

An Approach for Team Formation Using Modified Grey Wolf Optimizer

Sandip T. Shingade and Rajdeep Niyogi(✉)

Indian Institute of Technology Roorkee, Roorkee 247667, India
{stukaram,rajdeep.niyogi}@cs.iitr.ac.in

Abstract. The team formation problem is to find a subset of agents, referred to as a team, from a given set of agents such that the team satisfies some desirable property. The computational complexity of this problem is known to be NP-hard. In this paper, we suggest a modified grey wolf optimization approach to find a team with minimum communication cost. The proposed algorithm is evaluated with two real-world data sets, namely the ACM data set and the Academia Stack exchange data set. The experimental results show that the algorithm outperforms some existing approaches with respect to some parameters.

Keywords: Team formation · GWO · JAYA · IGWO

1 Introduction

Team formation is a vital task in many applications, such as hiring experts in the industry, working on project groups, and product development. Team formation is required in mission-oriented applications. For instance, in order to rescue people affected by an accident, a rescue team would typically consist of different types of agents, with distinct skills, such as policemen, ambulances and doctors, and cranes. The team should perform the rescue act and medical treatment with efficient collaboration using their expertise and within a specific deadline as per the severity and emergency of the accident. The team members should communicate among themselves and a team with minimum communication cost is desirable.

Given a set of agents, description of agents and tasks, the computational complexity to find a subset of the agents (team) with minimal cost for a task is known to be NP-hard. In this paper, we suggest a metaheuristic approach to finding a team with minimal communication cost. The approach is a modified grey wolf optimization technique.

The rest of the paper is organized as follows. In Sect. 2, we describe related work for team formation problems. In Sect. 3 we provide some motivations for using metaheuristic-based approaches. Some preliminary notions are given in Sect. 4. The proposed approach is discussed in Sect. 5. Experimental results are given in Sect. 6. Conclusions are drawn in Sect. 7.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
O. Gervasi et al. (Eds.): ICCSA 2023, LNCS 13956, pp. 440–452, 2023.
https://doi.org/10.1007/978-3-031-36805-9_29

2 Related Work

An effective team with a distance diameter and minimum spanning tree as a communication framework is suggested in [1]. In [6,7] the authors identify the distance for each team member with and without a team leader effectively and collaboratively in a social network. The expert team formation where the agents are connected using a social network is studied in [8,9].

A new approach based on a genetic algorithm for finding multiple teams to fulfill the requirements of multiple projects with socially connected people is proposed in [2]. A sociometric matrix has been used for calculating the positive relationship between people. In [11], the authors suggest a genetic algorithm-based approach for group formation in collaborative learning, considering multiple student characteristics to achieve groups. They experimented with college students considering three characteristics: an estimate of student knowledge levels, student communicative skills, and student leadership skills. The results obtained show that groups formed with the proposed method produced better outcomes than those formed with traditional methods like random assignment and self-organization.

In [5], the authors utilize the particle swarm optimization (PSO) algorithm for solving a team formation problem by minimizing the communication cost among the agents. The authors applied a new swap operator within PSO to ensure the consistency of the capabilities and the skills to perform the required project. They also applied the algorithm to different experiments with standard PSO by considering different numbers of agents and skills.

In [3], the authors proposed the Improved African Buffalo Optimization algorithm (IABO) where a new concept of swap sequence was applied to improve the performance by generating better team members that cover all the required skills. An improved JAYA algorithm with a modified swap operator is introduced by in [4]. They also compared the results with GA and PSO, ABO, and standard JAYA improved JAYA algorithm with a modified swap operator. All these papers apply modified swap operations and compare them with standard heuristic team formation algorithms.

In [13], the authors focus on a simple and new JAYA algorithm for solving constrained and unconstrained optimization problems. This algorithm is based on the concept that the solution obtained for a given problem should move toward the best solution and avoid the worst solution. It requires only the standard control parameters and does not require any algorithm-specific control parameters [12]. A self-adaptive multi-population-based JAYA (SAMP-JAYA) algorithm for solving the constrained and unconstrained numerical and engineering optimization problems is proposed by [13]. The search mechanism of the JAYA algorithm is upgraded by using the multi-population search scheme. The algorithm is compared and evaluated with the other latest algorithms, and JAYA showed better performance.

