User Modeling and User-Adapted Interaction (2023) 33:1113–1140
https://doi.org/10.1007/s11257-023-09360-3

A novel group recommender system for domain-independent decision support customizing a grouping genetic algorithm

Akrivi Krouska[1] · Christos Troussas[1] · Cleo Sgouropoulou[1]

Received: 15 July 2022 / Accepted in revised form: 17 March 2023 / Published online: 19 April 2023

Abstract

Group formation is a complex task requiring computational support to succeed. In the literature, there has been considerable effort in the development of algorithms for composing groups as well as their evaluation. The most widely used approach is the Genetic Algorithm, as, it can handle numerous variables, generating optimal solutions according to the problem requirements. In this study, a novel genetic algorithm was developed for forming groups using innovative genetic operators, such as a modification of 1-point and 2-point crossover, the gene and the group crossover, to improve its performance and accuracy. Moreover, the proposed algorithm can be characterized as domain-independent, as it allows any input regardless of the domain problem; i.e., whether the groups concern objects, items or people, or whether the field of application is industry, education, healthcare, etc. The grouping genetic algorithm has been evaluated using a dataset from the literature in terms of its settings, showing that the tournament selection is better to be chosen when a quick solution is required, while the introduced gene and group crossover operators are superior to the classic ones. Furthermore, the combination of up to three crossover operators is ideal solution concerning algorithm's accuracy and execution time. The effectiveness of the algorithm was tested in two grouping cases based on its acceptability. Both the students participated in forming collaborative groups and the professors participated in evaluating the groups of courses created were highly satisfied with the results. The contribution of this research is that it can help the stakeholders achieve an effective grouping using the presented genetic algorithm. In essence, they have the flexibility to execute the

✉ Christos Troussas
ctrouss@uniwa.gr

Akrivi Krouska
akrouska@uniwa.gr

Cleo Sgouropoulou
csgouro@uniwa.gr

[1] Department of Informatics and Computer Engineering, University of West Attica, 12243 Egaleo, Greece

genetic algorithm in different contexts as many times as they want until to succeed the preferred output by choosing the number of operators for either greater accuracy or reduced execution time.

Keywords Domain-independent genetic grouping · Grouping genetic algorithm · Group formation · Optimal group recommendation

1 Introduction

Group formation is a complex task that is crucial in different domains where the process of grouping is of great importance. Group formation is the process of building a team that is adequate in terms of meeting its output goals, such as reliability of outputs, functionality as well as the expectations of its members, or even its time objectives. The better the way a team is formulated, the better its performance can be achieved (Haq et al. 2021). As such, group formation is the forerunner of efficiency and effectiveness of this team.

Indeed, group formation can be seen as indispensable procedure for group development lifecycle. The idea of being applied automatically in different contexts and environments has been of ever-increasing interest of many researchers (Sklab et al. 2021). The atomic process of group formation is influenced by a number of factors. These factors can vary based on the traits of the group members, the setting of the grouping process, or the methods utilized to create the groups (Zheng et al. 2018).

It is possible to develop environments that encourage the occurrence of meaningful interactions, hence delivering strong outputs, by adequately choosing the members of a group (either animate or inanimate). According to several researchers (e.g., Cruz and Isotani 2014), poor group formation can demotivate animate members to collaborate or impede the proper formation of groups of inanimate members.

There are different cases of group formation. Group formation can be targeted at people to build adequate teams. For instance, the aim can be to create groups of students to work together in computer-supported collaborative learning environments. The optimal selection of students can improve the learning process and knowledge acquisition. Another case can be the grouping of employees to work together in their company environment. Through the adequate selection of individuals, employees can work in homogeneous teams with better professional results in less time. Moreover, a case in the field of medicine can be the grouping of patients or people who share common characteristics and can become ill by similar pathogens. If adequate groups are formulated, then people belonging to the same group and knowing that a member of their team got sick, they can protect themselves better (Mahdavi et al. 2021). In a similar way, another example of a case is the grouping of objects in an e-shop. By creating appropriate groups of objects, the customers can find the item they are looking to buy more easily, effortlessly and in less time. In all grouping cases, the main aim is to find the best way to create optimal groups regardless of group members' characteristics.

In the literature, several algorithmic methods and techniques have been presented to group formation problems. The most common computational techniques are probabilistic algorithms (e.g., genetic and swarm intelligence algorithms), data mining techniques and multi-agent approaches. Probabilistic algorithms are algorithms that rely on chance to determine the outcome or how the outcome is reached. Examples of probabilistic algorithms used for group formation are: Simulated Annealing Algorithm (Adinarayanan et al. 2018; Forghani et al. 2020; Liu et al. 2021), Memetic algorithm (Molina et al. 2010; Muruganandam et al. 2005; Yannibelli et al. 2011), Ant Colony Optimization algorithm (Chen et al. 2022; Kellel and Chniter 2019; Graf and Bekele 2006), Hill Climbing algorithm (Bonfert-Taylor and Miller 2020; Cavanaugh et al. 2004; Fronita et al. 2018), Particle swarm optimization algorithm (Liu et al. 2021; Haq et al. 2021; Ho et al. 2009) and Genetic algorithm (Krouska and Virvou 2019; Sukstrienwong 2021; Sánchez et al. 2021). Data mining is the process of identifying patterns and extracting information from big data sets using techniques that combine machine learning, statistics, and database systems. An example of data mining techniques used for group formation is the *k*-means algorithm (Troussas et al. 2019; Talavera-Mendoza et al. 2022; Ramos et al. 2021). Finally, multi-agent approaches include several intelligent interacting agents and can solve problems that are difficult for a single agent or a monolithic system to address; they have also been used in the literature for group formation purposes (Choi et al. 2010; Sklab et al. 2021; Tian et al. 2022).

However, in many review works (e.g., Cruz and Isotani 2014; Putro et al. 2018; Ge et al. 2018), the genetic algorithm seems to be the most appropriate technique for group formation, which brings considerable and positive outcomes. There are various studies that have proposed genetic algorithm techniques to create effective groups that maximize their output. They mainly focus on creating groups of humans, while the main target is to compose heterogeneous or mixed groups (Krouska and Virvou 2019; Sukstrienwong 2021; Sánchez et al. 2021; Putro et al 2020; Imbrie et al. 2020). When referring to group formation of humans, their knowledge level is the primary factor utilized to establish groups while communicative abilities and group-mate preferences are also used for this purpose (Moreno et al 2012; Contreras and Salcedo 2017; Zheng et al. 2018; Chniter et al. 2018; Flanagan et al. 2021; Wang et al 2011). On the other hand, when referring to objects, their common characteristics (e.g., category) seem to be the dominant determinants (Xiaohui 2022; Matviichuk et al. 2021; Hamrouni et al. 2020; Kumar and Sinha 2020; Chen et al. 2018; Tariq et al. 2009). Concerning the genetic settings, the majority of the studies use well-known crossover operators, such as order, one-point, or partially mapped, while other studies provide novel crossover strategies (Krouska et al. 2019; Ge et al. 2018; Putro et al. 2018; Cruz and Isotani 2014). Additionally, swap and displacement are the mutation operators that are employed the most (Krouska et al. 2019; Ge et al. 2018; Putro et al. 2018; Cruz and Isotani 2014).

Recommender systems can be used to support group formation. In the literature, recommender systems have been explored for different domains. Some examples include their use in e-learning settings (Marques et al. 2021; Amara and Subramanian 2020; Fernández-García et al. 2020; Chrysafiadi et al. 2018), entertainment websites (Kannikaklang et al. 2022; Gupta et al. 2020; Singla et al. 2020; Troussas et al. 2018), social environments (Yang et al. 2020; Wongkhamchan et al. 2019; Mughaid et al. 2019; Liang et al. 2019), digital repositories (Troussas et al. 2021; Borovič et al. 2020; Guan et al. 2019; Jomsri et al., 2018), tourism systems (Baker and Yuan 2021; Srisawatsakul and Boontarig 2020; Chen et al. 2020; Kbaier et al. 2017). The algorithmic techniques that have been primarily used in the research papers, described above, are collaborative filtering, model-based approaches, stereotypes, content-based filtering, machine learning and hybrid approaches. According to a study of 2019 (Al-Ghuribi and Mohd Noah 2019), for the majority of recommender systems, the primary source for the recommendation process is a single-criterion rating (overall rating). Another study of 2020 (Choi et al. 2020) shows that techniques that have been used extensively in the literature in the development of recommender systems, such as collaborative filtering, have a number of disadvantages, e.g., the cold start problem which prevents the system from drawing conclusions about persons/objects for whom/which insufficient information has been obtained. To this end, the need for more research to the direction of reinforcing recommender systems is apparent.

In this study, a novel genetic algorithm was developed for forming groups using innovative genetic operators to improve its performance and accuracy. In particular, our approach was designed in such a way that allows any input regardless of the domain problem; i.e., whether the groups concern objects, items or people, or whether the field of its application is industry, education, healthcare, etc. Therefore, the algorithm can be characterized as domain-independent. This feature is based on the fact that the meaning of the characteristics used by the genetic algorithms for producing the optimal solution is immaterial, since the algorithm just handles normalized numerical values. In more detail, the presented genetic algorithm is incorporated in a system that offers a big number of genetic operators, while the user has the possibility to select the settings of the algorithm. As such, the novelty of this paper is that our system can help the stakeholders achieve an effective grouping. They have the flexibility to execute the genetic algorithm as many times as they want until to succeed the output that they want. Also, the system offers a range of operators, introducing some novel ones apart from the classic ones, with the intention of optimizing the algorithm results. If they select a large number of operators, then the execution time will be high, while the accuracy may improve. If they select a small number of operators, then the execution time will be shorter, while the accuracy may be reduced slightly.

2 Domain-independent GGA: problem representation and algorithm description

In this study, a novel grouping genetic algorithm was developed incorporating innovative genetic operators to optimize its results. Moreover, the proposed algorithm can be considered as domain-independent, since its design allows any input regardless of the domain problem; i.e., whether the groups refer to objects, items or people, or the field of its application is industry, education, healthcare, etc. This feature arises from the fact that the genetic algorithms handle normalized numerical values without playing any role the meaning of the input data.

