Annales Mathematicae et Informaticae
63 (2026) pp. 1–17
DOI: 10.33039/ami.2026.06.001
URL: https://ami.uni-eszterhazy.hu

Fair team formation for STEAM education using exact and heuristic constraint-based solvers

Ali Adil Ali[a], **Gergely Kovásznai**[b], **Marcell Herbák**[b], **Mustafa Khalid**[c]

[a]University of Debrecen
ali.adil@inf.unideb.hu

[b]Eszterházy Károly Catholic University
kovasznai.gergely@uni-eszterhazy.hu, herbak.marcell@uni-eszterhazy.hu

[c]Imam Jaafar Al-Sadiq University
mustafa.khalid@ijsu.edu.iq

Abstract. STEAM (Science, Technology, Engineering, Arts, and Mathematics) education promotes interdisciplinary learning through collaboration and project-based activities. Constructing fair and balanced student teams under different limitations, due to their skills, gender, and academic training, is a difficult combinatorial issue. In this paper, we model the team formation task as a constraint satisfaction problem and provide a comparative analysis of the Z3 SMT solver, OR-Tools' CP-SAT constraint programming solver, and the heuristic Genetic Algorithm. We carry out experiments on synthetic datasets of varying sizes (50, 100, and 500 students) under identical constraints and evaluation criteria. Findings reveal an obvious trade-off between fairness and computational efficiency. Specifically, the two exact methods (Z3 and CP-SAT) both produce more balanced teams with lower gaps in fairness consistently, as compared to the Genetic Algorithm, which has great variability in solution quality. Based on performance, Z3 has the fastest solving time; however, its overall execution time increases dramatically for larger datasets. CP-SAT, on the other hand, tends to achieve more stable and scalable results across datasets of different sizes. These results aid in establishing a practical framework that supports the selection and implementation of the most effective strategies for creating educational teams.

Keywords: team formation, STEAM education, constraint satisfaction, Sat-

Submitted: March 31, 2026
Accepted: June 30, 2026
Published online: July 17, 2026

isfiability Modulo Theories (SMT), OR-Tools, CP-SAT, Genetic Algorithm, fairness optimization

2020 *Mathematics Subject Classification:* Primary 60F15, 60G42, 60G50; Secondary 60F10

1. Introduction

STEM and STEAM education refers to combinations of technology, engineering, arts, and arithmetic into a unified framework aimed at growing learners' multidisciplinary competencies [1]. With the rapid acceleration of virtual transformation, the call for competencies in those fields is developing exponentially, with projections indicating that STEM jobs will develop at a substantially faster price than other sectors in the course of the period 2021–2031 [15]. In this context, instructional projects including Coding for Kids Iraq [5] play a pivotal role in growing early programming talents and fostering computational questioning among students. In the domain of collaborative STEAM education, the formation of balanced and equitable learning groups plays an important role in efficiently achieving learning goals [10]. Traditional grouping methods, such as the use of randomness, self-selection, or inference based on the instructor's judgment, often do not optimize learning goals [9]. In addition, the use of guided group formation faces significant challenges in dealing with various overlapping constraints. Due to the combinatorial nature of the problem, it is classified as NP-hard, where the search space of the problem tends to increase exponentially along with the size of the dataset, thus calling for the use of advanced computational methods for efficient processing. To achieve this objective, a comparative analysis of three computational models of automated team formation is presented: the Satisfiability Modulo Theories (SMT) solver Z3, the constraint programming solver CP-SAT from Google's OR-Tools, and an evolutionary optimization approach, the Genetic Algorithm (GA). Through the evaluation of these models, the objective is to elucidate the effectiveness of each approach in the context of education. The following research questions are addressed by the paper:

RQ1. How successful is automated optimization in dealing with multi-dimensional constraints to ensure fairness in STEAM training contexts?

RQ2. What are the points of assessment among Constraint Programming (CP-SAT), Genetic Algorithms (GA), and SMT (Z3) in terms of adherence to constraints and efficiency?

RQ3. To what extent are the overall performance and scalability of the solving approaches (Z3, CP-SAT, and GA) affected by the growth in dataset size?

