Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
World Competitive Contests (WCC) algorithm: A novel intelligent optimization algorithm for biological and non-biological problems

Yosef Masoudi-Sobhanzadeh a, b, *, Habib Motieghader b

a Department of Computer Engineering, Payam Noor University, Tabriz, Iran
b Laboratory of Systems Biology and Bioinformatics (LBB), Institute of Biochemistry and Biophysics, University of Tehran, Tehran, Iran

A R T I C L E   I N F O

Article history:
Received 23 March 2016
Received in revised form 29 May 2016
Accepted 27 June 2016
Available online 28 June 2016

Keywords:
The motif discovery
Transcriptional factor binding sites
Optimization algorithms
World Competitive Contests

A B S T R A C T

Since different sciences face lots of problems which cannot be solved in reasonable time order, we need new methods and algorithms for getting acceptable answers in proper time order. In the present study, a novel intelligent optimization algorithm, known as WCC (World Competitive Contests), has been proposed and applied to find the transcriptional factor binding sites (TFBS) and eight benchmark functions discovery processes. We recognize the need to introduce an intelligent optimization algorithm because the TFBS discovery is a biological and an NP-Hard problem. Although there are some intelligent algorithms for the purpose of solving the above-mentioned problems, an optimization algorithm with good and acceptable performance, which is based on the real parameters, is essential. Like the other optimization algorithms, the proposed algorithm starts with the first population of teams. After teams are put into different groups, they will begin competing against their rival teams. The highly qualified teams will ascend to the elimination stage and will play each other in the next rounds. The other teams will wait for a new season to start. In this paper, we’re going to implement our proposed algorithm and compare it with five famous optimization algorithms from the perspective of the following: the obtained results, stability, convergence, standard deviation and elapsed time, which are applied to the real and randomly created datasets with different motif sizes. According to our obtained results, in many cases, the WCC’s performance is better than the other algorithms’.

© 2016 Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

Since there are a lot of problems with the theory of computational complexities, which need more and more time for being solved, we employ meta-heuristic optimization algorithms. Most of these algorithms can result in suitable solutions with less computational complexities. An evolutionary algorithm’s good performance, its stability in different types of execution with minimum errors and its elapsed time to produce acceptable results, are the main criteria that must be considered.

Many different optimization algorithms have been offered and various methods and new algorithms have been proposed to search for the motif discovery sites. Related works are examinable from two aspects: 1- proposing a new optimization algorithm aspect, and 2- the motif discovery aspect. Some of the imminent optimization algorithms which were proposed during the last decades are listed as follows. The genetic algorithm was inspired by living organisms’ evolutionary process [1]. A large number of modified genetic algorithms were proposed with a different crossover, mutation, and distinct selection operations [2]. A new optimization algorithm for a single hidden layer of feedback on a neural network, based on the genetic and convex combination algorithms, was introduced by Kwan et al. The most important characteristic of their algorithm was that it could get an acceptable answer in more reasonable time order compared with the others [3]. A new optimization algorithm, known as SOS, was introduced by Min-Yuan et al. to study living organisms. Obtained results proved the SOS’s good performance [4]. A combination of imperialist competitive and bee colony algorithms which was recently introduced led to a new method for an optimization problem. Improving a global exploration of a combination of floods of bee colony algorithm is mentioned as method’s goals. Moreover, it was applied to two real datasets [5]. The heat transfer optimization algorithm was inspired by both the law of heat transfer and thermodynamics [6]. Another optimization algorithm, inspired by the particle swarm optimization, was recently introduced by Gosciniak et al. Their algorithm was not influenced by the environment, but rather by the particle swarm optimization that was employed as a model of a predator and prey. Moreover, it
proved that their approach can be practicable in computer games [7]. The particle swarm optimization is an algorithm in which the first populations were created randomly. It is based on the social interaction among flocks of birds or schools of fish. Zhang et al. enhanced the search ability using the Bayesian techniques through exploiting past particles’ positions and using a new type of mutation in order to obtain better results. Their approach is known as the BPSO method [8]. Gomez et al. introduced new optimization algorithm for spatial variability analysis and showed high improvement in the NDT measurement position [9].

The other (The second) aspect of related works is the motif discovery. Binding sites, which indicate chemical specificity and affinity, are either part of a protein or part of DNA or RNA in ligands which form a chemical bond. One of the most important components of the biomolecules’ general characteristics is binding sites that can identify the other proteins. Binding sites also exist in antibodies as coded regions. A non-covalent bond between the two polypeptide chains is formed when a binding site of one protein determines another protein’s surface. Transcriptional factors (TF), are proteins which bind to specific DNA sequences and control the rate of the transcription of genetic information from DNA to RNA. TF are categorized as a sequence of specific DNA binding factors that are of great importance to the regulation of a gene expression. They contain one or more DNA-binding domains that bind to specific sequences of DNA that are contiguous to the genes which they regulate. DNA binding sites are distinguishable from the other algorithms because they include the targets of the other proteins and therefore are part of DNA- in other words, are bound by DNA-binding proteins. DNA binding sites divide into several categories that are as follows: transcriptional factor binding sites, recombination sites, and restriction sites. There is a need for the evolutionary algorithms because the motif discovery problem is an NP (nondeterministic polynomial) hard problem that cannot be solved in polynomial time order. Although these algorithms are not deterministic, they can get acceptable answers in reasonable time order.

We combine the protein motif search processes with the gene discovery methods in which the discovery of human new genes of cadherin superfamily proteins is the main purpose, to identify genes’ encoding protein containing specific domains [10]. Proposing a genetic algorithm for the motif discovery based on the statistical significance of the current research is another recent study carried out, along with the above-mentioned works. After their proposed algorithm was applied to several different datasets, the obtained motifs were compared with each other [11]. Research into the Bcl-2 family protein is among subjects that relate to the present discussion. The family protein comprises several homologous proteins as well as a collection of other proteins which consists of a sequence of similarities except for a Bcl-2 homology (BH)3 motif. One of the four BH motifs is required for membership in the Bcl-2 family. Abdel et al. evaluated Bcl-2 homology motifs [12]. Determining the cellular binding motif in the case of the acute respiratory syndrome coronavirus is essential because the respiratory syndrome coronavirus threatens human lives [13]. Parkinson, Alzheimer, and some other diseases are a result of aging and stress that are among the major causes of the protein damage. According to Bonham-Carter [14], oxidation of amino acids with a high rate of motifs is due to some protein damages. Most of the solutions for bioinformatics NP-hard problems are based on ACO and PSO [15,16]. Some of the other related works that we can refer to are as follows:

Specifications of the monocyte activating motif in the mycobacterium (mycobacterial) tuberculosis [17], motifs retrieval by the Secondary Structure terns occurrences [18], and the interaction of binding motif within the nucleocapsid protein of porcine reproductive and respiratory syndrome virus and the host cellular signaling proteins [19].

