Exploiting Geographical Location for Team Formation in Social Coding Sites

Yuqiang Han[1], Yao Wan[1,3], Liang Chen[2], Guandong Xu[3], and Jian Wu[1]

[1] College of Computer Science & Technology, Zhejiang University, Hangzhou, China
[2] School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China
[3] Advanced Analytics Institute, University of Technology, Sydney, Australia
{hyq2015, wanyao, wujian2000}@zju.edu.cn, jasonclx@gmail.com, guandong.xu@uts.edu.au

Abstract. Social coding sites (SCSs) such as GitHub and BitBucket are collaborative platforms where developers from different background (e.g., culture, language, location, skills) form a team to contribute to a shared project collaboratively. One essential task of such collaborative development is how to form a optimal team where each member makes his/her greatest contribution, which may have a great effect on the efficiency of collaboration. To the best of knowledge, all existing related works model the team formation problem as minimizing the communication cost among developers or taking the workload of individuals into account, ignoring the impact of geographical location of each developer. In this paper, we aims to exploit the geographical proximity factor to improve the performance of team formation in social coding sites. Specifically, we incorporate the communication cost and geographical proximity into a unified objective function and propose a genetic algorithm to optimize it. Comprehensive experiments on a real-world dataset (e.g., GitHub) demonstrate the performance of the proposed model with the comparison of some state-of-the-art ones.

Keywords: Team formation, geographical location, social coding sites, genetic algorithm

1 Introduction

With the prevalence of social networks in the world, social coding sites (SCSs) such as GitHub[4] and BitBucket[5] are changing software development toward a more collaborative manner by the way of integrating social media functionality and distributed version control tools. In SCSs, developers with different background (e.g., culture, language, location, skills) form a team and work collaboratively to contribute to a project, dramatically enhancing the efficiency of development when compared with individual development. One essential

[4] https://github.com
[5] https://bitbucket.org

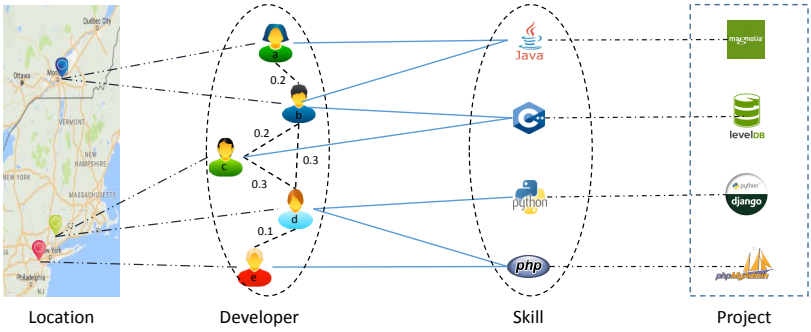


Fig. 1. Schema of developers' profiles and their corresponding skills in GitHub. The left part represents the geographical information of developers; the middle part represents a heterogeneous network among users and skills which can be constructed based on the collaborative development records of developers; the right part represents that each skill of developers can be extracted from his/her contributed projects in GitHub.

task of collaborative development is how to form a optimal team where each member makes his/her greatest contribution, which may have a great effect on the efficiency of collaboration. We called this kind of task as team formation problem.

There have been several related works [2,7,10,11,13] that try to address the team formation problem from different perspectives. In [10,13], the authors define several kinds of communication cost among teams and try to minimize the cost function. For example, the communication cost can be defined as the longest shortest path between any experts in team, the weight cost of the minimum spanning tree for subgraph, and the sum of all shortest paths between any two experts in team. This line of work optimize the team form the perspective of network structure of team. On the other hand, several works [2,7,11] take other factors such as the skill level, workload of individuals into account. The authors in these works aim to balance the workload of performing the tasks among people in the fairest possible way, on the condition that the required skills are covered. To the best of our knowledge, no existing work consider the geographical factor to boost the team formation performance especially in social coding sites. Where some works [17,18] demonstrate the importance of geographical proximity in some specific domain such as knowledge production and technological innovation. We believe that the geographical proximity may also affect the collaboration between developers in collaborative software development, and it is desirable to exploit the geographical proximity factor to improve the performance of team formation in social coding sites.