2. Related work

The problem of team formation has been studied in educational and organizational settings, and its goal has been to form balanced and effective teams based on different criteria such as skills, diversity, and collaboration, among others. Due to the satisfaction of different constraints, team formation problems are complex and can be categorized as NP-hard problems. To cope with computation complexity, different approaches have been proposed, such as metaheuristic, exact optimization, and hybrid models, among others. For instance, metaheuristic approaches such as evolutionary and genetic algorithms have been effectively used for team formation problems due to their ability to efficiently search through large solution spaces. For example, in [3], a team formation problem using an evolutionary metaheuristic approach has been proposed, considering different criteria such as team size, compulsory and forbidden combinations, among others, in a classroom setting. Similarly, in [12], a multi-objective genetic algorithm has been proposed for collaborative learning, considering different attributes such as knowledge, communication, and leadership, among others, for forming effective groups among students. The proposed approaches in these studies are highly scalable and flexible, and they do not guarantee optimal results, which might affect reproducibility depending on parameter tuning.

Recently, there have been some studies that investigate the use of computational algorithms in real classroom settings. For instance, [14] discusses various team formation algorithms and highlights the significance of choosing suitable algorithms depending upon the educational setting. Nevertheless, there has been a lack of thorough comparison between fundamentally diverse paradigms, especially between heuristic and exact optimization algorithms. To overcome the limitations associated with heuristic algorithms, exact optimization algorithms have been proposed. In [2], an SMT-based framework using the Z3 solver has been proposed to form fair teams considering various constraints, including team size, skill thresholds, gender, and diversity of backgrounds. The results show that SMT is capable of efficiently handling complex constraints; however, only small-scale problems have been considered in this study. Mathematical optimization techniques are also being implemented in team formation problems. In [4], the authors presented an integer linear programming model in forming teams of students while satisfying certain constraints like the size of the teams and pairing constraints. This model proves the achievement of optimal results for small and medium-sized problem sets. However, the problem's size affects the computational complexity of the problem.

More recent work has been done on fairness and diversity in team formation. In [8], an algorithm for diversity-aware team formation is proposed, where priority-based optimization and hill-climbing methods are utilized. In this work, intra-heterogeneity and inter-homogeneity metrics are proposed for evaluating the quality of formed teams. Despite its competitive performance, this method is based on heuristics and lacks comparison with other optimization approaches. On the whole, existing research works demonstrate the trade-off between the quality of the solution and the efficiency of the computation, as summarized in Table 1.

Nevertheless, there is a lack of unified experimental approaches to compare various solving paradigms under the same conditions, especially with regard to the aspects of fairness and scalability. This study aims to address the aforementioned problem and present a comparative analysis of the SMT approach (Z3), constraint programming (CP-SAT), and genetic algorithms.

Table 1. Comparison of related work in team formation.

Approach	Advantages	Limitations
Evolutionary Meta-heuristic [3]	Supports flexible team sizes; handles complex constraints (e.g., compulsory and forbidden conditions); demonstrates good scalability	Does not guarantee optimality; lacks comparison with exact optimization methods; fairness is not explicitly modeled
Genetic Algorithm (Multi-objective) [12]	Considers multiple student characteristics; validated through real-world classroom experiments; improves collaborative learning outcomes	Heuristic approach with no guarantee of optimality; no comparison with exact solvers; fairness is not explicitly quantified
SMT Solving (Z3) [2]	Guarantees strict constraint satisfaction; effective for fair team formation; fast on small datasets	Limited scalability evaluation; tested on small dataset; no comparison with other solvers
Integer Linear Programming [4]	Finds optimal solutions; supports rich constraints (pairing, size, composition); evaluated with multiple solvers	Scalability issues for large datasets; exponential growth in runtime; limited to ILP paradigm
Diversity-Aware Heuristic (Hill Climbing) [8]	Considers diversity and fairness; introduces evaluation metrics; tested on real and synthetic data; supports multiple constraints	Heuristic approach without optimality guarantee; lacks comparison with SMT and CP-based methods
Comparative Algorithm Study (Classroom-based) [14]	Multiple algorithms are evaluated in the classroom, offering practical validation, along with the differences in the quality of the solutions obtained	Theoretical analysis is limited, along with the lack of comparison between the exact methods and the constraint-based methods, and the scalability analysis was not deeply evaluated

2.1. Research gap and motivation

Despite the above-mentioned articles, there are many issues in the field of team building. Metaheuristic approaches show great scalability and flexibility; however, they do not ensure optimal solutions and, frequently, do not provide formal constraint satisfaction.

