Forming Heterogeneous Groups for Intelligent Collaborative Learning Systems with Ant Colony Optimization [*]

Sabine Graf[1] and Rahel Bekele[2]

[1] Vienna University of Technology, Austria
Women's Postgraduate College for Internet Technologies
`graf@wit.tuwien.ac.at`,
[2] Addis Ababa University, Ethiopia
Faculty of Informatics
`rbekele@sisa.aau.edu.et`

Abstract. Heterogeneity in learning groups is said to improve academic performance. But only few collaborative online systems consider the formation of heterogeneous groups. In this paper we propose a mathematical approach to form heterogeneous groups based on personality traits and the performance of students. We also present a tool that implements this mathematical approach, using an Ant Colony Optimization algorithm in order to maximize the heterogeneity of formed groups. Experiments show that the algorithm delivers stable solutions which are close to the optimum for different datasets of 100 students. An experiment with 512 students was also performed demonstrating the scalability of the algorithm.

1 Introduction

Cooperative learning is one of the many instructional techniques to enhance student performance described in the academic literature [4], [13], [19]. While the advantages of cooperative learning are very well documented [1], [11], [12], making it more efficient by creating heterogeneous groups has been given little attention. Researchers in the area of cooperative learning also claim that many of the unsuccessful outcomes of group work stem from the formation process (e.g., [14], [18]). Although group formation is said to play a critical role in terms of enhancing the success of cooperative learning ([12], [19]) and therefore increasing the learning progress of students, it is observed that there is only little research done that addresses the formation of groups in a heterogeneous way.

Moreover, the potentials of computer-based methods to assist in the group formation process have not been explored fully. Despite the popularity of computer-based tools to support collaborative learning [3], [8], [15], designers mainly focus on collaborative interaction to address the techniques of sharing information and resources between students. Inaba [10] incorporated the grouping aspect

---

and constructed a collaborative learning support system that detects appropriate situations for a learner to join in a learning group. Also Greer et al. [9] considered the formation of groups in tools that address the issue of peer-help.

While the above systems have proved to be appropriate in several contexts, they do not specifically reveal how the groups can be initially formed. It seems that considerations of the personality attributes are usually neglected in forming groups. The objective of this paper is, therefore, to address this limitation incorporating personality attributes as well as the performance level to form heterogeneous groups.

The research work has two main goals. The first one is the development of a mathematical model (Sect. 2) that addresses the group formation problem through the mapping of both performance and personality attributes into a student vector space. This serves as a foundation for the application of formal methods in the determination of heterogeneous groups. The second one is to provide a tool that implements the mathematical model. This tool can be used to supplement existing intelligent collaborative learning systems which do not consider the formation of heterogeneous groups so far. As a consequence, learners get more out of collaborative learning and their learning progress increases.

For maximizing the heterogeneity of the groups, the tool uses an Ant Colony Optimization algorithm described in Sect. 3. We describe in Sect. 4, how the algorithm is adopted in the group formation problem and experiments applying the algorithm with real-world data are presented in Sect. 5.

2 The Mathematical Approach of Group Formation

In this section, the conceptual framework for our mathematical model to form heterogenous groups of students is described.

2.1 The Student Space

For the definition of the student space, attributes whose values can be obtained from easily available indicators are selected based on expert opinion and discussion with colleagues. These attributes are *group work attitude, interest for the subject, achievement motivation, self-confidence, shyness, level of performance in the subject, and fluency in the language of instruction*. Each of these attributes has three possible values, where 1 indicates a low and 3 a high category value.

By applying the concepts of a vector space model, each student is represented in a multi-dimensional space by a vector whose components are made up of the values of personality and performance attributes. For instance, student S_1 may be represented by the vector $S_1(3, 1, 2, 1, 3, 3, 2)$, indicating that group work attitude is positive, interest of the subject is low and so on. For collecting the values of the attributes in order to apply the approach in real-world, a data collection instrument was designed and available at [2].

