Expert Systems With Applications 177 (2021) 114886

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A unified framework for effective team formation in social networks

Kalyani Selvarajah [a], Pooya Moradian Zadeh [a], Ziad Kobti [a], Yazwand Palanichamy [b], Mehdi Kargar [b,*]

[a] School of Computer Science, University of Windsor, Windsor, Ontario, Canada
[b] Ted Rogers School of Management, Ryerson University, Toronto, Ontario, Canada

ARTICLE INFO

Keywords:
Team formation
Social networks
Graph data management
Multi-objective optimization

ABSTRACT

Collaboration networks are social networks in which nodes represent experts, and edges represent the interactions between them. Team Formation Problem (TFP) in Social Networks (SN) is to construct a group of individuals to work on complex tasks. Teams should satisfy the skill set required by the tasks and can collaborate effectively under multiple constraints. Although many algorithms have been proposed to confront the TFP, most of them optimize different criteria and various parameters (e.g. communication cost or expertise level). There is no unified framework to incorporate the most significant parameters towards formulating effective teams of experts. We propose a unified framework for the TFP in SN based on a multi-objective cultural algorithm that involves the integration of essential cost functions such as communication cost, expertise level, collective trust score, and geological proximity. Since these are conflicting objectives, we return a set of Pareto front of teams that are not dominated by other feasible teams with regards to any of the objectives. Moreover, we examine the temporal nature of both communication costs and expertise levels in our model and introduce a new method to formulate them. We introduce a profile similarity formula to express the trust score. We then discuss the importance of emotional index in TFP. Our model is tested on a benchmark table, which is generated with various criteria of social networks. Our model is then compared with NSGA II, Graph-Based and Exhaustive search.

1. Introduction

In numerous circumstances, the use of team-based opportunities has become more commonplace. Such teams are often confined within a group or society, especially within the manufacturing, law, academia, healthcare sectors, in freelancer jobs such as Upwork[1] and Guru[2], and other professional organizations. In most of these situations, people form a network based on past collaboration with their colleagues. These connected networks can be considered as social networks that contain individuals (i.e. professionals) as nodes and edges that model the past collaboration among individuals. Furthermore, each individual posses a set of skills (i.e. expertise in particular areas).

A team is a group of individuals collaborating together to work on a task with expectations that the task should be completed successfully. Team Formation Problem (TFP) in social networks is to collect the group of individuals who match the requirements of given tasks. It is a challenging process because it needs to ensure that the team that is assembled is able to carry out the task effectively. Prior research studies have observed the TFP problem. However, this area of study requires newer and innovative approaches to improve it's accuracy. To identify productive or successful teams to work on projects, merely satisfying the bare-bone requirements of the projects is not enough. It requires several other prerequisite information that is related to the individuals in the teams, such as how they are connected in the underlying social network and the nature of the relationship that binds the individuals together in the team.

Most of the existing models regarding the TFP considered communication costs among teams as the predominant variable of interest, mainly because lower communication costs incurred among experts translates to increased efficient and effective delivery of tasks. In Lappas,

* Corresponding author.
E-mail addresses: selva111@uwindsor.ca (K. Selvarajah), pooya@uwindsor.ca (P.M. Zadeh), kobti@uwindsor.ca (Z. Kobti), ypalanichamy@ryerson.ca (Y. Palanichamy), kargar@ryerson.ca (M. Kargar).

[1] https://www.upwork.com/
[2] https://www.guru.com/

https://doi.org/10.1016/j.eswa.2021.114886
Received 24 June 2020; Received in revised form 5 December 2020; Accepted 7 March 2021
Available online 29 March 2021
0957-4174/

Liu, and Terzi, 2009, the authors mentioned that communication costs have a high impact when forming teams in social networks. Similarly, alternative studies further analyzed this problem and proposed various algorithms to minimize other functions for the communication cost (e.g., enhanced Steiner tree) Kargar and An, 2011; Kargar, An, and Zihayat, 2012; Anagnostopoulos, Becchetti, Castillo, Gionis, and Leonardi, 2012. Furthermore, there have also been subsequent investigations into the TFP that have been conducted to analyze different requirements which influence the formation of teams such as expertise score Anagnostopoulos et al., 2012; Farhadi, Sorkhi, Hashemi, and Hamzeh, 2011, geographical proximity Ponds, Van Oort, and Frenken, 2007, and density Juárez and Brizuela, 2018.

Existing research has undermined the notion of expertise level as a dynamic scoring option. For example, if the expert profile states that an individual possesses a set of skills, it does not mean that this person is an expert in every skill. Rather, it would be more appropriate to consider that in reality, people never maintain the same expertise level all the time, if they do not continuously work towards the betterment of that skill (as they might gradually become less competent over time due to forgetfulness or lack of practice). Thus, based on how often anyone works on any given skill, and the time of using the skill within a project, we can formulate and envision the current level of expertise as a dynamic score. Moreover, although existing research examined the communication cost of the team using different approaches, they failed to discuss the dynamic nature of it. As such, introducing the temporal features of the expertise score and communication score is significant towards further analyzing current constraints of the TFP. Therefore, we introduce two formulas to incorporate the dynamic behavior of expertise score and communication cost in our framework.

A team without trust is just a group of individuals working together, oftentimes making unsatisfactory progress. This study considers three factors to measure trust quantitatively: (1) explicit trust score, (2) profile similarity score, and (3) emotional intelligence index. We believe a direct trust score is not the only information sufficient enough to evaluate trust between any two members in a team because it might be biased. Scores that demonstrate how frequently a person shares similar interests with others and whether a person is emotionally fit to work in a team or not, are also significant intervening factors to take into consideration when establishing trust score. Although these scores are not deemed quantifiable enough to construct a computational model, we propose ideas in this study based on recent research in management.

Our primary objective is to model a unified framework for the team formation problem in social networks, to analyze various scenarios that when quantified, would help build a more effective team in the long run. Such scenarios are the following, but not limited to: How frequently have team members worked in the past, are they emotionally fit to work as a team, how often does each team member have faith in other members and are they in close proximity with one another in the network. To address these questions, we design a unified team formation model to study the formation of teams and their effect on workplace productivity. Since conflicting objectives are studied in this work, we return a set of Pareto front of teams that are not dominated by other feasible teams with regards to any of the objectives.

The remainder of this paper is organized as follows. We discuss related works in Section 2. Section 3 defines the problem definition for the TFP within a dynamic environment. Then we discuss the Multi-Objective Cultural Algorithm for the TFP in Section 4. In Section 5, we conduct extensive experiments on synthetic data-sets. Finally, we conclude this paper in Section 6.

2. Related Works

Researchers in different fields use a variety of methods to solve hard optimization problems, most of them are known to be NP-hard. These methods include heuristics (such as greedy algorithms), approximation algorithms with provable bounds, branch and bound, and evolutionary-based techniques. Here, we review some of the recent methods that solve optimization problems.

Mercorio, Mezzanzanica, Moscato, Picariello, and Sperli, 2019 used a modified graph-based algorithm to detect overlapped communities of authors from scholarly data while Pourasghar, Izadkhah, Isazadeh, and Lotfi, 2020 applied the same algorithm for clustering in order to help software designers in the software reverse engineering process. Kumar, 2020 and Ribas et al., Ribas, Companys, and Tort-Martorell, 2019 proposed greedy algorithms to solve problems related to speech enhancement and job scheduling respectively.

Branch-and-bound algorithms is one of the widely used choices for solving discrete optimization problems. Recent studies by Coniglio, Furini, and San Segundo, 2020 and Ebrahimi, Guda, Alamaniotis, Miserlis, and Jafari, 2020 applied branch-and-bound to solve knapsack and network optimization problems. Exact algorithms are guaranteed to find the optimal solution for any optimization problem. For example, Gil-Gala, Mencıa, Sierra, and Varela, 2020 proposed an exact algorithm to solve real-life scheduling problems. However, for an NP-hard problem, it is unlikely that the exact algorithm terminates in polynomial time. The run time of branch-and-bound algorithms are usually high too. Although greedy algorithms are fast, they usually find the nearest local optima.

Heuristic algorithms based on evolutionary optimization overcome these issues and approximate the global optimum, aiming to find a satisfactory solution (Abazari, Analoui, Takabi, & Fu, 2019; Bali, Ong, Gupta, & Tan, 2019; Wang & Li, 2019).

2.1. Team Formation

Lappas et al., 2009 first addressed the Team Formation Problem (TFP) in social networks. Their seminal research on the topic initially formulated the TFP using communication costs and proved that it was an NP-hard problem. In the last decade, TFP received a great deal of attention from many researchers (Kargar & An, 2011; Yang & Hu, 2013; Wang, Zhao, & Ng, 2015; Selvarajaha, Zadeha, Kargarb, & Kobtia, 2019). In addition to communication costs, some studies examined other aspects to form successful teams, including workload, expertise level, personnel cost, density, geographical proximity and trust score.

2.1.1. Communication Cost

The concept of communication cost (CommCost) is used to measure the effectiveness of collaboration within a team. Lappas et al., 2009 first suggested that it can be calculated using social network analysis, based on the interactions of the individuals. The success of a project relies on how well the experts in teams communicate and collaborate with others. The CommCost measures the closeness of the individuals in a social network $\mathscr{G}$. If two individuals V_i and V_i are adjacent with one another in the network, the CommCost is the weight of the edge (V_i, V_j); otherwise, the CommCost is the shortest path between V_i and V_i. Numerous studies focused on the concept of CommCost. Many studies in the field discussed various functions including minimum spanning tree (Lappas et al., 2009; Li & Shan, 2010; Anagnostopoulos et al., 2012; Majumder, Datta, & Naidu, 2012; Li, Shan, & Lin, 2015; Basiri, Taghiyareh, & Ghorbani, 2017; Chen, Yang, & Yu, 2017), the diameter distance (Lappas et al., 2009; Chen et al., 2017; Selvarajaha et al., 2019), and the sum of distance (Kargar & An, 2011; Kargar & An, 2011; Chen et al., 2017; Kargar, Zihayat, & An, 2013; Li, Huang, & Yan, 2018; Selvarajah, Bhullar, Kobti, & Kargar, 2018; Selvarajaha et al., 2019).

2.1.2. Work Load

In a multi-project scenario, adjusting the amount of work according to the capacity of the individuals is an essential factor to complete the work successfully. Some research works focused on balancing the load of the teams in the presence of multiple tasks (Anagnostopoulos et al., 2012; Majumder et al., 2012; Gutiérrez, Astudillo, Ballesteros-Pérez, Mora-Melià, & Candia-Véjar, 2016; Selvarajah et al., 2018). However,

Table 1
Team formation problem based on various parameters.