2. The proposed algorithm

Proposed optimization algorithm starts with the first population of teams [20]. Every team includes several players with random values. The mentioned teams are placed into different groups based on geographical distance and begin to compete with each other in it. This stage can be considered as a local optimization. Global optimization stage which is elimination stage will begin after grouping matches. Eliminated teams will not be removed from the contests, and they will wait for new season of matches. Here is a flowchart of WCC (Fig. 1).

WCC includes several main stages:
1. Generating initial teams: the number of teams that is generally generated randomly, depends on the exact nature of the problem’s variables. An example of a team with 9 players has been shown in Fig. 2. A team consists of a number of players with different roles. Every player has his own role that is very important for the algorithm’s good performance and convergence. Lots of various roles can be applied based on the nature of the problem. Every team which is a 1\((N+1)\) array, is a solution for the problem. The above-mentioned array in which \(N\) is a problem dimension and \(p_i\) is the \(i\)th player, is defined as follows

\[
\text{Team} = [p_1, p_2, p_3, ..., p_N, \text{Score}]
\]

2. Classifying: classification or grouping refers to organizing teams into different groups which compete against each other. A lot of methods, such as hamming distance, maximum likelihood, minimum likelihood and so many other methods, can be employed in this stage. The second stage is based on the nature of the problem. Fig. 3 represents eight groups as well as a team that has to be put into one of these groups. The values labeled on edges are resemblance rates between a team and groups. First group’s matches will be held after teams are put into groups.

3. Holding a match: as we know, there are a number of rules in a match, which must be followed. In this stage, how you define rules depends directly on the nature of the problem. Only do we consider one rule for the motif finding problem that is contest termination conditions. Two termination conditions are as follows: 1-time intervals, 2- the number of scoring function callings. Instead of considering “time’s up” a termination condition, we considered ”the number of scoring function callings” the termination of a contest. In a match, the rival teams usually imitate each other’s values and take measures to ameliorate their status by helping their players take more active roles and score more goals. The higher score, the better status. In our proposed algorithm, there is a ball that carries a value and delivers it to the destination. A referee will call the scoring function after a player receives the ball and then he will change his value in case his score improves. The above-mentioned roles (population changes) are as follows:

1. Shooting: shooting is a process in which a player throws a ball toward the rival team’s goal. In our proposed algorithm, the player who shoots throws several balls toward the rival team’s goal; Ball values are considered his teammate’s values. The rival team will update its players’ values if obtained values improve the scoring functions; otherwise, changes cannot be applied.

2. Attacking: in the attacking role like shooting role, a player throws the balls toward the rival team with different values which are created randomly. Although attacking and shooting roles are similar in some ways, they are different in other ways. A player in the shooting role chooses his teammate’s values whereas a player in the attacking role sends the randomly created values to the rival team.

3. Passing: passing is one of the main links among players. As we observed in Fig. 5, a player passes the ball to one of the randomly selected teammates and changes his value.

4. Crossing: a cross is a means of a long pass from a player to his teammate. In our proposed algorithm, a cross pass has an angle (the effect of the angle has been depicted in Fig. 4) named \(\alpha\). The more the \(\alpha\) value is, the more the distance range is. We implemented the cross pass as a left rotation as it is shown in Fig. 5.

4. The scoring functions: every team can be considered as an answer to the problem. As we know, there are many teams and each team has its own merits that are different from those of another team. A score value is a value that indicates how a developed answer is close to an optimal answer. As mentioned earlier, a team is composed of some players whose values are a motif starting position in a DNA sequence. The fitness functions...
have the main role in the convergence and correlation of an optimization algorithm and an algorithm either will converge toward a bad solution or will have difficulty in converging in case its design is not appropriate. Its speed is also an issue of the utmost importance, so it must be computed quickly. Considering the nature of the problem, the scoring functions can be defined as the maximization and minimization of the scoring functions. The scoring functions can be defined as follows

\[
\text{Score}(\text{team}) = \text{score}(P_1, P_2, P_3, \ldots, P_N), \text{Score}
\]

For the motif discovery problem, higher score values represent the meritorious teams- in other words, the scoring functions inserted in Eq. (1), is maximization scoring functions for this problem. Fig. 6 shows how a sample of scoring is computed for the motif discovery problem.

\[
\text{Score} = \sum_{i=1}^{l} \max(\text{count}(k, i))
\]

where \(k\) belongs to \(\{A,T,C,G\}\), \(l\) is the length of a motif, \(t\) is the number of a sample of a sequence of DNA.

5. The elimination stages: after first group’s matches are held, the elimination stages in which the most meritorious teams play each other in the final rounds, will begin. A team’s face to face rivals can be determined by disparate methods such as random selection, maximum resemblance, minimum resemblance, maximum likelihood, minimum likelihood and many other methods. These stages will continue until the champion is selected. The weak teams will wait until a new season starts. The weak teams have to change their players’ roles and values at the beginning of a new season. Fig. 7 depicts how an elimination tournament is held for sixteen meritorious teams.

The teams which fail to ascend to the elimination stages or those which are eliminated by the other teams in this stage must prepare for a new season of international contests.

6. The stop condition: as in other optimization algorithms’ case, we can select one of the following options as a termination condition:

1. The ending of predefined numbers of seasons.
2. The passing of the allocated time.
3. Reaching definite precision.
4. Keeping the best score for several seasons.
5. Calling the antiseptic numbers of scoring functions.
6. Using a combination of the above-mentioned options.

