I.J. Information Technology and Computer Science, **2018, 5, 16-29**
Published Online May 2018 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2018.05.02

An Improved African Buffalo Optimization Algorithm for Collaborative Team Formation in Social Network

Walaa H. El-Ashmawi
Computer Science Department, Faculty of Computers and Informatics, Suez Canal University, Ismailia, 41522, Egypt
E-mail: w.hashmawi@ci.suez.edu.eg

Received: 20 January 2018; Accepted: 20 March 2018; Published: 08 May 2018

***Abstract*—**Collaborative team formation in a social network is an important aspect for solving a real-world problem that requires different expert skills to achieve it. In this paper, we propose an improved African Buffalo Optimization algorithm integrated with discrete crossover operator conjointly with swap sequence for efficient team formation whose members can assist in solving a given problem with minimum communication cost. The proposed algorithm is called Improved African Buffalo Optimization algorithm (IABO). In IABO, a new concept of swap sequence applied to improve the performance by generating better team members that cover all the required skills. To the best of our knowledge, this is the first work that considers the African Buffalo Optimization algorithm for collaborative team formation in a social network of experts. A set of experiments have been done on two popular real-world benchmark datasets (i.e., DBLP and Stack Overflow) to determine the efficiency of the proposed algorithm in team formation. The results demonstrate the effectiveness of the IABO algorithm in comparison with GA, PSO and standard African Buffalo Optimization algorithm (ABO).

***Index Terms*—**African Buffalo Optimization, Team Formation, Social Network, Swap Sequence, Discrete crossover.

I. Introduction

Formation of collaborative teams in social network plays a significant role in practical applications that require looking for experts with diverse skills to achieve a given task in a collaborative manner. These applications are ranging from project development in a social network to various collaborative tasks. The problem is how to form such collaborative team with minimum communication cost among team members. It is well known that this problem can be formulated as an NP-hard problem [1] and can be modeled as an optimization problem. Therefore, the goal is seeking to meta-heuristic algorithms to solve the team formation problem in which they have proven their effectiveness to solve other large and complex problems [2]. The authors in [3] considered the problem of team formation in a social network by finding a team of experts that cover the required skills as a subgraph in social network. In their work, they have proven that problem is very complex to be achieved.

Ref.[4] illustrated different swarm intelligent algorithms and their applications in varies areas such as Aerospace technology, Telecommunication Networks, Civil Engineering Design Multi-Objective, Image processing , Entertainment and Control Engineering.

One of the recent optimization meta-heuristic techniques is African Buffalo Optimization (ABO) which developed by Odili and Kahar in 2015 [5]. The ABO was inspired by the behavior of African buffalos, a species of wild cows known from their migration lifestyle for searching of lush pastures. The strength of ABO relies on solving the problem of convergence or stagnation or that suffer from using of many parameters that exist in many existing algorithms such as genetic algorithm and particle swarm optimization in order to reach a better solution. All the research applied on ABO tested on continuous optimization problems or salesman's problem. The authors in [6] solve the continuous optimization problems using 21 benchmark numerical test cases ranging from multimodal to uni-modal, separable to non-separable search spaces and compared the results with Genetic Algorithm (GA) and the Improved Genetic Algorithm (IGA) while in [7] studies the convergence Analysis of the African Buffalo Optimization algorithm in terms of the trade-off between exploration and exploitation. ABO used to solve the traveling salesman's problem as in [8,9] which explain the working of the algorithm to solve such problem and compared with hybrid algorithm and ant colony algorithm while in [10] compared the ABO against randomized insertion algorithm that achieved better results to the solution but has proven that the ABO is much faster. To the best of our knowledge, there is no work has been done to solve the team formation problem based on ABO algorithm and a little bit work based on other meta-heuristic algorithms.

Most of the existing work in team formation based on approximation algorithms [11-13] that considered different communication costs such as diameter and minimum spanning tree [1] or the sum of the distance from team leader [14]. Other works in [15-17] generalize the problem by assigning each skill to a specific number of experts while in [18] consider the maximum load of experts according to different tasks without taking into consideration the communication cost of team formation. In [19], the formation of team based on the available work time and set of skills that each expert has associated with associated with a skill level indicating his competence in this skill.

Although a minimal research work has been done based on meta-heuristic algorithms especially the well-known one's particle swarm optimization (PSO) and Genetic Algorithm [20] in team formation, they have been successfully applied in an optimization method as in [21-23] for many real-world applications. In [24] a genetic algorithm presented to solve the problem of assigning project supervisors to students. The authors in [25] presented a group formation method based on genetic algorithm and each groups' members is generated according to the students' programming skill. While the authors in [26] used a genetic algorithm in team formation on the bases of Belbin team role that categorized individuals in nine roles regarding their speciality and attitude toward team working. The authors in [27] proposed a mathematical framework for treating the team formation problem explicitly incorporating social structure among experts. They used an LK-TFP heuristic that performs variable-depth neighborhood search and compared their results with a standard genetic algorithm.

Therefore, the main objective of this research lies on proposing an improved ABO algorithm for forming a collaborative team of experts for task achievement with minimum communication cost among team members by using a Discrete Crossover operator (DC) [28] to guarantee that the whole population moving towards the global optimum conjointly with a new concept of swap sequence operator to improve the performance of generating a collaborative team of experts. The proposed algorithm is called an Improved African Buffalo Optimization algorithm (IABO).

The structure of the paper is as follows. In Section 2, we illustrate the collaborative team formation in a social network. Section 3 introduces the proposed improved algorithm. In Section 4, we discuss the experimental results and performance analysis of the proposed algorithm against other existing meta-heuristic algorithms. Finally, we conclude the work in Section 5 and draw the future work.

II. Collaborative Team Formation in Social Network

Assuming that there exist a pool set of n experts $X = \{x_1, x_2, \ldots, x_n\}$ and a set of m skills $S = \{s_1, s_2, \ldots, s_m\}$. Each expert x_i has a set of skills denoted by $\mathfrak{S}(x_i) \subseteq S$ that represent the strength of a given expert with respect to a particular skill. For each skill s_j, the set of all experts having s_j denoted as $X(s_j) = \{x_i | s_j \in \mathfrak{S}(x_i)\}$. Given a task $T \subseteq S$ consists of a set of required skills that can be performed by a subset of experts $\tilde{X} \subseteq X$ is satisfied if $\forall s_j \in T: \exists x_i \in \tilde{X}, s_j \in \mathfrak{S}(x_i)$.

Let $G = (X, E)$ represents the collaborative relationship among experts in a social network. Each edge $e(x_i, x_j) \in E$ between expert x_i and x_j represents such collaboration and the weight on the edge represents that communication cost between two experts. The lower communication cost between two experts leads to more collaboration between them. The computation of communication cost among experts can be defined according to different criteria as mentioned in the previous section. In this research, the communication cost among experts can be defined by the collaboration between them. The edge between two experts in social network exists if the two experts have the ability to collaborate and the weight of the edge represents the relationship force between two experts and can be computed according to (1). Such relationships can be obtained from real-world scientific social networks such as DBLP or Stack Overflow.

$$e(x_i, x_j) = 1 - \frac{\mathfrak{S}(x_i) \cap \mathfrak{S}(x_j)}{\mathfrak{S}(x_i) \cup \mathfrak{S}(x_j)} \tag{1}$$

Fig.1. illustrates an example of collaborative social network of experts. It consists of a set of 5 experts (i.e., $X = \{x_1, x_2, x_3, x_4, x_5\}$ and a set of 8 skills (i.e., $S = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8\}$). The set of skills that each expert has (e.g., $(x_1) = \{s_1, s_2, s_4, s_5\}$) and the communication cost between experts are illustrated in the figure. Given a task $= \{s_2, s_3, s_5\}$, the goal is to find the team of collaborated experts that satisfied the required skills of a given task with least minimum communication cost among all possible teams. This collaboration team formation problem can be solved by a meta-heuristic algorithm.