Based on this intuition, our paper proposes to integrate the conventional communication cost and geographical proximity for team formation in social coding sites such as GitHub. The challenges of our paper lies in two fold. a) How to encode the geographical information of developers into our model. In

our GitHub scenario, the developer declares his/her location attribute via a string. It is challengable to determine the impact of geographical information for team formation and then encode it mathematically (e.g., via calculating distance according to latitude and longitude or encoding it into time zone). b) How to incorporate the geographical information and communication cost into a unified objective function and solve the optimization problem. The optimization problem in team formation issue has been proven to be NP-hard, it is challengable to devise a heuristic approach to solve the optimization problem.

To achieve this goal, in this paper, we firstly define the team formation task as finding a team of developers that cover the required skills while minimizing both the communication cost and geographical proximity, given a collaboration network and a task with a set of required skills where each skill is associated with a specific number of developers. Then, we incorporate the communication cost and geographical proximity into a unified objective function and propose a genetic algorithm to optimize it. Furthermore, we conduct comprehensive experiments on a real-world dataset to verify the effectiveness of our proposed model. Figure 1 gives an overview of developers' profile with geographical information and their corresponding skills which can be extracted from their contributed projects in GitHub. We can also note that a heterogeneous network among users and skills can be constructed based on the collaboration between users (see Section 3).

The main contributions of this paper are summarized as follows:

- To the best of our knowledge, this work is the first attempt to improve the performance of team formation in social coding sites by taking both communication cost and geographical proximity into consideration.
- We incorporate the communication cost and geographical location cost into a unified objective function and propose a genetic algorithm to optimize it.
- We crawl 36,701 users and 3,532,453 projects from GitHub as a real-world dataset to evaluate the performance our approach. Comprehensive experiments show the effectiveness of our model with the comparison of other baseline models

Organization. The remainder of this paper is organized as follows. In Section 2, we survey some works related to this paper. Section 3 shows some preliminaries. Section 4 presents details of our proposed geographical location aware model for team formation in social coding sites. Section 5 describes the real-world dataset (e.g., GitHub) we use in our experiments. Experimental results and analysis are shown in Section 6. Finally, we conclude this paper and propose some future directions in Section 7.

2 Related Work

Team Formation. The team formation problem is majorly studied in the field of collaborative social networks since it has an effect on the efficiency of

collaboration. Lappas et al. [13] are the first to address the issue of social team formation by considering the communication costs for organizing a team. They also prove the team formation problem is NP-hard. Based on the perspective of communication cost, some variants have been derived. In [10], Kargar et al. improve the communication cost function based on the sum of distances and leader distance. Ashenagar et al. [3] devise a new method to determine the distance between pairs of experts. Besides considering the communication cost between users, some other factor such as the cost of individuals [11,12] and the workload balance of team are also considered [1,2]. Majumder et al. [15] account for capacity constraints in the team formation problem so that no user is overloaded by the assignment. Farhadi et al. [7,8] suggest a skill-grading method to measure for the skill level of experts. Y. Yang and H. Hu [19] propose a new cost model to solve team formation with limited time. Avradeep Bhowmik et al. [4] take the Submodularity method to find a team of experts by relaxing the requirement of skill. Li and Shan [14] generalize the team formation problem by associating each required skill with a designated number of experts. Although many other factors have been considered, the geographical location of developers in social coding sites is not been considered yet.

Geographical Location. Another line of reasearch which are related to our paper is on exploiting the geographical location. Lot of literature suggest that geographical proximity is playing an increasing important role in many domains in spite of rapid development in telecommunications technology. In [17], the authors demonstrate that the geographical proximity in the creation of economically-useful knowledge appeared to be becoming even more important. Soon et al. [18] analyze patent citations and found that in contemporary knowledge production and innovation the role for geographical proximity was increasing. Ponds et al. [16] analyze the role of geographical proximity for collaborative scientific research and confirmed its significance. Brocco et al. [5] propose two different ways to integrate location using spatial operations and utilize the location-based solution to support team composition in different computer gaming scenarios.