2.1. Discussion

Although our proposed algorithm (WCC), outlined above, was inspired by human sports rules, most heuristic algorithms have been inspired by nature. The present paper presents a new evolutionary algorithm. The above-mentioned optimization algorithm can be applied to various fields such as economics, computer science, engineering etc., in which the classical algorithms fail to get acceptable answers in reasonable time order. One of the main steps in determining the optimization algorithms' practical applications is classifying their models into major categories such as continuous optimization versus discrete optimization, unconstrained optimization versus constrained optimization, single-objective optimization versus multi-objective optimization, and deterministic optimization versus stochastic optimization [22]. In discrete optimization, variables often contain a subset of integer values whereas, in continuous optimization, variables contain real values. The proposed algorithm can be used as both discrete and continuous optimization algorithms. While there are some constraints on variables in the constrained category, there are no constraints on variables in the unconstrained category. Such a difference between constrained and unconstrained categories results in the placement of the proposed algorithm (WCC) on an unconstrained category. Most of the optimization algorithms which are known as single-objective optimization algorithms have single-objective functions whereas others which are known as multi-objective optimization algorithms, have multi-objective functions. WCC which is based on the nature of the problem can be applied to either multi-objective or single-objective optimization problems. It is also assumed that WCC is practical for only deterministic optimization problems, in which the nature of problems is invariable, in contrast to stochastic optimization problems with variable natures. The main purpose of this study is to achieve good performance by reducing errors, reach a suitable convergence and appropriate stability, and get an acceptable execution time. A combination of the other optimization algorithms' dominant properties (such as being competitive in an imperialism competitive algorithm, the penalties and rewards of learning automata, and the evolutionary changes in the genetic algorithms) and the unification of them into the proposed algorithm are the other motivations behind the present paper. Creating teams with random values is the first step forward in the proposed algorithm. A team consists of some players that not only have different roles based on the exact nature of the problem but also play major roles in an algorithm's good performance. Teams that are put into different groups compete against each other. The highly qualified teams ascend to the elimination stage and continue competing depending on their players' different tasks. There is a referee who evaluates and scrutinizes teams' performance. If the end result is not acceptable or termination conditions are not fulfilled, a new season will begin. To carry out the present study, we applied WCC to the transcriptional factor binding sites (TFBS) problem which is categorized as an NP-hard problem, then compared it with several algorithms, including the genetic algorithm (GA), the imperialism competitive algorithm (ICA), the particles swarm optimization (PSO), the ant colony algorithm (ACO), and learning automata (LA) that are categorized as optimization algorithms.

Although there are some similarities between WCC and the other evolutionary algorithms, there are some important differences as well. In this section, we're going to compare WCC with the other evolutionary algorithms, so we begin with comparing WCC to the genetic algorithm in detail and indicating its terms in Table 1, along with the other algorithms' terms. The differences between WCC and the genetic algorithm are as follows:
1. All the chromosomes in the genetic algorithm are put into the current population whereas all the teams in WCC are put into different groups based on hamming distance, maximum likelihood, minimum likelihood, random grouping, and many other methods that are employed based on the exact nature of the problem.

2. While there are lots of various operations such as shooting, passing, attacking, kicking and crossing in WCC, there are only two operations in the genetic algorithm, which change the chromosomes that are categorized as crossover and mutation.

3. There are two types of competitions among teams in WCC: 1-the local competition and 2- the global competition. By contrast, there is not any competition among chromosomes in the genetic algorithm. The global competition guarantees that WCC does not remain in the local optimal points.

4. Only the most deserved chromosomes in the genetic algorithm have a chance to pass down to the next generation whereas all the teams in WCC will get a chance to attend the next season of matches while they are trying to ameliorate their performance.

5. In a generation, the genetic operations are applied with a definite rate which is categorized as crossover and mutation rates whereas all of these teams have lots of operations that can be applied in case match termination conditions are fulfilled.

6. A team’s values will be replaced with new values in case the newly earned score is better than the previous one. As we know, in a genetic algorithm, the fitness value for a chromosome is likely to change for the worse after crossover and mutation operations are applied.

7. The genetic algorithm was inspired by natural evolution but WCC was inspired by sports rules and roles which are applied to the games.

LCA (league competition algorithm) is an optimization algorithm which is recently introduced [23]. Although it seems that WCC and LCA are the same, they are two different algorithms and have main differences that are:

1. WCC is inspired by world competition and LCA is inspired by league competition, so they have difference roles and steps.

2. WCC has two steps in optimization which are local optimization and then global optimization, but LCA has only one step in optimization that is global optimization.

3. WCC has various operations such as shooting, crossing, passing and so on whereas there are operations like mutation and crossover in LCA.

4. The scoring function is not the same in two algorithms. There is no winner and loser in WCC while winner and loser are introduced in LCA. Winner gets 3 scores and loser does not get any score. Also, scores determine to play chance in WCC whereas scores have not any influence in competition in LCA.

5. A person who evaluates the player’s functionality is the referee and the referee evaluates each operation of players and gives them score in WCC, but a person who evaluates team performance is a coach and evaluates team functionality overall.

6. We can use from artificial intelligence methods in grouping or clustering of teams in WCC whereas there is no grouping in LCA and all teams are placed in one group and compete based on schedule table.

7. There is transfer operation at the end of a season in LCA, but this operation is not available in WCC and instead of, every team tries to get some players who were not participated in the world cup and now they can proper functionality relative to current players. Then the mentioned players are superseded with the players who have not a good performance.

8. There is elimination stage in WCC whereas this stage is not available in LCA.

In Table 1, the terms which are available to famous optimization algorithms are shown.

### Table 1

Parameters set for famous optimization algorithms.

| Algorithm name | Population individuals | Operations on population | Evaluation function | Inspired by |
|----------------|------------------------|--------------------------|---------------------|-------------|
| WCC            | Team                   | Shoot, pass, cross, attack | Scoring             | International sport matches |
| GA             | Chromosome             | Crossover, mutation, position and velocity update | Fitness             | Natural revolution |
| PSO            | Particle               | Absorb, revolution       | Solution evaluation | Social of birds |
| ICA            | Country                | Evaporation & Laying pheromone | Cost               | Imperialism competition |
| ACO            | Ant                    | Set of finite actions    | Fitness             | Behavior of Ants |
| LA             | Stochastic automata    |                           | Computable function | Cellular automata |

3. Experimental results

In this section, we are going to conduct two experiments in order to compare the proposed algorithm with the other famous algorithms. First, we scrutinize the TFBS problem. Then we examine the proposed algorithm’s performance on standard datasets.