The *student-score* for a particular student, used to measure heterogeneity, represents the total score of a student computed as the sum of all values of the student's attributes.
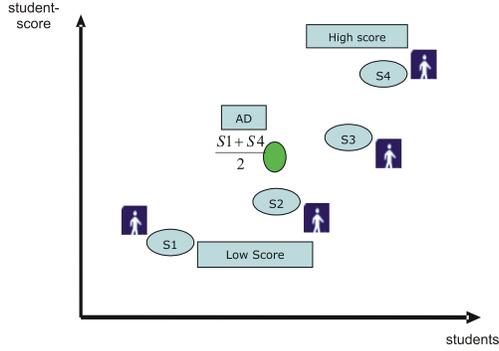
Fig. 1. Illustration of the measure of goodness of heterogeneity

2.2 Heterogeneity of Students

In heterogeneous groups, it is important that students have different values of the attributes considered. This may be measured by the Euclidean distance (ED) between two students.

Let $ED(S_1, S_2)$ be defined as the distance between the vectors representing two students in space. Applying the Euclidean distance, this becomes

$$ED(S_1, S_2) = \sqrt{\sum_{i=1}^{n}(A_i(S_1) - A_i(S_2))^2} \ , \tag{1}$$

where $A_i(S_j)$ represents the value for a particular attribute A_i for a student S_j and n represents the number of attributes.

2.3 Goodness of Heterogeneity in Groups

As shown in Fig. 1, a reasonably heterogeneous group refers to a group where student-scores reveal a combination of low, average and high student-scores. This is justified by the recommendation of Slavin [18] who proposes that students should work in small, mixed-ability groups of four members: one high achiever, two average achievers, and one low achiever. This idea is extended further and applied in student-scores.

The measure of goodness of heterogeneity (GH) is developed with the assumption that in a reasonably heterogeneous group, after taking the maximum and minimum student-score, the rest of the student-scores are expected to lie half way between the maximum and minimum score. In this case, the absolute difference of the average difference (AD) and the rest of the student-scores is minimal. Figure 1 illustrates the concept of the goodness of heterogeneity, assuming that each group has four members. In the following, we assume and also recommend a group size of four, as it is also suggested by Slavin [18]. Nevertheless, the group size can also be extended or reduced by increasing or decreasing the number of students with average score.

The measure of GH can be computed as follows. Let AD_i be the average of the maximum and the minimum student-score in the i-th group.

$$AD_i = \frac{\max scoreof(S_1, S_2, S_3, S_4) + \min scoreof(S_1, S_2, S_3, S_4)}{2} \ . \quad (2)$$

The measure of goodness of heterogeneity is then defined as

$$GH_i = \frac{\max scoreof(S_1, S_2, S_3, S_4) - \min scoreof(S_1, S_2, S_3, S_4)}{1 + \sum_j |AD_i - scoreof(S_{j(i)})|} \ , \quad (3)$$

where $S_{j(i)}$ is the student-score of the j-th student in group i, excluding the maximum and the minimum student-score.

Where a reasonable heterogeneity is experienced, the numerator in (3) should be greater than the denominator hence yielding a relatively high value of GH_i. It is trivial to show that $GH_i = 0$ when all students in a group have equal student-scores; $GH_i < 1$ when there is unreasonable heterogeneity in the group (meaning student-scores are at two extremes) and $GH_i > 1$ in reasonably heterogeneous groups. The greater GH_i, the better the heterogeneity.

2.4 Forming Heterogeneous Groups

An experiment by Bekele [2] shows that students who were grouped according to GH perform better than students grouped randomly or on a self-selection basis. But the GH deals on the basis of score values and does not distinguish between the individual characteristics. To address the limitation of GH, our approach additionally incorporates the Euclidean distance between the group members in the process of forming heterogeneous groups.

Considering the group building process as a whole, we have another aim regarding the goodness of heterogeneity. Aiming only at high GH values will result in some groups with very high GH and the remaining students will form groups with low GH. To form groups with a similar degree of heterogeneity, the deviation of GH values need to be considered additionally.