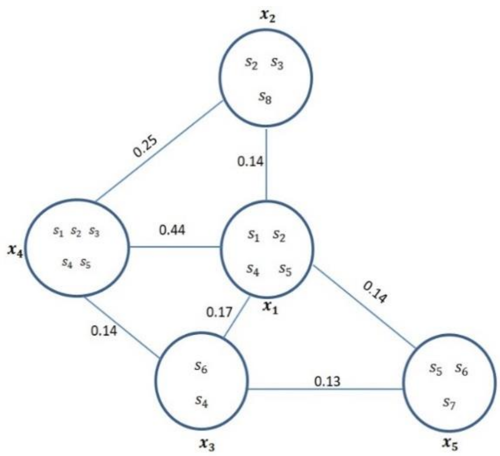

Fig.1. Example of collaborative social network of experts.

Some of teams can be formed that have the required skills to achieve the task such as $TM1 = \{x_1, x_2\}$, $TM2 = \{x_1, x_4\}$, $TM3 = \{x_1, x_2, x_5\}$ and $TM4 = \{x_1, x_2, x_4\}$. An expert may be responsible for more than required skill to be achieved in the team such that $x_{s_i} = x_{s_j}$ $for\ i \neq j$. Among these teams the proposed algorithm is seeking to find the most feasible team with minimum communication cost.

III. Proposed IABO Algorithm

In this section, we discuss the main concept of African Buffalo Optimization algorithm, they key factor of crossover operator and the new concept of swap sequence operator and how to integrate both in the proposed algorithm.

A. African Buffalo Optimization Algorithm

African Buffalo Optimization (ABO) is a recently meta-heuristic algorithm developed in 2015 [5] that emulate an d utilize the effective communication and management structure of herd during the migration lifestyle. They follow the voting behavior in their decision making and their movements are controlled by the majority decisions. In their movement, they use two vocalizations/sounds "*maaa*" and "*waaa*" for exploitation and exploration. The "*maaa*" sound summons the buffalos to stay on to exploit the present location since it has sufficient pasture and is safe while the "*waaa*" sound is used to explore other location because the current location may be lack of sufficient pasture . By using these sounds, buffalos are able to optimize their search to reach fruitful regions of food and can be represented mathematically according to (2) and (3) respectively.

$$m_{k+1} = m_k + lp1\,(bg_{max} - w_k) + lp2\,(bg_{max.k} - w_k) \tag{2}$$

Where m_k is a *maaa* sound with a particular reference to a buffalo k ($k = 1,2, \ldots n$) , bg_{max} is the location of the best buffalo in the herd, $bg_{max.k}$ is the best location found by an individual buffalo k, $lp1$ and $lp2$ are the learning parameters $\in [0,1]$.

By using (2), m_{k+1} is an indication for relocation of buffalo from current location m_k to a new location that reflects the extensive memory capacity in the migration lifestyle. The actual adjustment of the herd movement can be achieved according to (3).

$$w_{k+1} = \frac{(w_k + m_k)}{\lambda} \tag{3}$$

Where w_{k+1} represents the movement to a new location, w_k is the current exploration values that represent "*waaa"* sound while m_k is the current exploitation values and λ is a parameter that defines the unit of time interval over the movement of buffalo and usually is set to 1[6].

The algorithm below describes the ABO algorithm by initially placing random of the k-th buffalos in the solution space. The final best solution obtained based on adjusting the movement of buffalos during iterations. In each iteration, the fitness value of each buffalo obtained and the best one among all is assigned to bg_{max} (i.e., the best global one) while the best for each individual is assigned to $bg_{max.k}$ (i.e., best local one). Each buffalo updates its location and moves based on the best neighboring buffalo according to (2) and (3). Applying this update enables the movement of buffalos towards the best solution and tracking it.

The whole process of ABO algorithm can be illustrated in Fig. 2.

Algorithm: ABO algorithm

Step 1: Initialization

Randomly initialize k^{th} buffalos' location on solution space

Step 2: Evaluation of buffalos' fitness value and assigning the herd's best to bg_{max} and individual buffalo's best to $bg_{max.k}$

Step 3: Update (exploitation move)

Update the buffalos' fitness value according to (2)

$$m_{k+1} = m_k + lp1\,(bg_{max} - w_k) + lp2\,(bg_{max.k} - w_k)$$

Step 4: Update (exploration move)

Update the movement of buffalo according to (3)

$$w_{k+1} = \frac{(w_k + m_k)}{\lambda}$$

Step 5: Is bg_{max} updating? Yes, go to step 6. No, go to step 1.

Step 6: Check the validation of stopping criteria; if satisfied? Yes, go to step 7. No, go to step 2

Step 7: Return the best solution so far.

B. Swap Sequence Operator

A swap sequence SS is consists of one or more swap operators SO i.e., $SS = (SO_1, SO_2, \ldots, SO_n)$ that applied on a solution to produce new solution according to (4).

$$X_{new} = X_{old} + SS = X_{old} + (SO_1, SO_2, \ldots, SO_n) = (((X_{old} + SO_1) + SO_2) \ldots + SO_n) \tag{4}$$

Where X_{new} is the new solution that obtained from applying the swap operators on the old solution X_{old} in order. The basic concept of swap operator and swap sequence are discussed in [29-31] that considered a swap operator made-up of two parameters a *and* b (i.e., (a, b) . For example if we have a solution $X_{old} = (1,3,5,2,6)$ and $SO(2,4)$, then the new solution $X_{new} = X_{old} + SO(2,4) = (1,3,5,2,6) + SO(2,4) = (1, \mathbf{2}, 5, \mathbf{3}, 6)$, i.e., the values at index 2 and 4 are swapped. The swap operator applied in [32] with algorithms inspired from the animal group living behavior to solve traveling salesman problem (TSP).

In this research, we applied a new concept of swap operator $\widetilde{SO}\,(a, b)$ in the swap sequence which acts on an old solution to produce a final new solution that made-up of two parameters. The first parameter a represents the experts' skill_id (s_{id}) , and the second parameter represents new experts' index (x_{id}) that have s_{id} (i.e., $b \in X(s_{id})$). For example, $\widetilde{SO}$ (1,4) means for skill_id=1, swap/replace the current expert with expert at index 4. Therefore, the swap sequence operator consists of one or more $\widetilde{SO}$ that applied sequentially on the solution $SS = (\widetilde{SO}_1, \widetilde{SO}_2, \dots, \widetilde{SO}_n)$.

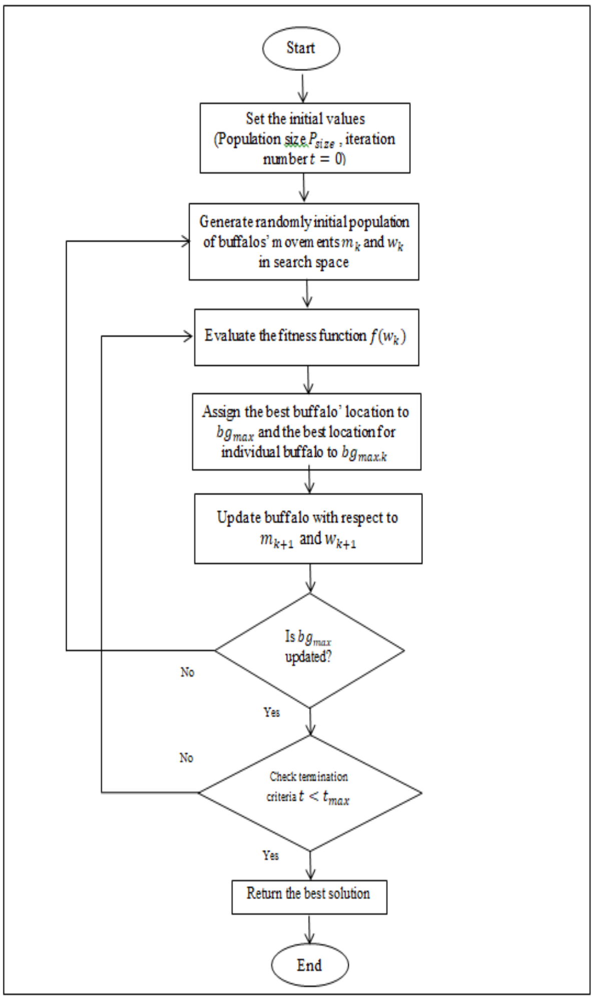

Fig.2. The whole process in ABO algorithm.

C. An Improved African Buffalo Optimization Algorithm (IABO)

In this subsection, the proposed algorithm is described in details and how can be applied in social network for collaboration among team members. The IABO algorithm can be obtained by integrating the DC operator into ABO algorithm along with the new concept of swap sequence operator SS as described in Algorithm 1 and summarized the main steps below. Given a task T requires a set of skills $s_j, j = \{1,2,\dots,d\}$ to be achieved where d is the number of required skills.