3 Preliminaries

We present the social coding network as an undirected graph $G = (V, E, w)$. Each vertex in V denoting an expert and the weight of each edge in E represents the communication cost between a pair of experts. We assume that (u, v) is an edge if developers u and v have participated in common projects before, and the weight of the edge is related to the fraction of projects they have worked on together, which is calculated by

$$w(u, v) = 1 - \frac{|N_u \cap N_v|}{|N_u \cup N_v|} \tag{1}$$

where N_u and N_v is the set of projects in which u and v are listed as contributors respectively. The communication cost is the sum of weights on the shortest path

between two developers in G. The lower the communication cost is, the more easily they can collaborate with each other. If two experts are not connected in G(directly or indirectly), the communication cost between them is ∞. Consider the social coding network in Figure 1, the communication costs between a and b, a and c are 0.2, 0.4, respectively. In the social coding network, each developer possess some skills such as programming languages. And each skill is related to some projects.

The geographical proximity is the distance between two regions, such as cities, countries and so on. It is related to the differences in culture, work habits of developers and so on. In order to quantify the geographical proximity between two developers, we extract the country of every developer, and define the geographical proximity between them as follows:

$$gp(u,v) = \begin{cases} 0, & \textit{If } u \textit{ and } v \textit{ in he same country} \\ 1. & \textit{Otherwise} \end{cases} \tag{2}$$

For example, in Figure 1, the geographical proximity between a and b, a and e are 0, 1 respectively.

Definition 1. ***(Team of Developers)*** *Given a social coding site and a project P with some requirement of skills(e.g. programming languages), a team of developers for P is a set of developers who can meet the requirement of P.*

4 Location-aware Model for Team Formation

In this section, we will model communication cost and geographical proximity, and then state the team formation problem followed by introducing the genetic algorithm based approach for solving the problem.

4.1 Model the Communication Cost

To evaluate the communication cost among the developers in a team T, we take the sum of communication costs among the selected developers of a team defined as follows, which is the same as [10].

Definition 2. ***(Sum of Communication Costs)*** *Given a social coding network G whose edges are weighted by the communication cost between two developers and a team T of developers from G, the sum of communication cost of T is defined as*

$$SCC(T) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} cc(e_i, e_j) \tag{3}$$

where $cc(e_i, e_j)$ is the communication cost of developer e_i and e_j (as defined earlier).

4.2 Model the Geographical Location

Based on the perspective of sum of communication cost, to measure the geographical proximity of the team of experts, we define the sum of geographical proximity of a team as follows:

Definition 3. ***(Sum of Geographical Proximity)*** *Given a team T of experts, where each having a location code, the sum of geographical proximity of T is defined as*

$$SGP(T) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} gp(e_i, e_j) \tag{4}$$

where $gp(e_i, e_j)$ is the geographical proximity between expert e_i and e_j which as defined above.

4.3 Objective Function

For finding a team of developers from a social coding network that minimize the sum of communication cost as well as the sum of geographical proximity, we combine the two objective functions into a single one to convert the bi-objective optimization problem into a single objective problem and define a new combined cost function as follows which is based on the linear combination of the sum of communication cost and sum of geographical proximity.

Definition 4. ***(Combined Cost Function)*** *Given a collaboration network and a trade-off λ between the sum of communication cost and sum of geographical proximity, we define the combined cost of the team T as*

$$ComCost(T) = (1 - \lambda) \times SCC(T) + \lambda \times SGP(T) \tag{5}$$

The parameter λ varying from 0 to 1 indicates the tradeoff between sum of communication and sum of geographical proximity.

Given the combined cost function, we now formally define the team formation problem in social coding networks as follows:

Team Formation by Minimizing the Combined Cost. Given a social coding network $G(V, E, w)$ where the developers are associated with specified skills, a project P with requirements of skills, the aim of team formation by minimizing the combined cost is to find a team $T \subseteq V$ so that each skill in P will be covered by the specified number of developers, each developer will cover and only cover one skill, and the combined cost $ComCost(T)$ defined in 4 among selected experts are as minimum as possible.

4.4 GA-based Optimization

Since the team formation by minimizing the combined cost is an NP-hard problem, we employ an genetic algorithm to find an optimal solution for the

team formation problem in the context of social coding networks. The details of GA-based model are presented in the following subsections.

Encoding. We consider each candidate team as a chromosome and each developer in the team as a gene. So each candidate team is a linear vector and composes of several partitions where each one represents a skill. An example of candidate team with four required skills is represented in Figure 2.

