Keys Uniform	(3)	0,34555	2,2946	2,67124	2,51640	1,63116	2,06780	1,80386
	(4)	0,04712	2,07614	2,50701	2,30293	1,02322	1,81396	1,49647

TABLE II.   COMPARATIVE ANALYSIS OF THE PERFORMANCE OF GENETIC ALGORITHM BASED ON FITNESS (3) VERSUS FITNESS (4) (POPULATION SIZE = 80, GENERATIONS = 200, MUTATION RATE = 0.001, $\alpha = 8$)

Random Data Set of 40 Students								
Crossover	Fitness	F_{best}	Avg_{min}	Avg_{max}	Avg_{median}	Min_{min}	Min_{max}	Min_{median}
Order	(3)	0,61242	2,04115	2,73540	2,27964	1,10697	2,19040	1,53068
	(4)	0,15982	1,91476	2,56528	2,20301	0,52619	2,01572	1,41403
R. Keys Single-point	(3)	0,58795	2,02214	2,61416	2,28030	1,16734	2,00016	1,48775
	(4)	0,19972	1,88904	2,53112	2,21207	0,64927	2,00527	1,36829
R. Keys Two-point	(3)	0,56642	2,01923	2,66400	2,29310	1,24155	2,10547	1,57232
	(4)	0,17039	1,90926	2,59815	2,26121	0,55352	2,04251	1,48417
R. Keys Uniform	(3)	0,58600	1,99183	2,64434	2,29607	1,14184	2,05865	1,49780
	(4)	0,17349	1,91332	2,54612	2,21842	0,62825	1,93114	1,37765

TABLE III. COMPARATIVE ANALYSIS OF THE PERFORMANCE OF GENETIC ALGORITHM BASED ON FITNESS (3) VERSUS FITNESS (4) (POPULATION SIZE = 120, GENERATIONS = 200, MUTATION RATE = 0.001, α = 12)

Random Data Set of 60 Students								
Crossover	Fitness	F_{best}	Avg_{min}	Avg_{max}	Avg_{median}	Min_{min}	Min_{max}	Min_{median}
Order	(3)	0,64552	1,92776	2,53707	2,23878	1,09650	2,07752	1,52831
	(4)	0,22394	1,82434	2,51451	2,18508	0,62140	2,02689	1,39662
R. Keys Single-point	(3)	0,68964	1,90218	2,58344	2,24333	1,05206	2,12256	1,52230
	(4)	0,30427	1,80189	2,52980	2,18551	0,59824	2,04232	1,41119
R. Keys Two-point	(3)	0,68300	1,88965	2,58525	2,22402	1,08020	2,12506	1,52543
	(4)	0,30175	1,84326	2,53305	2,17113	0,66259	2,02101	1,38816
R. Keys Uniform	(3)	0,66417	1,92619	2,56619	2,26264	1,15363	2,11301	1,56361

TABLE IV. SIMULATION RESULTS FOR TWO-POINT CROSSOVER AND UNIFORM CROSSOVER BASED ON RANDOM KEYS REPRESENTATION (POPULATION SIZE = 40, GENERATIONS = 600, MUTATION RATE = 0.001, α= 4).

Random Data Set of 20 Students								
Crossover	Fitness	F_{best}	Avg_{min}	Avg_{max}	Avg_{median}	Min_{min}	Min_{max}	Min_{median}
R. Keys Two-point	Fitness (3)	0,354453	2,319143	2,605539	2,462328	1,602172	2,014233	1,720088
	Fitness (4)	0,042657	2,080786	2,489469	2,283906	1,063818	1,756801	1,382646
R. Keys Uniform	Fitness (3)	0,34834	2,313022	2,648039	2,484857	1,598868	1,982582	1,766234
	Fitness (4)	0,034438	2,110269	2,548716	2,327067	1,043996	1,876398	1,49996

TABLE V. SIMULATION RESULTS FOR SINGLE-POINT CROSSOVER AND UNIFORM CROSSOVER BASED ON RANDOM KEYS REPRESENTATION (POPULATION SIZE = 80, GENERATIONS = 600, MUTATION RATE = 0.001, α= 8)

Random Data Set of 40 Students								
Crossover	Fitness	F_{best}	Avg_{min}	Avg_{max}	Avg_{median}	Min_{min}	Min_{max}	Min_{median}
Order	(3)	0,59161	2,02472	2,61923	2,30961	1,18193	2,00459	1,57977
	(4)	0,13544	1,92959	2,53868	2,24520	0,73276	1,96688	1,37998
R. Keys Two-point	(3)	0,57600	2,02383	2,61031	2,31733	1,21341	2,02746	1,53499
	(4)	0.17705	1.91723	2.50001	2.23312	0.69542	1.96258	1.40115
R. Keys Uniform	(3)	0,52780	2,06095	2,63358	2,31399	1,28536	2,04107	1,59964
	(4)	0.14670	1.93933	2.50318	2.24888	0.64332	1.99622	1.42142

TABLE VI. SIMULATION RESULTS FOR SINGLE-POINT CROSSOVER AND UNIFORM CROSSOVER BASED ON RANDOM KEYS REPRESENTATION (POPULATION SIZE = 120, GENERATIONS = 600, MUTATION RATE = 0.001, α= 12)