Step 1: Initialization

Randomly initialize the main parameters of IABO such as population size P_{size}, learning parameters $lp1$ and $lp2$ and the maximum number of iterations t_{max}. The IABO sets the movement (exploitation and exploration move) for all buffalos randomly (i.e.,w_k and m_k) , where each buffalo represents a vector of random skills that form a task and exploitation move is a swap sequence made-up of sequence of random swap operators $\widetilde{SO}$.

Step 2: *Evaluation of buffalos*

Each buffalo (i.e., solution) in the population evaluated by calculating its objective function $f(w_k^t)$ according to (5) based on (1) and assign the bg_{max}^t and $bg_{max.k}^t$.

$$f(w_k^t) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} e(x_i, x_j) \quad (5)$$

Where n is the number of team members, and $e(x_i, x_j)$ is the sum of communication cost between experts (i.e., x_i and x_j) in social network G.

Step 3: *Integration of discrete crossover operator*

In order to guarantee that the whole population moving towards the global optimum solution, we integrate a Discrete Crossover operator [26] that uses a random real number to create one child from two parents as shown below.

Parent 1:	1 1 1 0 1 0 0 1 0
Parent 2:	1 0 0 0 1 0 1 1 0
Child:	1 1 0 0 1 0 1 1 0

The child can be generated by selecting genes of both the parents uniformly depending on the random real number $u \in< 0,1 >$. The DC applied between the global herd's best solution bg_{max}^t and the current solution w_k^t to produce a new solution (i.e., child). The produced solution is denoted as $w_{k-cross}^t$.

Step 4: *Update (exploitation move)*

The initial buffalo's exploitation move m_k^t is made-up of random set of swap operators $\widetilde{SO}$ and can be updated according to (6)

$$m_k^{t+1} = m_k^t \oplus lp1 \otimes (w_{k-cross}^t - w_k^t) \oplus lp2 \otimes (bg_{max.k}^t - w_k^t) \quad (6)$$

Where $lp1$ and $lp2$ are learning parameters randomly between [0,1], $\oplus$ is a combining sequence operator of two swap operators, $\otimes$ is the probability of $lp1$ that all swap operators can be selected from the swap sequence $(w_{k-cross}^t - w_k^t)$ and the probability of $lp2$ that all swap operators can be selected from the swap sequence $(bg_{max.k} - w_k^t)$ which are denoted as SS_1 and SS_2 respectively. Therefore, $m_k^{t+1} = SS = (\widetilde{SO}_1, \widetilde{SO}_2, \dots, \widetilde{SO}_l)$ where l is the number of swap operators in m_k^t, SS_1 and SS_2.

Step 5: *Update (exploration move)*

In order to update the movement of current buffalo, we apply the sequence of swap operators to the current solution to obtain a new solution with a new movement according to (7).

$$w_k^{t+1} = \frac{(w_k^t + m_k^t)}{\lambda} \quad \forall k, k = \{1, \dots, P_{size}\} \tag{7}$$

Where λ is a unit time interval over the exploration move and set to 1.

Step 6: bg_{max}^t check

Check if the herd's best fitness value updated or not; if ($bg_{max}^{t+1} > bg_{max}^t$), then repeat the process from step 2 while the termination condition t_{max} not satisfied. Otherwise, back to step 1 and repeat the process again.

Step 7: *Output the best solution*

After a number of iterations, the global best solution with minimum communication cost among team members obtained as a final solution.

The algorithm below describes the whole process of improved African buffalo optimization algorithm.

Algorithm: An Improved African Buffalo Optimization Algorithm (IABO)

Input: Pool of experts X exists in social network $G = (X, E)$, a given task T associated with the set of required skills to achieve the task; $T = \{s_1, s_2, \dots, s_n\}$
Output: Best solution represents a team of experts perform the required skills for a given task with minimum communication cost.
Steps:
1. Set the initial population size P_{size}.
2. Set $t = 0$ and maximum number of iterations to t_{max}
3. Generate randomly m_k^t, w_k^t ; $k = 1, \dots, P_{size}$
4. Evaluate the solution fitness function $f(w_k^t)$ according to (5)
5. Set bg_{max}^t to the herd's best solution (best global solution in population)
6. Set $bg_{max.k}^t$ to the best individual's solution (best individual solution)
7. **While** $(t < t_{max})$
8. **For each** $w_k^t, m_k^t \in P_{size}$ **Do**
9. Apply Discrete crossover between bg_{max}^t and current solution w_k^t to obtain $w_{k-cross}^t$
10. Update m_k^t according to (6)
$$m_k^{t+1} = m_k^t \oplus lp1 \otimes (w_{k-cross}^t - w_k^t) \oplus lp2 \otimes (bg_{max.k}^t - w_k^t)$$
11. Update w_k^t according to (7)
$$w_k^{t+1} = \frac{(w_k^t + m_k^t)}{\lambda}$$
12. **End For each**
13. **If** $(f(w_k^{t+1}) \leq f(bg_{max}^t))$ **Then**
$bg_{max}^{t+1} = w_k^{t+1}$
End If
14. **If** $(f(w_k^{t+1}) \leq f(bg_{max.k}^t))$ **Then**
$bg_{max.k}^{t+1} = w_k^{t+1}$
Else
$bg_{max.k}^{t+1} = w_k^t$
End If
15. **If** $(bg_{max}^{t+1} < bg_{max}^t)$ **Then**
Set $t = t + 1$;
Go to step 3
Else
Go to step 4
End If
16. **End While**
17. Return the best solution $\boldsymbol{bg_{max}^t}$ so far

D. An illustrative example of IABO on collaborative team formation

According to Fig. 1, the relationship between experts is illustrated in the figure. The fitness function of the formed teams that satisfy the task is presented in table 1. Among these teams, the most collaborative one is $TM1$ (i.e., the one that has minimum communication cost among team members.

Table 1. Teams' fitness value

$T = \{s_2, s_3, s_5\}$	
TM	$F(TM)$
$TM1 = \{x_1, x_2\}$	**0.14**
$TM2 = \{x_1, x_4\}$	0.44
$TM3 = \{x_1, x_2, x_5\}$	0.28
$TM4 = \{x_1, x_2, x_4\}$	0.83

According to the exmple, each buffalo (i.e., solution) in the IABO algorithm is represented as an array list of size 1 × 3 where the first required skill is "s_2" , the second skill is "s_3" and the third skill is "s_5" as shown in Fig. 3. This figure represents the possible values for each index (i.e., start at index 0) of a solution in the IABO algorithm. As for required s_{id}=1, there are three experts that have this skill (i.e., x_1, x_2 and x_4).

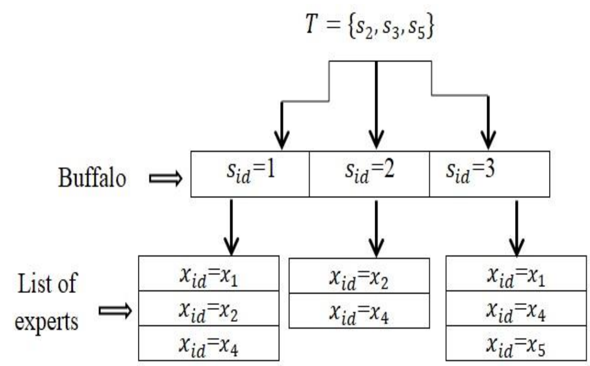

Fig.3. Solution representation in the IABO algorithm

According the main steps of proposed algorithm, the first step is the initial population which is generated randomaly (i.e., buffalo's exploitation and exploration movement) as shown in table 2.

The second step is evaluation of each solution in the population and assignment to bg_{max}^t and $bg_{max.k}^t$ as shown in table 3.

Table 2. Initial population of buffalos

Buffalo_id	w_k	m_k ($\widetilde{SO}_s$)
A	(x_1,x_2,x_4)	(3, 2),(2,1)
B	(x_2,x_4,x_5)	(1,1),(3,0)
C	(x_1,x_4,x_5)	(2,1),(3, 0)

Table 3. Buffalos' evaluation

Buffalo_ id	w_k	$f(w_k)$	bg_{max}^t	$bg_{max.k}^t$
A	(x_1,x_2,x_4)	0.83		0.83
B	(x_2,x_4,x_5)	**0.25**	**0.25**	0.25
C	(x_1,x_4,x_5)	0.58		0.58

In the third step, a DC operator applied between the best solution (i.e., B) and each solution (e.g., A) as shown in Fig. 4 to generate a new solution. In this case, suppose $u = 0.6$ (i.e., this means 60% of the genes of the first parent inherited in the child).