SMT guarantees strict constraint satisfaction, and so does integer programming, which also guarantees optimality but has scalability issues on large datasets. Moreover, the bulk of prior research on such issues as preference, skill, or diversity addresses certain factors alone, rather than integrating multiple factors in the context of unified criteria. Recent research has drawn attention to fairness and diversity in the form of teams, but there is a lack of standardized rubrics for evaluating and comparing team building, and so it is uncertain whether a variety of methodological methods are equally effective in different settings. Moreover, we find that there seems to be a lack of comparing fundamentally diverse solving paradigms, specifically exact methods and metaheuristic methods against common constraints and criteria. In light of these issues, this paper conducts a joint study of SMT (Z3), Constraint Programming (CP-SAT), and Genetic Algorithms for team formation in STEAM education. The presented approach assesses these techniques through common constraints which refer to skill thresholds, gender balance, and background diversity, and analyses their performance for scalability, execution time, and solution quality. The model, therefore, allows for a more pragmatic understanding of the trade-offs between optimality and scalability for suitable methods in real-world education settings.

3. Dataset and constraints

To experiment with multiple team formation approaches, a synthetic dataset [6, 7] was used, which was created to represent the characteristics of real-life students in order to simulate scenarios with a focus on STEAM-targeted activities and to represent realistic student demographics. This dataset contains representative content features in relation to gender, education status, and skill level across the main STEAM disciplines. It was rigorously designed to capture diversity; it made sure that any constraints were possible.

3.1. Dataset description

The dataset in Table 2 contains 50, 100, and 500 records of students. Each record is for a student possessing several skill attributes. The dataset can be used to construct teams that have people of different genders, backgrounds, and talents. Using this information, one can utilize exact and heuristic approaches to construct balanced teams that meet certain skill and fairness standards.

Table 2. Dataset attributes.

ID	Unique numeric identifier
Name	Student's first name (anonymized)
Gender	Female (F), Male (M)
Background	Arts, Engineering, Science
Skills	Rated from 1 (lowest) to 5 (highest): • Programming (technical ability) • Design (visual creativity) • Math (quantitative skills) • Creativity (original thinking) • Science (theoretical knowledge)

3.2. Fairness constraints

To make sure teams are fair, we defined a set of constraints that the solver must follow. These constraints help to create balanced and diverse teams.

- **Team Size:** Each team must have exactly 5 students.
- **Minimum Skill Totals:**
  - Programming ≥ 10
  - Design ≥ 10
  - Math ≥ 8

  - Creativity ≥ 6
  - Science ≥ 8

  These totals are the sum of individual student scores in each team. For each threshold, it was decided how many minimum skills should be considered in order to realistically represent the minimum competencies required from multidisciplinary teams working in STEAM. Considering the fact that the skills will be evaluated from 1 to 5 points, and each team consists of five members, the selected minimum levels of skills ensure an adequate base competence of the team as a whole while being feasible solutions.
- **Gender Balance:** Each team must include at least 2 males and at least 2 females.
- **Academic Background Diversity:** Each team must include at least one student from each of the 3 backgrounds (Arts, Engineering, Science).

These constraints help ensure a fair distribution of skills and expertise across teams, thereby supporting effective collaboration and promoting diversity in STEAM contexts.

4. Problem formulation

The team formation problem is formulated as a constrained optimization problem, where the objective is to assign students to teams under a set of predefined constraints. Given a set of $n > 0$ students, the goal is to partition them into T teams such that each team meets all requirements, such all students are assigned to exactly one team, and the teams satisfy fairness and diversity constraints. Moreover, a fairness metric is proposed to measure the balance of the teams in the distribution of total skill set. The above definition makes the assumption that the total number of students can be divided evenly among the specified size of teams, so that there are no students left out. The case where there are leftover students or different sizes of teams is to be considered in the future.

4.1. Decision variables

We define a binary decision variable:

$$x_{i,t} = \begin{cases} 1, & \text{if student } i \text{ is assigned to team } t, \\ 0, & \text{otherwise,} \end{cases}$$

where $i \in \{1, \dots, n\}$ and $t \in \{1, \dots, T\}$. This variable determines the assignment of students to teams.