The genetic algorithm is a meta-heuristic based on Darwin's theory of evolution, developed by John Holland in 1975 (Holland 1975); while in 1992, Emmanuel Falkenauer introduced the Grouping Genetic algorithm, overcoming the difficulties of traditional genetic algorithm in clustering issues (Falkenauer 1992). The genetic algorithm is based on the encoding of the potential solutions of the problem using a chromosome-like data structure, which consists of genes, and the application of several recombination operators on them for generating the optimal solution based on a cost function. It should be mentioned that a chromosome refers to one possible solution to the given problem, a gene is one element of a chromosome, while an allele is the value a gene has in a particular chromosome.

The group formation problem consists of a set of n objects $O = \{o_1, o_2, o_3, \ldots, o_n\}$ that should be grouped in a set of k groups $G = \{g_1, g_2, g_3, \ldots, g_k\}$. Each object has a number of z attributes $A_{\mathrm{i}} = \{a_{\mathrm{i},1}, a_{\mathrm{i},2}, a_{\mathrm{i},3}, \ldots, a\}$, indicating the values of the attributes of object i, based on which the groups are formed. Each object can be in exactly one group and each group should have a number of m members (group size). The group type can be: a. heterogeneous, where the attributes of the members should have different values, b. homogeneous, where the members should have the same attribute values, and c. mixed, where same attribute values should be different and others the same. The objective of the grouping is to generate the optimal groups based on the group type and objects' attributes by optimizing a cost function, $f(\mathrm{x})$. The problem involves minimizing this function, since the lower the value of cost function, the greater the fitness of the group. The optimal group has $f(\mathrm{x}) = 0$.

The steps of the proposed algorithm for creating the optimal groups are as follows:

1. *Import data* Firstly, the input dataset with the population attributes has to be uploaded to the system. The dataset should be a Comma Separated Values (CSV) file, namely a plain text file with the list of data separated by commas. The first row of the data should include the attributes' names, while each following row should include the corresponding attributes' values of each object belonged to the population. Moreover, the first attribute should refer to object key value, corresponded to a unique identifier of each object (i.e., row of dataset).
2. *Configure algorithm settings* For executing the grouping genetic algorithm, it is required the genetic parameters and operators to be set. Therefore, the user selects the preferred selection and crossover operators, as well as the population and group size, the number of generations, the probability of crossover and mutation, and the elitism percentage.
3. *Pre-process data* The algorithm handles attributes with numerical values in a pre-defined range. As such, firstly, a numerical discretization process is applied to all categorical attributes. Afterward, all attributes are normalized in order all data to fit in the same range for avoiding perturbations in the evaluation of fitness function. The formula used for falling values between 0 and 1, is the following:

$$a_{\mathrm{norm}} = (a - a_{\mathrm{min}})/(a_{\mathrm{max}} - a_{\mathrm{min}}) \tag{1}$$

   where a_{norm} is the normalized value of a value pertained to the attribute a_{i}, and a_{min} and a_{max} is the minimum and maximum values of the corresponding attribute.
4. *Generate initial population* In order the genetic operators to be applied, the grouping problem solution should be represented as chromosome, using a pre-defined data structure. In this case, each solution (i.e., chromosome) is encoded as an array, having size equal to the number of objects m. Each item of the array (namely gene) includes the object id, while its index indicates the group to which the object is

assigned. Thus, the formula given object's group is the following:

$$\text{object\_group} = (\text{object\_index div group\_size}) + 1 \tag{2}$$

Based on encoding scheme, the initial population is generated randomly. It should be noted that the population includes a certain number of chromosomes, namely feasible encoded solutions, equal to the defined population size. The random strategy is chosen, since the population diversity is promoted and the convergence to the best solution is enhanced.

5. *Calculate chromosomes' fitness* Once the population has produced, the fitness of each chromosome is computed, through calculating the fitness value of each chromosome's group. The fitness function is associated with the appropriateness of groups formed, aiming to minimize their total variance. In this work, the fitness function is defined based on Euclidean distance metric, which is one of the most basic distance metrics and commonly used in genetic algorithms (Krouska et al. 2019). The system provides the capability for creating mixed groups, meaning that homogeneity is applied for some attributes, while heterogeneity is applied for other ones. Thus, let q and r be the numbers of attributes in which homogeneity and heterogeneity is required, respectively; i.e., $q + r = z$ (z: total attributes). Homogeneous attributes mean that the objects grouped together should have similar values for these characteristics. Thus, the similarity between two objects, namely $\mathrm{o_i}$ and $\mathrm{o_j}$, for q attributes is given by the following formula:

$$\mathrm{OS}(o_i, o_j) = \sqrt{\sum_{c=1}^{q}(a_{i,c} - a_{j,c})^2} \tag{3}$$

where $a_{i,c}$ and $a_{j,c}$ indicate the attribute c of the objects o_i and o_j, respectively.

Figure 1 shows the encoding scheme, giving an example of 12 objects divided into three groups.

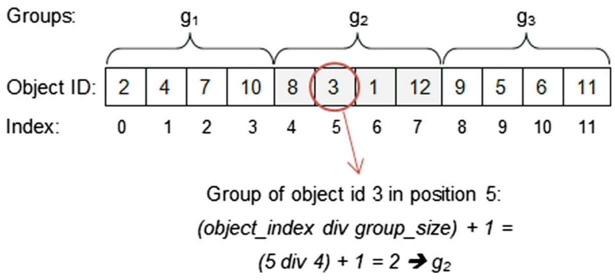

Groups:
g1
g2
g3
Object ID: 2 4 7 10 8 3 1 12 9 5 6 11
Index: 0 1 2 3 4 5 6 7 8 9 10 11
Group of object id 3 in position 5:
(object_index div group_size) + 1 =
(5 div 4) + 1 = 2 ➔ g2

Fig. 1 Encoding scheme

It is obvious that minimizing the distance, the optimal similarity is achieved. The homogeneity among the objects in a group is computed as:

$$\text{GHomo}\ (g_t) = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \text{OS}\big(o_i, o_j\big) \tag{4}$$

On the other hand, heterogeneous attributes mean that the objects grouped together should have different values for these characteristics. Thus, the difference between two objects for r attributes is given by the following formula:

$$\text{OD}\big(o_i, o_j\big) = 1 - \sqrt{\sum_{c=1}^{r}\big(a_{i,c} - a_{j,c}\big)^2} \tag{5}$$

Apparently, the greater the distance is, the more different is the values. So, according to the aforementioned formula, the optimal difference is obtained when OD function is minimized. The heterogeneity among the objects in a group is formulated as:

$$\text{GHete}\ (g_t) = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \text{OD}\big(o_i, o_j\big) \tag{6}$$

Based on the above, the group's fitness is computed as follows:

$$G\text{Fitness}(g_t) = G\text{Homo}(g_t) + G\text{Hete}(g_t) \tag{7}$$

It should also be noted that users can create homogeneous or heterogeneous groups, instead of mixed ones, if no such attributes are indicated; i.e., the q or r values are zero, so the GHomo and GHete are not calculated, correspondingly.

In order to achieve the optimal mixed group, the value of GFitness should be minimized. Consequently, the optimization objective of this grouping problem is to minimize chromosome fitness function, which is given by the following formula:

$$C\text{Fitness}(c) = \sum_{i=1}^{k} G\text{Fitness}(g_i) \tag{8}$$

As such, the lower value the CFitness has, the better the solution, which the chromosome c represents, is.

6. *Create new population* This is the most crucial step of the algorithm, since a new population is created including new fitter chromosomes, with the intention to converge toward the optimal solution. To achieve this, a series of genetic operations is performed probabilistically, namely elitism, selection, crossover and mutation. Regarding the elitism strategy used, the best chromosomes of the population in a number equal to the elitism percentage are transmitted to the next generation, ensuring that highly fit chromosomes remain during reproduction.

Selection plays an important role in the evolution of the population. The selection operators are based on the principle that the fittest survives and the suggestion that the fittest generates better offspring. Therefore, they select using several techniques the most suitable chromosomes, so they or a portion of them applying genetic operators are preserved in the new population.

Crossover is vital to exploit the search space, aiming to improve the new population. This operator combines two chromosomes, called parents chosen by selection process, exchanging their genetic material, in order to produce fitter offspring for the new generation. The two chromosomes with the best fitness between the parents and offspring are inserted into the next population. Crossover is based on the principle that suitable genes should preserve and performed with a crossover probability.

Finally, mutation is applied to a chromosome changing a few genes based on a mutation probability, in order to insert new genetic material into the next generation, avoiding converge to locally optimal solutions. There are several mutation operators in literature; however, in this work, the well-known swap mutation is used, where two randomly-selected genes exchange their positions. The reason why this operator is adopted is due to its simplicity and effectiveness, as well as the fact that it does not cause inconsistency in groups (e.g., groups with different number of members or duplicate genes).

The system developed provides a variety of selection and crossover operators described in the following section, as well as the possibility to choose the probability of genetic operations, in order users to setup the grouping process properly, according to their problem context and desired performance.

7. *Check termination condition* The algorithm continues reproducing generations until the search space is sufficiently explored. There are several termination conditions used; however, in this case, the most common stopping criterion used. As such, the algorithm is developed to terminate when a maximum number of iterations reached. The number of generations is determined by the user at configuration setup.
8. *Export results* After the execution of the developed grouping genetic algorithm based on user's configurations, the system exports the optimal solution of group formation found.

Figure 2 illustrates the flowchart of the grouping genetic algorithm.