Thus, the objective of building heterogeneous groups can be formulated as follows:

$$F = w_{GH} \cdot GH + w_{CV} \cdot CV + w_{ED} \cdot ED \rightarrow max \ , \quad (4)$$

where GH is the sum of the goodness of heterogeneity values, as defined in (3), of all groups. CV is the coefficient of variation based on all GH values and ED is the Euclidean distance of all groups, whereby the Euclidean distance of one group can be calculated by summing up the Euclidean distance between all combinations of group members according to (1). Each of these terms is weighted by the corresponding w. Aiming at a high heterogeneity, the fitness F should be maximized.

As can be seen, forming heterogeneous groups is not trivial. In a former experiment by Bekele [2], an iterative algorithm was developed to build heterogeneous groups based on GH. Euclidean distance is considered by the restriction

that ED between at least two students has to exceed a certain threshold. By extending the objectives of [2] and including the Euclidean distance and the coefficient of variation of GH values in the optimization process, the problem becomes even more complex. For this reason and also because the problem is an NP-hard problem, we developed a tool based on an artificial intelligence approach, namely Ant Colony Optimization. In the next section, Ant Colony Optimization is introduced.

3 Ant Colony Optimization

Ant Colony Optimization (ACO) [5] is a multi-agent meta-heuristic for solving NP-hard combinatorial optimization problems, e.g. the travelling salesman problem. In the following, a brief introduction into ACO as well as a description of the applied algorithm is provided.

3.1 Background

ACO algorithms are inspired by the collective foraging behaviour of specific ant species. When these species of ants are searching for food sources they follow a trail-laying trail-following behaviour. Trail-laying means that each ant drops a chemical substance called pheromone on its chosen path. Trail-following means that each ant senses its environment for existing pheromone trails and their strength. This information builds the basis for their decision which path to follow. If there is a high amount of pheromones on a path, the probability that the ant will choose it is also high. If there is a low amount of pheromones, the probability is low. The more often a path is chosen, the more pheromones are laid on it which increases the probability that it will be chosen again. Since the decision is based on probabilities, an ant does not always follow the way that has the highest pheromone concentration. Paths which are marked as poor are also chosen, but with lower probability. Pheromones evaporate over time, leading to the effect that rarely used trails will vanish. These strategies enable natural ants to build a map of pheromone trails which indicates the best paths to a food source.

Several ACO algorithms exist that model and exploit this behaviour for solving graph-based NP-hard combinatorial optimization problems. One of the biggest advantages of ACO algorithms is that they can be applied very well to different optimization problems. The only requirement is that the problem can be represented as a graph, where the ants optimize according to the best path through this graph.

3.2 Ant Colony System

Ant Colony System (ACS) [6] is one of the most successfully applied ACO algorithms. In [7], Dorigo and Gambardella compared ACS with other optimization algorithms, e.g., neural networks and genetic algorithms for different instances of

the travelling salesman problem. As a result, it is shown that ACS is competitive to the other algorithms and sometimes even finds better solutions.

The procedure of ACS is as follows. The first step is to represent the problem as graph where the optimum solution is a certain - e.g. the shortest - way through this graph. After initializing each edge of the problem graph with a small amount of pheromones and defining each ant's starting node, a small number of ants (e.g., 10) runs for a certain number of iterations. For every iteration, each ant determines a path through the graph from its starting node to the destination node. It does this by applying a so-called random proportional transition rule at each decision point. This rule decides which of all possible next nodes l included in the list J to choose, based on (1) the specific edge's amount of pheromones, also called global information τ, and (2) local information η representing the costs or utility of choosing the node. Equation (5) describes how to calculate the probability p that ant k goes from node i to node j.

$$p_{ij}^k = \frac{[\tau_{ij}] \cdot [\eta_{ij}]^\beta}{\sum_{l \in J_i^k} \left([\tau_{il}] \cdot [\eta_{il}]^\beta \right)} \quad . \tag{5}$$

The transition rule itself consists of two strategies. In the *exploring strategy* the ants act similar to natural ants by deciding according to the probabilities p_{ij}^k. In the *exploiting strategy* the already gathered knowledge about the problem is used straight forward, choosing the node that fits best according to its local and global information. Which strategy is used is decided randomly for each transition whereby the parameter $q0$ determines the probability.