Most of works used only one heuristic approach with methodologies such as MSO or GA. If the algorithm changes the parameters and applies an objective function, it always gives better results. In [14], the authors reduced the data by

applying state space reduction techniques for horizontal and vertical reduction. Further improvements on the earlier heuristic approaches based on MSO, GA can be obtained using GWO (Grey Wolf Optimizer) [10].

3 Metaheuristic-Based Approach for Team Formation

The motivation for using a metaheuristic-based approach is due to the following reasons: The metaheuristic-based approach is a simple evolutionary notion of studying animal behaviour. It appears like a black box testing only focusing on final optimized results. These techniques also provide a random solution without calculating derivation search space to find an optimal solution. This first derivative decides the direction, and the second derivative minimizes or maximizes optimization results. It also works on avoiding stagnation for a local solution and extensively searches for the entire solution; however, there is no metaheuristic approach to best solve all optimization problems [10].

3.1 Grey Wolf Optimization

A metaheuristic optimization technique called GWO [10] was developed as a result of studying how grey wolves hunt in the wild. GWO is a population-based method that begins with an initial population of wolves, which stand in for potential solutions to an optimization problem. There are four groups of wolves in the population: alpha, beta, delta, and omega. In a pack, the alpha wolf is the best one and the omega wolf is the worst. In order for the algorithm to function, the population's distribution of wolves is updated after each iteration. Based on the locations of the alpha, beta, and delta wolves, each wolf's position is updated.

3.2 JAYA Algorithm

A population-based metaheuristic approach called JAYA algorithm was sugggested in [12]. Since it does not require parameters to be set, unlike other metaheuristic algorithms, it is a simple approach that was created primarily for continuous optimization problems. The algorithm was developed with the intention of handling both constrained and unconstrained optimization functions. This algorithm blends features of evolutionary algorithms with swarm-based intelligence. The search process looks for the best solutions available in an effort to increase the likelihood of success and looks to prevent failure by avoiding the worst options. This method has several advantages over other population-based algorithms, including being straightforward to use and not requiring any algorithm-specific settings (i.e., the population size, and a maximum number of iterations). The objective function of JAYA algorithm is given below.

$$X_{j,k}^{t+1} = X_{j,k}^{t} + r_{1,j}^{t}[(X_{j,\text{best}}^{t}) - |(X_{j,k}^{t})|] - r_{2,j}^{t}[(X_{j,\text{worst}}^{t}) - |(X_{j,k}^{t})|]$$

Fig. 1. JAYA objective function.

4 Preliminaries

4.1 Team Formation with Minimum Communication Cost

The Team Formation (TF) problem is a combinatorial optimization problem that involves grouping individuals into teams based on their skills, preferences, and other characteristics. The goal is to form teams that are both effective and efficient, and meet specific constraints or objectives.

Definition 1 (Team Formation). *Given: a set of agents, skills of agents, and tasks. An agent has a set of skills (abilities). A task is defined by a set of skills. Team formation problem is to find a subset of agents whose combined skills (union of the skills of the individual agents) is a superset of the skills required for the task.*

Definition 2 (Communication Graph). *A communication graph* $G = (V, E)$ *is an undirected weighted graph where the set of vertices are the agents and an edge* (u, v) *is given a weight* w, $0 \leq w \leq 1$, *where low weight means that the agents can communicate easily.*

Definition 3 (Communication Cost (CC) of two agents). *Let the agents* A_i, A_j *have skills* S_i, S_j *respectively. The communication cost of the agents is given as:*

$$CC_{i,j} = 1 - |(S_i \cap S_j)|/|(S_i \cup S_j)| \tag{1}$$

Figure 2 shows the communication cost among the agents. The skills of the agents are given in Table 1.