4.2. Team size constraint

Each team must consist of a specific, fixed number of students, denoted by the symbol k, to ensure an even distribution among the teams:

$$\sum_{i=1}^{n} x_{i,t} = k \quad \forall t.$$

4.3. Student assignment constraint

Each student must be assigned to exactly one team:

$$\sum_{t=1}^{T} x_{i,t} = 1 \quad \forall i.$$

This guarantees that no student is assigned to multiple teams or left unassigned.

4.4. Skill constraints

Each team must meet minimum skill requirements for different skill categories such as Programming, Design, Math, Creativity, and Science. Let the binary decision variables skill_i^s represent the skill value of student i for skill s.

$$\sum_{i=1}^{n} x_{i,t} \cdot \mathrm{skill}_i^s \geq \mathrm{MinSkill}_s \quad \forall t, \forall s.$$

This ensures that each team has sufficient capability in all required skill areas.

4.5. Gender constraints

To ensure gender diversity, each team must include a minimum number of students from each gender category, where F and M represent female and male students, respectively.

$$\sum_{i \in F} x_{i,t} \geq F_{\min}, \quad \sum_{i \in M} x_{i,t} \geq M_{\min} \quad \forall t.$$

This constraint promotes balanced gender representation within each team.

4.6. Fairness gap

To evaluate the skill-based balance between teams, we define a fairness metric based on the total skill score of each team.

Let the total skill of team t be:

$$\mathrm{Score}_t = \sum_{i=1}^{n} \sum_{s} x_{i,t} \cdot \mathrm{skill}_i^s$$

The fairness gap is defined as:

$$\text{Gap} = \max_t(\text{Score}_t) - \min_t(\text{Score}_t).$$

A smaller gap means that the teams are more evenly matched, while a bigger gap means that the teams are not that balanced. This measure is used to see how well different solvers perform in assembling fair teams.

5. Experimental setup

The experimental setup, datasets, solver configurations, and evaluation process utilized to gauge the effectiveness of the suggested multi-solver strategy are all covered in this section.

5.1. Datasets

In the experiments to check for scalability and speed, three datasets of different sizes were used:

1. Dataset A of 50 students,
2. Dataset B of 100 students, and
3. Dataset C of 500 students.

Each dataset comprises records of students featuring various attributes such as skill scores (including Programming, Design, Math, Creativity, and Science), gender, and educational background, as explained in Section 3.1.

5.2. System workflow

The team formation problem from Section 4 is solved by three tools: an exact SMT solver called Z3, a constraint programming solver from OR-Tools called CP-SAT, and a heuristic optimization method known as the Genetic Algorithm. Figure 1 illustrates the whole workflow of the system. Z3, CP-SAT, and the Genetic Algorithm access the dataset after loading. Team assignments are generated individually by each solver and combined for further analysis. The team-specific information includes team members' composition, skill levels in general, gender distribution, and background diversity. In addition, the results are exported in Excel format for further analysis.

To make it possible for a fair comparison to be made, all the solvers were set up under consistent parameters and timeouts of 100 seconds. The time allocated to all the solvers was used uniformly so that an analysis could be conducted.

Z3. The SMT solver Z3 checks to see if a set of logical constraints can be met [13]. In this paper, Z3 is used to show the team formation problem as a problem of satisfying constraints. Using Boolean decision variables, all of the constraints are stored as QF_LIA logical formulas. Z3 ensures fairness if there is a satisfying assignment; otherwise, it returns UNSAT.

CP-SAT. OR-Tools' CP-SAT solver is a constraint programming solver intended for extensive combinatorial optimization problems [11]. This methodology makes use of binary decision variables to represent the assignment of students to a specific team. The constraints are formulated using linear integer equations, and the solver makes use of advanced techniques like presolve, constraint propagation, and parallel search. It is configured with multi-threading (parallel workers) and a time limit.

Genetic algorithm. The Genetic Algorithm (GA), as a heuristic optimization method, is inspired by the evolution principles of nature [16]. Solutions are encoded as chromosomes, representing team assignments, and they evolve over a series of generations by selection, crossover, and mutation operations. It is configured with population size, number of generations, and mutation/crossover probabilities.