Authors	CommCost	Work Load	Expertise Level	Personnel Cost	Density	Geo-Proximity	Trust Score
Lappas et al. (2009), Kargar and An (2011), Li and Shan (2010), Kargar and An (2011), Li et al. (2015), Basiri et al. (2017), Chen et al. (2017), Li et al. (2018), Selvarajah et al. (2018), Selvarajaha et al. (2019)	✓						
Anagnostopoulos et al. (2012), Majumder et al. (2012), Gutiérrez et al. (2016)	✓	✓					
Kargar et al. (2012), Kargar et al. (2013), Ashenagar et al. (2015)	✓			✓			
Selvarajah et al. (2018)	✓	✓				✓	
Han et al. (2017), Chen et al. (2019)	✓					✓	
Dorn and Dustdar (2010), Farhadi et al. (2011), Zhu et al. (2014), Neshati et al. (2014), Zhang et al. (2017)	✓		✓				
Bozzon et al. (2013)		✓					
Awal and Bharadwaj (2014)			✓				✓
Largillier and Vassileva (2012)							✓
Gutiérrez et al. (2016, 2013, 2012, 2018)					✓		

the work load has not been studied in the past in combination with other objectives.

2.1.3. *Expertise Level*

In the real world, knowledge structures of skills are quite diverse. Many people possess different types of competencies and their level of expertise can vary greatly. This is due to the fact that as specialization in every field increases, an individual who is an expert, has to maintain the level of expertise for a specialized skill in a certain discipline to perform a given task skillfully. In a given extensive social network, many individuals might have the ability to perform a specific task. Some people perform at a higher level than others. As such, there should be a method to measure the expertise score in any particular skill to find a suitable candidate to complete the task.

The skill involving the mastery level of the entire team decides the successful outcome of the project. Prior studies have investigated the expertise level and considered it as a binary value: if a person has a skill, it is 1; otherwise, it is 0. Several studies have combined the expertise level with the communication costs (Dorn & Dustdar, 2010; Farhadi et al., 2011; Zhu, Chen, Xiong, Cao, & Tian, 2014; Neshati, Hashemi, & Beigy, 2014; Zhang, Yu, & Lv, 2017). Some studies also examined the expertise level from another perspective, such as trust (Awal & Bharadwaj, 2014).

2.1.4. *Personnel Cost*

In many real scenarios such as Guru, Freelancer, and Upwork, people offer their services in exchange for a salary, which is the personnel cost of experts. Forming teams with affordable cost is beneficial for the project and recent work modeled the TFP with the combinations of CommCost and Personnel Cost (Golshan, Lappas, & Terzi, 2014; Kargar et al., 2012; Kargar et al., 2013; Ashenagar, Eghlidi, Afshar, & Hamzeh, 2015).

2.1.5. *Geological Proximity*

Collaboration and the exchange of knowledge are easier to obtain by way of geographical proximity. Although many studies in social science theoretically discussed the benefits of geographical proximity in collaboration of individuals, this direction has not been studied thoroughly within the IT and computer science disciplines (Han, Wan, Chen, Xu, & Wu, 2017; Chen et al., 2019; Selvarajah et al., 2018). In the era of web 2.0, one can argue that geological proximity is not necessarily required as a fundamental component when considering effective team dynamics. However, the chances of devising increasingly effective group collaborations are higher when experts have regular in person meetings.

2.1.6. *Trust Score*

Team trust is increasingly being recognized as an essential predictor of team performance. It has been generally agreed upon that a team with high collective trust is more successful than teams with lower trust (Cascio, 2000). Prior research has indicated that within high-trust environments, people commit towards doing the best work possible and are motivated to produce more successful outcomes. If team members make mistakes, everyone else in the team will support and share information to remediate any errors in the process, in order to better produce the outcome of the task in a streamlined manner.

The trust score indicates the level of trust that members of the team share with one another. Relying on trust becomes an important element regarding the success of social networks, a number of prior studies have closely examined the trust score in various perspectives (Sherchan, Nepal, & Paris, 2013). However, quantifying the trust value is a challenging process. The authors in (Ahn, DeAngelis, & Barber, 2007; Awal & Bharadwaj, 2014; Largillier & Vassileva, 2012) initially formulated a trust score for the TFP. Similarly, the authors of (Awal & Bharadwaj, 2014) used a ratings-based approach based on past interaction that was adopted and modified from (Dong-Huynha, Jennings, & Shadbolt, 2004).

2.1.7. *Density*

In a given graph G, the density method finds the densest sub-graph, which satisfies the skill requirements of a project. The theory behind the implementation of the density method, involves the understanding that multiple past collaboration among experts results in a strong and coherent team (Rangapuram, Bühler, & Hein, 2013; Gajewar & Das Sarma, 2012; Juárez & Brizuela, 2018).

Table 1 summarizes the existing related work in TFP based on various requirements to formulate successful teams. Many of these studies are considered multi-objective optimization as they optimize more than one objective in the TFP. However, many of these prior research studies have undermined the investigation and application of important requirements in one way or another. For example, if the study proposed a model using both communication cost and workload, it failed to analyze or take into account the expertise level or trust shared amongst team members. Moreover, the communication cost and expertise level do not remain the same over time. For instance, if a programmer who is an expert in Java, gets promoted to a managerial status, or begins to work in a different programming language such as Python, their expertise level in Java will not remain the same indefinitely. In this regard, we seek to remediate this limitation regarding the TFP by identifying and incorporating the temporal behavior of experts when evaluating their expertise score.

Similarly, regarding communication costs, if two experts have frequent collaborations in the past and no recent collaborations, the chances of collaborating in the future will become less likely of an occurrence. Previous work does not take this into account when modeling the TFP with dynamic communication score. We focus on these issues and propose a new unified framework for TFP by incorporating dynamic communication cost and dynamic expertise score.

2.2. Evolutionary Multi-objective Optimization Methods

Evolutionary Algorithms (EA) or variants of EAs are identified as powerful approaches for solving different multi-objective optimization problems (MOPs). In this section, we present recent advancement in this area and discuss studies that have a close relation to our topic.

Interval multi-objective optimization problems (IMOPs) have many applications in real-world. Authors of Gong, Sun, and Ji, 2013 propose a new preference-based evolutionary algorithm to deal with IMOPs. The algorithm works based on the theory of preference polyhedron and provides a novel mechanism to sort the solutions and guide the search direction to the decision maker's preferences. According to the experiments, the proposed method is capable of obtaining the most preferred solution highly suitable for the DMs preferences. Recently, Sun et al., 2020 propose a new method to deal with IMOPs by incorporating a number of local searches into an existing EAs and presenting a memetic algorithm (MA). The authors first use an existing model (imprecision-propagating multi-objective EA) to search the decision space and propose an activation mechanism based on the increment of the hyper-volume, to start the local search process. Then the initial population is made in a way that its center is an individual with a great contribution to the hyper-volume and a small level of uncertainty. According to the results, the proposed approach can obtain Pareto fronts with a small uncertainty and high convergence.

One of the related problems to this work is the blocking lot-streaming flow shop (BLSFS) scheduling problem. In BLSFS, the goal is to schedule a number of jobs on more than one machine without any intermediate buffers between adjacent machines. Note that BLSFS is related to the classical job shop scheduling problem, but it contains additional requirements. Recently, Gong, Han, and Sun, 2018 present a hybrid multi-objective discrete artificial bee colony algorithm to solve this problem. The authors study two conflicting requirements, the makespan and the earliness time, and proposed an initialization strategy using prior knowledge with the capability of producing high-quality solutions. They also develop two crossover operators using information obtained from all non-dominated solutions in the current population. Lastly, they propose a local search operator that works based on the Pareto dominance relation. In another work, the authors in (Han, Gong, Jin, & Pan, 2019) study the BLSFS problem under a common scenario where the machine breakdown occurs. They propose a novel multi-objective model for this problem by addressing the robustness and stability requirements. The authors design an evolutionary multi-objective robust scheduling algorithm, where solutions that are made by different single-objective heuristics are incorporated into population initialization. Furthermore, they propose a rescheduling strategy to reduce the negative effect of machine breakdowns.

Decomposition-based multi-objective evolutionary methods are one of the effective ways of solving MOPs (Luo et al., 2018; Cai, Mei, & Fan, 2018; Zhang, Gong, Sun, & Qu, 2018; Dong & Dai, 2019). In one of the recent works, Cai et al., 2018 propose a decomposition-based many-objective EA that contains two types of adjustments for the direction vectors in order to enhance the convergence of the solutions. The first type is designed to expand the number of direction vectors, while the second one aims to change the positions of the weak vectors. According to their results, the proposed model is effective in addressing multi-objective problems with irregular Pareto fronts.

To solve multi-objective optimization problems (MOPs), some studies convert the problem into a single objective optimization. However, every conflicting objective is equally important to the problem. Converting a multi-objective problem into a single objective requires trade-off parameters. In order to give equal importance to trade-off solutions, multi-objective evolutionary algorithm is a preferred method. In recent years, several works use multi-objective evolutionary algorithms and their variants such as NSGA II in various applications such as forming sports teams, collaborative networks and project management (Pérez-Toledano, Rodriguez, García-Rubio, & Ibañez, 2019; Crawford, Rahaman, & Sen, 2016; Juárez & Brizuela, 2018; Niveditha, Swetha, Poornima, & Senthilkumar, 2017; Miranda, Mello, & Nascimento, 2020).

EAs are also actively used for solving clustering (Alves, Campello, & Hruschka, 2006; Dutta, Sil, & Dutta, 2019), community detection (Guerrero, Montoya, Baños, Alcayde, & Gil, 2017; Liu, Liu, & Jiang, 2014; Zadeh & Kobti, 2015; Feng, Chen, Li, & Luo, 2020), engineering applications (Manne, 2019) and link prediction (Bliss, Frank, Danforth, & Dodds, 2014; Zadeh & Kobti, 2016). In recent years several approaches have been proposed based on genetic algorithms to form collaborative learning teams in an educational setting (Lescano, Costaguta, & Amandi, 2016; Zheng, Li, Liu, & Lu, 2018; Awal & Bharadwaj, 2014). However, recently, the works of (Selvarajah et al., 2018; Selvarajaha et al., 2019) which aim to deal with the team formation problem using a novel evolutionary search technique based on a higher-order cultural evolution, demonstrated a better performance than genetic algorithms.

3. Proposed Model

Our model aims to find experts with the highest level of expertise who can perform the tasks, which require a certain expertise level in a cost-effective manner. To address the inherent cost of team members in various aspects, we have to demand a new model for the Team Formation Problem(TFP). This section discusses the proposed comprehensive model for the TFP and outlines key definitions to formulate the model.

3.1. Preliminaries

The social network is modeled as an undirected weighted graph $\mathscr{G}(\mathscr{V},\mathscr{E})$, where $\mathscr{V}$ represents a set of individuals or experts and $\mathscr{E}$ represents the relationship between them. The weights of the edges $\mathscr{E}$ in graph $\mathscr{G}$ can be interpreted as a indicator to measure how efficiently the individuals work together in the past. We assume there is a set of n skills $\mathscr{S} = \{S_1, S_2, \ldots, S_n\}$ and a set of m individuals $\mathscr{V} = \{V_1, V_2, \ldots, V_m\}$. Each individual V_j is associated with a set of skills $\mathscr{S}_i$ ($\mathscr{S}_i(V_i) \subseteq \mathscr{S}$). Each skill in $\mathscr{S}_i$ is associated with a score based on the expertise level of the expert in a specific skill.