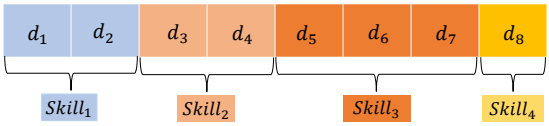


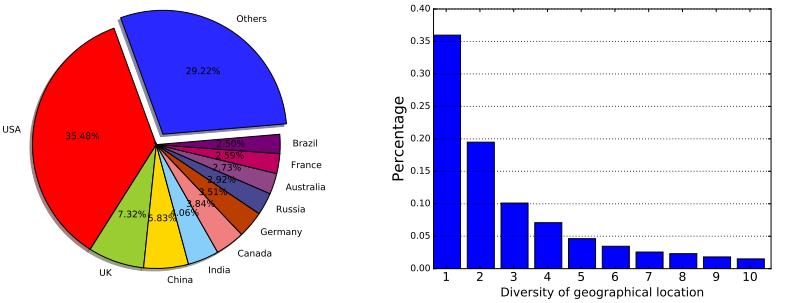
Fig. 2. An example of representation of the candidate team with four required skills

Initialize The random method to generating the initial population ensures a good level of genetic diversity in the population and thus prevents the premature convergence of the algorithm [6, 9]. So we take the random method to randomly generate the initial candidate teams fulfilling the requirement posed by the projects.

Genetic Operators. Crossover, mutation and selection are the three main types of genetic operators. They must work in conjunction with one another to ensure the success of the algorithm.

- **Crossover.** The crossover operator aims to preserve and combines the best characteristics of the parents to evolve better solutions [6, 9]. We have applied a two-point crossover here with the probability P_c to generate two new offspring solutions.
- **Mutation.** This operator is applied to the encoded solutions with the probability P_m to introduce genetic diversity into the population. In this paper, we have applied two types of mutation operators - substitution mutation and swap mutation. The substitution mutation operator involves the selection of a developer in a team with skill s_m, and replacing him with a developer at random from support set of s_m. Swap mutation operator randomly selects a developer from the team and swaps him with one in the team who covers the skill in his skill set at current.
- **Selection.** The new population at generation $k + 1$ is generated by the application of genetic operators at generation of k. We combine elitism and tournament to complete the selection, which means that the best teams in generation k are automatically transferred to the population of generation $k + 1$, and the rest teams will be chosen as the parents by the tournament method to generate new teams.

Sometimes, crossover and mutation operators may produce infeasible solutions. The reparation strategy is designed to ensure the new team is infeasible.



(a) Top 10 countries (b) Location diversity distribution

Fig. 3. An overview of geographical location distribution of GitHub developers. (a) Top 10 countries with the largest number of developers. (b) Distribution of location diversity distribution, considering the composition of teams.

Evaluation. We apply the opposite number of combined cost as the fitness function to simultaneously optimize the sum of communication cost and sum of geographical proximity.

5 Dataset

In our paper, we conduct experiments on a real-word dataset from GitHub, which is one of the most popular social coding sites and has gained much popularity among a large number of software developers around the world. In GitHub, users are encouraged to contribute to a share project collaboratively, which is coincided with our scenario of team formation.

GitHub publicizes its data via APIs. Crawling GitHub website by its API, we get 28,362,019 projects, 15,647,255 users and make out the relationships between them. We then filter out users who provide the geographical location information and obtain 36,701 users and 3,532,453 their contributed projects. Constructing the network by the way described in Section 3, we get 1,610,072 edges. Considering the programming languages of project as required skills, we obtain 273 distinct skills.

Figure 3 presents an overview of geographical location distribution of GitHub developers. Figure 3a lists the top 10 countries with the largest number of developers. From this figure, we observe that more than a third of developers are from the USA, accounting for the largest part. The following parts are developers from UK and China, which are also within our expectation. Figure 3b shows the distribution of location diversity distribution in terms of the number of countries the developers come from in composing a team. In this figure, we observe in most teams (nearly 55%), the developers come from no more than one or two countries. And the situation that members are from many different countries is uncommon. This phenomenon just verifies our intuition.

6 Experiments

6.1 Experimental Setup

For all experiments, we set the number of skill $k = 2$ and $\lambda = 0.5$. For GA algorithm, we set the population size as 200, the number of generation as 100, the crossover probability as 0.2 and runs for 10 iterations for each experiment.