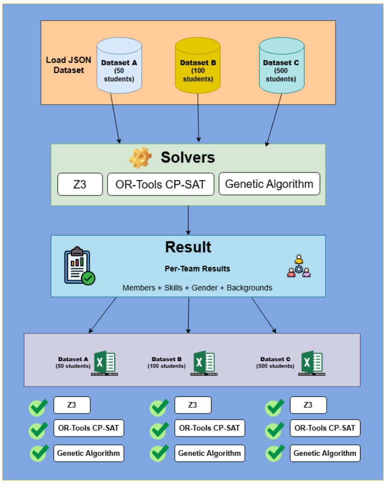

Figure 1. Integrated framework for multi-solver evaluation and comparison.

5.3. Evaluation metrics

The performance of the solvers was evaluated using the following metrics:

- **Solve Time:** Time required by the solver to find a solution.
- **Total Time:** Overall execution time including processing and extraction.
- **Fairness (MIN, MAX, GAP):** The minimum and the maximum team scores, and the difference between those two.

- **Average Skill Score:** Mean skill values across all teams.

This ensures that there is a comprehensive assessment of the efficiency of the computation and the quality of the solutions obtained using various solvers and datasets of different sizes. In Section 6, these measures are presented and analyzed on the results of the experiments. They include the fairness tests and comparisons.

6. Results

The evaluation of the experiments highlights both the operational efficiency and the fairness of the assembled teams. Extensive numerical data is available in Tables 3 and 4.

6.1. Runtime analysis

The runtime performance of the three solvers on the datasets of 50, 100, and 500 students is shown in Table 3, respectively. All solvers showed quick execution times on the smallest dataset of 50 students. Z3 took the lead, followed by CP-SAT and, finally, the Genetic Algorithm. Z3 and CP-SAT maintained their efficiency with a low solve time for the dataset of 100 students. Due to iterative optimization strategy, the Genetic Algorithm's runtime significantly increased by two orders of magnitude. Scalability issues were more noticeable with the largest dataset of 500 students. While Z3 demonstrated strong scalability, CP-SAT and the Genetic Algorithm had long solve time. Overall, Z3 consistently produced the fastest solve times for datasets of all sizes; however, as datasets grew in size, its overall execution time considerably deteriorated. In contrast, the Genetic Algorithm resulted in greater computational costs, due to a methodology in which multiple potential solutions are evaluated across several generations, providing the freedom to explore the solution space but requiring the longest time to compute. CP-SAT demonstrated more consistent and scalable performance in terms of overall runtime.

Table 3. Detailed solver runtime results for synthetic datasets of different sizes.

Size	Solver	Status	Solve (s)	Total (s)
50	Z3	Solved	0.04	1.54
50	CP-SAT	Solved	0.09	0.12
50	Genetic Algorithm	Solved	0.12	0.12
100	Z3	Solved	0.11	5.06
100	CP-SAT	Solved	0.28	0.42
100	Genetic Algorithm	Solved	12.62	12.62
500	Z3	Solved	2.99	123.23
500	CP-SAT	Solved	24.86	27.92
500	Genetic Algorithm	Solved	59.50	59.50

6.2. Fairness analysis

The evaluation of fairness using MIN, MAX, and GAP metrics across different datasets and solvers is shown in Table 4. All solvers produced comparatively equal teams on the dataset of 50 students. The Genetic Algorithm showed a somewhat higher GAP of 9, indicating a modest decline in balance, while Z3 and CP-SAT both recorded a GAP of 7. GAP increased for all solvers on the dataset of 100 students. While Z3 (18) and the Genetic Algorithm (19) produced similar results in terms of the GAP, the team balance was declining in the case of CP-SAT (22). With a GAP of 11, Z3 produced the best fairness results on the largest dataset of 500 students. CP-SAT came in second with a GAP of 12 and the Genetic Algorithm had a GAP of 14. In conclusion, Z3 consistently produced more balanced teams for datasets of all sizes. The reason for this is that Z3 strictly adheres to the constraints by ensuring all conditions are met without trade-offs. On the other hand, the Genetic Algorithm provided competitive fairness but showed less overall stability, because it is a heuristic algorithm and does not ensure an optimal solution. CP-SAT showed somewhat higher variability in the fairness outcomes.