Table 1. Skills of the agents

Agent	Agent Skills		
A_1	agent computing	distributed data mining	intrusion detection
A_2	speech acts	agent computing	-
A_3	security	social dynamics	multi agent systems
A_4	agent computing	model checking	verification
A_5	machine learning	multi agent systems	advice exchange

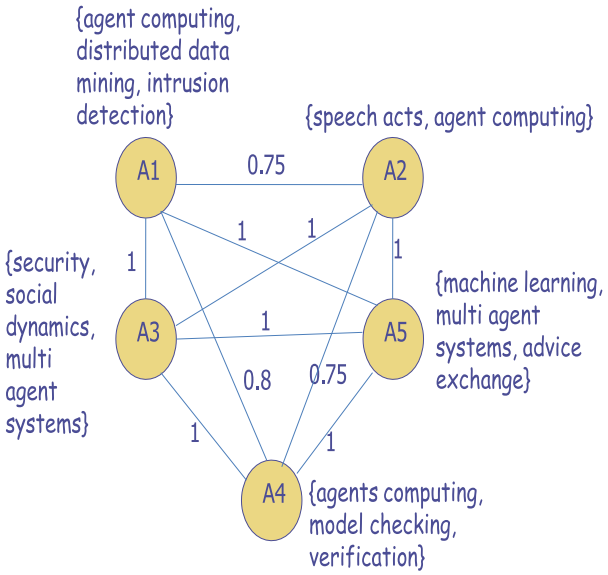

Fig. 2. ACM database example of team formation in social networks.

Definition 4 (Total Communication Cost of a team (TC)). *Let a team comprise of the agents* $A_1, \ldots, A_m$, $m > 1$. *Let the communication graph for the team be* $G = (V, E)$, *where* $V = \{A_1, \ldots, A_m\}$, *and* $CC_{i,j}$ *is the weight of the edge* (A_i, A_j). *Then TC is the sum of the weights of all the edges in G.*

A task requires the following skills: {*agent computing, multi-agent systems, model checking, machine learning* }. Some of the possible teams are: $T_1 = \{A_1, A_3, A_4, A_5\}$. The communication cost of the team is the sum of the weights of the edges connecting the team members. The edges are: (A_1, A_3), (A_1, A_4), (A_1, A_5),(A_3, A_4), (A_3, A_5), and (A_4, A_5). The weights of the edges are 1, 0.8, 1, 1, 1, and 1 respectively. Therefore the communication cost of the team is equal to 5.8. Similarly, for $T_2 = \{A_2, A_3, A_4, A_5\}$, $T_3 = \{A_1, A_4, A_5\}$, $T_4 = \{A_4, A_5\}$, the communication costs are 5.75, 2.8, and 1.0 respectively.

The goal of the proposed metaheuristic algorithm is to find the team with minimum communication cost. For the above example, the team T_4 is thus selected.

5 Proposed Improved Grey Wolf Optimization (IGWO) Algorithm

The pseudocode of our metaheuristic approach based on GWO that we refer to as IGWO, Improved Grey wolf optimization is shown in Fig. 3 and the flowchart is given in Fig. 4. We now give a detailed illustration of our algorithm using an example scenario taken from the Stack exchange dataset.

Algorithm: Improved Grey Wolf Optimization Algorithm (IGWO)

Input: P- Number of wolves in the population, MaxIter-Maximum number of iterations, S- Number of skills
Output: The best solution found (alpha)

1. Initialize the positions of alpha, beta, and delta wolves randomly within the search space
2. Initialize the iteration counter c := 0
3. for (i = 1; i <= S; i++)
4. for (j = 1; j <= P; j++)
5. Generate the initial population randomly x_{ij}^{c}
6. end for
7. end for
8. Evaluate the fitness function for the population $f(x)^{c}$ of bestAlpha solution
9. do
10. for (i = 1; i <= S; i++)
11. for (j = 1; j <= P; j++)
12. Apply crossover on the best solution $x_{alp\,ha}^{c}$ and current solution x_{ck}^{c}
13. Select the best offspring solution x_{best}^{c} and update the solution
14. if $f(x_{ij}^{c+1})$ <= $f(x_{alp\,ha}^{c})$ then
15. $x_{alp\,ha}^{c} = x_{ij}^{c+1}$
16. else
17. $x_{alp\,ha}^{c} = x_{alp\,ha}^{c}$
18. end if
19. end for
20. end for
21. Increment the iteration counter $c = c + 1$
22. Update the position of search agents alpha, beta and gama to determine optimized best alpha solution
23. While (c < MaxIter)
24. Get the best solution alpha

Fig. 3. Algorithm: Improved Grey Wolf Optimization Algorithm (IGWO).