We assume that a set of k tasks is $\mathscr{T} = \{t_1, t_2, \ldots, t_k\}$. A task t_i is simply a set of skills $\mathscr{S}_j$ required to perform a project, i.e $\mathscr{S}_j(t_i) \subseteq \mathscr{S}$. Similar to the expertise score of an expert, each required skill for a project is also associated with a score based on the required expertise level of the skill in that project.

Definition 1. (*Team of Experts*) For a given set of experts $\mathscr{V}$ and a given task T that requires a set of skills $\{S_{i_1}, S_{i_2}, \ldots, S_{i_r}\} \subseteq \mathscr{S}$, a *team of experts* for t_i is a set of r skill-expert pairs: $t_i = \{\langle S_{i_1}, V_{j_1}^{S_{i_1}}\rangle, \langle S_{i_2}, V_{j_2}^{S_{i_2}}\rangle, \ldots, \langle S_{i_r}, V_{j_r}^{S_{i_r}}\rangle\}$.

In the above definition, $V_{j_p}^{S_{i_q}}$ is an expert $V_{j_p} \in \mathscr{V}$ who posses skill S_{i_q}, where.

$S_{i_q} \in \mathscr{S}, \{V_{j_1}, V_{j_2} \ldots V_{j_r}\} \subseteq \mathscr{V}$.

3.2. Communication Cost

Kargar et al. Kargar and An, 2011 proposed the Sum of Distances to measure the communication cost of a team. This is the sum of distances

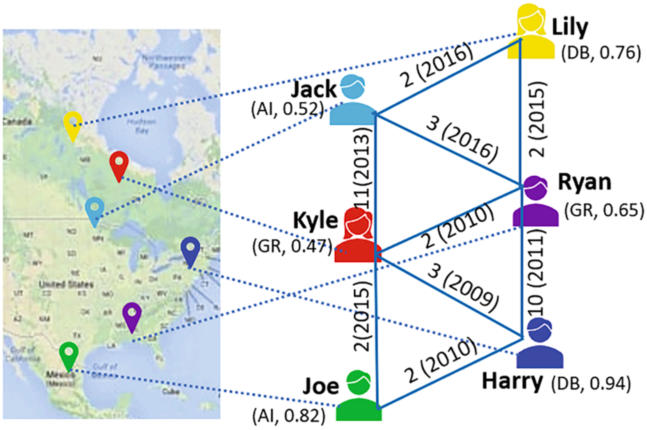

Fig. 1. A Social Network of experts with expertise skill followed by Emotional Intelligence Index, geographical location and the last time each pair collaborated together.

between each pair of skill holders. The expectation is that a team with smaller communication values would finish the project faster.

- **Sum of Distance:** *Given a team of experts* $\{\langle S_{i_1}, V_{j_1}^{S_{i_1}}\rangle, \langle S_{i_2}, V_{j_2}^{S_{i_2}}\rangle, \ldots, \langle S_{i_r}, V_{j_r}^{S_{i_r}}\rangle\}$, *the sum of distances in the team is defined as:*

$$SumDist = \sum_{p=j_1}^{j_r} \sum_{q=j_2}^{j_r} dist(V_p, V_q) \quad (1)$$

where $dist(V_p, V_q)$ is the shortest distance between V_p and V_q in G. i.e., the sum of weights on the shortest path between V_p and V_q.

Note that for the above cost function, the distance between two experts is defined over the entire graph G as a static parameter. In reality, the communication cost shows the dynamic nature and changes over time. Therefore, we enhance the above cost function in order to incorporate the dynamic nature of the communication cost. Time difference from current to last collaboration on a project is incorporated with the shortest path distance to adopt the temporal feature.

Definition 2. (*Dynamic Communication Cost*) Given a team of experts $\{\langle S_{i_1}, V_{j_1}^{S_{i_1}}\rangle, \langle S_{i_2}, V_{j_2}^{S_{i_2}}\rangle, \ldots, \langle S_{i_r}, V_{j_r}^{S_{i_r}}\rangle\}$, the dynamic sum of distances in the team is defined as:

$$DCC = \sum_{p=j_1}^{j_r} \sum_{q=j_2}^{j_r} (dist(V_p, V_q) + \alpha(t' - t)) \quad (2)$$

where $dist(V_p, V_q)$ is the shortest distance between V_p and V_q in G. t' is the current time and t is last time the two experts work on a task together. α is an attenuation factor.

To motivate our approach, we illustrate a social network of experts in Fig. 1. This diagram represents the information that our model takes into account for finding effective teams: geographical location of each expert, the skills of each expert, the emotional intelligence index of each expert, and the the number of times and year of last collaboration between any two experts. The following example represents the concept of dynamic communication cost that we use in this work. Assume we need to perform a project which needs expertise in three areas: artificial intelligence (AI), databases (DB), and graphics (GR). We can form two teams $A = \{Jack, Lily, Ryan\}$ and $B = \{Kyle, Joe, Harry\}$. The communication cost of both A and B is the same 7 if we do not consider the time of their last collaboration. On the other hand, if we consider the last collaboration time, we can evaluate the communication cost using Eq. 2 as below:

Table 2
The history of skills by each expert in a given social network G.

Experts	Skills of projects	Worked Year
Jack	AI	2009, 2011
Kyle	GR	2009,2010
Ryan	GR	2015,2017
Harry	DB	2009, 2010, 2011
Joe	AI	2014, 2017
Lily	DB	2016, 2015
Ryan	AI	2013

$$DCC_A = (2 + 0.1(2019 - 2016)) + (3 + 0.1(2019 - 2016)) + (2 + 0.1(2019 - 2015))$$

where we assume that the current year as 2019 and $\alpha = 0.1$. The dynamic communication cost of teams A and B would be 8.0 and 9.3, respectively. A smaller communication cost would indicate a more cohesive team of experts with frequent and recent collaboration attempts. Therefore, team A has a higher likelihood of collaborating more effectively in the future.

3.3. Team Skill Mastery Level

To the best of our knowledge, a limited amount of research has closely examined the expertise level regarding the TFP Awal and Bharadwaj, 2014; Farhadi et al., 2011. However, throughout the literature, this scale has been interpreted as a static score. The level of expertise of an individual never remains the same if he/she does not work on a specific field continuously. Therefore, considering the temporal behavior of the expertise is an important issue to take into consideration. Past experiences are used to obtain the expertise score of an individual. The difference in time from the present to the last time an expertise was used by an expert is incorporated within the expertise score to adopt the temporal feature. By incorporating this difference in recorded time from whence the last skill was used, we will be able to more accurately record the temporal nature of the expertise score for a given individual over the lifetime of a project.

The expertise score of an expert $v_i \in \mathscr{V}$ can be defined as the ratio between the total number of projects or published papers $|ES_{V_i}^{S_j}|$ of the expert V_i in a skill $S_j \in \mathscr{S}$ and the total number of projects or published papers of the experts in a given network G in the skill S_j.

$$\mathscr{ES}^{S_j}_{V_i} = \frac{|ES^{S_j}_{V_i}|}{\sum_{p=1}^{r}|ES^{S_j}_{V_p}|} - \alpha(t' - t) \tag{3}$$

where r is the number of experts who interact with projects or papers with the skill S_j, t' is the current time, and t is last time the experts works on a task with skill S_j. α is an attenuation factor.

For instance, let us consider the network of 7 people and the history of skills by each expert in a given social network G, as shown in Table 2. The expertise score of *Jack* and *Joe* for *AI* is the same value 0.4 if we do not consider the temporal nature. But their score is different with temporal nature, 0.24 and 0.36 respectively if the current year is 2019 and $\alpha = 0.02$.

Definition 3. (*Team Skill Mastery Level*) Given a network G, if the expertise level of an expert $V_i \in \mathscr{V}$ in skill S_j is $\mathscr{ES}^{S_j}_{V_i}$, the collective expertise level of a team is the sum of the expert level (ExpLevel) of the team of experts $\{\langle S_{i_1}, V^{S_{i_1}}_{j_1}\rangle, \langle S_{i_2}, V^{S_{i_2}}_{j_2}\rangle, \ldots, \langle S_{i_r}, V^{S_{i_r}}_{j_r}\rangle\}$, and can be defined as:

$$EL = \sum_{p=j_1, q=i_1}^{i_r, j_r} \mathscr{ES}^{S_q}_{V_p} \tag{4}$$

3.4. Geographical Proximity

Geographical proximity plays a more subtle and indirect role in influencing collaboration and knowledge exchange, Howells, 2002. Experts who are located closely with others have a higher chance to collaborate due to many reasons such as cultural background, language, and time zone Chen et al., 2019. Therefore, considering geographical proximity is an effective feature in team formation.

Definition 4. (*Geographical Proximity*) The Geographical Proximity (GeoCost) of the team of experts $\{\langle S_{i_1}, V^{S_{i_1}}_{j_1}\rangle, \langle S_{i_2}, V^{S_{i_2}}_{j_2}\rangle, \ldots, \langle S_{i_r}, V^{S_{i_r}}_{j_r}\rangle\}$, can be defined as the sum of geographical distance between every pairs in the team.

$$GC = \sum_{p=j_1}^{j_r} \sum_{q=j_2}^{j_r} geodist(V_p, V_q) \tag{5}$$

where $geodist(V_p, V_q)$ is the geographical proximity between V_p and V_q in G.

3.5. Trust Score

The trust mechanism indicates feelings of trustworthiness that an individual feels towards others, and has been studied in greater depth over the years. The trust score can be an explicit value that an individual directly gives a score to another based on their experience, such as in Epinion.com Richardson, Agrawal, and Domingos, 2003; Selvarajah, Kobti, and Kargar, 2019. The explicit value might be biased. In addition to this, if two individuals have never interacted with one another in the past, there should be an effective mechanism in place to gauge the level of trust between them. Moreover, if an individual A trust individual B, this does not imply that the individual B should trust A. In other words, the property of the trust is asymmetrical Zhan and Fang, 2010. At the same time, trust does not hold the transitivity property. (i.e.) If person A trusts person B and B trusts person C, we cannot imply that A trusts C.

To compute trust between any two individuals in a given network G, we incorporate three types of information: explicit trust score, profile similarity score, and emotional intelligence index.

3.5.1. Explicit Trust

The concept of explicit trust score is simple, and it is the value given by an individual to another. It can be a number between -1 and 1. Based on the people's experience, they can rate the trust value. If a person A trusts person B, trust can be 1, if he/she does not trust the other one, it can be -1, and if they have never interacted in the past, it can be 0.