Evaluation Metrics For evaluation, three evaluation metrics which are commonly adopted in conventional studies are used in this paper. The three evaluation metric are listed as follows:

- **Sum of Geographical Proximity.** This metric measures the geographical proximity of the team. It reveals how closely the developers of the team in terms of geographical location.
- **Sum of Communication Cost.** This metric measures the communication cost of the team. It reveals the efficiency of the communication between developers. It is also taken as an evaluation metric in some previous works.
- **Combined Cost.** This metric is the combination of sum communication cost and sum of geographical proximity.

Performance Comparison By following [11], we compare our proposed model against the following three baselines.

- **Random Algorithm.** Random algorithm randomly creates 1,000 teams and selects the one with the minimal combined cost for the required set of skills as the optimal team.
- **Approximation Rare Algorithm.** Approximation rare algorithm selects the skill with least supporters as the initial skill. Firstly, an expert with the initial skill is selected as a seed expert followed by an expert added with the minimum communication cost to the seed expert with each of other required skills into the team. Then, the team with the minimum costs is selected among the entire candidate teams.
- **Minimum Cost Contribution Rare Algorithm.** MCC-rare algorithm chooses an expert with the skill who has rarest supporters as the initial member of candidate team, and then adds a new team member by considering its communication cost in comparison to all current team members.

All the experiments in this paper are implemented with Python 2.7, and run on a computer with an 2.2 GHz Intel Core i7 CPU and 64 GB 1600 MHz DDR3 RAM, running Debian 7.0.

6.2 Experimental Results

Figure 4 shows the performance of different models on different metrics. From this figure, we have the following observations:

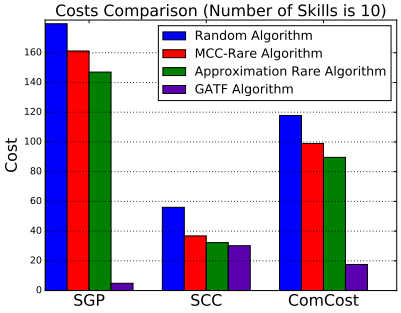


Fig. 4. Performance comparison.

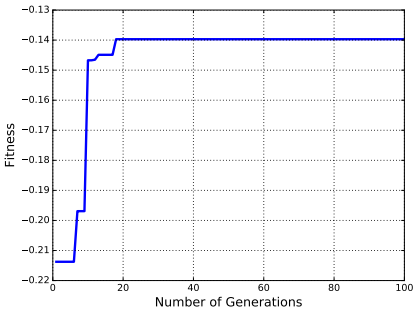
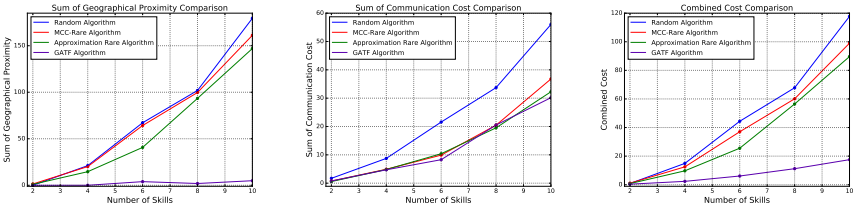


Fig. 5. Convergence of GA.

- On the sum of geographical proximity evaluation metric, the proposed GA-based model achieves better performance, random algorithm gets the worst. This is because the GA-based model considers the sum of geographical proximity during the process of finding a optimal team. Other three algorithms do not consider the geographical proximity factor.
- On the sum of communication cost evaluation metric, the proposed GA-based model also achieves better performance and random algorithm worst. This is because GA-based model has a larger search space while MCC-Rare algorithm and approximation rare algorithm has a smaller one. The random algorithm do not consider the sum of communication cost factor.
- On the combined cost evaluation metric, the proposed GA-based model also achieves better performance. This is because GA-based algorithm consider both the sum of graphical proximity and sum of communication cost. MCC-Rare algorithm and approximation rare algorithm consider communication cost only. The random algorithm only covers the basic requirements of projects, including neither geographical proximity nor communication cost.

6.3 Parameter Analysis

Impact of Skills Number In our model, the skills number controls the team size. To study the impact of skills number on the performance, we set skills number $k \subseteq \{2, 4, 6, 8, 10\}$. And for each k, we generate 10 random projects to take the average result. The experimental results are show in Figure 6. Figure 6(a) shows that all algorithms will get high sum of geographical proximity with the increasing of task number. But proposed GA-based model can always achieve better performance on sum of geographical proximity. The similar are Figure 6(b) and Figure 6(c), where all algorithms will get high sum of communication cost and combined cost with the increasing of skills number. Our proposed GA-based model always achieve better performance.