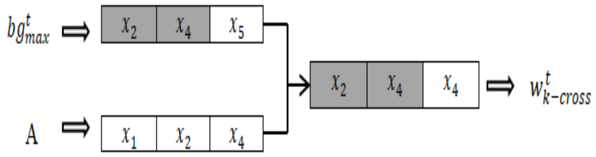

Fig.4. Discrete Crossover operator on solution

According to step 4, we suppose $lp1$and $lp2$ are equal 1 for simplicity, then obtain the update of exploitation move according to (6).

$$m_k^{t+1} = m_k^t \oplus lp1 \otimes (\, w_{k-cross}^t - w_k^t) \oplus lp2 \otimes (bg_{max.k}^t - w_k^t)$$

For the term $(w_{k-cross}^t - w_k^t)$; it means A1-A (i.e., the transformation sequence of $\widetilde{SO}_s$ that transform from A2 to A): $\widetilde{SO}_1 = (1{,}0), \widetilde{SO}_2 = (2{,}0)$. For the term $(bg_{max.k}^t - w_k^t)$; it means A-A=0 (i.e., identical solutions). Therefore, the update of buffalo's exploitation moves as follows:

$$m_k^{t+1} = SS = \big((3{,}2),(2{,}1)\big) \oplus \big((1{,}0),(2{,}0)\big) = ((3{,}2),(2{,}1),(1{,}0),(2{,}0))$$

At step 5, we considered $\lambda = 1$ then the update of buffalo's exploration move according to (7).

$$w_k^{t+1} = (w_k^t + m_k^t)$$

$$\begin{aligned} w_k^{t+1} &= (x_1, x_2, x_4) + SS \\ &= (x_1, x_2, x_4) + \big((3{,}2),(2{,}1),(1{,}0),(2{,}0)\big) \\ &= (x_1, x_2, x_5) + \big((2{,}1),(1{,}0),(2{,}0)\big) \\ &= (x_1, x_4, x_5) + \big((1{,}0),(2{,}0)\big) \\ &= (x_1, x_4, x_5) + \big((2{,}0)\big) = (x_1, x_2, x_5) \end{aligned}$$

Therefore, solution A(x_1, x_2, x_4) updated to (x_1, x_2, x_5) and $f(A^{t+1}) = 0.28$ (i.e., it updated from 0.83 to 0.28). The above process applied for other solutions (i.e., B and C).

After that, check bg_{max}^t if updated to better minimization value (i.e., $bg_{max}^{t+1} > bg_{max}^t$) then, repeat all the above process till termination condition satisfied and report the best solution that represent the most feasible collaborative experts to achieve a task. Otherwise, re-initialize a population and the same procedure applied for all solutions.

IV. Experimental Results And Performance Analysis

We have demonstrated a series of experiments to examine the effectiveness and the accuracy of the proposed IABO algorithm in terms of collaborative team formation with minimum communication cost. The proposed IABO algorithm was compared against GA, PSO and standard ABO (ABO). In addition, the performance of the proposed algorithm is examined on two popular benchmark real-life dataset (DBLP and Stack Overflow). Each experiment considered a number of skills that control the team members. We set the skill number $M \subseteq \{2{,}4{,}6{,}8{,}10\}$. For each skill N, we set a number of initial population that are generated randomly (i.e., each individual in the population represents a possible solution of collaborative team) and a number of iteration $N \subseteq \{5{,}10{,}15{,}20{,}25\}$ that focuses on iteratively minimizing the communication cost among collaborative team members as summarized in table 4.

Table 4. Experiment's parameters

Exp. #	M	P_{size}	N
1	2	4	5
2	4	4	10
3	6	6	15
4	8	6	20
5	10	8	25

The experiments were implemented by Eclipse Java Neon V-1.8 running on Intel(R) core i7 CPU- 2.80 GHz with 8 GB RAM and (Windows 10). In the following sub-sections, we presented the parameter setting of both real-life dataset and the performance measure.

A. DBLP Dataset

In this research, we used the DBLP datasets (i.e., It is one of the most popular benchmark real-life dataset on social network), which has been extracted from DBLP XML released on July 2017. The DBLP consists of 4 main tables and associated fields as listed below in table 5.

Table 5. Extracted DBLP dataset

Table name	Fields	Number of records
Author	name, paper key	6054672
Paper	title, year, conference, paper key	1992157
Citation	paper cite key, paper cited key	79002
Conference	conf key, name, detail	33953

Due to the diversity of dataset, we focused on papers that are published on 2017 (i.e., 22364 records) on the following main five fields in computer science as follows: Databases, Data Mining, Theory, Software Engineering and Artificial Intelligence. The following steps applied on the extracted data to obtain the DBLP information.

- For each paper in the above conferences, the authors' names (i.e., experts) and the title of the paper were identified.
- The expert set consists of authors that have at least 3 papers published 2017 in DBLP (i.e., 77 authors have published papers > 3).
- Each author has a skill set that is extracted from his published papers' title using StringTokenizer in Java (i.e., extracted from 267 papers) and considered the most important shared skills between experts.
- Two authors are connected if they share author's skills where the communication cost $e(x_i, x_j)$ between two authors x_i and x_j is computed as shown in (1). The more common skills between experts, the less communication cost between them.

Each experiment is conducted and the average results are taken over 50 runs to ensure the accuracy of the proposed algorithm against others as described in the next sub-section.

A.1 Numerical results of IABO against other meta-heuristic algorithms with DBLP dataset

Several numerical results based on DBLP dataset illustrate and prove the efficiency of proposed algorithm against other meta-heuristic algorithms (e.g., GA [33], PSO [34] and ABO [5]) in terms of different number of skills. The numerical results such as minimum (Min.), maximum (Max.), average (Avg.) and standard deviation (St.d.) of communication cost over 50 runs of four algorithms are shown in table 6.

Table 6. Numerical results of IABO and others in DBLP dataset

Exp. #	*M*		**GA**	**PSO**	**ABO**	**IABO**
1	2	Min.	0.85	0.63	0.5	0.5
		Max.	0.97	0.96	0.93	0.89
		Avg.	0.9248	0.9094	0.7944	0.6026
		St.d.	0.025253	0.050159	0.136967	0.13407
2	4	Min.	5.07	5.07	4.39	3.39
		Max.	5.74	5.72	5.56	5.33
		Avg.	5.4972	5.4056	5.1622	4.8234
		St.d.	0.164937	0.180918	0.23261	0.454235
3	6	Min.	12.85	12.74	12.66	11.14
		Max.	14.44	14.44	13.97	13.8
		Avg.	13.86818	13.73340	13.39113	12.9286
		St.d.	0.3421768	0.336147	0.322960	0.534213
4	8	Min.	24.87	24.82	24.22	22.34
		Max.	28.65	27.8	26.56	25.85
		Avg.	26.4832	26.0556	25.4496	24.7546
		St.d.	0.826145	0.525932	0.531709	0.678264
5	10	Min.	40.68	40.68	39.87	37.67
		Max.	43.87	43.87	42.57	41.6
		Avg.	42.3998	42.17857	41.11469	40.33673
		St.d.	0.764903	0.823623	0.606775	0.728235

Fig. 5(a), (b), (c), (d) and (e) illustrated the communication cost vs. iterations according to different number of skills $M \subseteq \{2,4,6,8,10\}$ respectively. The solid line represents the results of proposed IABO algorithm while other dotted lines represent the other meta-heuristic algorithms. The improved African buffalo optimization algorithm (IABO) has proven its effectiveness when compared with other existing algorithms in terms of the communication cost (i.e., minimization of fitness value) where increasing the number of iterations lead to decreasing of the communication cost.