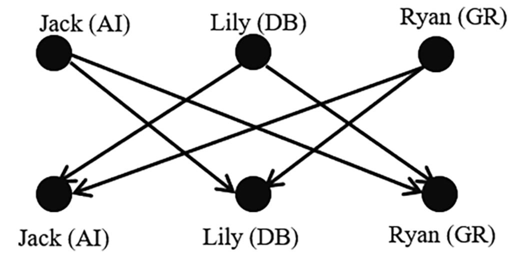

Fig. 2. The representation for the method to calculate collective trust score of a team.

3.5.2. Profile Similarity

A number of social psychology research state that people with similar interests like to communicate and work together Lazarsfeld et al., 1954; McPherson, Smith-Lovin, and Cook, 2001. Further, Ziegler et al. Ziegler and Golbeck, 2007 investigate the relationship between trust and profile similarity, and introduce a framework to quantify the trust using profile similarity when other types of trust evidence is absent.

In this work, since we primarily focus on the expert networks, we consider the expert's skill descriptions to calculate the profile similarity. If everybody knew to what degree every expert is a specialist in each field, they could potentially utilize this knowledge to discover researchers with trust and recommend future collaborations Pavlov and Ichise, 2007. The profile similarity between two experts can be calculated by taking into account the relationship between any pair of experts, where a pair is formed by elements from the corresponding expertise expert list. For every such pair, a suitable value $\mathscr{V} \in [0,1]$, that best reflects the strength of the relationship between the two concepts, can be associated with their profile similarity. For example, $\mathscr{V} = 1$, when the corresponding concepts are identical and respectively, $\mathscr{V} = 0$ if they are not related. Thus, the expertise similarity $\mathscr{SF}_{i,j}$ between two expert profiles, a_i and a_j, can be defined by the following equation.

Definition 5. (*Score of Skill Similarity*) Assume a network $\mathscr{G}\langle\mathscr{V}, \mathscr{E}\rangle$and a set of experts $V = \langle v_1, v_2, \ldots, v_m\rangle$with a set set of skills $\mathscr{S} = \langle s_1, s_2, \ldots, s_n\rangle$ is given. The skill set of any two experts v_i and v_j can be $\mathscr{S}_{v_i} = \langle v_i^{s_{p1}}, v_i^{s_{p2}}, \ldots, v_i^{s_{pr}}\rangle$ and $\mathscr{S}_{v_j} = \langle v_j^{s_{q1}}, v_j^{s_{q2}}, \ldots, v_j^{s_{ql}}, \rangle$ respectively. If $v_i^{s_{px}}$ and $a_j^{s_{qy}}$ are the same skills, $\mathscr{V} = 1$, otherwise, $\mathscr{V} = 0$.

$$ProSim_{v_i, v_j} = \sum_{k=1}^{n} \frac{\mathscr{V}_k}{n} \tag{6}$$

where n is the total number of expert's skill list of v_i and v_j,$n = |S_{v_i} \cup S_{v_j}|$.

Let us consider an example wherein expert A has a list of skills $\{ML, DB, CN, WM\}$ and another expert B has a list of skills $\{ML, DB, SE\}$. The profile similarity will be 0.4, because the total number of skills is $|S_A \cup S_B| = 5$ and number of common skills is $|S_A \cap S_B| = 2$.

3.5.3. Emotional Intelligence Index

Emotional intelligence (EI) of a person is playing a key role in team performance, displaying the mental experiences, or deploying actual human behaviors. A good amount of management research has paid great attention to EI Lee and Wong, 2017. Emotional intelligence fosters trust, which can be built through EI Cooper, 1997. In this study, we incorporate the EI index alongside the trust score.

Emotional intelligence can be defined in several ways, including the ability to understand emotions in oneself and others to make decisions, solve problems, and communicate with others. It requires a strenuous

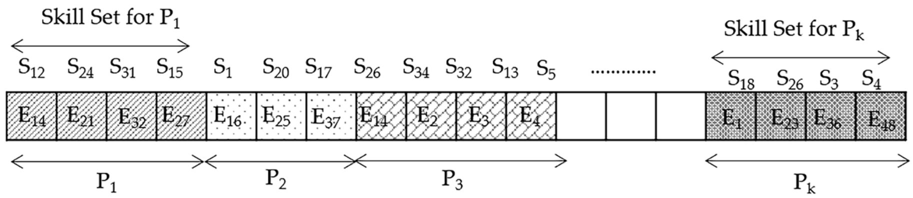

Fig. 3. The representation of an individual.

effort to quantify as a score. To measure EI, many studies have used standard models or metrics such as Myers-Briggs Type Indicator (MBTI) and Five-Factor Model (FFM) and discovered significant associations between personality factors and the way of thinking Kosti, Feldt, and Angelis, 2014. These model give a quantified score for emotional intelligence.

3.5.4. *Computing Final Trust Score*

This section aims to calculate a final trust score for two connected individuals on social networks. As was discussed in the aforementioned sections, the final score is the combination of explicit trust score, profile similarity and emotional index. The following equation provides how the expert v_i trusts expert v_j.

$$Tr_{V_i,V_j} = \alpha_1 ExpT_{V_i,V_j} + \alpha_2 ProSim_{V_i,V_j} + \alpha_3 EI_{V_j} \tag{7}$$

Where EI_{v_j} is the emotional index of the expert v_j. α_1, α_2 and α_3 are three balancing factors such that $\alpha_1 + \alpha_2 + \alpha_3 = 1$. Note that, the trust expert v_j has on v_i might not be the same as per the property of asymmetry.

Definition 6. (*Collective Trust Score*) Given network G, the trust score of an expert $V_i \in \mathscr{V}$ has on $V_j \in \mathscr{V}$ is $TrustScore_{v_i,v_j}$, and V_j has on V_i is $TrustScore_{V_j,V_i}$. Note that $TrustScore_{v_i,v_j} \neq TrustScore_{V_j,V_i}$. The collective trust score of a team is the sum of the trust score (ColTrust) of the team of experts $\{\langle S_{i_1}, V_{j_1}^{S_{i_1}}\rangle, \langle S_{i_2}, V_{j_2}^{S_{i_2}}\rangle, \dots, \langle S_{i_r}, V_{j_r}^{S_{i_r}}\rangle\}$, and can be defined as:

$$CT = \sum_{p=j_1}^{j_r}\sum_{q=j_2}^{j_r} Tr_{V_p,V_q} + \sum_{p=j_1}^{j_r}\sum_{q=j_2}^{j_r} Tr_{V_q,V_p} \tag{8}$$

As shown in Fig. 2, collective trust of the team evaluates the trust between every pair of individuals separately. For example, the trust between *Jack* and *Lily* are $Tr_{jack,lily}$ and $Tr_{lily,jack}$, where $Tr_{jack,lily} \neq Tr_{lily,jack}$.

4. Multi Objective Cultural Algorithm for TFP

The proposed TFP framework is a multi-objective optimization problem. Hence, every objective is equally important to the problem. Applying the trade-off method favors some objectives than objectives. Therefore, to solve the multi-objective TFP, we use the Multi-Objective Cultural Algorithm (MOCA) Framework. It is important because solutions to TFP are regarded from a variety of perspectives and cannot be expressed using only one objective.

The MOCA is the extended version of Cultural algorithms(CA), which is a class of evolutionary algorithms and is inspired by social learning in the society Reynolds, 1994. It is modeled using two pieces of information: the Population Space, keeping a set of individual solutions, and the Belief Space, keeping various knowledge (e.g., Normative, Situational, Historic, etc.) collected from the population. The two pieces communicate through Accept and Influence functions. The Accept function is to permit a selected population to the Belief space, which extracts the knowledge from this population. The Influence function creates new individuals by applying the obtained knowledge. Different steps of MOCA is presented in Algorithm 1. Details of each step is discussed in the following sub-sections.

Algorithm 1: MOCA for TFP.

Input: Social Network of Graph $\mathscr{G}$ ($\mathscr{A}_{\mathscr{N}\times\mathscr{N}}$); Popularity Scores of each experts; Score of Skill expertise; Explicit Trust score; Profile Similarity; Emotional Intelligence Index; Geographical Proximity
Output: Highest Rank Pareto Set

$i \leftarrow 0$ number of iterations
$\mathscr{N} \leftarrow$ numberofpopulation
$\mathscr{P} \leftarrow \varnothing$population
$\mathscr{Q} \leftarrow \varnothing$ offspringpopulation
$k \leftarrow$ numberofProjects
$kb \leftarrow$ numberofteamstobuildknowledge $-$basedbeliefspace
$el \leftarrow$ numberofeliteteamsfornextgeneration
$\mathscr{P}_0[1\dots\mathscr{N}] \leftarrow$ Randomqualifiedexpert $-$idwhosatisfyskillsfor k projects
while $i \leqslant$ TerminationCondition **do**
  ComputeValuesofObjectivefunctionfor $\mathscr{P}_i[1\dots\mathscr{N}]$
  $\mathscr{P}_i[1\dots \mathrm{bi}] \leftarrow$ CollectbinumberofBestIndividualbysortingeachobjective
  $PN[1\dots\mathscr{N}] \leftarrow$ initializenewpopulation
  $k \times bi$ expertsids $\leftarrow$ ChromosomeBreaker($\mathscr{P}_i[1\dots \mathrm{bi}]$)
  $\mathscr{P}_i[1,2\dots,el] \leftarrow k \times bi$ expertsids
  $PN[1\dots el] \leftarrow \mathscr{P}_i[1\dots el]$
  $SP \leftarrow \mathscr{P}_i[1\dots kb]$
  $BS \leftarrow SP^T$
  for $j \leftarrow (el+1)$ **to** $\mathscr{N}$
    if rand() $\leqslant$ 80% **then**
      Offspring[j] $\leftarrow$ new team from belief space (BS)
      if rand() $\leqslant$ 80% **then**
        Offspring[j] $\leftarrow$ Crossover($\mathscr{P}_m, \mathscr{P}_n$)
      else
        Offspring[j] $\leftarrow$ Mutation($\mathscr{P}_m$)
  $\mathscr{R} \leftarrow \mathscr{P} \cup \mathscr{Q}$
  $\mathscr{R}[1\dots\mathscr{N}] \leftarrow$ fastNondominated $-$sort($\mathscr{R}$)
  EvaluateCrowdingDistance($\mathscr{R}[1\dots\mathscr{N}]$)
  Rank $\mathscr{P}$
return $\mathscr{P}[1] \leftarrow$ HighestRankParetoSet

4.1. *Initial Population*

In the setting of cultural algorithms, an initialization method generates individuals randomly. The individuals or chromosomes are solutions to the team formation problem. The individual represents the teams of expert-id who qualified to perform the list of projects, as shown in Fig. 3. In our framework, we define the level of expertise needed from each skill, that is required to complete a task or project. At the same time, we already set the expertise level of experts, as explained in the previous section. When we assign the experts to the required skill of a project, we randomly choose from the qualified list of the experts who satisfy the level of expertise that is required by the project.