### Table 2

Algorithms’ parameters for the randomly created datasets.

| Algorithm name | Parameters |
|----------------|------------|
| WCC            | Number of teams = 30, number of seasons = 10, number of match trying = 35, number of groups = 2 |
| PSO            | Number of particles = 200, number of iteration = 100, number of best p = 5 |
| ACO            | Number of ants = 200, number of iteration = 60, pheromones actor = 5, evaporation factor = 5 |
| LA             | Size of boundary operation = 10, number of iteration = 8000 |
| ICA            | Number of countries = 1000, number of imperialism = 10, number of iteration = 1 |
| GA             | Size of first population = 200, number of generation = 100, mutation and crossover rate = 30% |

### Table 3

Algorithms’ parameters for the real datasets.

| Algorithm name | Parameters |
|----------------|------------|
| WCC            | Number of teams = 100, number of season = 20, number of match trying = 50, number of groups = 10 |
| PSO            | Number of particles = 400, number of iteration = 400, number of best p = 10 |
| ACO            | Number of ants = 750, number of iteration = 150, pheromones actor = 5, evaporation factor = 5 |
| LA             | Size of boundary operation = 10, number of iteration = 8000 |
| ICA            | Number of countries = 1000, number of imperialism = 7, number of iteration = 10 |
| GA             | Size of first population = 400, number of generation = 1000, mutation and crossover rate = 30% |
Table 4
obtained results from the randomly created datasets.

| Algorithm | Motif Length — 10 | Motif Length — 11 | Motif Length — 12 |
|-----------|------------------|------------------|------------------|
|           | Best | Worst | Avg | SD | Best | Worst | Avg | SD | Best | Worst | Avg | SD | Best | Worst | Avg | SD | Best | Worst | Avg | SD |
| GA        | FIT 57 | 49 | 52.6 | 75.1 | 91 | 79 | 85.4 | 10.69 | 126 | 111 | 118.9 | 13.44 | 152 | 141 | 148.3 | 11.04 |
|           | TIME 0.82 | 1.24 | 0.92 | 0.39 | 1.63 | 2.18 | 1.85 | 0.57 | 1.73 | 2.77 | 2.18 | 0.87 | 2.57 | 3.29 | 2.82 | 0.7 |
| LA        | FIT 65 | 60 | 62.8 | 5.79 | 115 | 107 | 110.3 | 4.94 | 155 | 145 | 149 | 8.6 | 183 | 171 | 178 | 10.29 |
|           | TIME 1.98 | 1.16 | 1.09 | 0.37 | 1.45 | 1.37 | 1.77 | 0.72 | 1.57 | 2.36 | 1.99 | 0.75 | 1.78 | 2.98 | 2.4 | 11 |
| WCC       | FIT 65 | 60 | 62.2 | 4.52 | 124 | 116 | 120.5 | 8.51 | 182 | 167 | 174.3 | 14.57 | 233 | 221 | 226.4 | 11.24 |
|           | TIME 0.99 | 1.05 | 1.02 | 0.06 | 1.43 | 1.72 | 1.55 | 0.26 | 1.57 | 1.79 | 1.69 | 0.22 | 1.89 | 2.23 | 2.01 | 0.34 |
| PSO       | FIT 65 | 53 | 59.3 | 5.99 | 117 | 109 | 111.7 | 14.76 | 171 | 156 | 164 | 15.87 | 228 | 211 | 215.7 | 14.49 |
|           | TIME 1.07 | 1.43 | 1.12 | 0.14 | 1.27 | 1.17 | 1.91 | 0.61 | 1.89 | 2.31 | 1.64 | 0.54 | 2.11 | 2.39 | 2.61 | 0.7 |
| ACO       | FIT 63 | 58 | 60.4 | 5.13 | 120 | 111 | 114.6 | 8.02 | 171 | 160 | 166 | 11.48 | 228 | 207 | 214 | 19.08 |
|           | TIME 0.97 | 1.04 | 0.98 | 0.14 | 1.69 | 2.13 | 1.86 | 0.38 | 1.83 | 2.38 | 2.13 | 0.51 | 2.45 | 3.04 | 2.75 | 0.45 |
| ICA       | FIT 57 | 51 | 55.1 | 6.23 | 104 | 88 | 93.1 | 15.12 | 141 | 123 | 129.7 | 15.23 | 182 | 160 | 169.5 | 22.81 |
|           | TIME 1.01 | 1.07 | 1.03 | 0.06 | 1.4 | 1.67 | 1.57 | 0.23 | 1.51 | 1.78 | 1.66 | 0.31 | 1.87 | 2.33 | 1.99 | 0.41 |

benchmark functions and compare it with FOA (Forest Optimization Algorithm), PSO (Particles Swarm Optimization Algorithm), GA (Genetic Algorithm). To carry out the study, WCC, the ant colony optimization algorithm (ACO), PSO, learning automata (LA), GA, and the imperialist competitive algorithm (ICA) are implemented in the MATLAB environment in a system with a 2.2 GHz of CPU, 2 GB of RAM and WINT (64 bit) operating system, then four datasets are created randomly. We employ four real datasets [24] and present the obtained results in tables and diagrams. A web site is also available, "bwbccir". You can find some information about implemented algorithms and applied databases on the web. The above-mentioned algorithms run 100 times for each dataset with a different size of the motif length. The best, the worst, and the average score values, the elapsed time (based on second), the score values standard deviation, and the elapsed time standard deviation (based on second) are determined. The standard deviation (SD) that is commonly used for algorithms’ confidence is a criterion that shows how much data scattered around the mean. The fewer SD is, the more reliable algorithms are. The algorithms’ parameters for the randomly created datasets are shown in Table 2 and the algorithms’ parameters for the real datasets in Table 3. Moreover, the obtained results from both the randomly created datasets and the real datasets are shown in Tables 4 and 5, respectively. The figures are shown in Tables 4 and 5 indicate that dataset 1 size is 10 × 100, dataset 2 size is 20 × 200, dataset 3 size is 30 × 300, and dataset 4 size is 40 × 400. According to figures, all the real datasets are the same size (in a size 40 × 60). As we mentioned earlier, the conditions for all algorithms are so similar that they are compared to the same terms and conditions. The end of the predefined numbers of iterations or seasons will bring an end to the algorithms.

Figs. 9 and 8 depict Tables 4 and 5 as diagrams with over 100 times of independent execution. The horizontal pivot represents amounts of execution and the vertical pivot represents the fitness values. The best, the worst, the average and the standard deviation which is inserted in Tables 3 and 4, are obtained from over 100 times of independent execution for each algorithm with different motif sizes. Bold values for every group in Tables 3 and 4 indicate the best results compared with the other algorithms. Fig. 8 shows
the average stability of algorithms for all randomly created and all real datasets by implementing over 100 times of individual execution. Algorithms’ stability which indicates a fluctuation in the current outcome compared with the previous and subsequent outcomes, is one of the main criteria for a comparison between meta-heuristic algorithms. From this point of view, an algorithm whose results are in a straight forward line with maximum fitness values and a minimum fluctuation index is considered as a proper algorithm [25]. The error bars which indicate the validity of the algorithms are represented in Fig. 8. The above-mentioned criteria determines how similar or different measurements are. The error bars with a minimum height show the suitability of an algorithm as the benchmark that can be used in scientific papers with experimental results shown on graphs [26].