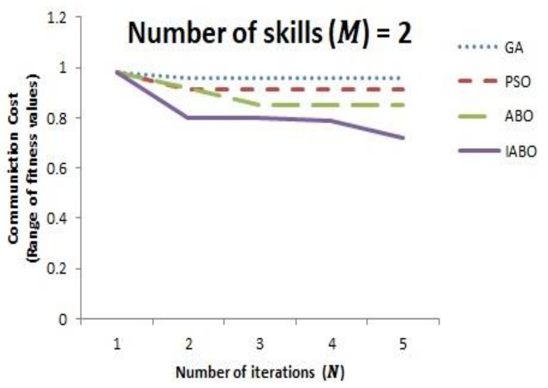

Fig.5(a). Communication cost vs. iteration number DBLP (M=2)

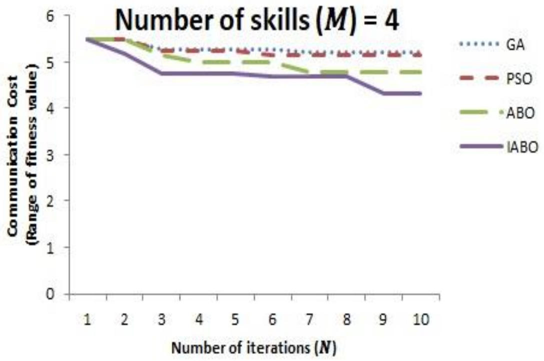

Fig.5(b). Communication cost vs. iteration number DBLP (M=4)

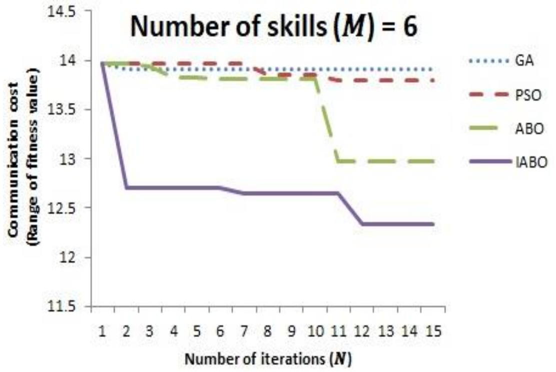

Fig.5(c). Communication cost vs. iteration number DBLP (M=6)

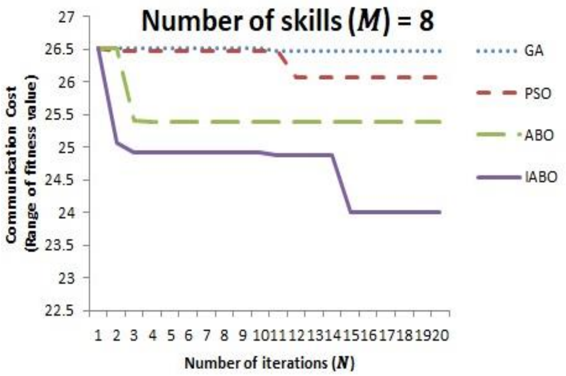

Fig.5(d). Communication cost vs. iteration number DBLP (M=8)

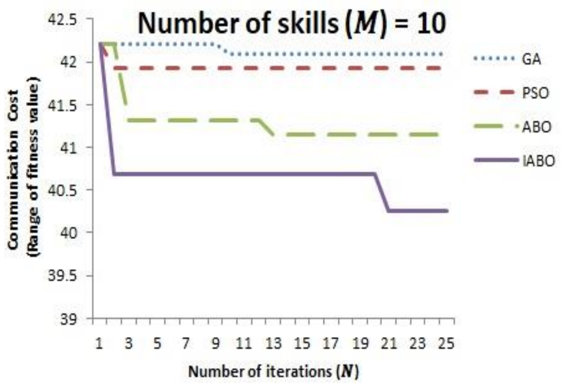

Fig.5(e). Communication cost vs. iteration number DBLP (M=10)

In addition, the percentage improvement of IABO algorithm over other algorithms can be computed according to (8).

$$Improvement\ (\%) = (\mathcal{C}_{alg} - \mathcal{C}_{HIABOSS})/\ \mathcal{C}_{alg} \quad (8)$$

Where, $alg = \{GA, PSO, ABO\}$, $\mathcal{C}_{alg}$ is the communication cost result of each of the existing algorithms and $\mathcal{C}_{HIABOSS}$ is the communication cost result of the proposed algorithm.

From Fig. 5(a) with small number of iterations ($N = 5$), the proposed IABO has achieved minimum communication cost when compared with other algorithms (e.g., GA, PSO, and ABO). The IABO results reached to 25% better than GA, 21% better than PSO and 15% better than standard ABO. During the iterations of proposed algorithm, the results improved to be reached to 27% in terms of minimization of communication cost.

While with ($M = 4$) in Fig. 5(b), the improved ABO algorithm is better than GA, PSO and standard ABO in reaching minimum range of fitness value iteratively. Although the standard ABO is near to the PSO in the first iterations, it has proven its efficiency at the last iterations while the proposed IABO has proven its efficiency during iterations with improving performance within range 5% ~ 21%. In addition, it is better than GA with 17%, better than PSO with 16% and better than ABO with 9%.

The proposed algorithm reached to best results with respect to number of skills ($M = 6$) as shown in Fig. 5(c) faster than other algorithms and within performance improvement up to 11% better than both GA and PSO, and 4.93% better than standard ABO. With increasing number of iterations, the range of fitness value of IABO algorithm improved to 12% iteratively.

Fig. 5(d) illustrated that, with increasing number of skills ($M = 8$) (the proposed IABO algorithm obtained communication cost results better than GA with 9%, better than PSO with 8% and better than standard ABO with 5%. with increased number of iterations, the IABO improved the performance of fitness value from 5.47% in the first iteration to 9% at the last iterations.

With large number of skills ($M = 10$) and iterations ($N = 25$) as shown in Fig. 5(e), the proposed IABO achieved a practical solution to a collaborative team formation in social network with respect to increase number of skills and iterations. it achieved better results than GA and PSO with 4%. Although the ABO algorithm achieved minimum communication cost during iterations when compared with GA and PSO, the proposed algorithm has proven its efficiency than ABO with improved results 2% and reached to 5% during the number of iterations.

The performance analysis of the proposed algorithm against other meta-heuristic algorithm can be measured by computing the 95% confidence interval (95% CI) on average communication cost according to (9).

$$CI = mean\ \pm margin\ of\ error \quad (9)$$

The 95% CI on the average communication cost of IABO and other algorithms are shown in figure 6 associated with table 7.

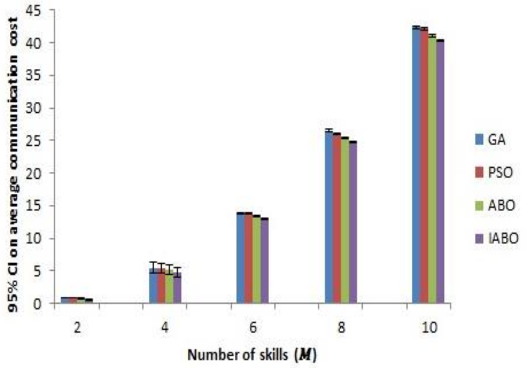

Fig.6. 95% Confidence Interval on communication cost (DBLP dataset)

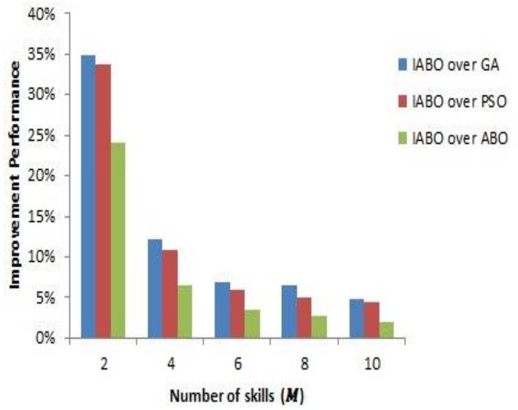

Fig.7. Improvement performance of IABO (DBLP dataset)

The improvement performance of the proposed IABO over other meta-heuristic algorithms (GA, PSO and ABO) for different number of skills are shown in Fig. 7 which proves the efficiency and effectiveness of proposed algorithm in terms of minimization of communication cost.

Moreover, the main objective of the proposed algorithm is to reach the minimum fitness value through iterations that have been done on the population as shown in figure 8 which considered the performance of proposed algorithm and other meta-heuristic algorithms with different population size.