4.2. *Objective Functions*

In the spirit of the multi-objective optimization paradigm, we have defined four conflicting objectives to consider when forming a team: (1) The communication cost, (2) The expertise, (3) The geographical proximity, and (4) The collective trust. It automates a unified approach to assemble teams. The multi-objective team formation problem in social networks can be defined as the following: Given a social graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ and a task $\mathscr{T}$ requiring k skills.

$$\underset{\mathscr{V}' \subseteq \mathscr{V}}{Minimize}\ \{DCC(\mathscr{V}'), GC(\mathscr{V}')\} \tag{9}$$

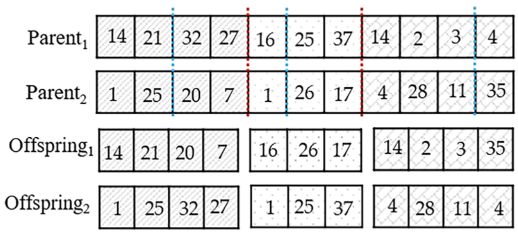

Fig. 4. The process of crossover, where red line indicates the breaking process from ChromosomeBreaker() function and blue line indicates the random one point crossover on sub chromosome.

$$\underset{\mathscr{V}'\subseteq\mathscr{V}}{Maximize}\ \{EL(\mathscr{V}'), CT(\mathscr{V}')\} \tag{10}$$

$$Suchthat\sum|V(t_i)| = |\mathscr{V}'|$$

4.3. Generating Offspring

To generate new individuals or offspring for each generation, we utilize the benefits of CA that uses the information of knowledge and the benefits of genetic operators.

- **The Belief Space:** To influence an individual in the population with the knowledge sources, we collect the best individual based on each best objective separately. We then breakdown the chromosome into the sub-chromosome, which best represents the solution for a single project t_i from the set of projects $\mathscr{T}$. The sub-chromosomes for the same projects are collected as the best team of experts. The normative knowledge information, what is believed to be good areas to search in each dimension, is defined as the transpose matrix of the selected sub-population composed by the selected teams. To update the topological knowledge component, we mutate from the unexplored expert list. We explained this under the Genetic operator section.
- **Genetic Operators:** The genetic operations in MOCA involve crossover and mutations, which help to generate offspring. First, the genetic operator, crossover, or mutation was selected. Following this, the individuals or chromosome is chosen randomly. The chromosome then sends to the ChromosomeBreaker() function to break down into a team of experts per project $\{t_1, t_2, \ldots, t_k\}$. One-point crossover is then applied on each sub chromosomes, as shown in Fig. 4.
- **Mutation Operators:** The mutation operator begins with ChromosomeBreaker() function. It then mutates an expert from a randomly selected skill for each project. This random expert will be chosen from an unexplored qualified expert list, which is different from qualified experts to a set of experts in the current population. After every genetic operator, sub chromosomes are combined using ChromosomeCombiner() function and then objective values of the new chromosome are evaluated.

4.4. Non-Dominated Sorting

For a multi-objective problem, MOCA uses the widely used fast non-dominated sorting approach to compare solutions. We use the ranking procedure as described in Deb, Pratap, Agarwal, and Meyarivan, 2002. Objective values of each solution are compared with every other solution in the population, to classify them as either dominated or non-dominated. All individuals in the first Pareto front are ranked to 1. To find the individuals in the next Pareto front, the first front is removed temporarily, and the above procedure is repeated until they reach the termination condition. In order to find the valid Pareto front, the objective functions are considered collectively when knowledge information influences individuals in the belief space.

Table 3
Set of experimental settings where P represents the number of tasks and S represents numbers of skills for each task.

Experiments	No of Projects	No of Skills
No 1	1	3
No 2	1	5
No 3	1	8
No 4	3	{2,3,5}
No 5	3	{4,5,6}
No 6	4	{2,3,4,5}
No 7	4	{3,4,5,6}
No 8	4	{5,4,8,6}

5. Experiments

This section is to evaluate and analyze the solution produced by MOCA for TFP. To have a unified framework for TFP, we consider highly preferred parameters to discover teams. To the best of our knowledge, there is no real-world dataset that aligns accordingly to the proposed unified team formation model. Therefore, we examine our experiments on the synthetic dataset that is similar to real-world collaboration networks.

5.1. Dataset

The synthetic social network is generated from LFR benchmark Lancichinetti, Fortunato, and Radicchi, 2008 with 200 nodes of 10,136 edges. We consider the nodes as experts and edges are the past collaborations. We assign random years from 2010 to 2019 to each collaboration because it requires to evaluate the dynamic nature of communication cost. We collect 50 skill sets as a string array. We then assign a random range of expert lists to each skill with their mastery level score. The emotional index and geographical proximity were generated randomly between 0 and 1. Direct trust value is assigned randomly, either 1 or -1. If any experts never collaborate in the past, we assign 0. To generate the combined trust score, we take into consideration three components: emotional index, direct trust, and skill similarity, we use the Trade-off solutions. Since both direct trust and emotional index are random values, they receive less importance in our model. Therefore, we assign $\alpha_1 = 0.25, \alpha_2 = 0.5$ and $\alpha_3 = 0.25$.

To evaluate the communication cost, we first generate a graph file that contains a detailed list of the connection between each expert with the weight of the connection. Two experts are connected in the network if they have interacted with one another previously, at least in two projects. The weights on edges are computed as:

$$W_{v_i,v_j} = 1 - \frac{|T_{v_i} \cap T_{v_j}|}{|T_{v_i} \cup T_{v_j}|} \tag{11}$$

where T_{v_i}(resp.T_{v_j}) is the set of collaboration by V_i (resp. V_j).

The shortest path distance between two experts is computed by an efficient indexing method called 2-hop cover index Cohen, Halperin, Kaplan, and Zwick, 2002. This indexing technique returns the value of the shortest path between any pair of experts in graphs almost instantly.

5.2. Experimental Setup

The MOCA for the TFP is implemented in Java (1.8), and the experiments are performed on an Intel(R) Core(TM) CPU (64 bits), Windows 10 machine with 16 GB of memory. The configuration details for MOCA are set to the following values: belief space probability = 0.8, crossover probability = 0.16 and mutation probability = 0.04. We test our experiments by changing the number of population and generation. For comparison, we implement the exhaustive search algorithm (or

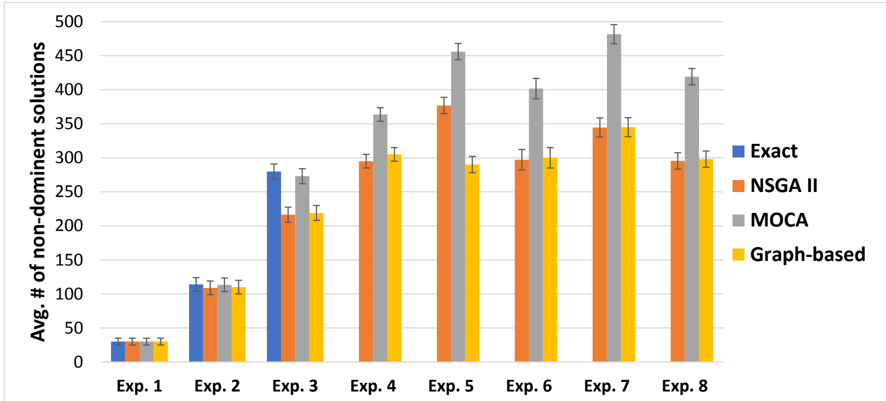

Fig. 5. Average number of non-dominated solutions for different algorithms and different experimental settings of Table 3. Exact algorithm did not termina.te after Exp. 3.

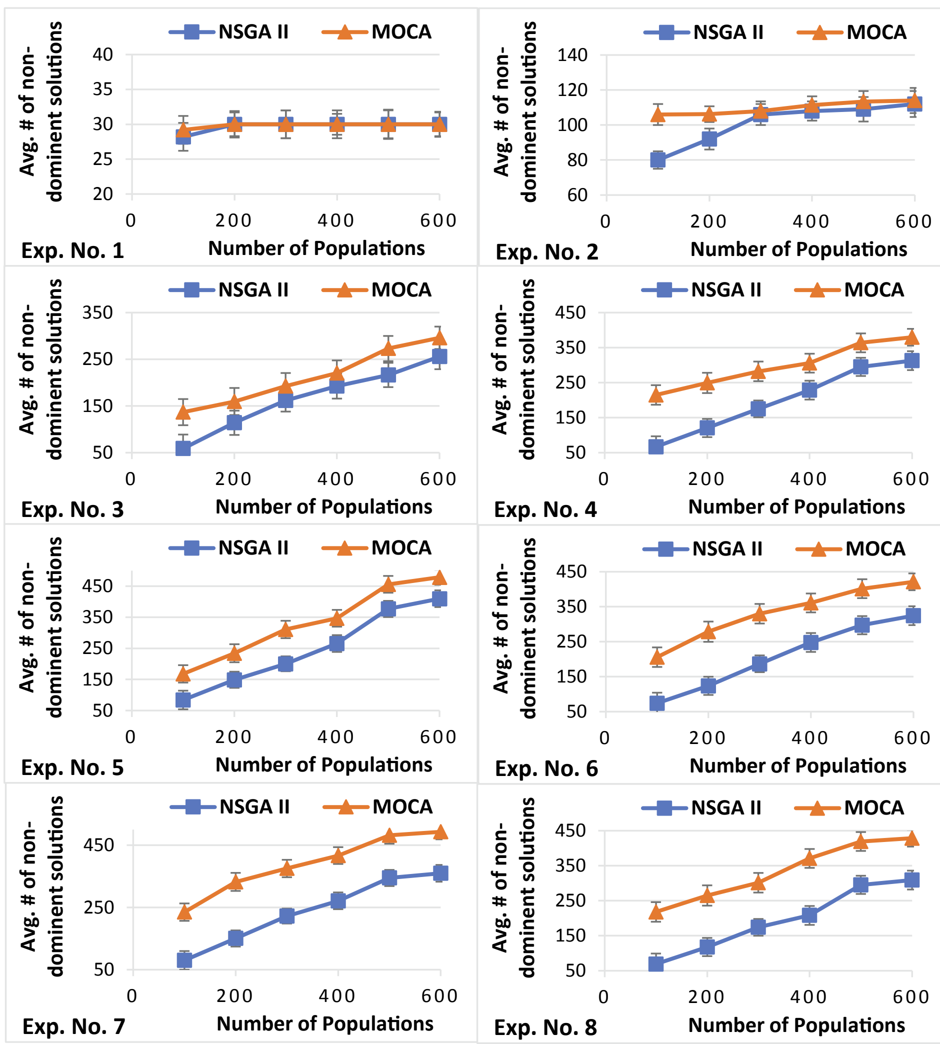

Fig. 6. Average number of non-dominated solutions when the number of population changes for NSGA II and MOCA for experimental settings of Table 3.