Table 4. Minimum and maximum total STEAM skill scores and fairness gap across datasets.

Dataset Size	Solver	MIN	MAX	GAP
50	Z3	76	83	7
50	CP-SAT	75	82	7
50	Genetic Algorithm	74	83	9
100	Z3	72	90	18
100	CP-SAT	69	91	22
100	Genetic Algorithm	71	90	19
500	Z3	74	85	11
500	CP-SAT	75	87	12
500	Genetic Algorithm	70	84	14

6.3. Scalability considerations

This difference in scalability became more apparent as the size of the datasets increased. Even with a dataset size of 500 students, Z3 exhibited significant scalability. This further suggests that the optimizations made in the solver and the way the constraints are being implemented are well-suited for the application. It is vital to note that solvers such as Z3 may experience scalability issues with increased problem size due to the exponential size of the search space. On the other hand, the Genetic Algorithm presented itself as a viable solution for large-scale problems where an approximate solution is acceptable, despite the lower operational speed associated with increased dataset sizes. CP-SAT also presented itself as a viable solution through the significant scalability and solution quality. Memory consumption was not explicitly measured in this study. A detailed comparison of memory requirements among SMT, CP-SAT, and Genetic Algorithm approaches remains an interesting direction for future research.

6.4. Practical implications

The selection of the solver is dependent on the particular needs of the application. Z3 is the best choice in situations with is a need for stringent fairness and adherence to set limits. In situations where data is large in volume and resources limited in processing capacity, the Genetic Algorithm offers an alternative. At the same time, CP-SAT strikes an optimal balance between efficiency and scalability. The data shows that no particular solver can be considered better than another. Each method has unique advantages depending on the size of the problem and solution accuracy desired.

7. Conclusion and future work

In summary, the research presented in the paper is based on the multi-solution approach, exact (Z3, CP-SAT) and heuristic (Genetic Algorithm), for the team formation problem. The problem has been formulated as a constraint satisfaction problem with the set of criteria, such as team size, skills, gender, and diversity in terms of academic background. The experimental results for the exact approaches show promising results with the balance between the quality of the solutions and the efficiency of the computation compared to the heuristic approach. Z3 demonstrated promising results with the ability to achieve the best solution with the shortest solve time, with low GAP values. On the contrary, CP-SAT demonstrated the ability to achieve the balance between fairness and the efficiency of computation. The Genetic Algorithm approach also demonstrated the ability to achieve the balance, but with higher computation time. The research also demonstrated the need for choosing the most suitable approach based on the scale of the problem to solve.

There exist several possible directions for further research. The most promising future direction is the development of a hybrid framework that combines exact optimization methods with heuristic search techniques, aiming to achieve both high-quality solutions and strong scalability for large-scale team formation problems. The application of the real-life characteristics, such as the preferences of students, the personality, or the changing team, will also enhance the practical application of the model.

Data availability

The datasets generated and used during this study are available from the corresponding author, Ali Adil Ali (ali.adil@inf.unideb.hu), upon reasonable request.

The synthetic datasets used in this study are publicly available at `https://github.com/AliProgrammer85/Three-Solver-Team-Formation`. The repository contains the datasets with 50, 100, and 500 students that were used in the experimental evaluation of the proposed team formation approaches.

References

[1] N. O. Abdulwahid, S. Fakhfakh, I. Amous: *Simulating and Predicting Students' Academic Performance Using a New Approach based on STEAM Education*, JUCS - Journal of Universal Computer Science 28.12 (2022), pp. 1252–1281, issn: 0948-695X, doi: 10.3897/jucs.86340.

[2] A. A. Adil, G. Kovásznai, M. Khalid: *Automated fair team formation in STEAM activities using Satisfiability Modulo Theories (SMT)*, Annales Mathematicae et Informaticae 61 (2025), pp. 1–14, doi: 10.33039/ami.2025.10.001, url: https://ami.uni-eszterhazy.hu.