When the ant arrives at the destination node, the fitness of the newly found solution is calculated. In case the newly found solution outperforms the existing solutions, it is saved to memory as the currently best one. Additionally, to avoid that succeeding ants chose the same path, a local pheromone trail update rule is applied, decreasing the amount of pheromones on the found path slightly.

After all ants have found a solution, the ant which found the best one so far spreads pheromones according to the pheromone trail update rule. Furthermore, the amount of pheromones on each edge is reduced by the evaporation factor ρ.

ACS can be improved by additionally combining it with a local search method. This local search method can be embedded in different ways. The most usual way is to apply it to each found solution [6].

4 Forming Groups with Ants

In the following, we describe how we applied the ACS algorithm to the group forming problem and the necessary modifications to solve the problem with ACS.

4.1 Representing the Group Forming Problem as Graph

As already mentioned, the only requirement to use ACO algorithms is to represent the problem as graph. The representation form we used is based on the idea
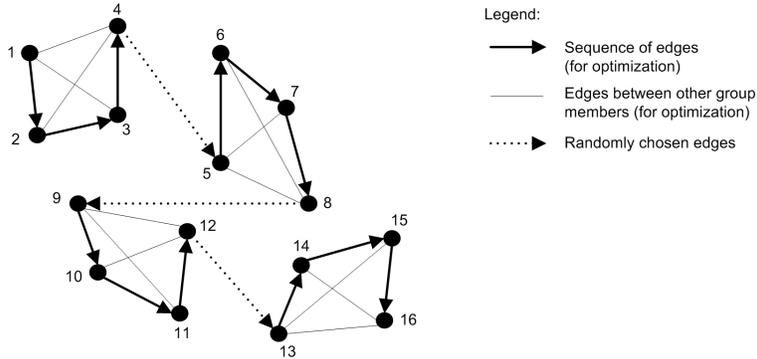
Fig. 2. Representation of the grouping problem as graph (group size = 4)

of ordering students comparable to the travelling salesman problem. The first m students belong to the first group, the second m student to the second group and so on, whereby m is the maximum number of students per group. Figure 2 shows this representation for a group size of four students, whereby the order is indicated by arrows. Having in mind that edges are used for pheromones and therefore indicate how good this edge is, within a group each newly assigned group member is linked not only to the last assigned group member but also to all other members of the group (indicated by solid lines in Fig. 2). This is because the important information for optimization is not the order in which the students are assigned to a group but the fact that exactly these m students belong together. Therefore, also the decision which student starts a new group is performed randomly (see dotted arrows in Fig. 2).

4.2 Applying ACS

For applying ACS to our grouping problem, we need to decide how to measure the local information of an edge. In our case, local information means the benefit to add a specific student to a group to which some group members already are assigned. As described in Sect. 2, heterogeneity of a group depends on the Euclidean distance between all group members and the GH of the group. Regarding ED, the benefit of adding a specific student is the sum of the ED of the new student and all already assigned group members. Because GH can be only calculated if the group is completed, the benefit for adding a student is based on the difference between the scores of the students in the best possible group and the scores of the students in the current group incorporating the specific positions of each student (one high score, one low score, and two average scores). Both local information values are normalized so that a high value indicates a good result and all values are between 0 and 1. For calculating the overall local information of an edge, both information values are weighted and summed up.

The global information is mainly calculated according to ACS. The only modification which has to be done for the grouping problem is that updating

pheromones, in both pheromone trail update rules, is done for the edges between the newly assigned student and all other group members rather than only for the edge between the newly assigned student and the student which is assigned last. The amount of pheromones is for each of these edges equal.

The measurement of the quality of a solution is calculated according to the objective function described in (4). The objective of the algorithm is to maximize the heterogeneity of all groups based on the GH value of all groups, the coefficient of variation of these GH values, and the overall Euclidean distance. To improve the performance of ACS, a local search method called 2-opt [16] is applied to each solution an ant found.