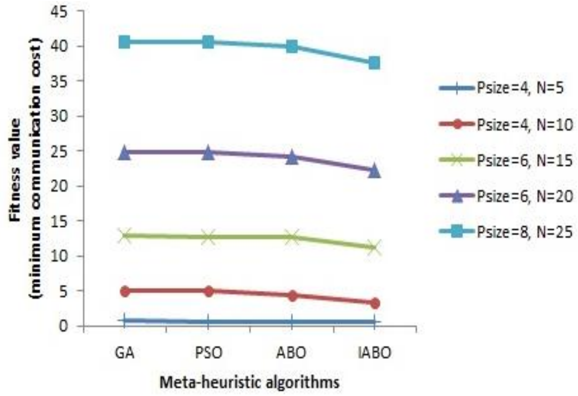

Fig.8. Performance of IABO with different P_{size} (DBLP dataset)

Table 7. 95% CI on average communication cost for IABO vs. other meta-heuristic algorithms (DBLP dataset)

M	GA	PSO	ABO	IABO
2	0.9248 ±0.007	0.9094 ±0.0139	0.7944 ±0.03796	0.6026 ±0.03716
4	5.4972 ±0.79353	5.4056 ±0.78808	5.1622 ±0.7636	4.8234 ±0.76
6	13.86818 ±0.09484	13.7334 ±0.09317	13.39113 ±0.08952	12.9286 ±0.14807
8	26.4832 ±0.22899	26.0556 ±0.14578	25.4496 ±0.14738	24.7546 ±0.188
10	42.3998 ±0.21202	42.17857 ±0.22829	41.11469 ±0.16819	40.33673 ±0.20185

B. Stack Overflow dataset

We validate the performance of the proposed IABO algorithm based on another popular benchmark real-life dataset on social network (i.e., Stack Overflow), which has been extracted from Stack Overflow XML released on August 2017. The extracted Stack Overflow dataset consists of 5 main tables associated with varying fields for each table as listed in table 8.

Table 8. Extracted Stack Overflow dataset

Table name	Fields	Number of records
Users	Id, Reputation, CreationDate, DisplayName, LastAccessDate, WebsiteUrl, Location, AboutMe, Views, UpVotes, DownVotes, Age, AccountId	7617191
Posts	Id, PostTypeId, AcceptedAnswerId, CreationDate, Score, ViewCount, Body, OwnerUserId, LastEditorUserId, LastEditDate, LastActivityDate, Title, Tags, AnswerCount, CommentCount, FavoriteCount	37215528
Comments	Id, PostId, Score, Text, CreationDate, UserId	157864
Postlinks	Id, CreationDate, PostId, RelatedPostId, LinkTypeId	9830
Tags	WikiPostId, ExcerptPostId, Count, TagName, Id	400
Posthistory	Id, PostHistoryTypeId, PostId, RevisionGUID, CreationDate, UserID, Text	180620
Votes	CreationDate, VoteTypeId, PostId, Id	703546
Badges	TagBased, Class, Date, Name, UserId, Id	116925

Due to the huge data extracted from the Stack Overflow dataset, we have focused on users' posts on 2017 (i.e., 1696954 rows) on different topics. The following steps applied on the extracted Stack Overflow to obtain the most relevant information (i.e., expert set and skill set) that can be used in this research.

- The expert set consists of users that each one has posted not less than 50 posts in stack overflow on 2017 (i.e., 738 users).
- The extracted total number of posts of expert set is 61893 posts.
- The skill set for each expert extracted from his/her posts' tags using StringTokenizer in java.
- Two experts are connected if they share expert's skills where the communication cost $e(x_i, x_j)$ between two experts x_i and x_j is computed as shown in (1). The more common skills between experts, the less communication cost between them.
- In this research, we have considered the most important tags that are extracted from popular posts in Stack Overflow.

Each experiment in table 4 is conducted on the above setting and the average results are taken over 50 runs to ensure the validity and accuracy of the proposed algorithm against other standard algorithms.

B.1 Numerical results of IABO against other meta-heuristic algorithms with Stack Overflow dataset

Another test has been done on most popular social network dataset (Stack Overflow) that is slightly denser than DBLP in order the prove the accuracy of the proposed algorithm in terms of collaborative team formation in social network with minimum communication cost among team members. The numerical results are including the minimum (Min.), maximum (Max.), average (Avg.) and standard deviation (St.d.) over 50 runs for each algorithm is shown in table 9

Table 9. Numerical results of IABO and others in Stack Overflow dataset

Exp. #	M		**GA**	**PSO**	**ABO**	**IABO**
1	2	Min.	0.79	0.79	0.79	0.58
		Max.	0.94	0.93	0.92	0.92
		Avg.	0.905918367	0.894489796	0.874081633	0.825306122
		St.d.	0.032462694	0.030758551	0.035702379	0.071708381
2	4	Min.	4.7	4.7	4.61	3.76
		Max.	5.66	5.64	5.56	5.52
		Avg.	5.4846	5.4372	5.2586	4.9666
		St.d.	0.181683847	0.171440952	0.295199759	0.417937648
3	6	Min.	11.45	11.45	11.45	10.74
		Max.	14.08	13.99	13.87	13.5
		Avg.	13.6516	13.5598	13.2452	12.794
		St.d.	0.446872853	0.432276792	0.39947537	0.466926555
4	8	Min.	25.11	25.11	24.46	23.17
		Max.	26.54	26.54	26.15	25.64
		Avg.	25.8574	25.7214	25.3142	24.7736
		St.d.	0.332147066	0.334499382	0.368610685	0.509848957
5	10	Min.	40.64	40.39	40.01	38.05
		Max.	42.69	42.02	41.77	40.95
		Avg.	41.6148	41.3544	40.9328	39.9668
		St.d.	0.452400581	0.4347843	0.37067528	0.644371043

Fig. 9(a), (b), (c), (d) and (e) illustrated the communication cost vs. iterations according to different number of skills $M \subseteq \{2,4,6,8,10\}$ respectively on stack overflow dataset. The solid line represents the results of proposed IABO algorithm while other dotted lines represent the other meta-heuristic algorithms. The improved African buffalo optimization algorithm (IABO) has proven its effectiveness when compared with other existing algorithms in terms of minimization the fitness value especially when dealing with huge dataset where increasing the number of iterations lead to decreasing of the communication cost. The percentage improvement of IABO algorithm over other algorithms on stack overflow dataset can be computed according to (8).

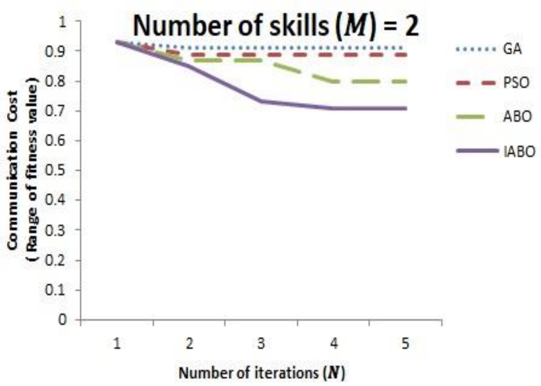

Fig.9(a). Communication cost vs. iteration number Stack Overflow (M=2)

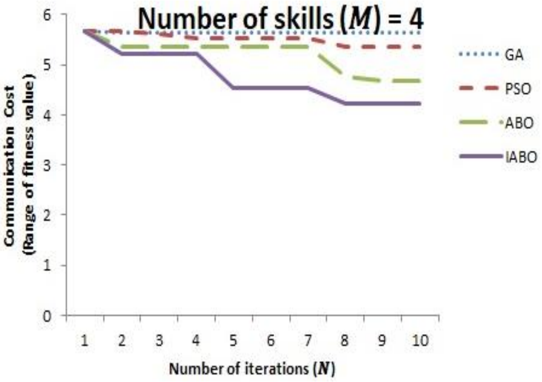

Fig.9(b). Communication cost vs. iteration number Stack Overflow (M=4)