[3] G. Candel, V. Sánchez-Anguix, J. M. Alberola, V. Julián, V. Botti: *An evolutionary metaheuristic for forming teams in the classroom with constraints*, Neurocomputing 638 (2025), p. 130138, issn: 0925-2312, doi: 10.1016/j.neucom.2025.130138, url: https://www.sciencedirect.com/science/article/pii/S0925231225008100.

[4] G. Candel, V. Sánchez-Anguix, J. M. Alberola, V. Julián, V. Botti: *An Integer Linear Programming Model for Team Formation in the Classroom with Constraints*, in: Hybrid Artificial Intelligent Systems (HAIS), vol. 14001, Lecture Notes in Artificial Intelligence (LNAI), Springer, 2023, pp. 397–408, doi: 10.1007/978-3-031-40725-3_34.

[5] Coding for Kids Iraq: *Coding for Kids Iraq Initiative*, Accessed: 2026-03-20, 2024, url: https://www.coding4kidsiraq.com/.

[6] M. Dorodchi, E. Al-Hossami, A. Benedict, E. Demeter: *Using Synthetic Data Generators to Promote Open Science in Higher Education Learning Analytics*, in: 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 4672–4675, doi: 10.1109/BigData47090.2019.9006475.

[7] B. Flanagan, R. Majumdar, H. Ogata: *Fine Grain Synthetic Educational Data: Challenges and Limitations of Collaborative Learning Analytics*, IEEE Access 10 (2022), pp. 26230–26241, doi: 10.1109/ACCESS.2022.3156073.

[8] B. Hui, O. Adeyemi, K. Phan, J. Schoenit, S. Akins, K. Khademi: *Diversity Considerations in Team Formation Design, Algorithm, and Measurement*, in: Proceedings of the 15th International Learning Analytics and Knowledge Conference (LAK), ACM, 2025, pp. 1–11, doi: 10.1145/3706468.3706473.

[9] A. Jenkins, E. Rainbow, I. Jaimoukha, F. S. Ng, L. Stankovic, M. Dakovič, D. Mandic, D. Mandic: *Fair and Skill-Diverse Student Group Formation: A graph-theoretic approach*, IEEE Signal Processing Magazine 43.1 (2026), Cited by: 1, pp. 83–97, doi: 10.1109/MSP.2025.3627096, url: https://www.scopus.com/inward/record.uri?eid=2-s2.0-105030855588&doi=10.1109%2fMSP.2025.3627096&partnerID=40&md5=fc13d3c31d02887705b7673b6dc40319.

[10] K. Krasnić, P. Žigman, T. Babić: *Team Building Elements and Their Importance According to STEAM Students*, in: 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), 2022, pp. 739–744, doi: 10.23919/MIPRO55190.2022.9803519.

[11] B. Lennartson: *Optimization of Timed Petri Nets using CP-SAT*, IFAC-PapersOnLine 58.1 (2024), 17th IFAC Workshop on discrete Event Systems WODES 2024, pp. 90–95, issn: 2405-8963, doi: 10.1016/j.ifacol.2024.07.016, url: https://www.sciencedirect.com/science/article/pii/S2405896324000946.

[12] J. Moreno, D. A. Ovalle, R. M. Vicari: *A genetic algorithm approach for group formation in collaborative learning considering multiple student characteristics*, Computers & Education 58.1 (2012), pp. 560–569, issn: 0360-1315, doi: 10.1016/j.compedu.2011.09.011, url: https://www.sciencedirect.com/science/article/pii/S0360131511002284.

[13] L. de Moura, N. Bjørner: *Z3: An Efficient SMT Solver*, in: Tools and Algorithms for the Construction and Analysis of Systems, ed. by C. R. Ramakrishnan, J. Rehof, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 337–340.

[14] S. Peuker, A. Hill: *A Novel Approach to Purposeful Team Formation*, in: 2024 ASEE Annual Conference and Exposition, 2024, doi: 10.18260/1-2--46466.

[15] U.S. Department of Labor: *STEM Day: Explore Growing Careers*, Accessed: 2026-03-20, 2022, url: https://blog.dol.gov/2022/11/04/stem-day-explore-growing-careers.

[16] C. Zhamri, Z. Che Ani, A. Yasin, M. Z. Husin, Z. Hamid: *A Method for Group Formation Using Genetic Algorithm*, International Journal on Computer Science and Engineering 02 (2010), pp. 3060–3064.