3 Combining genetic operators to optimize grouping decision-making

In the presented group recommender system, the user can upload any dataset in the format of containing at least two columns as attributes (i.e., object id and one characteristic) and rows as objects' data, more than the group size. Afterward, the system shows the genetic grouping algorithm settings. The user can choose how to configure them, depending on problem context and target group. For example, the user can select the groups to have only heterogeneous features, creating heterogeneous groups,

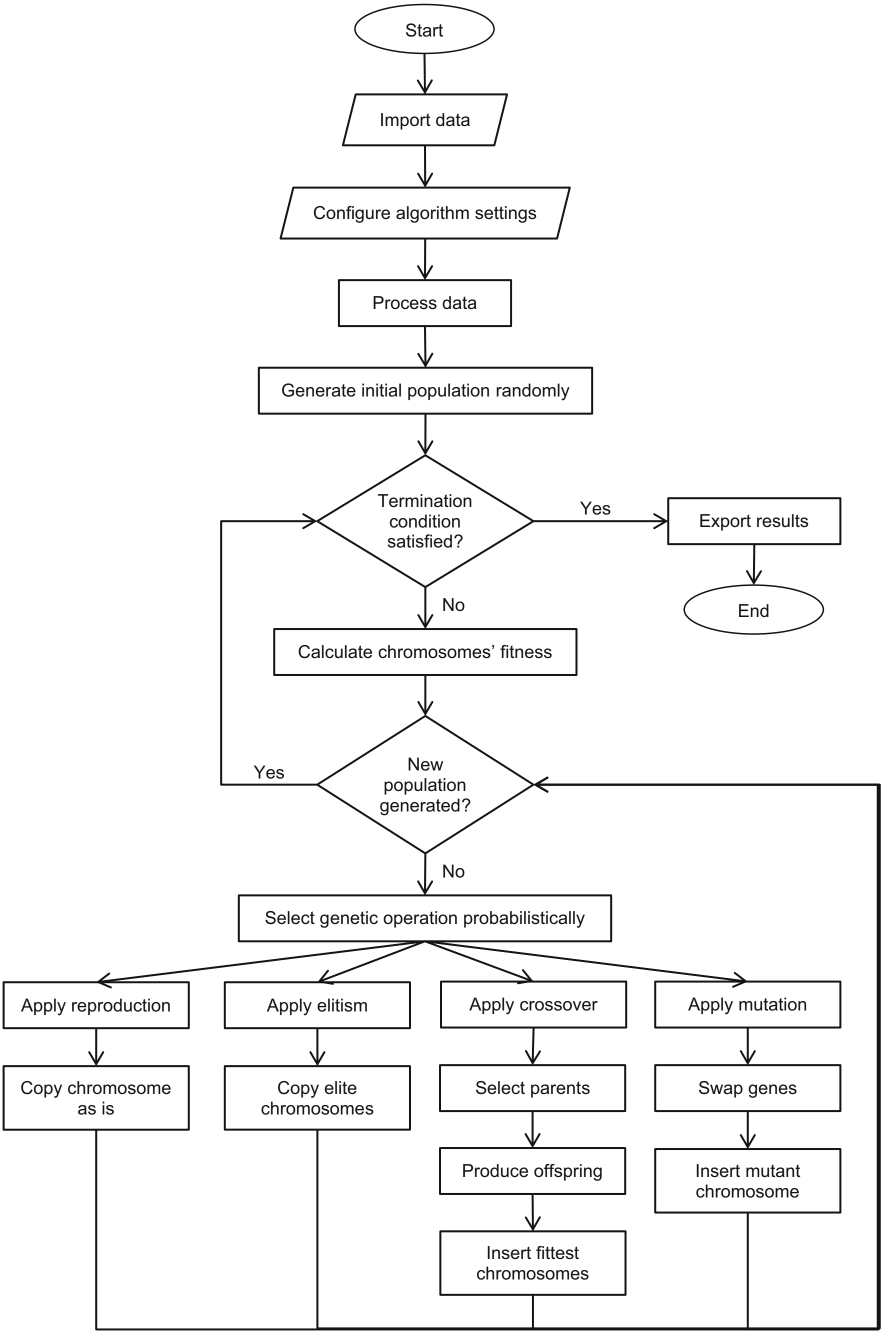
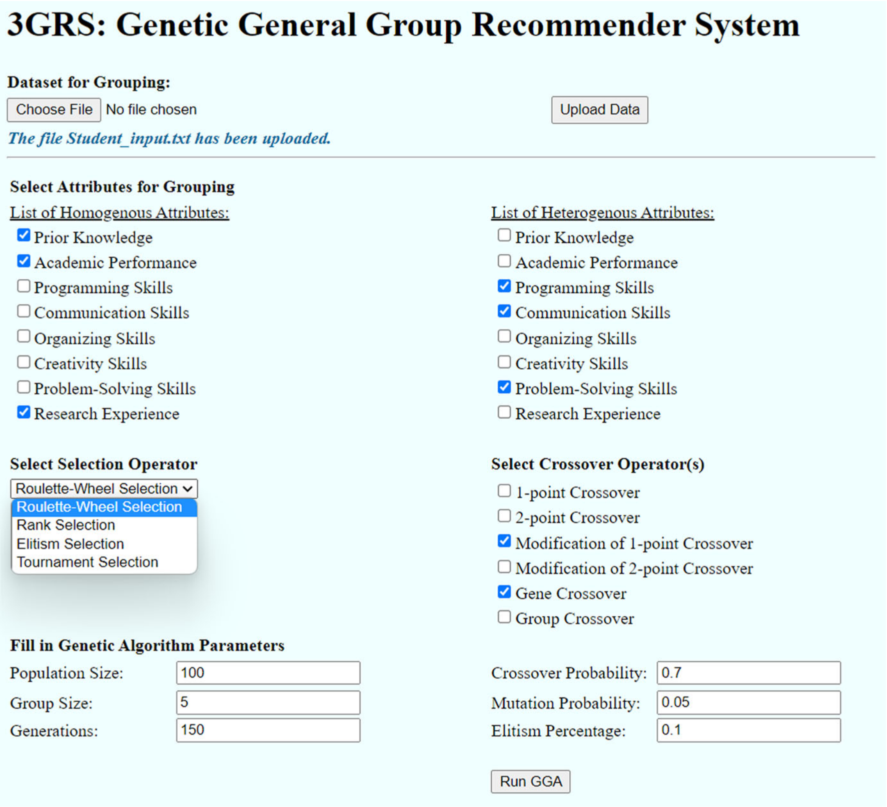


Fig. 2 The flowchart of grouping genetic algorithm



3GRS: Genetic General Group Recommender System

Dataset for Grouping:

Choose File No file chosen Upload Data

The file Student_input.txt has been uploaded.

Select Attributes for Grouping

List of Homogenous Attributes:

☑ Prior Knowledge
☑ Academic Performance
□ Programming Skills
□ Communication Skills
□ Organizing Skills
□ Creativity Skills
□ Problem-Solving Skills
☑ Research Experience

List of Heterogenous Attributes:

□ Prior Knowledge
□ Academic Performance
☑ Programming Skills
☑ Communication Skills
□ Organizing Skills
□ Creativity Skills
☑ Problem-Solving Skills
□ Research Experience

Select Selection Operator

Roulette-Wheel Selection
Roulette-Wheel Selection
Rank Selection
Elitism Selection
Tournament Selection

Select Crossover Operator(s)

□ 1-point Crossover
□ 2-point Crossover
☑ Modification of 1-point Crossover
□ Modification of 2-point Crossover
☑ Gene Crossover
□ Group Crossover

Fill in Genetic Algorithm Parameters

Population Size:	100
Group Size:	5
Generations:	150
Crossover Probability:	0.7
Mutation Probability:	0.05
Elitism Percentage:	0.1

Run GGA

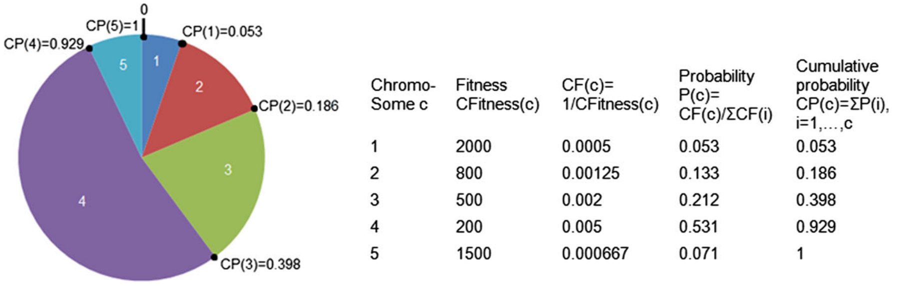
Fig. 3 Screenshot of group recommender system settings

or only homogeneous features, creating homogeneous groups, or can combine them, creating mixed groups. It is worth noting that it is not necessary for the user to use all the attributes of the input dataset, but to select at least one in order the cost function to be able to be calculated. Consequently, the genetic operators should be determined. In particular, the user should choose one selection operator and at least one crossover operator. Regarding the selection operator, the system supports the roulette-wheel selection, the rank one, the elitism and the tournament one. Concerning the crossover operators, one novelty of the system is the development of a variety of classic and modified crossover operators, as well new ones, for optimizing the algorithm results. As such, the system supports the 1-point crossover, the 2-point crossover, a modification of 1-point crossover, a modification of 2-point crossover, the gene crossover and the group crossover. Finally, the genetic algorithm parameters should be defined, i.e., the population size, the group size, the generations, the crossover/ mutation/ elitism percentage. Figure 3 illustrates a screenshot of the system.