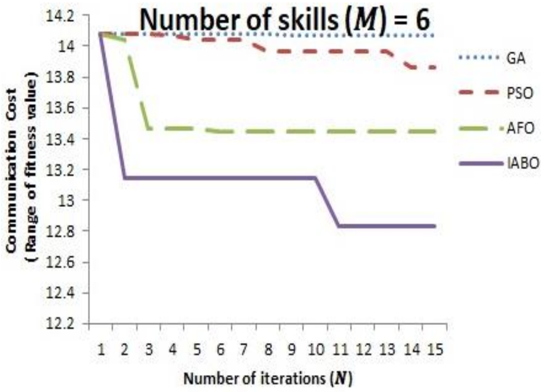

Fig.9(c). Communication cost vs. iteration number Stack Overflow (M=6)

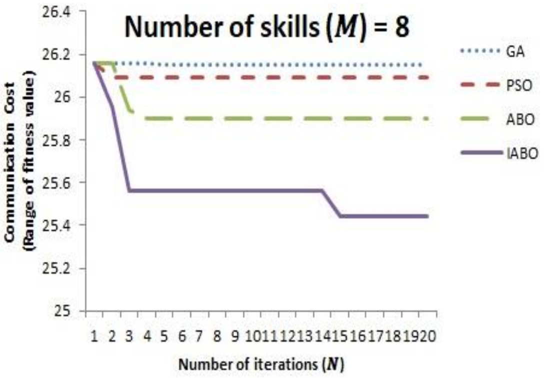

Fig.9(d). Communication cost vs. iteration number Stack Overflow (M=8)

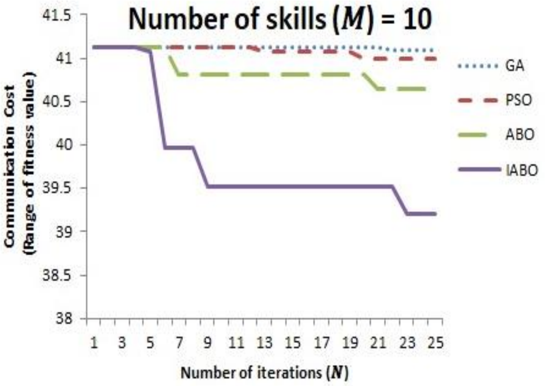

Fig.9(e). Communication cost vs. iteration number Stack Overflow (M=10)

According to Fig. 9(a) with small number of skills ($M = 2$), the GA has a little bit change of fitness value during iterations while PSO is better than it to achieve minimum communication cost. Although the standard ABO obtain results better than GA and PSO, the improved ABO algorithm achieved better results than ABO reached to 11%, better than GA with 22% and better than PSO with 20%. From first iteration to last iteration, the IABO achieved improvement in the performance reached to 24%.

With increasing the number of skills as in Fig. 9(b), the proposed algorithm has proven its efficiency for reaching to minimum communication cost during iterations ranged from 8% to 26 %. The improvement performance of IABO is 25% when compared against GA, 21% when compared with PSO and 10% when compared with standard ABO algorithm.

With increasing the number of iterations ($N = 15$) and skills ($M = 6$) as shown in Fig. 9(c), the PSO achieved better results than GA to reach better fitness value while AFO achieved better results than GA and PSO from the first iterations. The IABO improves the performance of collaborative team with respect to number of skills during iterations and when compared with GA reached to 9%. In case of PSO, the IABO reached 7% over it and 5% over standard AFO.

Fig. 9(d) illustrated that, the IABO algorithm reached better fitness value faster than other algorithms with performance improvement up to 3% better than GA, 2% better than PSO and ABO. With large number of iterations the proposed algorithm minimizes the communication cost iteratively within range 1% ~ 3%.

The fast decreasing in the range of fitness value of the proposed IABO algorithm as shown in Fig. 9(e) with ($M = 10$) and ($N = 25$) has proven the efficiency of the proposed algorithm when compared with others. IABO obtained minimum communication cost results better than GA with 5% and better than PSO and ABO with 4% while the improvement performance of the proposed algorithm during iteration ranged from 3%~5%.

We have used the 95% confidence interval (95% CI) on average communication cost to measure the performance analysis of the IABO algorithm against GA, PSO and ABO in case Stack Overflow based on (9).

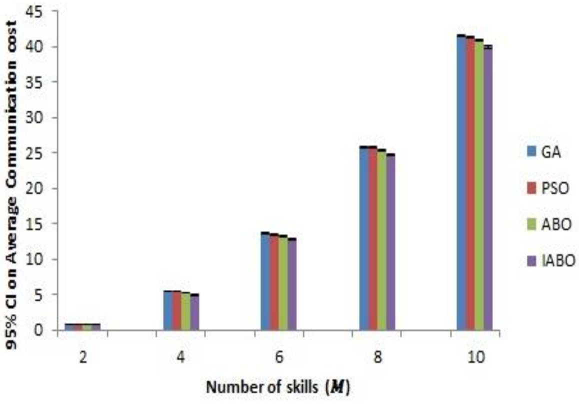

Fig.10. 95% Confidence Interval on communication cost (Stack Overflow dataset)

The 95% CI on the average communication cost of IABO and other algorithms are shown in Fig. 10 associated with table 10.

In addition, the improvement performance of the proposed IABO over other meta-heuristic algorithms (GA, PSO and ABO) for testing different number of skills in case of Stack Overflow dataset are shown in Fig. 11 which proves the efficiency and effectiveness of proposed algorithm in terms of minimization of communication cost.

Moreover, the main objective of the proposed algorithm is to reach the minimum fitness value through iterations that have been done on the population as shown in Fig. 12 which considered the performance of proposed algorithm and other meta-heuristic algorithms with different population size.

Table 10. 95% CI on average communication cost for IABO vs. other meta-heuristic algorithms (Stack Overflow dataset)

M	GA	PSO	ABO	IABO
2	0.9059 ±0.009	0.8945 ±0.00853	0.8741 ±0.0099	0.8253 ±0.01988
4	5.4846 ±0.05036	5.4372 ±0.04752	5.2586 ±0.08182	4.9666 ±0.11584
6	13.6516 ±0.12386	13.5598 ±0.11982	13.2452 ±0.11073	12.794 ±0.12942
8	25.8574 ±0.09206	25.7214 ±0.09272	25.3142 ±0.10217	24.7736 ±0.14132
10	41.6148 ±0.1254	41.3544 ±0.12051	40.9328 ±0.10274	39.9668 ±0.17861

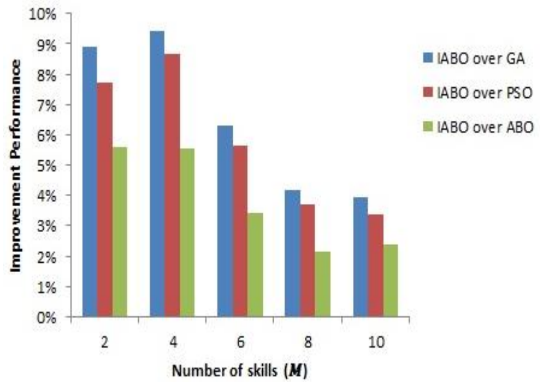

Fig.11. Improvement performance of IABO (Stack Overflow dataset)

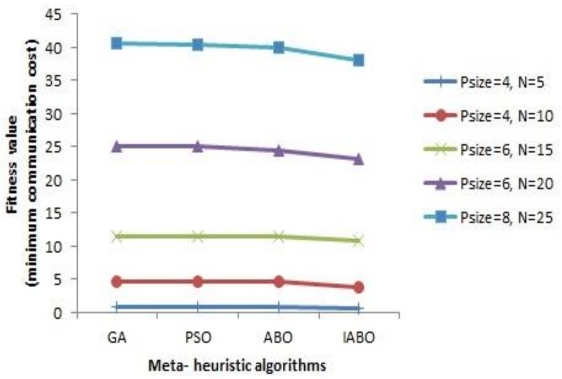

Fig.12. Performance of IABO with different P_{size} (Stack Overflow dataset)

V. Conclusion and Future Work

In this paper, we have considered the problem of collaborative team formation in social network where the objective is to form an efficient collaborative team of experts with minimum communication cost between team members to achieve a given task. Such formation of team expert can be considered an optimization problem that require meta-heuristic algorithm to solve it. Therefore, this research provides an improvement of meta-heuristic algorithm with the main strength of discrete crossover operator and a new concept of swap sequence. The proposed algorithm called an Improved African Buffalo Optimization (IABO). In IABO, we integrate a discrete crossover DC operator along with a new concept of swap sequence that consists of one or more $\widetilde{SO}$ that applied sequentially to improve the solution. The performance analysis of the proposed algorithm is tested on two popular real-life dataset (DBLP and Stack Overflow) and the results are compared with other existing meta-heuristic algorithms (GA, PSO and standard ABO). The achieved results have proven the efficiency of it with respect to reach a best solution with minimum communication cost and faster than other compared meta-heuristic algorithms. In the future work, we will integrate the proposed algorithm with other new meta-heuristic algorithms to improve the performance of team formation problem in social network to achieve a given goal.