exact algorithm) and NSGA II. The exhaustive search algorithm iterates through the entire search space to generate every possible combination of teams. NSGA II is a basic fast non-dominated searching algorithm by Deb, Agrawal, Pratap, and Meyarivan, 2000. We can either set the skills and required expert-level manually or generate them randomly. We also compare our results with the results of Zihayat, Kargar, and An, 2014, which is a graph-based algorithm for finding Pareto-optimal teams. We adopt this system to cover the objectives of this work. We refer to this

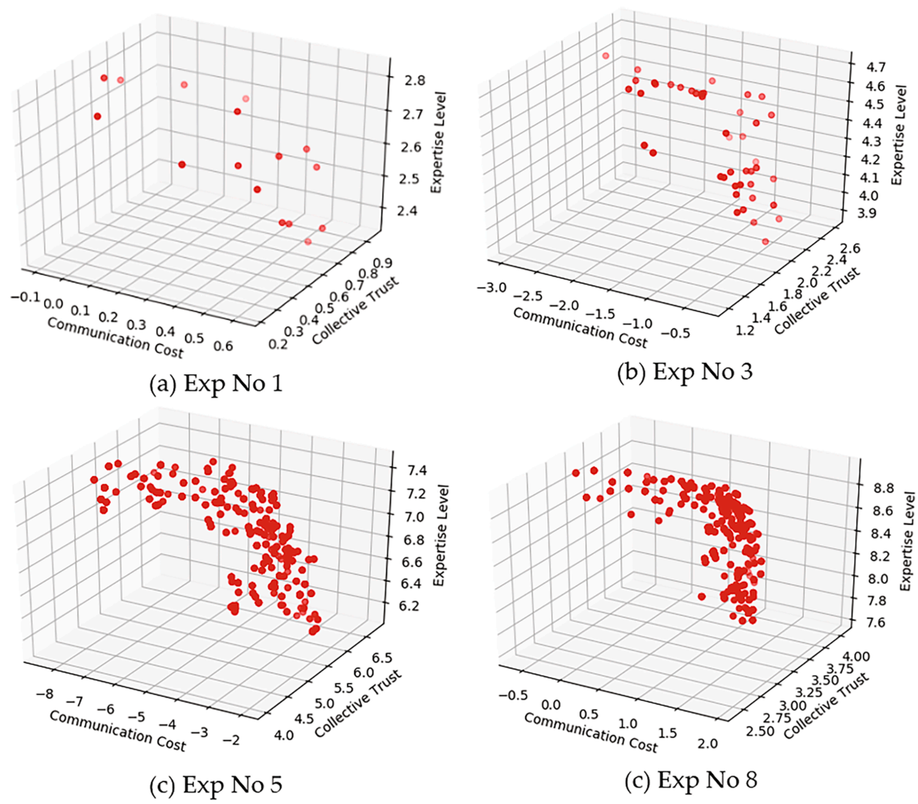

Fig. 7. Pareto non-dominated fronts from multiple test instances when we consider 3 objectives: Communication Cost, Expertise Level, and Collective Trust.

work as Graph-based in our experiments.

We create a benchmark table, as shown in Table 3, to perform the experiments. We test our framework with each settings of the table. For each setting, 100 instances are randomly generated and the average of the results are reported. For NSGA II and MOCA, default number of population and iterations are set to 500 and 50 respectively. The required skills and expertise level for these project *"Machine Learning","Python","Social Networks", "Artificial Intelligence","Statistics","Project Management","Big Data", "Data Mining"* are set to $\{0.7, 0.9, 0.8, 0.5, 0.7, 0.6, 0.7, 0.5\}$.

5.3. Effectiveness Results

We first show the average number of non-dominated solutions for different algorithms and different experimental settings of Table 3. This is presented in Fig. 5. Note that the Exact algorithm did not terminate after the third experiment. This result suggest that MOCA outperforms NSGA II and Graph-based, and its output is close to Exact.

Next, we study the effect of changing the number of population on the outcome of NSGA II and MOCA. Fig. 6 shows the average number of non-dominated solutions when the number of population changes for NSGA II and MOCA. Clearly, MOCA outperforms NSGA II. Also, the higher value of population results in higher number of non-dominated solutions. The outcome of MOCA and NSGA II becomes stable when the numbe of population is equal to 500, and this is why we set the default number of population to 500 in our experiments.

To check the effectiveness of the expertise level, we observe the solutions with and without the expertise level of the project requirement. For the first case, our framework always searched within the qualified expert list for the experts that satisfy the required skill, as well as the level of expertise needed. However, in the second case, the search space is the list of experts who satisfy the required skill. The average non-dominated sorting provides a totally different set of teams for both cases. At the same time, if the expertise level of any required skill is not satisfied with the available expert's list, our model does not form any group until it meets the expertise level.

Any given solution $\mathscr{V}'$ can be evaluated through multiple criteria, such as shortest path distance, diameter distance, minimum spanning tree, and the combination of various objectives. We then run our model by turning off some objectives and observe the non-dominated solutions. We randomly plot a few instances from our benchmark to visualize the Pareto front in Fig. 7 and Fig. 8.

We compare MOCA solutions with those obtained by NSGA II, Graph-based and exhaustive algorithms over various criteria.

- DCC($\mathscr{V}'$)↓- Dynamic shortest path distance
- CC-Dia($\mathscr{V}'$)↓ - Diameter distance
- GD($\mathscr{V}'$)↓- Geological proximity of the team
- CT($\mathscr{V}'$)↑- Trust value of the team
- EL($\mathscr{V}'$)↑- Expertise level of the team

Some of the above criteria need to be maximized, while others need to be minimized in order to discover successful teams. Apart from the multi objective optimization, we also want to know how our proposed algorithm perform when the goal is to optimize only one objective (and we compare our results with other works when they also optimize only one of the objectives). Table 4 presents this result. Each column represents one of the the above criteria, and the value obtained by each algorithm when the algorithm is set to only optimize that particular objective. For MOCA and NSGA II, each experiment is conducted with 500 number of populations and 50 number of iterations. When the number of skills and the number of projects increases, exhaustive algorithm is not terminated. At the same time, our model outperforms NSGA II and Graph-based.

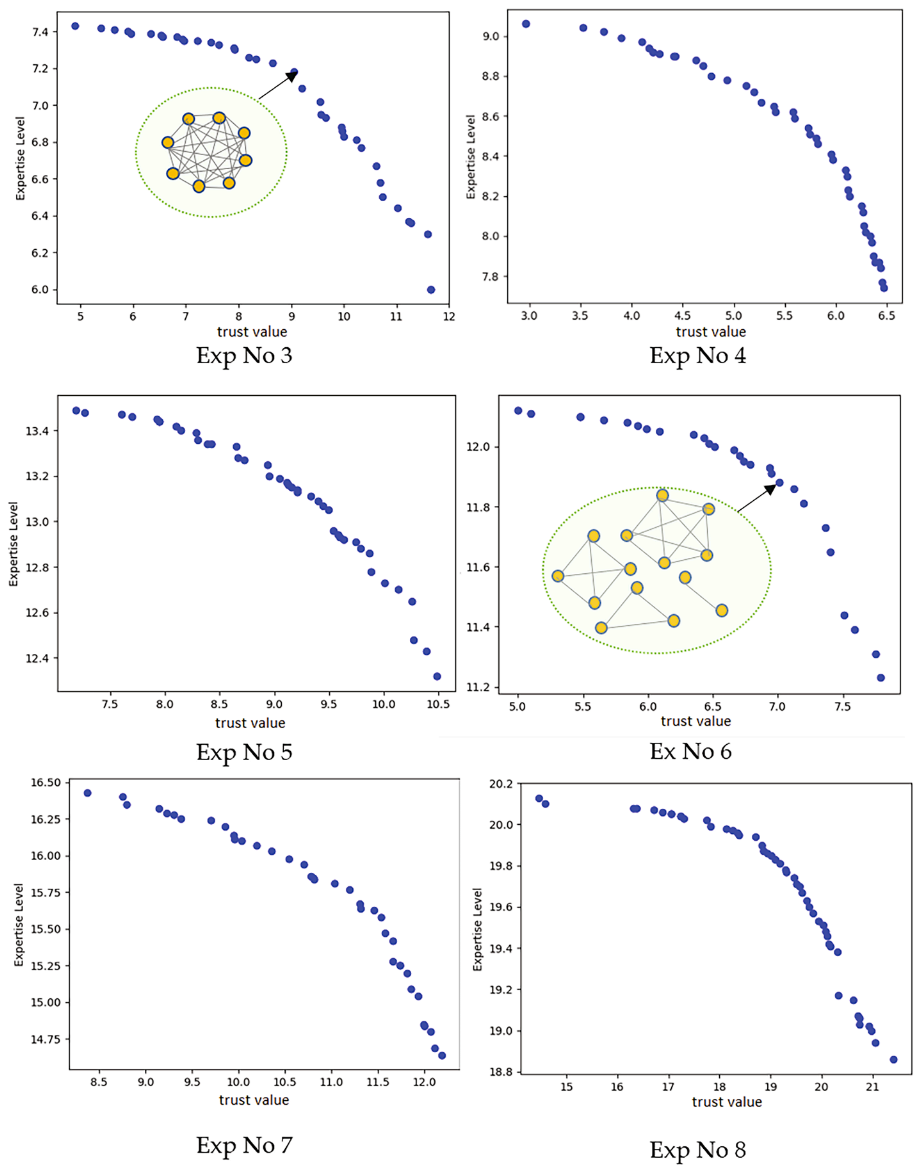

Fig. 8. Pareto non-dominated fronts from multiple test instances when we consider 2 objectives: Trust Value and Expertise Level and the topology of some of benchmark's instances.

5.4. Run Time and Scalability

In this section, we discuss the run time of the algorithms in different scenarios. The run time for each algorithm is shown in Fig. 9. In experiment No. 1, the run time of the exhaustive search was shorter than the other algorithms. It then increases suddenly for other instances. Eventually, with larger instances (after the third experiment), the exact algorithm did not terminate. As shown in Fig. 9, MOCA, and NSGA II have almost the same run time. However, MOCA takes a bit longer than NSGA II because MOCA needs to perform extra operations to extract knowledge and pass that to the next generation. In terms of the number of iterations, MOCA begins to converge sooner than NSGA II. Graph-based is slower than both MOCA and NSGA II since it needs to perform expensive graph operations to form the answers. The results of the scalability study of our proposed algorithm is shown in Fig. 10. As the results suggest, by increasing the size of the graph, the run time of MOCA increases linearly for all eight experimental settings.

6. Conclusion

In this paper, we proposed a unified framework for the TFP in social networks using a multi-objective formulation that optimizes the communication cost, team skill mastery level, the collective trust score, and geographical proximity. We introduced two objective functions, the dynamic communication cost, and expertise level that consider the temporal behavior of the social networks.

Moreover, we discussed the importance of integrating the emotional intelligence index within the TFP. In addition to this, we evaluated the trust score using various parameters such as EI, profile similarity and direct trust score. We introduced a new formula for the profile similarity based on similar skills. We solved this problem by using the Multi-Objective Cultural Algorithm (MOCA) framework for which normative and topological knowledge are extracted to generate the next population.