5   Experiments and Results

This section demonstrates that the group formation based on ACS works effectively using real-world data. Based on 512 student data records we created five randomly chosen datasets of 100 records to demonstrate that the proposed algorithm works not only for a specific dataset but also for randomly chosen real-world data. Additionally, we show one experiment with all 512 records to show the scalability of our approach.

Each experiment consists of 20 runs. The parameter for ACS are assumed according to literature [6] or based on experiments. We assumed $\beta=1$, $\rho=0.1$, $q0=0.9$, and the number of ants=10. The weights are decided as follows: $w_{GH}=$ 0.35, $w_{CV}=0.15$, and $w_{ED}=0.5$. Because the coefficient of variation impacts the GH values, we assumed w_{GH} and w_{CV} together as important as w_{ED}.

A run with 100 students stops after at least 100 iterations and only when the solution has not changed over the last $t*2/3$ iterations where t is the number of already calculated iterations. In all experiments, the GH values and the CV values were stable, indicating that the best values were already found, and the values of ED varied only slightly per run. Looking at Tab. 1, this can also be seen by the small CV values of the fitness. These values show that the solutions of each run are similar and indicate that the algorithm finds solutions which are stable and close to the optimum for all datasets with 100 students.

Table 1. Results of different datasets

Dataset	No. of students	Average GH	Average CV	Average ED	Average Fitness	SD Fitness	CV Fitness
A	100	129.813	39.223	363.936	52.141	0.033	0.064
B	100	117.200	35.182	377.415	51.558	0.029	0.057
C	100	114.234	41.906	374.147	49.422	0.033	0.067
D	100	132.176	31.344	354.588	52.584	0.027	0.050
E	100	131.958	31.437	372.214	54.870	0.046	0.084
F	512	537.595	45.552	1915.024	46.704	0.370	0.793

Because of the NP-hard nature of the problem, some modifications for running the experiments with 512 students were necessary. The main issue of scal-

ability is the local search method. Therefore, we modified it by applying 2-opt not for all students but only for 20 % of the students which were randomly selected for each solution. This approach is also used successfully by Lo et al. [17]. Furthermore, the general goal changed from looking for a solution which is close to the optimum to finding a good solution. Therefore, the termination condition changed to stopping after 200 iterations. As can be seen in Tab. 1 the CV value of the fitness is higher than for the experiments with 100 students but it is still less than 1. This indicates that the found solutions are stable, good solutions but not that close at the optimum than for the experiments with 100 students.

Comparing the result of the experiment with 512 students with the result of the iterative algorithm in [2], aimed at finding heterogeneous groups according to the goodness of heterogeneity, it can be seen that the proposed algorithm delivers much better results. The iterative algorithm results in an average GH value of 1.6 per group while the proposed algorithm found an average GH value of 4.2. Regarding ED, the iterative algorithm considers only the maximum difference of two students in a group while the proposed algorithm includes the ED values of all combinations of group members. Nevertheless, the average ED values of the proposed algorithm are slightly higher which indicated a much better heterogeneity.

6 Conclusions and Future Work

In this paper we have presented a mathematical approach for forming heterogeneous groups of students based on their personality traits and performance. The approach is based on the different characteristics of the students, a general measure of the goodness of heterogeneity of the groups, and its coefficient of variation. The second aim of this paper was to present a tool that implements the proposed mathematical approach by using an Ant Colony Optimization algorithm. Experiments were performed, showing that the algorithm finds stable solutions close to the optimum for different datasets, each consisting of 100 students. An experiment with 512 students was performed demonstrating the scalability of the algorithm.

Because building heterogeneous groups improves the learning progress in collaborative learning, future work will deal with combining the tool with online learning systems, especially collaborative intelligent tutoring systems. We plan to develop a mediator agent that facilitates the group formation process, and to implement it in an already existing system. Another issue for future work is to provide the users with more options to adjust the algorithm, for example, to allow the user to determine a certain duration of running the algorithm or also a certain quality of solution.