A. Algorithmic details

Z3 solver

When using the Z3 solver, the team formation problem is modelled as a set of logical constraints and searches for a solution that satisfies all requirements exactly. It guarantees correctness if a solution exists.

Algorithm 1 Z3-based team formation

Require: Students S, Requirements R
Ensure: Team assignments or UNSAT

Initialize solver
Define binary variables $x_{i,t}$
Add constraint: each student assigned to one team
Add constraint: each team has fixed size
for each team t **do**
    Apply skill constraints
    Apply gender constraints
    Apply background constraints
end for
Run solver
if solution exists **then**
    Extract team assignments
else
    Return UNSAT or TIMEOUT
end if

OR-Tools' CP-SAT solver

The CP-SAT solver formulates the problem using integer constraints and applies efficient search and propagation techniques to find feasible solutions with good scalability.

Algorithm 2 OR-Tools CP-SAT team formation

Require: Students S, Requirements R
Ensure: Team assignments or UNSAT
Initialize CP-SAT model
Define binary variables $x_{i,t}$
Add constraint: each student assigned to one team
Add constraint: each team has fixed size
for each team t **do**
    Apply skill constraints
    Apply gender constraints
    Apply background constraints
end for
Solve model with time limit
if solution found **then**
    Extract team assignments
else
    Return UNSAT or TIMEOUT
end if

Genetic algorithm

The Genetic Algorithm is a heuristic approach that iteratively improves solutions using evolutionary operators such as selection, crossover, and mutation. It is suitable for large-scale problems where approximate solutions are acceptable.

Algorithm 3 Genetic algorithm for team formation

Require: Students S, Requirements R
Ensure: Best team assignment
Initialize population P
Evaluate fitness of each solution
while not termination condition **do**
    Select parents
    Apply crossover
    Apply mutation
    Evaluate new solutions
    Update population
end while
Return best solution found

B. Example team formation

This section presents typical team configurations generated by the three solvers (Z3, CP-SAT, and GA) using a dataset of 50 students in order to illustrate the utility of the proposed framework.

Each constructed team meets the established criteria and limitations. Each team has five students representing the different areas of STEAM. All the teams are composed of students of diverse backgrounds and genders. In the results, it is indicated that all the strategies can yield feasible solutions. However, there is a difference in the distribution of the overall skill scores of the teams. Table 5, Table 6, and Table 7 show the exemplary team assignments generated by each solver.

Table 5. A fragment of the team formation generated by Z3 (for 50 students).

Team	Members	F	M	Total Skills
Team1	Kara, Mia, Zane, Max, Nora	3	2	76
Team2	Jack, Quinn, Ryan, Uma, Yusuf	2	3	79
Team3	Grace, Victor, Wendy, Jake, Umar	2	3	81
Team4	Ivy, Noah, Xavier, Elena, Xena	3	2	81
Team5	Yara, Gina, Iris, Quincy, Sam	3	2	78

Table 6. A fragment of the team formation generated by CP-SAT (for 50 students).

Team	Members	F	M	Total Skills
Team1	Eve, Ivy, Owen, Tara, Yusuf	3	2	80
Team2	Jack, Noah, Gina, Iris, Quincy	2	3	79
Team3	Bob, Paul, Ryan, Elena, Luna	2	3	81
Team4	Alice, Charlie, Grace, Finn, Vera	3	2	79
Team5	Diana, Olivia, Yara, Zane, Max	3	2	79

Table 7. A fragment of the team formation generated by GA (for 50 students).

Team	Members	F	M	Total Skills
Team1	Tara, Riley, Ryan, Ben, Sam	2	3	82
Team2	Vera, Olivia, Max, Hank, Tom	2	3	77
Team3	Clara, Will, Henry, Nora, Jack	2	3	79
Team4	Xavier, Quincy, Grace, Noah, Uma	2	3	83
Team5	Xena, Elena, Piper, Frank, Jake	3	2	79

The results show that the exact (Z3 and CP-SAT) and heuristics solvers (GA) are able to satisfy the necessary constraints, such as the balance of genders and diversity in terms of academics; however, the distribution of the skills among the teams is not uniform, with more consistency observed in the results of Z3 and OR-Tools than in the genetic algorithm.