An algorithm’s convergence is considered another criterion for a comparison between evolutionary algorithms. When the number of seasons, iterations or generations are augmented or when the allocated time to algorithms is augmented, the heuristic algorithms have to approach an optimal answer. An example of the above-mentioned algorithms’ convergence proved that these algorithms have a proper convergence. Since the number of generations or the number of iterations is not identical for the algorithms, we have to employ the allocated time (or the number of fitness functions callings) to compare different algorithms. The convergence of algorithms for all randomly created datasets as well as all real datasets is depicted in Fig. 9. In Fig. 9, the horizontal pivot represents the number of fitness functions callings and the vertical pivot represents the fitness values. Dash lines in Fig. 9 relate to the mean values of algorithms. The algorithms which are at the top of the dash lines are definitely better than the algorithms which are at the bottom of the dash lines.

In conclusion, we’d like to refer to some practical results of the present paper:

1. Table average values shown in Table 6 which depict the mean values of all randomly created and all real datasets provide a solid base for the proposed algorithm’s good performance. Two bold values in every column indicate the best state.

### Table 5: Obtained results from the real datasets.

| Algorithm | Dataset5 | Best | Worst | Avg | SD |
|-----------|----------|------|-------|-----|----|
| GA        | 218      | 202  | 210.90| 17.28| 224 |
| TIME      | 23.34    | 30.58| 29.84 | 1.09 | 29.82 |
| LA        | 303      | 286  | 293.30| 22.62| 314 |
| TIME      | 15.65    | 15.92| 15.82 | 0.38 | 15.47 |
| WCC       | 312      | 295  | 303   | 19.69| 314 |
| TIME      | 17.27    | 18.29| 17.62 | 0.88 | 16.75 |
| PSO       | 279      | 263  | 273.70| 15.03| 259 |
| TIME      | 14.08    | 14.57| 14.35 | 0.57 | 14.05 |
| ACO       | 314      | 284  | 298.90| 31.25| 307 |
| TIME      | 15.57    | 16.33| 15.96 | 0.88 | 15.84 |

| Algorithm | Dataset6 | Best | Worst | Avg | SD |
|-----------|----------|------|-------|-----|----|
| GA        | 218      | 202  | 210.90| 17.28| 224 |
| TIME      | 23.34    | 30.58| 29.84 | 1.09 | 29.82 |
| LA        | 303      | 286  | 293.30| 22.62| 314 |
| TIME      | 15.65    | 15.92| 15.82 | 0.38 | 15.47 |
| WCC       | 312      | 295  | 303   | 19.69| 314 |
| TIME      | 17.27    | 18.29| 17.62 | 0.88 | 16.75 |
| PSO       | 279      | 263  | 273.70| 15.03| 259 |
| TIME      | 14.08    | 14.57| 14.35 | 0.57 | 14.05 |
| ACO       | 314      | 284  | 298.90| 31.25| 307 |
| TIME      | 15.57    | 16.33| 15.96 | 0.88 | 15.84 |

### Table 6: Obtained results from the real datasets.

| Algorithm | Dataset5 | Best | Worst | Avg | SD |
|-----------|----------|------|-------|-----|----|
| GA        | 218      | 202  | 210.90| 17.28| 224 |
| TIME      | 23.34    | 30.58| 29.84 | 1.09 | 29.82 |
| LA        | 303      | 286  | 293.30| 22.62| 314 |
| TIME      | 15.65    | 15.92| 15.82 | 0.38 | 15.47 |
| WCC       | 312      | 295  | 303   | 19.69| 314 |
| TIME      | 17.27    | 18.29| 17.62 | 0.88 | 16.75 |
| PSO       | 279      | 263  | 273.70| 15.03| 259 |
| TIME      | 14.08    | 14.57| 14.35 | 0.57 | 14.05 |
| ACO       | 314      | 284  | 298.90| 31.25| 307 |
| TIME      | 15.57    | 16.33| 15.96 | 0.88 | 15.84 |

| Algorithm | Dataset6 | Best | Worst | Avg | SD |
|-----------|----------|------|-------|-----|----|
| GA        | 218      | 202  | 210.90| 17.28| 224 |
| TIME      | 23.34    | 30.58| 29.84 | 1.09 | 29.82 |
| LA        | 303      | 286  | 293.30| 22.62| 314 |
| TIME      | 15.65    | 15.92| 15.82 | 0.38 | 15.47 |
| WCC       | 312      | 295  | 303   | 19.69| 314 |
| TIME      | 17.27    | 18.29| 17.62 | 0.88 | 16.75 |
| PSO       | 279      | 263  | 273.70| 15.03| 259 |
| TIME      | 14.08    | 14.57| 14.35 | 0.57 | 14.05 |
| ACO       | 314      | 284  | 298.90| 31.25| 307 |
| TIME      | 15.57    | 16.33| 15.96 | 0.88 | 15.84 |
2. The GA algorithm is of great research value because it provides a solid base for the other evolutionary algorithms’ good performance. However, its results are worse than the others’.

3. Although LA has good performance for small datasets and lies at the top of the dash lines, it is not suitable for big datasets because it lies at the bottom of the dash lines.

4. ICA is better than GA, although it does not have proper performance compared with the other algorithms. As we observed earlier, it lies at the bottom of the dash lines.

5. There is a close competition among ACO, PSO, and WCC and all of them are at the top of the dash lines. WCC is better than two others and ACO is relatively better than PSO.
In another experiment, in order to examine the real quality of the proposed algorithm, we employ eight benchmark functions listed in Table 7 [27,28]. Benchmark functions’ names, variable ranges, optimum target values, benchmark functions equations, and the number of variables are all presented in Table 7. These functions can be used either as maximization or as minimization problems. We consider them to be minimization problems.

F1, shown in Fig. 10, is a benchmark function with a minimum value $f = -18.5547$ for $x = 9.0389$ and $y = 8.6678$.