References

1. Ames, C., and Ames, R. (eds.): Research on Motivation in Education. Academic Press Inc., Orlando, USA (1985)

2. Bekele, R.: Computer Assisted Learner Group Formation Based on Personality Traits. Ph.D Dissertation, University of Hamburg, Hamburg, Germany (2005). Retrieved 5 January, 2006, from http://www.sub.uni-hamburg.de/opus/volltexte/2006/2759

3. Collins, A., Brown, J.S.: The Computer As A Tool For Learning Through Reflection. In: Mandl, H., Lesgold, A. (eds.): Learning Issues For Intelligent Tutoring Systems, Springer Verlag, New York (1988) 1–18

4. Dansereau, D., Johnson, D.: Cooperative learning. In: Druckman, D., Bjork, R.A.: Learning, Remembering, Believing: Enhancing Human Performance, National Academy Press, Washington, D.C. (1994) 83–111.

5. Dorigo, M., Di Caro, G.: The Ant Colony Optimization Meta-Heuristic. In: D. Corne, M. Dorigo, F. Glover (eds.), New Ideas in Optimization, McGraw-Hill, London, UK (1999) 11–32

6. Dorigo, M., Gambardella, L.M.: Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. IEEE Transactions on Evolutionary Computation, Vol. 1, No. 1 (1997) 53–66

7. Dorigo,M and Gambardella, L.M.: Ant Colonies for the Traveling Salesman Problem, BioSystems, Vol. 43, No. 2 (1997) 73-81

8. Florea, A.M.: An Agent-Based Collaborative Learning System: Advanced Research In Computers and Communications in Education. In Proc. of the Int. Conf. on Computers in Education, IOS press (1999) 161–164

9. Greer, J., McCalla, G., Cooke, J., Collins, J., Kumar, V., Bishop, A., Vassileva, J.: The Intelligent Help Desk: Supporting Peer-Help in a University Course. In Proc. of the Int. Conf. on Intelligent Tutoring Systems, Springer-Verlag (1998) 494–503.

10. Inaba, A., Supnithi, T., Ikeda, M., Mizoguchi, R., Toyoda, J.: How Can We Form Effective Collaborative Learning Groups? In Proc. of the Int. Conf. on Intelligent Tutoring Systems, Springer-Verlag (2000) 282–291.

11. Jacobs, G.: Cooperative Goal Structure: A Way to Improve Group Activities. ELT Journal. Vol. 42, No. 2 (1988) 97–100.

12. Johnson, D.W., Johnson, R.T.: Cooperative Classrooms. In: Brubacher, M. (eds.): Perspectives on Small Group Learning: Theory And Practice, Rubican Publishing Ind., Ontario (1990)

13. Johnson, D.W., Johnson, R.T.: Leading the Cooperative School. Interaction Book, Edina, MN (1989)

14. Johnson, D.W., Johnson, R.T.: The Internal Dynamics of Cooperative Learning Groups. In: Slavin, R., et al. (eds.): Learning to Cooperate, Cooperating to Learn, Plenum, New York (1985) 103–124

15. Krejins, K., Kirschner, P.A., Jochems, W.: The Sociability of Computer-Supported Collaborative Learning Environments. Educational Technology and Society, Vol. 5, No. 1 (2002) 26–37

16. Lin, S.: Computer Solutions of the Traveling Salesman Problem, Bell Systems Journal, Vol. 44, (1965) 2245–2269

17. Lo, C.D., Srisa-an, W., Chang, J.M., Chern, J.C.: The Effect of 2-opt and Initial Population Generation on Solving the Traveling Salesman Problem unsing Genetic Algorithms. In Proc. of World Multiconference on Systemics, Cybernetics and Informatics (2001) 282–287

18. Slavin, R.E.: Developmental and Motivational Perspectives on Cooperative Learning: A Reconciliation. Child Development, Vol. 58, No. 5., Special Issue on Schools and Development (1987) 1161–1167

19. Slavin, R.E.: When Does Cooperative Learning Increase Achievement? Psychological Bulletin, Vol. 94 (1983) 429–445.