3.1 Selection operators

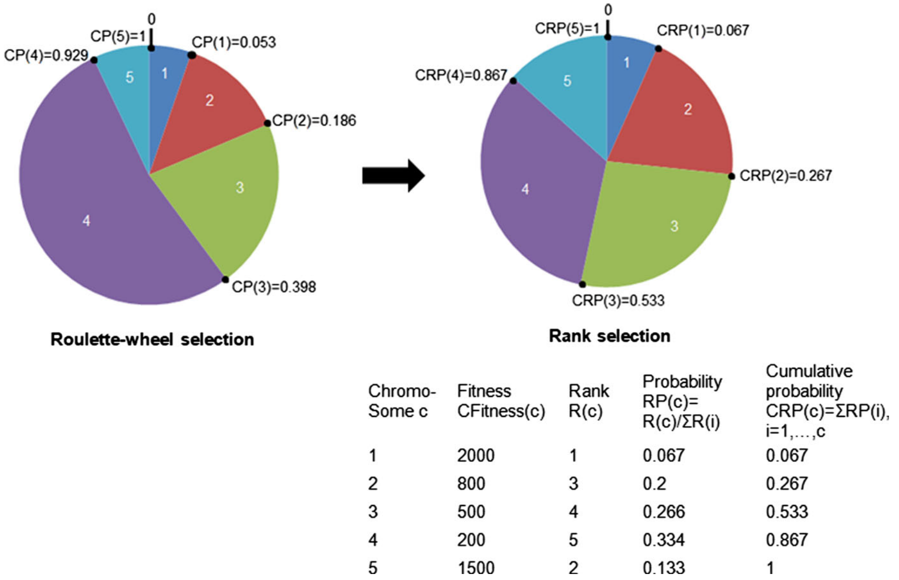
The selection operators incorporated into the developed system in order to choose the fittest chromosomes as parents to be evolved are the following:

- *Roulette-wheel selection*: Roulette-wheel selection is a stochastic technique, where the probability for choosing a chromosome is proportional to its fitness. In particular, for each chromosome, a selection probability is assigned, defined as the ration of chromosome's fitness to the total fitness of the population. Moreover, a cumulative probability is calculated as the sum of chromosome's selection probability and this of all the previous chromosomes in the population. It should be noted that the cumulative probability of the population is 1. As such, each chromosome gets the corresponding portion of the roulette. Afterward, a random number between 0 and 1 is generated and depending on the roulette's portion it belongs to, the corresponding chromosome is chosen for evolution. It is obvious that the better fitness a chromosome has, the bigger portion it possesses, and therefore it is more likely to be selected. Figure 4 represents a case of roulette-wheel selection.
- *Rank selection*: Rank selection is an explorative method for selecting chromosomes. It performs like the roulette-wheel selection, with the difference that the selection probability, during its computation, considers also the rank of the chromosome in the population. As such, the selection probability is adjusted properly, assigning fairer portions of the pie at chromosomes than roulette-wheel selection, giving the chance to all the chromosomes to be selected. Therefore, this method overcomes the limitation of roulette-wheel selection when chromosomes' fitness differs a lot; e.g., if a chromosome has with very high fitness, it possess a large portion of the roulette, resulting on selecting it more times and consequently reducing the performance of selection. Figure 5 represents the roulette of Fig. 4 before and after adjusting the selection probability based on chromosomes' ranking.
- *Elitism selection*: Elitism is the selection of the best chromosomes to be included in the new population. As such, the elitism selection creates a pool with the elite chromosomes, namely the chromosomes with the best fitness value. As the population includes different chromosomes in each iteration, this pool is redefined. Thus, the first parent is always picked from this list, while the other parent is selected randomly from the population. Since the one parent is one of the best chromosomes, the probability to generate optimal offspring is increased. Figure 6 shows the operation of elitism selection.
- *Tournament selection*: In this operator, "tournaments" run in order to select the parents. In particular, a tournament size is set, determining the number of chromosomes that are chosen randomly from the population to compete against each other. The winner is the chromosome with the best fitness, constituting the parent. All chromosomes return into the population and are eligible to participate again in other tournaments. Figure 7 represents how tournament selection works.



Chromo-Some c	Fitness CFitness(c)	CF(c)= 1/CFitness(c)	Probability P(c)= CF(c)/ΣCF(i)	Cumulative probability CP(c)=ΣP(i), i=1,…,c
1	2000	0.0005	0.053	0.053
2	800	0.00125	0.133	0.186
3	500	0.002	0.212	0.398
4	200	0.005	0.531	0.929
5	1500	0.000667	0.071	1

Fig. 4 Roulette-wheel selection



Chromo-Some c	Fitness CFitness(c)	Rank R(c)	Probability RP(c)= R(c)/ΣR(i)	Cumulative probability CRP(c)=ΣRP(i), i=1,…,c
1	2000	1	0.067	0.067
2	800	3	0.2	0.267
3	500	4	0.266	0.533
4	200	5	0.334	0.867
5	1500	2	0.133	1

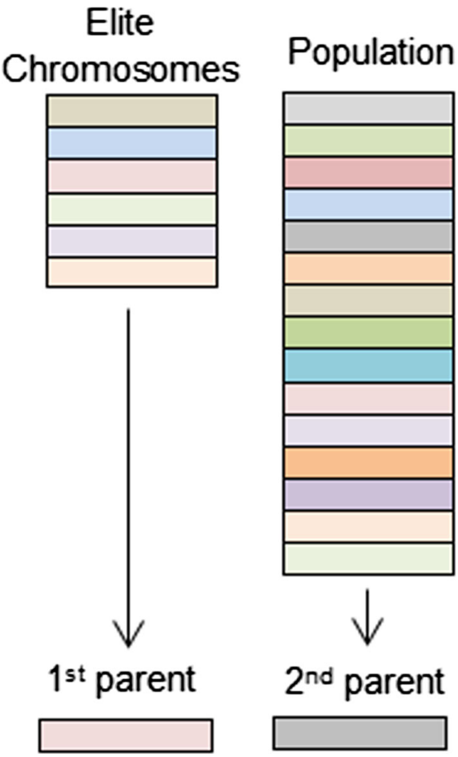
Fig. 5 Rank selection

3.2 Crossover operators

The crossover operators provided by the system to combine parents' genetic information to produce the offspring are the following. It should be noted that apart from the well-known crossover, namely 1-point and 2-point, modified and new approaches have been incorporated with the intention to improve the quality of the offspring.

- *1-point crossover*: A crossover point on both parents is selected randomly, swapping the genetic material to the right of this point. As such, each offspring includes one parent's left part defining by this point and the right one of the other parent. In case the chromosomes consist of unique genes that should not be repeated, the genes of the right part that are not appeared in the left part are swapped, and then the

Fig. 6 Elitism selection



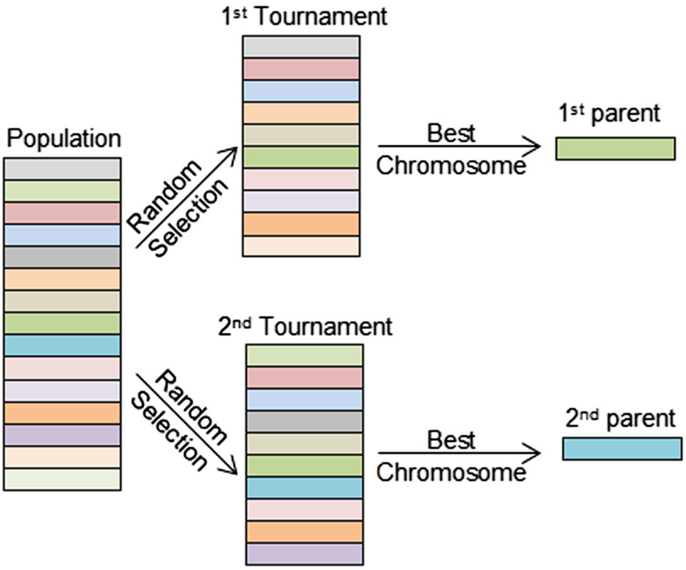


Fig. 7 Tournament selection

empty genes are filled by the remaining values. Figure 8 illustrates how the 1-point crossover operator works.

- *2-point crossover*: This operator performs like 1-point crossover, with the difference that two crossover points are selected randomly on both parents and the genes between these points are swapped. Figure 9 shows 2-point crossover.
- *Modified 1-point crossover*: In order to improve the effectiveness of 1-point crossover, a modification is introduced where the crossover point is not a random position in the whole chromosome, but it is picked from the worst group of parents considering the groups' fitness. This modification on point selection strategy may lead to better groups, since at least the worst group will recombined and groups with better fitness will preserve into next generation. In Fig. 10, an example of the proposed modified 1-point crossover is presented.
- *Modified 2-point crossover*: Similarly to the modified 1-point crossover, this operator selects randomly the first crossover point from the worst group of one parent and the other point from the worst group of the other parent. This approach may prevent to lose fit groups, recombining at the same time the worst ones. Figure 11 depicts the functionality of the proposed modified 2-point crossover.
- *Gene crossover*: In this method, a random number is corresponded to each gene, like rolling a die. If this number is even, then the parents swap the genes belonged to this position; otherwise, the genes remain the same in the offspring. Similarly to the other crossover operators, the genes cannot be repeated after the completion of the process. Figure 12 illustrates the introduced gene crossover.
- *Group crossover*: The scope of this new operator is to expand the problem space and avoid getting stuck at local minimum, introducing new genetic material into next generation. As such, a multi-point crossover was developed, called group crossover, where the genes of the worst groups between the parents are swapped. In particular, firstly, the number of groups participated in the crossover is defined, and afterward the corresponding worst groups of each parent are selected for crossover. In this work, the 50% of chromosome's groups are chosen for crossover. Hence, for these groups, the offspring include the half genes of each parent, aiming this recombination to improve the total fitness. Figure 13 shows an example of the new group crossover.

4 Evaluation results and discussion

To evaluate the effectiveness and efficiency of the proposed grouping genetic algorithm, a series of experiments assessing the algorithm performance considering the genetic settings was conducted, aiming to investigate the optimal configurations. Furthermore, the algorithm was applied on two grouping problems measuring its performance through the acceptability of group formation.