Random Data Set of 60 Students								
Crossover	Fitness	F_{best}	Avg_{min}	Avg_{max}	Avg_{median}	Min_{min}	Min_{max}	Min_{median}
R. Keys Two-point	Fitness (3)	0,63032	1,96704	2,591758	2,247427	1,13809	2,09044	1,590941
	Fitness (4)	0,19903	1,85369	2,529114	2,179855	0,623051	2,08470	1,353187
R. Keys Uniform	Fitness (3)	0,63269	1,95224	2,564983	2,270216	1,202023	2,13042	1,586091
	Fitness (4)	0,26463	1,78407	2,49969	2,172122	0,608483	2,03858	1,362632

V. Conclusion and Future Work

This paper proposes a GA approach considering multi-student characteristics for forming inter- and intra-homogeneous groups. The proposed fitness function aimed to achieve the two objectives of minimizing the difference between the average characteristics of each group and the average of the students' characteristics and maximizing the minimum of characteristics of all the groups. In fact, the second objective is of assistance in reducing the gaps in the levels of the students within the same group. Consequently, it improves the intragroup homogeneity, which the first objective does not achieve.

The reported simulation results show that the quality of the groupings improved remarkably when using the proposed fitness function compared to the quality obtained via the fitness function based only on the first objective. Since the genotypes were coded by permutation, the crossover operator based on the random keys representations was employed. Specifically, the uniform crossover provides better grouping quality than do the single-point crossover and the two-point crossover. In addition, the analysis of the obtained simulation results demonstrates the uniform crossover based on random keys representation is more efficient than is the order crossover that is generally used with permutations.

In the future, we plan to test the proposed approach on real data sets in order to further emphasize the performance of the proposed approach for group formation and, eventually propose further adjustments of the GA parameters.

References

[1] Dai-Yi Wang, Sunny S.J. Lin, Chuen-Tsai Sun, DIANA (2007). A computer-supported heterogeneous grouping system for teachers to conduct successful small learning groups. Computers in Human Behavior , 23, 1997–2010.

[2] Odo, C., Masthoff, J., & Beacham, N. (2019). Group formation for collaborative learning: A systematic literature review. In S. Isotani, E. Millán, A. Ogan, P. Hastings, B. McLaren, & R. Luckin (Eds.), Artificial Intelligence in Education: Proceedings of the 20th International Conference, AIED, Springer-Verlag, 2, pp. 206-212. https://doi.org/10.1007/978-3-030-23207-8_39.

[3] Kravitz, D. A., & Martin, B. (1986). Ringelmann rediscovered: The original article. Journal of Personality and Social Psychology, 50(5), 936–941.

[4] Maqtary, N., Mohsen, A. & Bechkoum, K. (2019). Group Formation Techniques in Computer-Supported Collaborative Learning: A Systematic Literature Review. Tech Know Learn, 24, 169–190.

[5] Julián Moreno, Demetrio A. Ovalle, Rosa M. Vicari. (2012) A genetic algorithm approach for group formation in collaborative learning considering multiple student characteristics. Computers & Education, 58(1), 560-569.

[6] P. I. Ciptayani, K. C. Dewi and I. W. B. Sentana. (2016). Student grouping using adaptive genetic algorithm. International Electronics Symposium (IES), Denpasar, 375-379.

[7] Fan, Zhi-Ping & Ma, Jian & Zeng, Shuo. (2011). A hybrid genetic algorithmic approach to the maximally diverse grouping problem. JORS, 62, 1423-1430.

[8] Anon Sukstrienwong, (2017). A Genetic-algorithm Approach for Balancing Learning Styles and Academic Attributes in Heterogeneous Grouping of Students. International journal of emerging technology in Learning, 12(3).

[9] Jihene Kaabi & Youssef Harrath. (2019). Permutation rules and genetic algorithm to solve the traveling salesman problem. Arab Journal of Basic and Applied Sciences, 26(1), 283-291.

[10] James C. Bean. (1994). Genetic Algorithms and Random Keys for Sequencing and Optimization. ORSA Journal on Computing, 6(2), 154-160. http://dx.doi.org/10.1287/ijoc.6.2.154.

[11] Reeves C. (2003). Genetic Algorithms. In: Glover F., Kochenberger G.A. (eds) Handbook of Metaheuristics. International Series in Operations Research & Management Science, 57. Springer, Boston, MA. https://doi.org/10.1007/0-306-48056-5_3.

[12] Lei, D. (2010). Solving fuzzy job shop scheduling problems using random key genetic algorithm. Int J Adv Manuf Technol, 49, 253–262.

[13] Chen, Rong-Chang, Chen, Shih-Ying, Fan, Jyun-You, Chen, Yen-Ting. (2012). Grouping Partners for Cooperative Learning Using Genetic Algorithm and Social Network Analysis. Procedia Engineering, 29, 3888-3893.

[14] Atidel Ben Hadj-Alouane, James C. Bean. (1997). A Genetic Algorithm for the Multiple-Choice Integer Program. Operations Research, 45(1), 92-101. http://dx.doi.org/10.1287/opre.45.1.92.

[15] R. Saraçoğlu and A. F. Kazankaya, "Developing an adaptation process for real-coded genetic algorithms," Computer Systems Science and Engineering, vol. 35, no.1, pp. 13–19, 2020.