Initialization step:

- We take an initial population of size 100 and the maximum number of iterations is 25.
- Each agent's id and its skills are represented as key : [value] pair. For example, 160429 : [writing, writing-style, peer-review] where 160429 is the agent id and [writing, writing-style, peer-review] are the skills.
- We consider a task and the number of skills required to fulfill the task. We identify the agents for each skill. For example, the four skills that are required to complete a task are **[computer science, graduate-admissions, graduate-school, research process]**. The agents for each skill are: computer science: [14842, 149199, 139056, 173659,.....] graduate-admissions: [11702, 16206, 14251, 14844,.....] graduate-school: [16766, 21628,

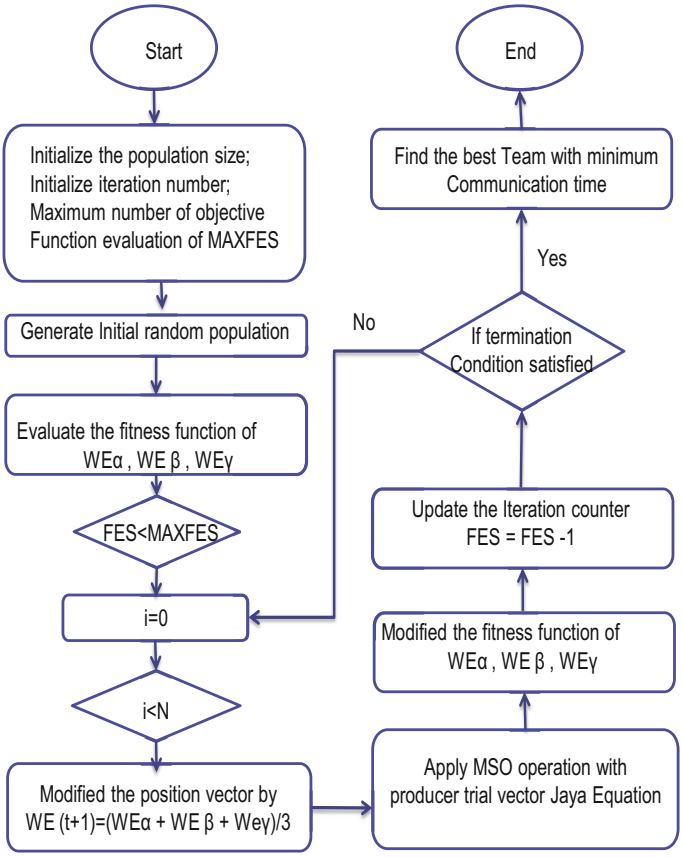

Fig. 4. Flow chart of IGWO algorithm for minimization communication time.

151956, 11849, 25785,....] research-process: [184, 534, 43201, 48875, 11498, 16270,...]

Evaluation step:

- Generate a random population of the given list of agents as shown in Table 2. We consider a population size of 100 and each solution (team) has four agents (team size).
- Find the communication cost between any two agents according to Definition 4. Determine the communication cost of each team according to Definition 4, shown in Table 3. The team for population number 83 has minimum communication cost 0.63.