First, we list the results of test functions, then describe F1 in detail. We select 10 teams, 40 playing chances in a match, 4 groups and 9 sessions for WCC, along with roles that we mentioned earlier. Fig. 11 depicts not only teams’ numerical values but also how players’ values are changing. In Fig. 11, there is a table next to each picture; every table shows the exact value of a player. In addition, scores values are available. The best team is bolded after a session. The obtained values are considerable after 1026 callings of the scoring functions.
Table 6
The average results of all randomly created datasets and all real datasets.

| Algorithm | All random datasets | All real datasets | All datasets |
|-----------|---------------------|------------------|--------------|
| F       | Best | Worst | Avg | SD | Best | Worst | Avg | SD | Best | Worst | Avg | SD |
| GA      | FIT  | 123.25 | 112.25 | 117.46 | 11.26 | 243.81 | 222.18 | 231.92 | 21.41 | 191.15 | 174.16 | 181.94 | 17.00 |
| TIME    | 1.10 | 1.38 | 1.18 | 0.26 | 31.11 | 32.41 | 31.63 | 1.29 | 17.08 | 17.91 | 17.39 | 0.82 |
| LA      | FIT  | 148.018 | 137.00 | 142.43 | 11.23 | 319.00 | 290.25 | 302.99 | 27.28 | 243.56 | 222.69 | 232.18 | 20.10 |
| TIME    | 1.23 | 1.51 | 1.33 | 0.28 | 16.51 | 17.33 | 16.96 | 0.80 | 9.38 | 9.96 | 9.67 | 0.92 |
| WCC     | FIT  | 168.18 | 156.37 | 162.33 | 11.29 | 328.37 | 303.37 | 316.28 | 23.09 | 258.54 | 239.35 | 249.19 | 17.91 |
| TIME    | 1.22 | 1.30 | 1.25 | 0.10 | 17.67 | 18.28 | 17.87 | 0.59 | 9.99 | 10.36 | 10.12 | 0.36 |
| PSO     | FIT  | 161.68 | 149.25 | 154.95 | 12.48 | 303.18 | 276.62 | 289.40 | 24.85 | 241.91 | 221.58 | 231.21 | 19.44 |
| TIME    | 1.30 | 1.50 | 1.38 | 0.23 | 15.02 | 15.68 | 15.25 | 0.63 | 8.63 | 9.08 | 8.79 | 0.49 |
| ACO     | FIT  | 160.37 | 148.81 | 154.58 | 12.13 | 323.75 | 296.62 | 310.78 | 26.55 | 252.17 | 231.98 | 242.39 | 20.17 |
| TIME    | 1.24 | 1.46 | 1.33 | 0.24 | 16.57 | 17.52 | 17.00 | 0.93 | 9.42 | 10.03 | 9.69 | 0.61 |
| ICA     | FIT  | 142.50 | 131.93 | 136.33 | 10.10 | 298.12 | 281.37 | 289.64 | 16.27 | 229.62 | 215.41 | 222.03 | 13.69 |
| TIME    | 1.17 | 1.36 | 1.22 | 0.23 | 16.36 | 17.06 | 16.70 | 0.57 | 9.38 | 9.74 | 9.48 | 0.42 |

Table 7
Test functions.

| Function | Name  | Equation | Search range | Dimension | Optimum |
|----------|-------|----------|--------------|-----------|---------|
| F1       | Ghaemi | \(x\sin(4x) + 1.1y\sin(2y)\) | \(0 < x, y < 10\) | 2 | -18.5547 |
| F2       | Griewangk | \(\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos \left(\frac{x_i}{\sqrt{i}}\right) + 1\) | \(-600 \leq x_i \leq 600\) | 5, 10 | 0 |
| F3       | SODPF | \(\sum_{i=1}^{n} |x_i|^{i+1}\) | \(-1 \leq x_i \leq 1\) | 5, 10 | 0 |
| F4       | Rastrigin | \(10n + \sum_{i=1}^{n} \left[ x_i^2 - 10 \cos \left(2\pi x_i\right) \right] \) | \(-5.12 \leq x_i \leq 5.12\) | 5, 10 | 0 |
| F5       | step | \(\sum_{i=1}^{n} (x_i + 0.5)^2\) | \(-100 \leq x_i \leq 100\) | 5, 10 | 0 |
| F6       | Schwefel’s p.2.22 | \(\sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i|\) | \(-10 \leq x_i \leq 10\) | 5, 10 | 0 |
| F7       | Quadric noise | \(\sum_{i=1}^{n} x_i^4 + \text{rand}(0,1)\) | \(-1.28 \leq x_i \leq 1.28\) | 5, 10 | 0 |
| F8       | quadric | \(\sum_{i=1}^{n} (x_i^2 - 1)^2\) | \(-100 \leq x_i \leq 100\) | 5, 10 | 0 |

Fig. 10. 3D diagram of F1 [29].

GA, PSO [30] and the forest optimization algorithm (FOA) [31], are among algorithms which have been compared to WCC in the second experiment. The experimental results obtained from over 30 separate execution, are listed in Table 7 as follows: benchmark functions’ alias names, the number of variables, algorithms’ names, the first population, the level of acceptable accuracy, the number of fitness functions callings in which an algorithm reaches a definite level of accuracy at its worst, at its best, and its average states, the average of obtained results, the average of obtained results and the elapsed time in which an algorithm is successful in reaching the level of accuracy. Reaching definite accuracy is considered a stop condition (Table 8).

In order to compare the above-mentioned algorithms, we will use \(F_k(i)\), where \(k\) is the number of benchmark functions and \(i\) is a dimension or the number of variables. As you can see in Table 3, GA does not have proper performance, so it is not comparable with the other algorithms. Since there is a close competition among WCC, PSO, and FOA, they can be compared in two ways:

First, we compare the algorithms with an average number of evaluation functions callings. The results are as follows: FOA is better than the others for \(F1(2), F2(10), F3(5), F4(5), F5(5)\) and \(F7(5)\); PSO is better than the other algorithms for \(F2(5)\) and \(F7(10)\); and WCC is better than the others for \(F3(10), F4(10), F5(10), F6(5), F6(10), F8(5)\) and \(F8(10)\).

Then we compare the algorithms with an average number of the elapsed time, with the result that GA is better than the others for \(F2(5)\); PSO is better than the others for \(F3(10), F5(10), F6(5), F6(10), F8(5)\) and \(F8(10)\).