4.1 Evaluation of genetic operators

The aim of this evaluation process is to explore the effect of genetic operators on algorithm performance. As such, firstly the values of the genetic parameters were set

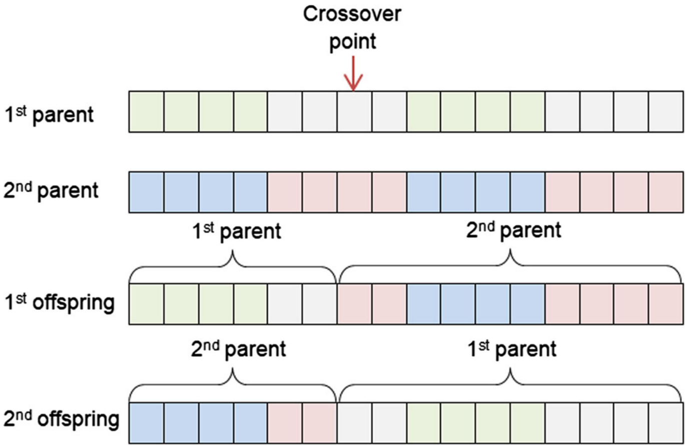


Fig. 8 1-point crossover

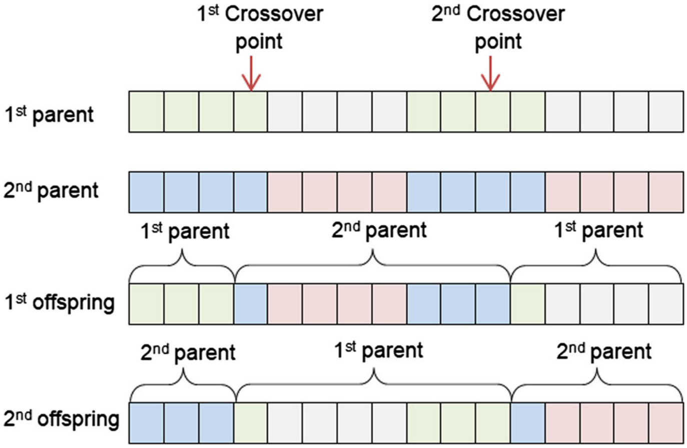


Fig. 9 2-point crossover

based on the literature, and afterward a sensitivity analysis approach was used in order to evaluate algorithm output in response to changing operator settings. In particular, the algorithm was run with different selection and crossover operators individually, while the rest parameters remained stable. As such, the operators' relative impact on algorithm performance is estimated. The algorithm performance was measured based on the execution time and fitness value, since the goal of the grouping formation is to generate the optimal groups, meaning to minimize the cost function, in a short time. Moreover, in order the experiments' results to be comparable, the same computer machine was used (Intel Core i7 12,700/16 GB DDR5/512 GB SSD), ensuring equal computational resources, as well as, the same dataset was utilized (Krouska et al. 2020). In particular, the dataset used consisted of 100 objects and 5 attributes. The purpose

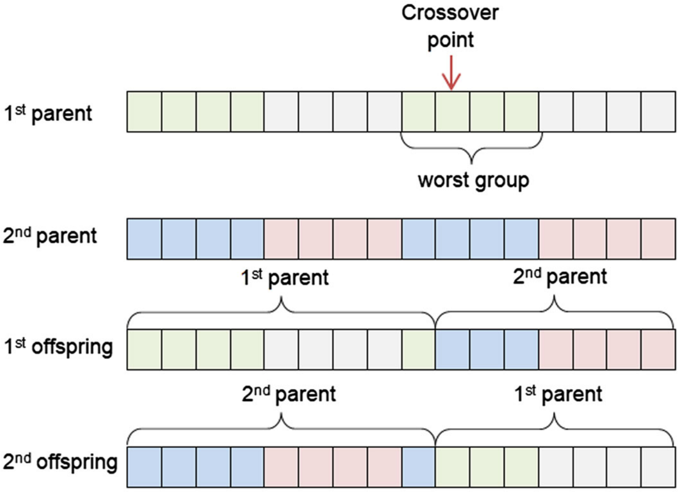


Fig. 10 Modified 1-point crossover

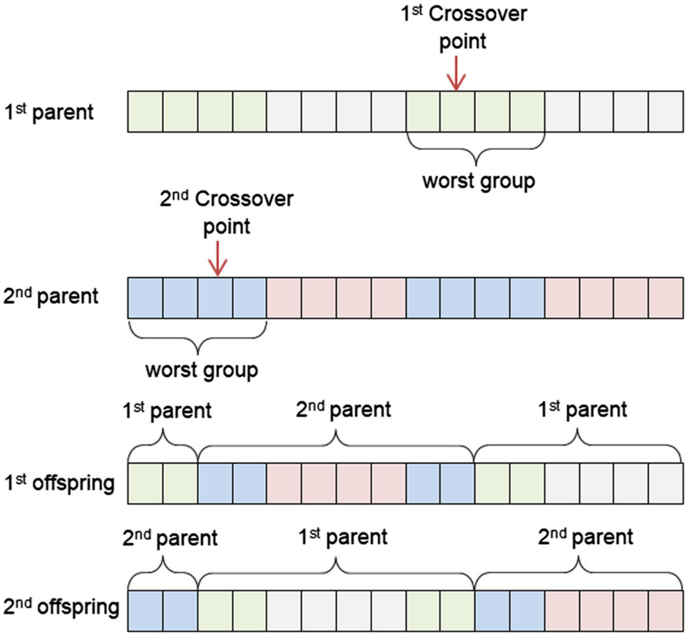


Fig. 11 Modified 2-point crossover

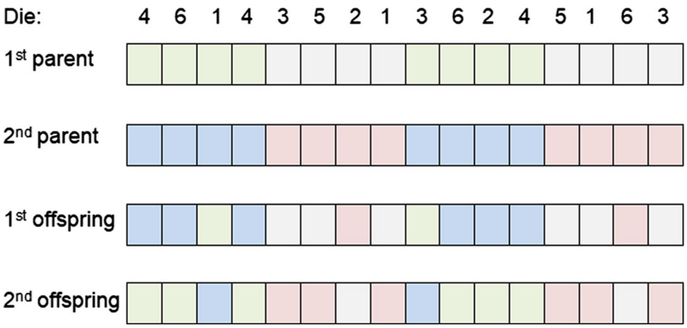


Fig. 12 Gene crossover

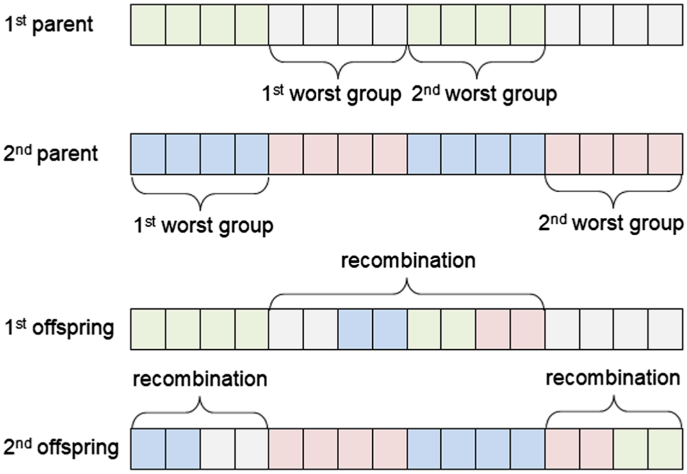


Fig. 13 Group crossover

of the problem was to form mixed groups of 5 members, having three heterogeneous attributes and 2 homogeneous ones. The further description of the dataset is out of this evaluation scope, where the algorithm's settings are tested.

In the literature, there are many studies evaluating the genetic parameters regarding the number of generations, the size of population, the crossover and mutation rate (Krajčovič et al. 2019; Krouska and Virvou 2019; Mills et al. 2015). Based on these researches, the values chosen for genetic parameters to conduct our experiments are 200 generations, 150 population size, 80% crossover rate and 5% mutation rate. These values are considered as valid settings that cannot affect negatively the algorithm performance, but can lead algorithm to optimal results. Moreover, as the grouping genetic algorithm is a meta-heuristic approach exporting different solutions each time it runs, each experiment was executed 15 times and the mean of the performance measures was used in the evaluation process. This number of executions is chosen based on literature, being an acceptable one giving reliable results (Krouska and Virvou 2019; Zheng et al. 2018).

Sensitivity analysis led to a large number of experiments emerged from the combination of selection and crossover operators provided by the system. However, in Table 1, the most meaningful results were shown. Regarding the selection operators, roulette-wheel and rank selection is more time consuming operators than elitism and tournament one. Moreover, especially the rank selection outperforms the other operators, producing solutions with better fitness. This may be due to the larger diversity in problem space that rank selection creates, as it readjusts the chromosome selection probabilities, allowing more different chromosomes to be selected. However, since tournament selection also leads to adequate group formation considering the fitness

Table 1 Evaluation results of genetic operators

Crossover operator	Selection operator							
	Roulette-wheel		Rank		Elitism		Tournament	
	(1)	(2)	(1)	(2)	(1)	(2)	(1)	(2)
Gene	167	1519	171	1419	114	1547	109	1523
Group	196	1479	202	1327	132	1532	125	1504
1-point + Gene	278	1432	299	1387	214	1478	232	1462
2-point + Gene	305	1401	315	1314	256	1437	269	1429
Mod. 1-point + Gene	394	1378	398	1365	325	1401	356	1411
Mod. 2-point + Gene	418	1346	401	1304	367	1392	369	1376
1-point + Group	448	1385	456	1345	425	1387	439	1389
2-point + Group	491	1335	489	1324	434	1361	442	1358
Mod. 1-point + Group	457	1331	465	1317	440	1378	438	1377
Mod. 2-point + Group	463	1312	471	1301	459	1352	451	1348
Gene + 1-point + Group	536	1308	531	1251	501	1302	514	1298
Gene + Mod. 2-point + Group	543	1289	529	1224	516	1274	526	1265
1-point + Mod. 2-point + Gene	478	1305	506	1297	469	1332	476	1325
1-point + Mod. 2-point + Group	481	1314	499	1303	478	1314	494	1305
1-point + Mod. 2-point + Gene + Group	705	1331	726	1314	657	1349	673	1337

(1) Execution time (sec), (2) Fitness

output in shorter time, this operator may be a better choice when a quick solution is required and/or more crossover operators are combined.

Comparing the crossover operators, the results reveal that the introduced gene and group crossover can be characterized as effective operators, since they generate fit groups in good execution time. Better fitness results are observed when combined three crossover operators, and especially in current experiments the use of gene, modified 2-point and group crossover outperforms the other combinations. Whereas, blending more than three operators does not significantly improve the quality of groups, but greatly increases the execution time, making this option prohibitive. Moreover, it seems that when combining more crossover operators, the selection operator plays a minor role in results. It should be noted that using more crossover operators can lead to a better solution; however, it greatly increases the completion time. Therefore, users should consider the completion time of the algorithm and the quality of the results they want to achieve according to the problem to be solved, in order to choose the most suitable combination of operators that will produce the desired results. It is worth mentioning that the algorithm performs very well giving reliable results even though a small number of crossover operators combined.