- Determine the best search agent (alpha), second best agent (beta) and third best agent (delta). The alpha solution is **0.66** at position 61. The beta solution

Table 2. Random population/teams of the given list of agents

Population No.	Solution			
1	161227	32336	161676	13455
2	145368	13491	14515	54913
3	36965	17778	161676	186739
–	–	–	–	–
–	–	–	–	–
99	173150	77220	38259	170157
100	37428	154770	58514	5580

Table 3. Total communication cost of a team

Population No.	Solution				TC
1	107149	140092	169846	40326	0.95
2	12076	44002	127021	34316	0.98
3	61014	173938	9359	180514	1.03
–	–	–	–	–	
–	–	–	–	–	
83	47634	42379	123167	11264	**0.63**
–	–	–	–	–	
–	–	–	–	–	
99	156469	184696	155922	86273	1.06
100	162140	88080	6054	76265	1.04

is **0.73** at position 45. The delta solution is **0.74** at position 48. This is obtained from Table 3.

- Apply a swap operation that works as follows. Consider the jth column of any two populations numbered, say P_1 and P_2; $swap(P_1, P_2, j)$ means the agents at position j corresponding to these populations are interchanged. Accordingly the total communication cost of the teams would change.
- Apply the objective function of JAYA algorithm (Fig. 1).
- Update the positions of each search agent. The Alpha solution Instance value is **0.63** at position 83. The Beta solution Instance value is **0.66** at position 66. The Delta solution Instance value is **0.69** at position 92.

Output step:

- The team obtained for the given task is 47634, 42379, 123167, 11264 with a final cost of **0.63**.

6 Experimental Results

For the implementations we used Python with Jypytor notebook with i7 core CPU having 16 GB RAM. We have compared the performance of IGWO with other algorithms JAYA, GWO, and modified JAYA. We have used two real-world data sets namely Academia stack exchange data set and ACM data set for the evaluations.

ACM Dataset. To evaluate our technique, we used the real-world ACM dataset. We collect data on author collaboration based on the agent's research expertise and construct a table for this data, that has 3856 lines of data.

Stack Exchange Dataset. To evaluate our approach, we employed another real-world Academia stack exchange dataset. We collect data with author cooperation skills in this data produced table that has 5730 lines of data, and if another author cites with his article, we state that they will connect with data. We also consider it related if any author shares their content with another author. We will look at the most crucial abilities, such as computer science, graduate entrance, graduate school, research procedure, conferences, and journal articles.

6.1 Parameter Setting

For all experiments we consider a population size of 100. We do five trials in succession with a variety of skills, 2 with 5 iterations, 3 with 10 iterations, 4 with 15 iterations, 5 with 20 iterations, and 6 with 25 iterations.

Table 4. Communication cost for ACM dataset.

Exp No	No of skill	Iterations	JAYA	IJMSO	GWO	IGWO
1	2	5	0.02	0.02	0.02	**0.02**
2	3	10	0.08	0.07	0.07	**0.06**
3	4	15	0.17	0.16	0.22	**0.18**
4	5	20	0.19	0.21	0.23	**0.18**
5	6	25	0.43	0.42	0.41	**0.33**

Table 5. Communication cost for academia stackexchange dataset.

Exp No	No of skill	Iterations	JAYA	IJMSO	GWO	**IGWO**
1	2	5	0.1	0.1	0.1	**0.1**
2	3	10	0.32	0.32	0.32	**0.3**
3	4	15	0.74	0.7	0.66	**0.66**
4	5	20	1.16	1.2	1.24	**1.13**
5	6	25	1.64	1.7	1.87	**1.55**

In all the simulations, IGWO algorithm outperformed JAYA, IJMSO, and GWO algorithms on both the datasets in terms of communication cost, as demonstrated in Tables 4 and 5. The IGWO technique yielded communication cost parameters of 0.02 and 0.1 for experiment 1, 0.06 and 0.3 for experiment 2, 0.18 and 0.66 for experiment 3, 0.18 and 1.13 for experiment 4, and 0.33 and 1.55 for experiment 5. For the other three techniques, the communication cost parameter values ranged from 0.02 to 0.42 for the ACM dataset and from 0.1 to 1.55 for academia stackexchange dataset.

Table 6. Communication cost for ACM dataset (Sample Mean).