In conclusion, we have to pay attention to a kind of trade between the elapsed time and the number of evaluation functions callings. The first method is appropriate for problems that need more time for evaluation functions, and the second method is appropriate for problems that need less time for evaluation functions.
|      | P1     | P2     | Score   |
|------|--------|--------|---------|
| a    | 0.9048 | 1.9597 | -1.9284 |
|      | 5.5542 | 7.9049 | -2.1280 |
|      | 5.2086 | 3.4503 | 6.96530 |
|      | 2.6671 | 5.4486 | 8.4902  |
|      | 0.9868 | 5.7701 | -6.1402 |
|      | 1.5889 | 1.8360 | -0.9069 |
|      | 1.5565 | 0.0118 | -0.0888 |
|      | 1.5002 | 8.8374 | -9.3867 |
|      | 5.3096 | 8.2439 | -2.7466 |
|      | 0.3330 | 8.4251 | -8.1064 |
| b    | 7.6037 | 9.3280 | -8.3754 |
|      | 9.1780 | 8.4878 | -16.5698 |
|      | 7.6018 | 8.3885 | -14.5217 |
|      | 9.2203 | 8.7835 | -15.9905 |
|      | 2.3173 | 8.5285 | -8.7920 |
|      | 5.8766 | 5.7574 | -11.3660 |
|      | 7.5371 | 5.6705 | -13.0618 |
|      | 8.8860 | 8.8334 | -16.4057 |
|      | 8.9829 | 8.9773 | -16.5140 |
|      | 7.4904 | 8.4251 | -15.8695 |
| c    | 7.6037 | 2.2152 | -8.7431 |
|      | 9.1780 | 8.4878 | -16.5698 |
|      | 7.6018 | 8.3885 | -14.5217 |
|      | 8.9486 | 8.6694 | -17.9731 |
|      | 5.8766 | 5.6500 | -11.7969 |
|      | 7.5371 | 8.7068 | -16.6833 |
|      | 8.8860 | 8.8334 | -16.4057 |
|      | 8.9829 | 8.9773 | -16.5140 |
|      | 8.9118 | 8.4251 | -16.3294 |
| d    | 7.6037 | 8.8196 | -15.4805 |
|      | 9.1175 | 8.6147 | -18.0554 |
|      | 7.6018 | 8.3885 | -16.4217 |
|      | 7.4260 | 8.5864 | -16.7442 |
|      | 8.9486 | 8.6694 | 17.9731 |
|      | 7.5019 | 5.4253 | -13.3086 |
|      | 7.5371 | 8.7068 | -16.6833 |
|      | 8.8860 | 8.8334 | -16.4057 |
|      | 8.9829 | 8.9773 | -16.5140 |
|      | 8.9118 | 8.4251 | -16.3294 |

**Fig. 11.** The location and values of teams for some of the iterations.
4. Conclusion

In the current study, a new evolutionary algorithm, known as WCC, which was inspired by human sports rules was introduced. As we know, there are lots of interesting events such as competitions, award ceremonies, victory parties and so many others in sports. Every team has a set of rules and some players who are put into different groups; every player performs a different task. Different teams compete against each other and qualified ones ascend to the next stage and continue their competition. At the end of a season, a champion is introduced and the other teams prepare for a new season. We evaluated the performance of the proposed algorithm on a motif discovery problem and eight benchmark functions. Moreover, we compared WCC with GA, ICA, PSO, ACO and LA, based on the above-mentioned problems. The experimental and numerical results indicated that the proposed algorithm’s performance was better than the other algorithms’ performance. The proposed algorithm had reasonable execution time, acceptable precision, and more satisfactory results in comparison to the other algorithms. According to our results, WCC was

| Function | Dim | Algorithm | Population | Worst | Best | Num_eval | Acc | Result | Iter | el.time |
|----------|-----|-----------|------------|-------|------|----------|-----|--------|------|---------|
| F1       | 2   | WCC       | 10         | 33,723| 13,301| 23,558.1| 0.001| – 18.55457| 7    | 0.4929  |
| F1       | 2   | FOA       | 30         | 3281  | 368   | 1092.9   | 0.001| – 18.55457| 141.6| 2.995   |
| F1       | 2   | PSO       | 30         | 31,862| 12,567| 22,288.3| 0.001| – 18.55457| 104  | 0.5217  |
| F1       | 2   | GA        | 30         | 62,937| 24,823| 40,000.5| 0.5  | 0.0312  | 1402.3| 0.3836  |
| F2       | 5   | WCC       | 40         | 9172  | 1804  | 8471.3   | 0.5  | 0.3590  | 71   | 0.3877  |
| F2       | 5   | FOA       | 30         | 15,043| 4400  | 8847.3   | 0.5  | 0.3590  | 124.2| 0.8069  |
| F2       | 5   | PSO       | 30         | 10,440| 2640  | 4320.3   | 0.5  | 0.3590  | 34.9 | 0.0991  |
| F2       | 5   | GA        | 30         | 11,613| 4806  | 9911.3   | 0.5  | 0.4111  | 324.53| 0.2227  |
| F3       | 5   | WCC       | 10         | 1147  | 64    | 357      | 0.001| 0.0003  | 1.4  | 0.2217  |
| F3       | 5   | FOA       | 30         | 881   | 108   | 351.9    | 0.001| 0.0003  | 44.9 | 0.0991  |
| F3       | 5   | PSO       | 30         | 3680  | 660   | 1756     | 0.001| 0.0005  | 29.26| 0.2143  |
| F3       | 5   | GA        | 30         | 1550  | 601   | 720.3    | 0.001| 0.0001  | 24.8 | 0.0803  |
| F4       | 5   | WCC       | 15         | 6775  | 3261  | 4747.3   | 0.001| 0.0001  | 1.85 | 0.0227  |
| F4       | 5   | FOA       | 30         | 5516  | 2655  | 3865.4   | 0.001| 0.0001  | 610.73| 0.2299  |
| F4       | 5   | PSO       | 30         | 52,410| 25,227| 36,722.4 | 0.001| 0.0001  | 611.01| 2.1838  |
| F4       | 5   | GA        | 30         | 1147  | 64    | 357      | 0.001| 0.0003  | 1.4  | 0.2217  |
| F5       | 5   | WCC       | 10         | 2387  | 1686  | 2218.9   | 0.001| 0.0001  | 5    | 0.2809  |
| F5       | 5   | FOA       | 30         | 4065  | 1129  | 2220.6   | 0.001| 0.0067  | 651.80| 0.3501  |
| F5       | 5   | PSO       | 30         | 12,180| 6000  | 8331     | 0.001| 0.0084  | 137.85| 0.2039  |
| F5       | 5   | GA        | 100        | 7959  | 4961  | 6202.8   | 0.001| 0.0001  | 61.01| 0.2138  |
| F6       | 5   | WCC       | 15         | 824   | 51    | 433.75   | 0.001| 0.0049  | 1    | 0.0561  |
| F6       | 5   | FOA       | 30         | 455   | 200   | 266.15   | 0.001| 0.0046  | 44.45| 0.1091  |
| F6       | 5   | PSO       | 30         | 3150  | 2950  | 3050     | 0.001| 0.0030  | 1.85 | 0.0227  |
| F6       | 5   | GA        | 100        | 15,465| 3876  | 10,667   | 0.001| 0.0093  | 404.5 | 0.9385  |
| F7       | 5   | WCC       | 15         | 2387  | 1686  | 2218.9   | 0.001| 0.0001  | 71   | 0.3846  |
| F7       | 5   | FOA       | 30         | 4065  | 1129  | 2220.6   | 0.001| 0.0001  | 1288.7| 0.2059  |
| F7       | 5   | PSO       | 30         | 12,180| 6000  | 8331     | 0.001| 0.0084  | 137.85| 0.2039  |
| F7       | 5   | GA        | 100        | 7959  | 4961  | 6202.8   | 0.001| 0.0001  | 61.01| 0.2138  |
| F8       | 5   | WCC       | 15         | 824   | 51    | 433.75   | 0.001| 0.0049  | 1    | 0.0561  |
| F8       | 5   | FOA       | 30         | 455   | 200   | 266.15   | 0.001| 0.0046  | 44.45| 0.1091  |
| F8       | 5   | PSO       | 30         | 3150  | 2950  | 3050     | 0.001| 0.0030  | 1.85 | 0.0227  |
| F8       | 5   | GA        | 100        | 15,465| 3876  | 10,667   | 0.001| 0.0093  | 404.5 | 0.9385  |