(a) SGP (b) SCC (c) ComCost

Fig. 6. Impact of skills number.

Impact of Iterations In our model, the iterations is directly affect the search result. To study the impact of iterations, we set the task number to 2 and generate 10 random projects to track the convergence of the algorithm. The experimental results are show in Figure 5. As we can see from the result, the proposed GA-based model can converge after 20 iterations when the task number is 2.

7 Conclusion and Future Work

In this paper, we exploit the geographical location of developers to boost the performance of team formation in social coding sites. We incorporate the communication cost and geographical proximity into a unified objective function and propose a genetic algorithm to optimize it. Experiments on a real-world dataset (e.g., GitHub) illustrate the effectiveness of our proposed approach.

In our future work, we plan to investigate the impact of social media on the performance of team formation. For example, we can also take the social network of developers in social media (e.g., Twitter) into consideration to boost the performance of team formation. Furthermore, we will exploit the interaction patterns for the accurate interpretation of link strength between developers.

References

1. Anagnostopoulos, A., Becchetti, L., Castillo, C., Gionis, A., Leonardi, S.: Power in unity: forming teams in large-scale community systems. In: Proceedings of the 19th ACM international conference on Information and knowledge management. pp. 599–608. ACM (2010)
2. 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. ACM (2012)
3. Ashenagar, B., Eghlidi, N.F., Afshar, A., Hamzeh, A.: Team formation in social networks based on local distance metric. In: Fuzzy Systems and Knowledge Discovery (FSKD), 2015 12th International Conference on. pp. 946–952. IEEE (2015)

4. Bhowmik, A., Borkar, V.S., Garg, D., Pallan, M.: Submodularity in team formation problem. In: SDM. pp. 893–901. SIAM (2014)
5. Brocco, M., Woerndl, W.: Location-based team recommendation in computer gaming scenarios. In: Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Querying and Mining Uncertain Spatio-Temporal Data. pp. 21–28. ACM (2011)
6. Eiben, A.E., Smith, J.E.: Introduction to evolutionary computing, vol. 53. Springer (2003)
7. Farhadi, F., Sorkhi, M., Hashemi, S., Hamzeh, A.: 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 (2011)
8. Farhadi, F., Sorkhi, M., Hashemi, S., Hamzeh, A.: An effective framework for fast expert mining in collaboration networks: a group-oriented and cost-based method. Journal of Computer Science and Technology 27(3), 577–590 (2012)
9. Golberg, D.E.: Genetic algorithms in search, optimization, and machine learning. Addion wesley 1989, 102 (1989)
10. 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. ACM (2011)
11. Kargar, M., An, A., Zihayat, M.: Efficient bi-objective team formation in social networks. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 483–498. Springer (2012)
12. Kargar, M., Zihayat, M., An, A.: Finding affordable and collaborative teams from a network of experts. In: Proceedings of the SIAM International Conference on Data Mining (SDM). pp. 587–595. SIAM (2013)
13. 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. ACM (2009)
14. Li, C.T., Shan, M.K.: Team formation for generalized tasks in expertise social networks. In: Social Computing (SocialCom), 2010 IEEE Second International Conference on. pp. 9–16. IEEE (2010)
15. Majumder, A., Datta, S., Naidu, K.: 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 (2012)
16. Ponds, R., Van Oort, F., Frenken, K.: The geographical and institutional proximity of research collaboration. Papers in regional science 86(3), 423–443 (2007)
17. Sonn, J.W., Storper, M.: The increasing importance of geographical proximity in technological innovation. What Do we Know about Innovation? in Honour of Keith Pavitt, Sussex, 13-15 November 2003 (2003)
18. Sonn, J.W., Storper, M.: The increasing importance of geographical proximity in knowledge production: an analysis of us patent citations, 1975–1997. Environment and Planning A 40(5), 1020–1039 (2008)
19. Yang, Y., Hu, H.: Team formation with time limit in social networks. In: Mechatronic Sciences, Electric Engineering and Computer (MEC), Proceedings 2013 International Conference on. pp. 1590–1594. IEEE (2013)