The aforementioned results are in accordance with those emerged from the literature, where more complex and modified operators boost algorithm performance (Krouska and Virvou 2019; Zheng et al. 2018; Ramos-Figueroa et al. 2021; Krouska et al. 2020). Optimizing the process of genetic algorithm requires an efficient examination of the search space to identify the most promising chromosomes, as well as to introduce more promising genes in the population. The proposed algorithm incorporates, apart from the classic genetic operators, novel ones that ensure the population diversity and the introduction of new and optimal chromosomes in each interaction. Moreover, they have developed combining the genetic material in such way, so that the convergence in local optima to be avoided. As such, the exploration and exploitation strategies used by the proposed algorithm increase the possibilities of coming to a high-quality solution, as well as of outperforming other grouping genetic algorithm approaches which run with simple genetic operators and without combining them.

4.2 Applications on grouping problems

In order to evaluate the user acceptance of group formation, the proposed algorithm was applied to two case studies. After creating the groups, the users were asked to assess the groups through a 5-point Likert scale online questionnaire (1: Strongly disagree, 2: Disagree, 3: Neutral, 4: Agree, 5: Strongly agree). As such, the responses were collected electronically and all the participants, in both experiments, completed the questionnaire successfully, resulting in a 100% response rate.

In the first experiment, the system was used to create collaborative teams for the completion of a project assigned to them as part of the teaching process of the course "Python programming" in a Greek University. The participants were 60 s year undergraduate students, namely 35 males and 25 females, separating into 15 groups of 4 members. All students had approximately equal age, ranging from 19 to 20 years old, and previous knowledge, as being in the same year of study and having passed

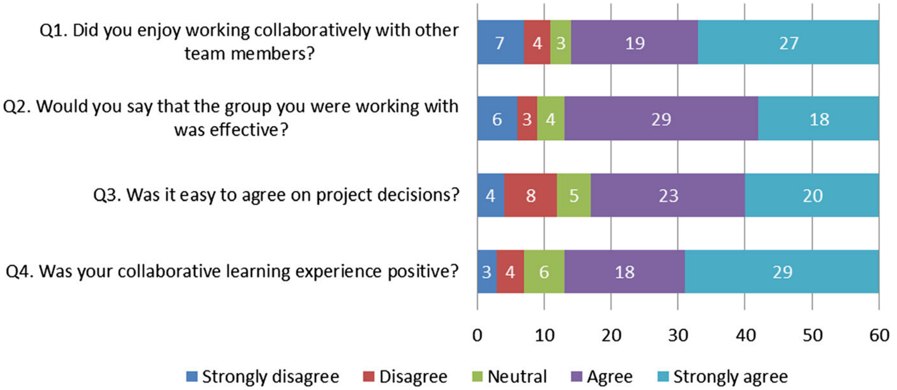


Fig. 14 Survey results on collaborative group formation using the proposed algorithm

successfully the courses of the first year of their studies. Furthermore, they had the same motivation in completing the project in order to pass the course, by achieving the highest possible grade.

The structure of the groups was selected to be mixed, having three homogeneous and three heterogeneous attributes as shown in Fig. 3, in order to enhance group dynamics for effective collaboration and better learning outcomes. The two professors of the course, having about 10 years teaching experience, determined the structure of the groups based on their requirements on teams' characteristics. The teams formed were working together for a semester on the project assigned to them, and all the teams delivered the project successfully. After the completion of the project, the students filled in the questionnaire regarding their experience on the collaborative team created by the proposed grouping genetic algorithm. Figure 14 illustrates the survey results, and Table 2 presents the descriptive statistics.

The results showed a high acceptance of the method used in forming the groups, supporting the effectiveness of the developed system. In particular, 81.67% of students stated that they enjoyed working collaboratively with other team members, indicating a pleasant collaborative environment, essential to achieve their learning goals. The mean of rating the enjoyment working collaboratively (Q1) is 3.92/5, the median is 4 and the mode is 5. The responses of Q1 had a standard deviation of 1.357. Moreover, the formation of adequate groups using the proposed grouping genetic algorithm is suggested by the fact that over 80% of participants found both the collaboration and the decision-making among team members effective. The mean of rating the effectiveness of collaboration (Q2) and the ease of decision-making (Q3) is 3.83/5 and 3.78/5 respectively, the median and mode is 4 in both cases, while the standard deviation is 1.209 and 1.236 correspondingly. Finally, the overall collaborative learning experience was rated with the high degree of 4.1/5, having the value of 4 as median, 5 as mode and 1.145 as standard deviation, indicating the positive attitude of students toward the group formation.

The second experiment concerns the grouping of courses in order to provide integrated learning outcomes and develop skills and competences. In particular, the 55

Table 2 Descriptive statistics on results of collaborative group formation using the proposed algorithm

	Q1	Q2	Q3	Q4
Mean	3.92	3.83	3.78	4.1
Standard Error	0.175	0.156	0.159	0.148
Median	4	4	4	4
Mode	5	4	4	5
Standard Deviation	1.357	1.209	1.236	1.145
Sample Variance	1.840	1.463	1.529	1.312
Kurtosis	0.164	0.836	− 0.214	1.013
Skewness	− 1.193	− 1.273	− 0.907	− 1.322
Range	4	4	4	4
Minimum	1	1	1	1
Maximum	5	5	5	5
Sum	235	230	227	246
Count	60	60	60	60

courses being included in the curriculum of the Department of Informatics of a Greek University were classified into 11 groups of 5 courses, aiming to create learning pathways that allow students to acquire integrated knowledge and improve their learning outcomes. As such, 10 professors of the aforementioned department were participated in this case study, determining the attributes of the courses and the structure of the group, as well as evaluating the groups formed. Regarding the teaching experience of the professors, 2 of them had up to 10 years, 5 of them had 10–20 years, while the rest of them had more than 20 years. Moreover, their experience has been acquired at the university where the experiment took place. Each professor has taught 5–8 courses of the total 55 ones during their working experience. Based on the above, the professors had the asset to categorize the courses and evaluate the groups formed.

The groups decided to consist of courses with 2 homogeneous characteristics, namely prerequisite knowledge and subject area, and 2 heterogeneous ones, namely difficulty level and provided skills. After forming the groups, the professors were asked to evaluate the grouping regarding its appropriateness and acceptance, as well as their attitude toward the proposed automatic group formation. Figure 15 depicts the survey results, and Table 2 presents the descriptive statistics.

As shown, almost all the professors found the groups of courses suitable and they were satisfied with the results of group formation. In particular, the mean of rating group formation is 3.9, the median and the mode is 4, and the standard deviation is 0.994. The mean of rating user satisfaction is 3.9, with the value of 4 as median, 5 as mode, and 1.101 as standard deviation. Finally, they stated that they intend to use the system in future for automatically creating groups, indicating the system acceptance (mean: 3.8, median: 4, mode: 3, standard deviation: 1.033). The above findings suggest that the algorithm created adequate groups which met the characteristics set by the

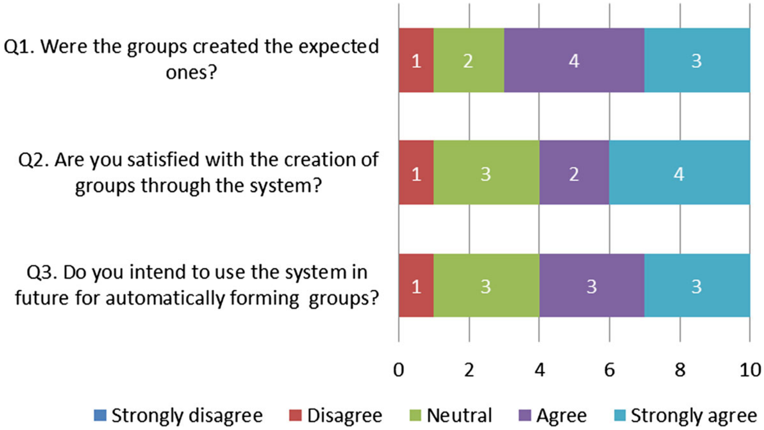


Fig.15 Survey results on objects' group formation using the proposed algorithm

professors. In addition, its performance met the professors' requirements, making them intend to use it in other circumstances as well (Table 3).

The proposed algorithm can be characterized as an effective solution in grouping problems, since it provides the capability to define the grouping characteristics, as well as to configure the genetic settings, according to problem requirements. This flexibility leads to the formation of efficient groups and enables it to be applied to a multitude of problems. Moreover, the algorithm performs effectively to different case studies, since it can handle efficiently data regardless of the problem domain.

Table 3 Descriptive statistics on results of objects' group formation using the proposed algorithm

	Q1	Q2	Q3
Mean	3.9	3.9	3.8
Standard Error	0.314	0.348	0.327
Median	4	4	4
Mode	4	5	3
Standard Deviation	0.994	1.101	1.033
Sample Variance	0.989	1.211	1.067
Kurtosis	– 0.157	– 1.236	– 0.896
Skewness	– 0.610	– 0.388	– 0.272
Range	3	3	3
Minimum	2	2	2
Maximum	5	5	5
Sum	39	39	38
Count	10	10	10

5 Conclusions

In this paper, a novel genetic algorithm for group formation is presented employing innovative genetic operators, such as a modification of 1-point crossover, a modification of 2-point crossover, the gene crossover and the group crossover, for greater performance and accuracy. Specifically, it is a domain-independent approach, since it is designed to allow any input regardless of the domain problem; i.e., whether the groups concern items or people, or the field of its application is industry, education, healthcare, etc. This feature is based on the idea that since the genetic algorithm only works with normalized numerical values, the significance of the traits employed by the algorithm to produce the best solution is irrelevant.