References

[1] T. Lappas, K. Liu, E. Terzi. "Finding a team of experts in social networks". *In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and datamining*. 2009, pp. 467-47.

[2] E.G. Talbi. "Metaheuristics: from design to implementation". Wiley, 2009, Vol. (74).

[3] Theodoros Lappas, Kun Liu, Evimaria Terzi. "A survey of algorithms and systems for expert location in social networks". *Social network data analytics*, Springer, 2011, pp. 215–241.

[4] Sangita Roy, Samir Biswas, Sheli Sinha Chaudhuri, "Nature-Inspired Swarm Intelligence and Its Applications". *IJMECS*, vol.6, no.12, pp.55-65, 2014.DOI: 10.5815/ijmecs.2014.12.08.

[5] J. B Odili, M. N. M. Kahar. "African Buffalo Optimization (ABO): a New Meta-Heuristic Algorithm". *Journal of Advanced & Applied Sciences*, April 2015, Vol. (03): 101-106.

[6] J. B Odili, M. N. M. Kahar. "Numerical Function Optimization Solutions Using the African Buffalo Optimization Algorithm (ABO)". *British Journal of Mathematics & Computer Science*, 2015,10(1): 1-12.

[7] J. B Odili, M. N. M. Kahar, A. Noraziah. "Convergence Analysis of the African Buffalo Optimization Algorithm". In *International Journal of Simulation: Systems, Science & Technology* (IJSSST), April 2017, 17(33).

[8] J. B Odili, M. N. M. Kahar. "Solving the Traveling

Salesman's Problem Using the African Buffalo Optimization". *Computational Intelligence and Neuroscience*, Hindawi Publishing Corporation, 2016.

[9] J. B Odili, M. N. M. Kahar, S. Anwar, M. Ali. "Tutorials on African Buffalo Optimization for Solving the Traveling Salesman Problem". In *International Journal of Software Engineering and Computer Systems* (IJSECS), February 2017, Vol.(3):120-128.

[10] J. B Odili, M. N. M. Kahar, A. Noraziah, E. A. Odili. "African Buffalo optimization and the randomized insertion algorithm for the asymmetric travelling salesman's problems". *Journal of Theoretical and Applied Information Technology*, May 2016,Vol.87(3).

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

[12] A. Gajewar, , A.D. Sarma. "Multi-skill collaborative teams based on densest subgraphs". *In SDM*, 2012, pp. 165–176.

[13] M. Kargar, M. Zihayat, A. An. "Finding affordable and collaborative teams from a network of experts". In *Proceedings of the 2013 SIAM International Conference on Data Mining*, Society for Industrial and Applied Mathematics, 2013, pp. 587-595.

[14] M. Kargar, A. An, "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*, 2011, pp. 985-994.

[15] Ana Paula Appel, Victor F. Cavalcante, Marcos R. Vieira,. Vagner F. de Santana, RA de Paula, Steven K Tsukamoto. "Building socially connected skilled teams to accomplish complex tasks," In *Proceedings of the 8th Workshop on Social Network Mining and Analysis*, 2014.

[16] Li, CT., Shan, MK. "Team formation for generalized tasks in expertise social networks," IEEE Second International Conference on Social Computing (SocialCom), pp. 9-16, 2010.

[17] CT. Li, MK. Shan, SD. Lin. "On team formation with expertise query in collaborative social networks". Knowledge and Information Systems, 2015,42(2): 441-463.

[18] Aris Anagnostopoulos , Luca Becchetti, Carlos Castillo, Aristides Gionis, Stefano Leonardi. "Power in unity: forming teams in large-scale community systems". In *Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010, pp. 599-608.

[19] J. Huang, X. Sun, Y. Zhou, H. Sun. "A Team Formation Model with Personnel Work Hours and Project Workload Quantified" .The Computer Journal, 2017, pp. 1-13.

[20] RL. Haupt, SE. Haupt. "Practical genetic algorithms," John Wiley & Sons, 2004.

[21] C. Blum, A. Roli. "Metaheuristics in combinatorial optimization: Overview and conceptual comparison". ACM Computing Surveys (CSUR), 2003, 35(3): 268-308.

[22] K. Pashaei, F. Taghiyareh, K. Badie. "A recursive genetic framework for evolutionary decision-making in problems with high dynamism". *International Journal of Systems Science*, 2015, 46(15): 2715-2731.

[23] D. Sedighizadeh, E. Masehian. "Particle swarm optimization methods, taxonomy and Applications". *International Journal of Computer Theory and Engineering*, 2009, 1(5): 486-502.

[24] Hamza O. Salami, Esther Y. Mamman, "A Genetic Algorithm for Allocating Project Supervisors to Students". *International Journal of Intelligent Systems and Applications (IJISA)*, Vol.8, No.10, pp.51-59, 2016. DOI: 10.5815/ijisa.2016.10.06

[25] JW. Zhang, WJ. Si. "Improved enhanced self-tentative PSO algorithm for TSP". In *Sixth International Conference on Natural Computation* (ICNC), 2010, Vol.(5):2638-2641.

[26] M. Nadershahi, RT. Moghaddam. "An application of genetic algorithm methods for team formation on the basis of Belbin team role". Archives of Applied Science Research, 2012, 4(6): 2488-2496.

[27] Mohammad Fathian, Mohamad Saei-Shahi , Ahmad Makui. "A New Optimization Model for Reliable Team Formation Problem Considering Experts Collaboration Network". IEEE Transactions on Engineering Management, 2017.

[28] T. G. Dominik. "Genetic Algorithms Reference". Volume –I, Poland: Tomasz Gwiazda, 2006.

[29] KP. Wang, L. Huang, CG. Zhou, W. Pang. "Particle swarm optimization for traveling salesman problem". In *International Conference on Machine Learning and Cybernetics*, 2003, Vol. (3): 1583-1585.

[30] X. Wei, Z. Jiang-wei, Z. Hon-lin. "Enhanced Self-Tentative Particle Swarm Optimization Algorithm for TSP". *Journal of north china electric power university*, 2009, 36(6): 69-74.

[31] JW. Zhang, WJ. Si. "Improved enhanced self-tentative PSO algorithm for TSP". In *Sixth International Conference on Natural Computation* (ICNC) , 2010, Vol.(5):2638-2641.

[32] M. A. H. Akhand, Pintu Chnadra Shill, Md. Forhad Hossain, A. B. M. Junaed, K. Murase, "Producer-Scrounger Method to Solve Traveling Salesman Problem". *International Journal of Intelligent Systems and Applications (IJISA)*, vol.7, no.3, pp.29-36, 2015. DOI:10.5815/ijisa.2015.03.04.

[33] M. Mitchell. An Introduction to Genetic Algorithms. Cambridge, MA: MIT Press (1996).

[34] RC. Eberhart, Y. Shi, J. Kennedy. "Swarm Intelligence". The Morgan Kaufmann Series in Evolutionary Computation, 2001.

Authors' Profiles

Walaa H. El-Ashmawi is an Assistant Professor at Suez Canal University, Faculty of Computers and Informatics, CS. Department, Egypt. She has obtained her PhD from College of Information Science and Engineering, Hunan University, China on 2013. She coauthored some journal publications in (World Applied Sciences Journal/2017, Asian Journal of Applied Sciences/2017, IAJIT/2015, IJCA/2014, JCIS/2014, JCIS/2013, JCP-Academy Publisher/2013, IJACT/2012 and IJJCS/2007) and others in conference proceedings. Her research interests include Artificial Intelligence and swarm intelligence, intelligent agent and multi-agent systems, cloud computing.

How to cite this paper: Walaa H. El-Ashmawi, "An Improved African Buffalo Optimization Algorithm for Collaborative Team Formation in Social Network", International Journal of Information Technology and Computer Science(IJITCS), Vol.10, No.5, pp.16-29, 2018. DOI: 10.5815/ijitcs.2018.05.02