Table 8
Experimental results for benchmark functions.
better than the others and from the perspective of the elapsed time standard deviation, it was better than all the other algorithms. The WCC’s elapsed time was a little longer than the others’, except GA, however if we consider the WCC’s elapsed time a termination condition for the other algorithms, its obtained results will be better than those of the others, too. The ICA’s results standard deviation was the best in comparison to the others’, but as we observed earlier, its best result was worse than the ACO’s. A trade discovery between WCC parameters that are as follows: playing chances, the number of groups, the number of teams and the number of seasons is an important topic for future research or combining WCC with the other metaheuristic algorithms and applying it to different optimization problems can be considered an interesting topic for research.

References

[1] John H. Holland, adaptation in natural and artificial systems. Cambridge, MA: MIT Press; 1992.
[2] Ugolotti R, et al. Particle swarm optimization and differential evolution for model-based object detection. Appl Soft Comput 2013;13(6):3092–105.
[3] Li LK, Shao S, Yiu K-FC. A new optimization algorithm for optimal spatial sampling during non-destructive testing of concrete structures. Eng Struct 2015;88:92–9.
[4] Fong CW, et al. A new hybrid imperialist swarm-based optimization algorithm for university timetabling problems. Inf Sci 2014;283:1–21.
[5] Gosciniak I. A new approach to particle swarm optimization algorithm. Expert Syst Appl 2015;42(2):844–54.
[6] Zhang L, et al. A new particle swarm optimization algorithm with adaptive inertia weight based on Bayesian techniques. Appl Soft Comput 2015;28:138–49.
[7] Gomez-Cardenas C, et al. New optimization algorithm for optimal spatial sampling during non-destructive testing of concrete structures. Eng Struct 2015;88:92–9.
[8] Høng JC, et al. Identification of new human cadherin genes using a combination of protein motif search and gene finding methods. J Mol Biol 2004;337(2):307–17.
[9] Gutierrez JB, Frith M, Nakai K. A genetic algorithm for motif finding based on statistical significance. In: Bioinformatics and Biomedical Engineering. Springer; 2015. p. 438–49.
[10] Höng JC, et al. Identification WCC as we observed earlier, its best result was worse than the ACO’s. A trade discovery between WCC parameters that are as follows: playing chances, the number of groups, the number of teams and the number of seasons is an important topic for future research or combining WCC with the other metaheuristic algorithms and applying it to different optimization problems can be considered an interesting topic for research.

References

[1] John H. Holland, adaptation in natural and artificial systems. Cambridge, MA: MIT Press; 1992.
[2] Ugolotti R, et al. Particle swarm optimization and differential evolution for model-based object detection. Appl Soft Comput 2013;13(6):3092–105.
[3] Li LK, Shao S, Yiu K-FC. A new optimization algorithm for optimal spatial sampling during non-destructive testing of concrete structures. Eng Struct 2015;88:92–9.
[4] Fong CW, et al. A new hybrid imperialist swarm-based optimization algorithm for university timetabling problems. Inf Sci 2014;283:1–21.
[5] Gosciniak I. A new approach to particle swarm optimization algorithm. Expert Syst Appl 2015;42(2):844–54.
[6] Zhang L, et al. A new particle swarm optimization algorithm with adaptive inertia weight based on Bayesian techniques. Appl Soft Comput 2015;28:138–49.
[7] Gomez-Cardenas C, et al. New optimization algorithm for optimal spatial sampling during non-destructive testing of concrete structures. Eng Struct 2015;88:92–9.
[8] Høng JC, et al. Identification of new human cadherin genes using a combination of protein motif search and gene finding methods. J Mol Biol 2004;337(2):307–17.
[9] Gutierrez JB, Frith M, Nakai K. A genetic algorithm for motif finding based on statistical significance. In: Bioinformatics and Biomedical Engineering. Springer; 2015. p. 438–49.
[10] Höng JC, et al. Identification WCC as we observed earlier, its best result was worse than the ACO’s. A trade discovery between WCC parameters that are as follows: playing chances, the number of groups, the number of teams and the number of seasons is an important topic for future research or combining WCC with the other metaheuristic algorithms and applying it to different optimization problems can be considered an interesting topic for research.

References

[1] John H. Holland, adaptation in natural and artificial systems. Cambridge, MA: MIT Press; 1992.
[2] Ugolotti R, et al. Particle swarm optimization and differential evolution for model-based object detection. Appl Soft Comput 2013;13(6):3092–105.
[3] Li LK, Shao S, Yiu K-FC. A new optimization algorithm for optimal spatial sampling during non-destructive testing of concrete structures. Eng Struct 2015;88:92–9.
[4] Fong CW, et al. A new hybrid imperialist swarm-based optimization algorithm for university timetabling problems. Inf Sci 2014;283:1–21.
[5] Gosciniak I. A new approach to particle swarm optimization algorithm. Expert Syst Appl 2015;42(2):844–54.
[6] Zhang L, et al. A new particle swarm optimization algorithm with adaptive inertia weight based on Bayesian techniques. Appl Soft Comput 2015;28:138–49.
[7] Gomez-Cardenas C, et al. New optimization algorithm for optimal spatial sampling during non-destructive testing of concrete structures. Eng Struct 2015;88:92–9.
[8] Høng JC, et al. Identification of new human cadherin genes using a combination of protein motif search and gene finding methods. J Mol Biol 2004;337(2):307–17.
[9] Gutierrez JB, Frith M, Nakai K. A genetic algorithm for motif finding based on statistical significance. In: Bioinformatics and Biomedical Engineering. Springer; 2015. p. 438–49.