The experimental evaluation of our method over different tasks shows a diverse set of competitive solutions from four objectives. MOCA

Table 4
Comparison results of solutions obtained for each experiment instances of Table 3 over various criteria from MOCA, NSGA II, Graph-based and Exhaustive algorithm. Note that the results of this table are obtained when the goal of each algorithm is set to optimize only one objective.

Algorithms	DCC($\mathscr{V}'$)	CC-Dia($\mathscr{V}'$)	GD($\mathscr{V}'$)	CT($\mathscr{V}'$)	EL($\mathscr{V}'$)
No 1					
MOCA	**0.3602**	**0.1493**	**2.5932**	**0.8601**	**2.7613**
NSGA II	**0.3602**	**0.1493**	**2.5932**	**0.8601**	**2.7613**
Graph-based	**0.3602**	**0.1493**	**2.5932**	**0.8601**	**2.7613**
Exhaustive	**0.3602**	**0.1493**	**2.5932**	**0.8601**	**2.7613**
No 2					
MOCA	**1.4894**	**0.2479**	**7.0160**	**2.3441**	**4.5312**
NSGA II	**1.4894**	**0.2479**	**7.0160**	**2.3441**	**4.5312**
Graph-based	**1.4894**	**0.2479**	**7.0160**	**2.3441**	**4.5312**
Exhaustive	**1.4894**	**0.2479**	**7.0160**	**2.3441**	**4.5312**
No3					
MOCA	**1.2918**	**0.1116**	**21.6713**	**8.6448**	**7.2634**
NSGA II	2.8179	0.2974	26.4526	6.8821	**7.2634**
Graph-based	2.9874	0.3117	27.1182	7.6208	**7.2634**
Exhaustive	**1.2918**	**0.1116**	**21.6713**	**8.6448**	**7.2634**
No 4					
MOCA	**1.3388**	**0.3157**	**11.2794**	**3.9831**	**9.2343**
NSGA II	1.4842	0.4592	11.8213	3.0438	8.5809
Graph-based	1.6204	0.4872	12.1397	3.3527	8.7841
Exhaustive	n/a	n/a	n/a	n/a	n/a
No 5					
MOCA	**2.4063**	**0.4614**	**21.7146**	**7.8237**	**13.1715**
NSGA II	3.5522	0.5539	24.5103	7.7308	12.3911
Graph-based	3.6392	0.6327	25.9523	7.7405	12.6308
Exhaustive	n/a	n/a	n/a	n/a	n/a
No 6					
MOCA	**1.1418**	**0.2694**	**14.8037**	**5.6907**	**12.9526**
NSGA II	2.0421	0.5787	16.8631	5.0422	11.5140
Graph-based	2.1257	0.6024	17.2503	5.3011	12.0523
Exhaustive	n/a	n/a	n/a	n/a	n/a
No 7					
MOCA	**2.4105**	**0.4884**	**25.6325**	**8.8307**	**16.2816**
NSGA II	3.6805	0.7177	27.1238	7.6624	14.9901
Graph-based	3.8423	0.8413	29.0614	8.1008	15.0027
Exhaustive	n/a	n/a	n/a	n/a	n/a
No 8					
MOCA	**4.9812**	**0.6106**	**36.7541**	**14.9241**	**20.5816**
Graph-based	7.5	0.9375	45.3054	13.2304	18.8613
NSGA II	7.8521	0.9873	47.8203	13.953	19.1038
Exhaustive	n/a	n/a	n/a	n/a	n/a

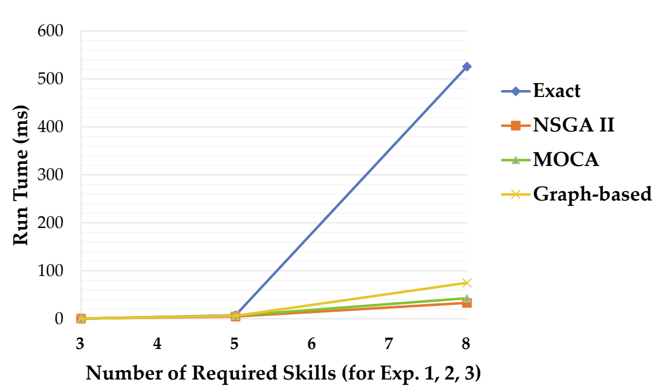

Fig. 9. The run time of different algorithms to run Exp. 1, Exp. 2, and Exp. 3.

was compared with other algorithms, NSGA II, Graph-based and Exhaustive method. MOCA outperformed NSGA II and Graph-based, and delivered accurate solutions that closely resembled the solutions as per generated by the Exhaustive search.

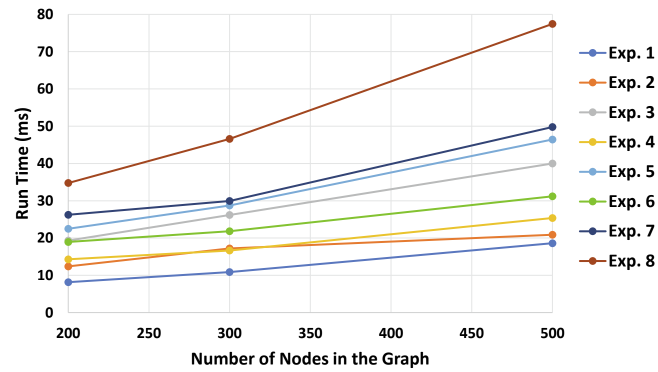

Fig. 10. Scalability of our proposed algorithm (MOCA) when the size of the graph changes.

CRediT authorship contribution statement

Conceptualization, Methodology, Validation, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to thank the anonymous reviewers for their comments that greatly improved the quality and presentation of this manuscript.

References

Abazari, F., Analoui, M., Takabi, H., & Fu, S. (2019). Mows: multi-objective workflow scheduling in cloud computing based on heuristic algorithm. *Simulation Modelling Practice and Theory, 93*, 119–132.

Ahn, J., DeAngelis, D., & Barber, S. (2007). Attitude driven team formation using multi-dimensional trust. In *2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT'07* pp. 229–235). IEEE.

Alves, V. S., Campello, R. J., & Hruschka, E. R. (2006). Towards a fast evolutionary algorithm for clustering. In *2006 IEEE International Conference on Evolutionary Computation* (pp. 1776–1783). IEEE.

Anagnostopoulos, A., Becchetti, L., Castillo, C., Gionis, A., & Leonardi, S. (2012). Online team formation in social networks. In Proceedings of the 21st international conference on World Wide Web (pp. 839–848). ACM.

Ashenagar, B., Eghlidi, N. F., Afshar, A., & Hamzeh, A. (2015). Team formation in social networks based on local distance metric. In *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)* (pp. 946–952). IEEE.

Awal, G. K., & Bharadwaj, K. (2014). Team formation in social networks based on collective intelligence–an evolutionary approach. *Applied intelligence, 41*, 627–648.

Bali, K. K., Ong, Y.-S., Gupta, A., & Tan, P. S. (2019). Multifactorial evolutionary algorithm with online transfer parameter estimation: Mfea-ii. *IEEE Transactions on Evolutionary Computation, 24*, 69–83.

Basiri, J., Taghiyareh, F., & Ghorbani, A. (2017). Collaborative team formation using brain drain optimization: a practical and effective solution. *World Wide Web, 20*, 1385–1407.

Bliss, C. A., Frank, M. R., Danforth, C. M., & Dodds, P. S. (2014). An evolutionary algorithm approach to link prediction in dynamic social networks. *Journal of Computational Science, 5*, 750–764.

Bozzon, A., Brambilla, M., Ceri, S., Silvestri, M., & Vesci, G. (2013). Choosing the right crowd: expert finding in social networks. In Proceedings of the 16th International Conference on Extending Database Technology (pp. 637–648). ACM.

Cai, X., Mei, Z., & Fan, Z. (2018). A decomposition-based many-objective evolutionary algorithm with two types of adjustments for direction vectors. *IEEE Transactions on Cybernetics, 48*, 2335–2348.

Cascio, W. F. (2000). Managing a virtual workplace. *Academy of Management Perspectives, 14*, 81–90.

Chen, L., Ye, Y., Zheng, A., Xie, F., Zheng, Z., & Lyu, M. R. (2019). Incorporating geographical location for team formation in social coding sites. World Wide Web, (pp. 1–22).

Chen, W., Yang, J., & Yu, Y. (2017). Analysis on communication cost and team performance in team formation problem. In *International Conference on Collaborative Computing: Networking, Applications and Worksharing* (pp. 435–443). Springer.

Cohen, E., Halperin, E., Kaplan, H., & Zwick, U. (2002). Reachability and Distance Queries via 2-hop Labels. In SODA (pp. 937–946).

Coniglio, S., Furini, F., & San Segundo, P. (2020). A new combinatorial branch-and-bound algorithm for the knapsack problem with conflicts. *European Journal of Operational Research*.

Cooper, R. K. (1997). Applying emotional intelligence in the workplace. *Training & development, 51*, 31–39.

Crawford, C., Rahaman, Z., & Sen, S. (2016). Evaluating the efficiency of robust team formation algorithms. In *International Conference on Autonomous Agents and Multiagent Systems* (pp. 14–29). Springer.

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In *International conference on parallel problem solving from nature* (pp. 849–858). Springer.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation, 6*, 182–197.

Dong, N., & Dai, C. (2019). An improvement decomposition-based multi-objective evolutionary algorithm using multi-search strategy. *Knowledge-Based Systems, 163*, 572–580.

Dong-Huynha, T., Jennings, N., & Shadbolt, N. (2004). Fire: An integrated trust and reputation model for open multi-agent systems. In ECAI 2004: 16th European Conference on Artificial Intelligence, August 22–27, 2004, Valencia, Spain: including Prestigious Applicants [sic] of Intelligent Systems (PAIS 2004): proceedings (p. 18). volume 110.

Dorn, C., & Dustdar, S. (2010). Composing near-optimal expert teams: a trade-off between skills and connectivity. In *OTM Confederated International Conferences On the Move to Meaningful Internet Systems* (pp. 472–489). Springer.

Dutta, D., Sil, J., & Dutta, P. (2019). Automatic clustering by multi-objective genetic algorithm with numeric and categorical features. *Expert Systems with Applications, 137*, 357–379.

Ebrahimi, N., Guda, T., Alamaniotis, M., Miserlis, D., & Jafari, A. (2020). Design optimization of a novel networked electromagnetic soft actuators system based on branch and bound algorithm. *IEEE Access, 8*, 119324–119335.

Farhadi, F., Sorkhi, M., Hashemi, S., & Hamzeh, A. (2011). An effective expert team formation in social networks based on skill grading. In *2011 IEEE 11th International Conference on Data Mining Workshops* (pp. 366–372). IEEE.