Exp No	No of skill	Iterations	JAYA	IJMSO	GWO	IGWO
1	2	5	0.0608	0.0656	0.0644	**0.0583**
2	3	10	0.1732	0.1755	0.1682	**0.1841**
3	4	15	0.3657	0.3913	0.4028	**0.3816**
4	5	20	0.6484	0.6554	0.668	**0.6683**
5	6	25	1.1113	1.1114	1.1166	**1.196**

Table 7. Communication cost for academia stackexchange dataset (Sample Mean).

Exp No	No of skill	Iterations	JAYA	IJMSO	GWO	**IGWO**
1	2	5	0.1493	0.1583	0.1521	**0.1506**
2	3	10	0.4945	0.5006	0.4891	**0.5106**
3	4	15	1.0252	0.1074	1.0267	**0.9961**
4	5	20	1.6476	1.6576	1.6394	**1.6393**
5	6	25	2.4332	2.4442	2.4501	**2.4339**

In all simulations utilising Sample Mean, the IGWO algorithm technique outperformed the other three algorithms (JAYA, IJMSO, and GWO) in terms of the communication cost parameter, as demonstrated in Table 6 and 7. The communication cost parameter of the IGWO algorithm was 0.0583 and 0.1506 in experiment 1, 0.1841 and 0.5106 in experiment 2, 0.3816 and 0.9961 in experiment 3, 0.6683 and 1.6393 in experiment 4, and 1.196 and 2.4339 in experiment 5. During the same experiments, the communication cost parameter values for the other three algorithms ranged from 0.0608 to 1.1116 for the ACM dataset, and from 0.1493 to 2.4501 for the academia stackexchange dataset.

In all simulations utilising Standard deviatiaon, the IGWO algorithm technique outperformed the other three algorithms (JAYA, IJMSO, and GWO) in terms of the communication cost parameter, as demonstrated in Table 8 and Table 9. The communication cost parameter of the IGWO algorithm was 0.0318 and 0.045 in experiment 1, 0.0823 and 0.1199 in experiment 2, 0.1564 and 0.1744 in experiment 3, 0.2364 and 0.2073 in experiment 4, and 0.4364 and 0.2863 in

Table 8. Communication cost for ACM dataset (Std. Deviation).

Exp. No.	No. of skill	Iterations	JAYA	IJMSO	GWO	IGWO
1	2	5	0.0361	0.0328	0.0363	**0.0318**
2	3	10	0.085	0.0813	0.0671	**0.0823**
3	4	15	0.1151	0.1395	0.1324	**0.1564**
4	5	20	0.2134	0.2234	0.2074	**0.2364**
5	6	25	0.415	0.455	0.3751	**0.4364**

Table 9. Communication cost for academia stackexchange dataset (Std. Deviation).

Exp. No.	No. of skill	Iterations	JAYA	IJMSO	GWO	**IGWO**
1	2	5	0.0468	0.0534	0.0437	**0.045**
2	3	10	0.1124	0.1155	0.1081	**0.1199**
3	4	15	0.1737	0.1854	0.1796	**0.1744**
4	5	20	0.2295	0.2195	0.2244	**0.2073**
5	6	25	0.2847	0.2888	0.3142	**0.2863**

experiment 5. During the same experiments, the communication cost parameter values for the other three algorithms ranged from 0.0328 to 0.0328 for the ACM dataset, and from 0.0468 to 0.3142 for the academia stackexchange dataset.

Table 10. Computing time for ACM dataset.

Exp. No.	No. of skill	Iterations	JAYA	IJMSO	GWO	**IGWO**
1	2	5	0.02332	0.076825	0.010185	**3.6970048**
2	3	10	0.050389	0.355808	0.022542	**7.3005062**
3	4	15	0.096243	0.657781	0.061759	**10.7778158**
4	5	20	0.296295	0.306295	0.189573	**14.8616639**
5	6	25	2.266233	2.356233	1.461107	**21.3694483**