Regarding system evaluation, a series of experiments was conducted applying deferent generic grouping settings, in order to evaluate the performance and the accuracy of the algorithm. The findings revealed that tournament selection is better to be chosen when a quick solution is required and/or more crossover operators are used. Moreover, the introduced gene and group crossover operators produce adequate groups in good execution time comparing to classic ones. Combining three crossover operators can generate fitter groups; while the combination of more ones is useless since the accuracy is not improved significantly and the execution time is greatly increased. Concerning problem's domain, the algorithm was applied to two case studies, namely grouping people and items, and evaluated based on the acceptability of group formation by the individuals involved. In first case, the students participated in creating collaborative groups expressed their full satisfaction on group formation, enjoying the collaboration with other members and working effectively together. In second case, the professors who evaluated the groups of courses generated, found them suitable learning paths and stated that they intend to use the system in other circumstances as well, where automatic grouping is needed. To sum up, the experimentation results are very encouraging and report great effectiveness of the presented group recommender system in terms of the genetic settings and its application in different domains and grouping problems.

The future work of this research includes the refinement of the algorithm to improve its execution time and resources. More extensive experiments investigating the performance of operators will take place. In addition, the scope of the algorithm will be further expanded to test its performance in different contexts. Finally, the comparison of the proposed algorithm with similar recommender systems, freely available on the web, using accuracy, precision, recall, rank and other famous metrics, is part of our future plans.

Author contributions AK and CT wrote the main manuscript text. All authors reviewed the manuscript.

Funding Open access funding provided by HEAL-Link Greece.

Declarations

Conflict of interest The authors declare no competing interests.

References

Adinarayanan, A., Uthayakumar, M., Prabhakaran, G.: Machine cell formation using simulated annealing algorithm in cellular manufacturing system. Int. J. Comput. Aided Eng. Technol. **10**(1/2), 111–125 (2018)

Al-Ghuribi, S.M., Mohd Noah, S.A.: Multi-criteria review-based recommender system-the state of the art. IEEE Access **7**, 169446–169468 (2019). https://doi.org/10.1109/ACCESS.2019.2954861

Amara, S.,Subramanian R.R.: Collaborating personalized recommender system and content-based recommender system using TextCorpus. In: 2020 6th IEEE International Conference on Advanced Computing And Communication Systems (ICACCS), pp. 105–109, (2020) doi: https://doi.org/10.1109/ICACCS48705.2020.9074360.

Baker, O., Yuan, Q.: Machine learning: factorization machines and normalized discounted cumulative gain for tourism recommender system optimisation. IEEE Int. Conf. Comput. (ICOCO) **2021**, 31–36 (2021). https://doi.org/10.1109/ICOCO53166.2021.9673502

Bonfert-Taylor P, Miller, C.: Improving automated group assignments in an academic setting. In: Paper presented at the ASEE Annual Conference and Exposition, Conference Proceedings, (2020)

Borovič, M., Ferme, M., Brezovnik, J., Majninger, S., Kac, K., Ojsteršek, M.: Document recommendations and feedback collection analysis within the slovenian open-access infrastructure. Information **11**(11), 497 (2020)

Cavanaugh, R., Ellis, M., Layton, R., Ardis, M.: Automating the process of assigning students to cooperative-learning teams. In: Paper presented at the ASEE Annual Conference Proceedings, pp. 1037–1050 (2004).

Chen, C., Wu, X., Chen, J., et al.: Dynamic grouping of heterogeneous agents for exploration and strike missions. Front Inform Technol Electron Eng **23**, 86–100 (2022). https://doi.org/10.1631/FITEE.2000352

Chen, C. H., Chiang, B. Y., Hong, T. P.: An approach for optimizing group stock portfolio using multi-objective genetic algorithm. In: 2018 5th International Conference on Behavioral, Economic, and Socio-Cultural Computing (BESC) (pp. 213–215). IEEE (2018)

Chen, X., Liu, Q., Qiao, X.: Approaching another tourism recommender. In: 2020 IEEE 20th IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 556–562, (2020) doi: https://doi.org/10.1109/QRS-C51114.2020.00097.

Chniter, M., Abid, A., Kallel, I.: Towards a Bio-inspired ACO approach for building collaborative learning teams. In: 2018 17th International Conference on Information Technology Based Higher Education and Training (ITHET) (pp. 1–8). IEEE (2018)

Choi, K., Kim, K.Y., Kwon, O.: A need-awaring multi-agent approach for ad hoc need identification and group formation in nomadic community computing. Cybern. Syst.: an Int. J. **41**(3), 216–244 (2010)

Choi, S.-M., Jang, K., Lee, T.-D., Khreishah, A., Noh, W.: Alleviating item-side cold-start problems in recommender systems using weak supervision. IEEE Access **8**, 167747–167756 (2020)

Chrysafiadi, K., Troussas, C., Virvou, M.: A framework for creating automated online adaptive tests using multiple-criteria decision analysis. In: Proceedings of the 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 226–231, Miyazaki, Japan, (2018)

Contreras, R., Salcedo, P.: Genetic algorithms as a tool for structuring collaborative groups. Nat. Comput. **16**(2), 231 (2017)

Cruz, W.M., Isotani, S.: Group formation algorithms in collaborative learning contexts: a systematic mapping of the literature. In: Baloian, N., Burstein, F., Ogata, H., Santoro, F., Zurita, G. (eds) Collaboration and Technology. CRIWG 2014. Lecture Notes in Computer Science, vol. 8658. Springer, Cham (2014)

Falkenauer, E.: The grouping genetic algorithms-widening the scope of the GAs. Belg. J. Op. Res. Stat. Comput. Sci. **33**(1), 2 (1992)

Fernández-García, J., Rodríguez-Echeverría, R., Preciado, J.C., Manzano, J.M.C., Sánchez-Figueroa, F.: Creating a recommender system to support higher education students in the subject enrollment decision. IEEE Access **8**, 189069–189088 (2020). https://doi.org/10.1109/ACCESS.2020.3031572

Flanagan, B., Liang, C., Majumdar, R., Ogata, H.: Towards explainable group formation by knowledge map based genetic algorithm. In: 2021 International Conference on Advanced Learning Technologies (ICALT) (pp. 370–372). IEEE (2021)

Forghani, K., Fatemi Ghomi, S.M.T., Kia, R.: Solving an integrated cell formation and group layout problem using a simulated annealing enhanced by linear programming. Soft Comput. **24**, 11621–11639 (2020). https://doi.org/10.1007/s00500-019-04626-8

Fronita, M., Gernowo, R., Gunawan, V.: Comparison of genetic algorithm and hill climbing for shortest path optimization mapping. In E3S Web of Conferences (Vol. 31, p. 11017). EDP Sciences (2018)

Ge, Z., Qi, Z., Yang, L.: A literature review of grouping solutions in collaborative learning. In: 2018 IEEE International Conference on Progress in Informatics and Computing (PIC) (pp. 393–397). IEEE (2018).

Graf, S., Bekele, R.: Forming heterogeneous groups for intelligent collaborative learning systems with ant colony optimization. In: Ikeda, M., Ashley, K.D., Chan, TW. (eds) Intelligent Tutoring Systems. ITS 2006. Lecture Notes in Computer Science, vol. 4053. Springer, Berlin, Heidelberg (2006). Doi: https://doi.org/10.1007/11774303_22

Guan, B., Hu, L., Liu, P., Xu, H., Fu, Z., Wang, Q.: dpSmart: a flexible group based recommendation framework for digital repository systems. In: Proceedings of the 2019 IEEE International Congress on Big Data (BigDataCongress), pp. 111–120, Milan, Italy, (2019)

Gupta, M., Thakkar, A., Aashish, Gupta, V., Rathore, D.P.S.: Movie recommender system using collaborative filtering. In Proceedings of the 2020 IEEE International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 415–420, Coimbatore, India, (2020)

Hamrouni, A., Ghazzai, H., Alelyani, T., Massoud, Y.: An evolutionary algorithm for collaborative mobile crowdsourcing recruitment in socially connected IoT systems. In 2020 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT) (pp. 1–6). IEEE (2020).

Haq, I.U., Anwar, A., Rehman, I.U., Asif, W., Sobnath, D., Sherazi, H.H.R., Nasralla, M.M.: Dynamic group formation with intelligent tutor collaborative learning: a novel approach for next generation collaboration. IEEE Access **9**, 143406–143422 (2021)

Ho, T., Shyong, J. S., Wang, F., Li, C. T.: Composing high-heterogeneous and high-interaction groups in collaborative learning with particle swarm optimization. In Paper Presented at the 2009 WRI World Congress on Computer Science and Information Engineering, CSIE 2009, vol. 4 pp. 607–611 (2009) doi:https://doi.org/10.1109/CSIE.2009.876

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

Imbrie, P. K., Agarwal, J., Raju, G.: Genetic algorithm optimization of teams for heterogeneity. In 2020 IEEE Frontiers in Education Conference (FIE) (pp. 1–5). IEEE (2020).

Jomsri, P.: FUCL mining technique for book recommender system in library service. Procedia Manuf. **22**, 550–557 (2018)

Kallel, I., Chniter, M.: Building Collaborative e-learning teams in a smart education environment. In 2019 IEEE International Smart Cities Conference (ISC2) (pp. 324–329). IEEE (2019).

Kannikaklang, N., Wongthanavasu, S., Thamviset, W.: A hybrid recommender system for improving rating prediction of movie recommendation. In 2022 19th IEEE International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 1–6, (2022) Doi: https://doi.org/10.1109/JCSSE54890.2022.9836257.

Kbaier, M. E. B. H., Masri, H., Krichen, S.: A personalized hybrid tourism recommender system. In 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), pp. 244–250, (2017) doi: https://doi.org/10.1109/AICCSA.2017.12.