Feng, Y., Chen, H., Li, T., & Luo, C. (2020). A novel community detection method based on whale optimization algorithm with evolutionary population. Applied Intelligence, (pp. 1–20).

Gajewar, A., & Das Sarma, A. (2012). Multi-skill collaborative teams based on densest subgraphs. In Proceedings of the 2012 SIAM International Conference on Data Mining (pp. 165–176). SIAM.

Gil-Gala, F. J., Mencıa, C., Sierra, M. R., & Varela, R. (2020). Exhaustive search of priority rules for on-line scheduling. In *European Conference on Artificial Intelligence (ECAI)*.

Golshan, B., Lappas, T., & Terzi, E. (2014). Profit-maximizing cluster hires. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 1196–1205). ACM.

Gong, D., Han, Y., & Sun, J. (2018). A novel hybrid multi-objective artificial bee colony algorithm for blocking lot-streaming flow shop scheduling problems. *Knowledge-Based Systems, 148*, 115–130.

Gong, D., Sun, J., & Ji, X. (2013). Evolutionary algorithms with preference polyhedron for interval multi-objective optimization problems. *Information Sciences, 233*, 141–161.

Guerrero, M., Montoya, F. G., Baños, R., Alcayde, A., & Gil, C. (2017). Adaptive community detection in complex networks using genetic algorithms. *Neurocomputing, 266*, 101–113.

Gutiérrez, J. H., Astudillo, C. A., Ballesteros-Pérez, P., Mora-Melià, D., & Candia-Véjar, A. (2016). The multiple team formation problem using sociometry. *Computers & Operations Research, 75*, 150–162.

Han, Y., Gong, D., Jin, Y., & Pan, Q. (2019). Evolutionary multiobjective blocking lot-streaming flow shop scheduling with machine breakdowns. *IEEE Transactions on Cybernetics, 49*, 184–197.

Han, Y., Wan, Y., Chen, L., Xu, G., & Wu, J. (2017). Exploiting geographical location for team formation in social coding sites. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 499–510). Springer.

Howells, J. R. (2002). Tacit knowledge, innovation and economic geography. *Urban studies, 39*, 871–884.

Juárez, J., & Brizuela, C. A. (2018). A multi-objective formulation of the team formation problem in social networks: preliminary results. In Proceedings of the Genetic and Evolutionary Computation Conference (pp. 261–268). ACM.

Kargar, M., & An, A. (2011a). 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). ACM.

Kargar, M., & An, A. (2011). Teamexp: Top-k team formation in social networks. In *2011 IEEE 11th International Conference on Data Mining Workshops* (pp. 1231–1234). IEEE.

Kargar, M., An, A., & Zihayat, M. (2012). Efficient bi-objective team formation in social networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 483–498). Springer.

Kargar, M., Zihayat, M., & An, A. (2013). 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.

Kosti, M. V., Feldt, R., & Angelis, L. (2014). Personality, emotional intelligence and work preferences in software engineering: An empirical study. *Information and Software Technology, 56*, 973–990.

Kumar, B. (2020). Comparative performance evaluation of greedy algorithms for speech enhancement system. Fluctuation and Noise Letters, (p. 2150017).

Lancichinetti, A., Fortunato, S., & Radicchi, F. (2008). Benchmark graphs for testing community detection algorithms. *Physical review E, 78*, Article 046110.

Lappas, T., Liu, L., & Terzi, E. (2009). Finding a Team of Experts in Social Networks. In KDD (pp. 467–476).

Largillier, T., & Vassileva, J. (2012). Using collective trust for group formation. In *International Conference on Collaboration and Technology* (pp. 137–144). Springer.

Lazarsfeld, P. F., Merton, R. K., et al. (1954). Friendship as a social process: A substantive and methodological analysis. *Freedom and control in modern society, 18*, 18–66.

Lee, C., & Wong, C.-S. (2017). The effect of team emotional intelligence on team process and effectiveness. *Journal of Management & Organization*, 1–16.

Lescano, G., Costaguta, R., & Amandi, A. (2016). Genetic algorithm for automatic group formation considering student's learning styles. In *2016 8th Euro American Conference on Telematics and Information Systems (EATIS)* (pp. 1–8). IEEE.

Li, C.-T., Huang, M.-Y., & Yan, R. (2018). Team formation with influence maximization for influential event organization on social networks. *World Wide Web, 21*, 939–959.

Li, C.-T., & Shan, M.-K. (2010). Team formation for generalized tasks in expertise social networks. In *2010 IEEE Second International Conference on Social Computing* (pp. 9–16). IEEE.

Li, C.-T., Shan, M.-K., & Lin, S.-D. (2015). On team formation with expertise query in collaborative social networks. *Knowledge and Information Systems, 42*, 441–463.

Liu, C., Liu, J., & Jiang, Z. (2014). A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks. *IEEE transactions on cybernetics, 44*, 2274–2287.

Luo, J., Yang, Y., Li, X., Liu, Q., Chen, M., & Gao, K. (2018). A decomposition-based multi-objective evolutionary algorithm with quality indicator. *Swarm and evolutionary computation, 39*, 339–355.

Majumder, A., Datta, S., & Naidu, K. (2012). Capacitated team formation problem on social networks. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 1005–1013). ACM.

Manne, J. R. (2019). *Swarm intelligence for multi-objective optimization in engineering design. In Advanced Methodologies and Technologies in Artificial Intelligence, Computer Simulation, and Human-Computer Interaction* (pp. 180–194). IGI Global.

McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual review of sociology, 27*, 415–444.

Mercorio, F., Mezzanzanica, M., Moscato, V., Picariello, A., & Sperli, G. (2019). Dico: a graph-db framework for community detection on big scholarly data. *IEEE Transactions on Emerging Topics Computing*.

Miranda, P. B., Mello, R. F., & Nascimento, A. C. (2020). A multi-objective optimization approach for the group formation problem. Expert Systems with Applications, (p. 113828).

Neshati, M., Hashemi, S. H., & Beigy, H. (2014). Expertise finding in bibliographic network: Topic dominance learning approach. *IEEE transactions on cybernetics, 44*, 2646–2657.

Niveditha, M., Swetha, G., Poornima, U., & Senthilkumar, R. (2017). A genetic approach for tri-objective optimization in team formation. In *2016 Eighth International Conference on Advanced Computing (ICoAC)* (pp. 123–130). IEEE.

Pavlov, M., & Ichise, R. (2007). Finding experts by link prediction in co-authorship networks. *FEWS, 290*, 42–55.

Pérez-Toledano, M.Á., Rodriguez, F. J., García-Rubio, J., & Ibañez, S. J. (2019). Players selection for basketball teams, through performance index rating, using multiobjective evolutionary algorithms. *PloS one, 14*, Article e0221258.

Ponds, R., Van Oort, F., & Frenken, K. (2007). The geographical and institutional proximity of research collaboration. *Papers in regional science, 86*, 423–443.

Pourasghar, B., Izadkhah, H., Isazadeh, A., & Lotfi, S. (2020). A graph-based clustering algorithm for software systems modularization. Information and Software Technology, (p. 106469).

Rangapuram, S. S., Bühler, T., & Hein, M. (2013). Towards realistic team formation in social networks based on densest subgraphs. In Proceedings of the 22nd international conference on World Wide Web (pp. 1077–1088). ACM.

Reynolds, R. G. (1994). An introduction to cultural algorithms. In *Proceedings of the third annual conference on evolutionary programming* (pp. 131–139). World Scientific.

Ribas, I., Companys, R., & Tort-Martorell, X. (2019). An iterated greedy algorithm for solving the total tardiness parallel blocking flow shop scheduling problem. *Expert Systems with Applications, 121*, 347–361.

Richardson, M., Agrawal, R., & Domingos, P. (2003). Trust management for the semantic web. In *International semantic Web conference* (pp. 351–368). Springer.

Selvarajah, K., Bhullar, A., Kobti, Z., & Kargar, M. (2018). Wscan-tfp: weighted scan clustering algorithm for team formation problem in social network. In *The Thirty-First International Flairs Conference*.

Selvarajah, K., Kobti, Z., & Kargar, M. (2019). A cultural algorithm for determining similarity values between users in recommender systems. In *International Conference on the Applications of Evolutionary Computation (Part of EvoStar)* (pp. 270–283). Springer.

Selvarajah, K., Zadeh, P. M., Kobti, Z., Kargar, M., Ishraque, M. T., & Pfaff, K. (2018). Team formation in community-based palliative care. In *2018 Innovations in Intelligent Systems and Applications (INISTA)* (pp. 1–7). IEEE.

Selvarajaha, K., Zadeha, P. M., Kargarb, M., & Kobtia, Z. (2019). Identifying a team of experts in social networks using a cultural algorithm. *Procedia Computer Science, 151*, 477–484.

Sherchan, W., Nepal, S., & Paris, C. (2013). A survey of trust in social networks. *ACM Computing Surveys (CSUR), 45*, 47.

Sun, J., Miao, Z., Gong, D., Zeng, X., Li, J., & Wang, G. (2020). Interval multiobjective optimization with memetic algorithms. *IEEE Transactions on Cybernetics, 50*, 3444–3457.

Wang, J., & Li, D. (2019). Task scheduling based on a hybrid heuristic algorithm for smart production line with fog computing. *Sensors, 19*, 1023.

Wang, X., Zhao, Z., & Ng, W. (2015). A comparative study of team formation in social networks. In *International conference on database systems for advanced applications* (pp. 389–404). Springer.

Yang, Y., & Hu, H. (2013). Team formation with time limit in social networks. In *Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)* (pp. 1590–1594). IEEE.

Zadeh, P. M., & Kobti, Z. (2015). A multi-population cultural algorithm for community detection in social networks. In ANT/SEIT (pp. 342–349).

Zadeh, P. M., & Kobti, Z. (2016). In In FoIKS (Ed.), *A knowledge based framework for link prediction in social networks* (pp. 255–268). Springer.

Zhan, J., & Fang, X. (2010). A computational trust framework for social computing (a position paper for panel discussion on social computing foundations). In *2010 IEEE Second International Conference on Social Computing* (pp. 264–269). IEEE.

Zhang, J., Yu, P. S., & Lv, Y. (2017). Enterprise employee training via project team formation. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (pp. 3–12). ACM.

Zhang, Y., Gong, D., Sun, J., & Qu, B. (2018). A decomposition-based archiving approach for multi-objective evolutionary optimization. *Information Sciences, 430*, 397–413.

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

Zhu, H., Chen, E., Xiong, H., Cao, H., & Tian, J. (2014). Ranking user authority with relevant knowledge categories for expert finding. *World Wide Web, 17*, 1081–1107.

Ziegler, C.-N., & Golbeck, J. (2007). Investigating interactions of trust and interest similarity. *Decision support systems, 43*, 460–475.

Zihayat, M., Kargar, M., & An, A. (2014). Two-phase pareto set discovery for three-objective team formation. In *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)* (pp. 304–311). IEEE.