According to the experimental table data, IGWO algorithm required more computation time than the other three algorithms (JAYA, IJMSO, and GWO) for all tests in both the datasets, as demonstrated in Tables 10 and 11. IGWO algorithm required time around 3.697 to 21.369 s for ACM dataset and 3.737 to 27.976 s for academia stackexchange dataset. The computation time for the other three algorithms ranged from 0.023 to 1.461 s for ACM dataset and from 0.029 to 8.6 s for academia stackexchange dataset. IGWO is more complex than the other three algorithms and so it performs more computations to arrive at an optimal solution that is better than that obtained by the other three algorithms. For example, when the communication cost of IGWO is 0.33, the minimum cost

Table 11. Computing time for academia stackexchange dataset

Exp No	No of skill	Iterations	JAYA	IJMSO	GWO	**IGWO**
1	2	5	0.028943	0.071224	0.107659	**3.7368715**
2	3	10	0.037418	0.240848	0.010023	**7.3201007**
3	4	15	0.061079	0.461732	0.019232	**10.8939794**
4	5	20	0.221946	0.251246	0.11478	**14.6149771**
5	6	25	11.08188	12.13188	8.600857	**27.9755678**

obtained by the other algorithms is 0.41 (Table 4); similarly when when the communication cost of IGWO is 1.55, the minimum cost obtained by the other algorithms is 1.64 (Table 5).

7 Conclusion

In this paper, we proposed an improved grey wolf optimization technique to find a team with minimal communication cost. For the evaluation of the algorithm, we used two real-world data sets (Stack exchange data set, ACM data set). The output of our suggested algorithm shows a significant improvement in communication cost compared to state-of-the-art algorithms. As part of future work, we would like to develop metaheuristic approaches for team formation with multiple objectives.

Acknowledgement. The second author was in part supported by a research grant from Google.

References

1. Anagnostopoulos, A., Becchetti L., Castillo, C., Gionis, A., Leonardi, S.: Online team formation in social networks. In: Proceedings of the 21st international conference on World Wide Web, pp. 839–848 (2012)
2. Baghel, V.S., Bhavanim S.D.: Multiple team formation using an evolutionary approach. In: 2018 Eleventh International Conference on Contemporary Computing (IC3), pp. 1–6. IEEE (2018)
3. El-Ashmawi, W.H.: An improved african buffalo optimization algorithm for collaborative team formation in social network. Int. J. Inf. Technol. Comput. Sci. **10**, 16–29 (2018)
4. El-Ashmawi, W.H., Ali, A.F., Slowik, A.: An improved jaya algorithm with a modified swap operator for solving team formation problem. Soft. Comput. **24**(21), 16627–16641 (2020)
5. El-Ashmawi, W.H., Ali, A.F., Tawhid, M.A.: An improved particle swarm optimization with a new swap operator for team formation problem. J. Indust. Eng. Internat. **15**(1), 53–71 (2019)

6. Kargar, M., An, A.: Discovering top-k teams of experts with/without a leader in social networks. In: Proceedings of the 20th Acm International Conference On Information And Knowledge Management, pp. 985–994 (2011)
7. Kargar, M., Zihayat, M., An, A.: Finding affordable and collaborative teams from a network of experts. In Proceedings of the 2013 SIAM International Conference On Data Mining, pp. 587–595. SIAM (2013)
8. Lappas, T., Liu, K., Terzi, E.: Finding a team of experts in social networks. In: Proceedings of the 15th ACM SIGKDD International Conference On Knowledge Discovery and Data Mining, pp. 467–476 (2009)
9. Li, C.-T., Shan, M.-K.: Team formation for generalized tasks in expertise social networks. In: 2010 IEEE second International Conference on Social Computing, pp. 9–16. IEEE (2010)
10. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. **69**, 46–61 (2014)
11. 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)
12. Rao, R.: Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int. J. Ind. Eng. Comput. **7**(1), 19–34 (2016)
13. Rao, R.V., Saroj, A.: A self-adaptive multi-population based jaya algorithm for engineering optimization. Swarm Evolut. Comput. **37**, 1–26 (2017)
14. Rehman, M.Z., et al.: A novel state space reduction algorithm for team formation in social networks. PLoS ONE **16**(12), e0259786 (2021)