Krajčovič, M., Hančinský, V., Dulina, Ľ., Grznár, P., Gašo, M., Vaculík, J.: Parameter setting for a genetic algorithm layout planner as a toll of sustainable manufacturing. Sustainability **11**(7), 2083 (2019)

Krouska, A., Virvou, M.: An enhanced genetic algorithm for heterogeneous group formation based on multi-characteristics in social-networking-based learning. IEEE Trans. Learn. Technol. **13**(3), 465–476 (2019)

Krouska, A., Troussas, C., Virvou, M.: Applying genetic algorithms for student grouping in collaborative learning: a synthetic literature review. Intell. Decis. Technol. **13**(4), 395–406 (2019)

Krouska, A., Troussas, C., Sgouropoulou, C.: Applying genetic algorithms for recommending adequate competitors in mobile game-based learning environments. In Intelligent Tutoring Systems: 16th International Conference, ITS 2020, Athens, Greece, June 8–12, 2020, Proceedings 16 (pp. 196–204). Springer International Publishing (2020).

Kumar, N., Sinha, R.: Genetic algorithm optimized structured dictionary for discriminative block sparse representation. IEEE Access **8**, 19058–19073 (2020)

Liang, B., Xu, B., Wu X., et al.: A community-based collaborative filtering method for social recommender systems. In Proceedings of the 2019 IEEE International Conference on Web Services (ICWS), pp. 159–162, Milan, Italy, (2019)

Liu, S., Huang, F., Yan, B., Zhang, T., Liu, R., Liu, W.: Optimal design of multimissile formation based on an adaptive SA-PSO algorithm. Aerospace **9**(1), 21 (2021)

Mahdavi, M., Choubdar, H., Zabeh, E., Rieder, M., Safavi-Naeini, S., Jobbagy, Z., et al.: A machine learning based exploration of COVID-19 mortality risk. PLoS ONE **16**(7), e0252384 (2021). https://doi.org/10.1371/journal.pone.0252384

Marques, G.A., Rigo, S.J., Alves, I.M.D.R.: Graduation mentoring recommender - hybrid recommendation system for customizing the undergraduate student's formative path. XVI IEEE Latin Am. Conf. Learn. Technol. (LACLO) **2021**, 342–349 (2021). https://doi.org/10.1109/LACLO54177.2021.00107

Matviichuk, O., Nosovets, O., Linnik, M., Davydko, O., Pavlov, V., Nastenko, I.: Class-oriented features selection technology in medical images classification problem on the example of distinguishing between tuberculosis sensitive and resistant forms. In 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT) (Vol. 1, pp. 385–389). IEEE (2021)

Mills, K.L., Filliben, J.J., Haines, A.L.: Determining relative importance and effective settings for genetic algorithm control parameters. Evol. Comput. **23**(2), 309–342 (2015)

Molina, D., Lozano, M., Garcia-Martinez, C., Herrera, F.: Memetic algorithms for continuous optimisation based on local search chains. Evol. Comput. **18**(1), 27–63 (2010)

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)

Mughaid, I., Obeidat, B., Hawashin, S., AlZu'bi, D., Aqel.: A smart geo-location job recommender system based on social media posts. In: Proceedings of the 2019 Sixth IEEE International Conference on Social Networks Analysis, Management and Security (SNAMS), pp. 505–510, Granada, Spain, (2019)

Muruganandam, A., Prabhaharan, G., Asokan, P., et al.: A memetic algorithm approach to the cell formation problem. Int. J. Adv. Manuf. Technol. **25**, 988–997 (2005). https://doi.org/10.1007/s00170-003-1912-7

Putro, B. L., Rosmansyah, Y.: Group formation in smart learning environment: a literature review. In 2018 International Conference on Information Technology Systems and Innovation (ICITSI) (pp. 381–385). IEEE (2018)

Putro, B. L., Rosmansyah, Y., Agustine, S. S.: Intelligent agent to form heterogeneous group based on personality traits with genetic algorithm. In 2020 International Conference on Information Technology Systems and Innovation (ICITSI) (pp. 294–299). IEEE (2020)

Ramos, I.M.M., Ramos, D.B., Gadelha, B.F., de Oliveira, E.H.T.: An approach to group formation in collaborative learning using learning paths in learning management systems. IEEE Trans. Learn. Technol. **14**(5), 555–567 (2021)

Ramos-Figueroa, O., Quiroz-Castellanos, M., Mezura-Montes, E., Kharel, R.: Variation operators for grouping genetic algorithms: a review. Swarm Evol. Comput. **60**, 100796 (2021)

Sánchez, O.R., Ordóñez, C.A.C., Duque, M.Á.R., Pinto, I.I.B.S.: Homogeneous group formation in collaborative learning scenarios: an approach based on personality traits and genetic algorithms. IEEE Trans. Learn. Technol. **14**(4), 486–499 (2021)

Singla, R., Gupta, S., Gupta, A., Vishwakarma, D. K.: FLEX: a content based movie recommender. In Proceedings of the 2020 IEEE International Conference for Emerging Technology (INCET), pp. 1–4, Belgaum, India, (2020)

Sklab, Y., Aknine, S., Shehory, O., Ariouat, H.: Coalition Formation with Multiple Alternatives of Interdependent Tasks. In: Nouri, A., Wu, W., Barkaoui, K., Li, Z. (eds) Verification and Evaluation of Computer and Communication Systems. VECoS 2021. Lecture Notes in Computer Science, vol 13187. Springer, Cham (2022) Doi: https://doi.org/10.1007/978-3-030-98850-0_6

Srisawatsakul, C., Boontarig, W.: Tourism recommender system using machine learning based on user's public instagram photos. In 5th IEEE International Conference on Information Technology (InCIT), pp. 276–281, (2020) Doi: https://doi.org/10.1109/InCIT50588.2020.9310777.

Sukstrienwong, A.: Group formation based on students' learning styles by circular genetic algorithm. TEM Journal **10**(3), 1016–1021 (2021). https://doi.org/10.18421/TEM103-02

Talavera-Mendoza, F., Atencio-Torres, C.E., Del Carpio, H., Deza, D.A., Cayro, A.R.: Usability analysis and clustering model in e-learning from the user experience perspective. Int. J. Inf. Edu. Technol. **12**(2), 108–115 (2022). https://doi.org/10.18178/ijiet.2022.12.2.1593

Tariq, A., Hussain, I., Ghafoor, A.: A hybrid genetic algorithm for machine-part grouping. Comput. Ind. Eng. **56**(1), 347–356 (2009)

Tian, L., Hua, Y., Dong, X., Lv, J., Ren, Z.: Distributed time-varying group formation tracking for multi-agent systems with switching interaction topologies via adaptive control protocols. IEEE Trans. Ind. Inf. **18**(12), 8422–8433 (2022)

Troussas, C., Krouska, A., Virvou, M.: Multi-algorithmic techniques and a hybrid model for increasing the efficiency of recommender systems. In Proceedings of the 2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 184–188, Volos, Greece, (2018)

Troussas, C., Krouska, A., Virvou, M.: Adaptive e-learning interactions using dynamic clustering of learners' characteristics. In 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA) (pp. 1–7). IEEE (2019)

Troussas, C., Krouska, A., Sgouropoulou, C.: Enhancing human-computer interaction in digital repositories through a MCDA-based recommender system. Adv. Hum.-Comput. Interact. (2021). https://doi.org/10.1155/2021/7213246

Wang, Y.H., Li, Y.C., Liao, H.C.: Using a genetic algorithm to determine optimal complementary learning clusters for ESL in Taiwan. Expert Syst. Appl. **38**(12), 14832–14837 (2011)

Wongkhamchan, T., Namvong, A., Surawanitkun, C.: Personalized recommender system using a social network based collaborative filtering technique. In Proceedings of the 2019 16th IEEE International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), pp. 846–849, Pattaya, Thailand, (2019)

Xiaohui, T.: An adaptive genetic algorithm-based background elimination model for English text. Soft Comput (2022). https://doi.org/10.1007/s00500-022-07204-7

Yang, X., Dong, M., Chen, X., Ota, K.: Recommender system-based diffusion inferring for open social networks. IEEE Trans. Comput. Soc. Syst. **7**(1), 24–34 (2020)

Yannibelli, V., Amandi, A.: A memetic algorithm for collaborative learning team formation in the context of software engineering courses. In: Cipolla-Ficarra, F., Veltman, K., Verber, D., Cipolla-Ficarra, M., Kammüller, F. (eds) Advances in New Technologies, Interactive Interfaces and Communicability. ADNTIIC 2011. Lecture Notes in Computer Science, vol 7547. Springer, Berlin, Heidelberg (2012). Doi: https://doi.org/10.1007/978-3-642-34010-9_9

Zheng, Y., Li, C., Liu, S., Lu, W.: An improved genetic approach for composing optimal collaborative learning groups. Knowl.-Based Syst. **139**, 214–225 (2018)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Akrivi Krouska is a Postdoctoral Researcher in the Department of Informatics and Computer Engineering, University of West Attica, Greece. She has received a Ph.D. Degree in Informatics from the Department of Informatics, University of Piraeus, Greece, a M.Sc. degree in "Information Systems" from Athens University of Economics and Business, Greece and a B.Sc. degree in Informatics from University of Piraeus, Greece. She has published a significant number of articles in international conferences, books and journals. Her research interests are in the areas of social networking services, sentiment analysis, learning analytics, user modeling and artificial intelligence in education.

Christos Troussas is Assistant Professor with the Department of Informatics and Computer Engineering at the University of West Attica. He has been PostDoc Researcher in the field of Software Engineering at the same Department. He has received a PhD degree in Informatics, a MSc degree with specialization in "Intelligent Technologies of Human-Computer Interaction" and a BSc degree in Informatics from the Department of Informatics, University of Piraeus. He has participated in national and international research projects. He has been Editorial Board member in many international journals, and Program Committee chair or member in many international conferences. He has published extensively in international journals, books and conference proceedings. His current research interests are in the areas of personalized software technology, humancomputer interaction and adaptive hypermedia systems.

Cleo Sgouropoulou is Professor with the Department of Informatics and Computer Engineering at the University of West Attica. She holds a Diploma in Electrical andComputer Engineering from the National Technical University of Athens, and a PhD from the same university, in Software Engineering with Emphasis on Learning Technology Systems. Her research interests lie in the fields of design, development and standardisation of Learning Technology and Research Information Systems. Prof. Sgouropoulou has published her work in several scientific journals and conference proceedings and has participated as coordinator or main researcher in numerous national and European-funded projects.