Ambidextrous Socio-Cultural Algorithms

José Lemus-Romani[1(✉)], Broderick Crawford[1], Ricardo Soto[1], Gino Astorga[2], Sanjay Misra[3], Kathleen Crawford[1], Giancarla Foschino[2], Agustín Salas-Fernández[1], and Fernando Paredes[4]

[1] Pontificia Universidad Católica de Valparaíso, Valparaso, Chile
{jose.lemus.r,juan.salas.f}@mail.pucv.cl, {broderick.crawford, ricardo.soto}@pucv.cl, kathleen.cra@gmail.com
[2] Universidad de Valparaíso, Valparaso, Chile
gino.astorga@uv.cl, giancarla.fos@gmail.com
[3] Covenant University, Ota, Nigeria
sanjay.misra@covenantuniversity.edu.ng
[4] Universidad Diego Portales, Santiago, Chile
fernando.paredes@udp.cl

Abstract. Metaheuristics are a class of algorithms with some intelligence and self-learning capabilities to find solutions to difficult combinatorial problems. Although the promised solutions are not necessarily globally optimal, they are computationally economical. In general, these types of algorithms have been created by imitating intelligent processes and behaviors observed in nature, sociology, psychology and other disciplines. Metaheuristic-based search and optimization is currently widely used for decision making and problem solving in different contexts. The inspiration for metaheuristic algorithms are mainly based on nature's behaviour or biological behaviour. Designing a good metaheurisitcs is making a proper trade-off between two forces: Exploration and exploitation. It is one of the most basic dilemmas that both individuals and organizations constantly are facing. But there is a little researched branch, which corresponds to the techniques based on the social behavior of people or communities, which are called Social-inspired. In this paper we explain and compare two socio-inspired metaheuristics solving a benchmark combinatorial problem.

Keywords: Social-inspired metaheuristics · Socio-cultural inspired metaheuristics · Human-based algorithm · Teaching–learning-based optimization · Twitter optimization · Ambidextrous metaheuristics

1 Introduction

From time to time humanity is faced with various problems and within them is the scarcity of resources or problems of daily life that are difficult to solve by the human mind. In this context, optimization constitutes an alternative to give solution to these problems. In general the optimization methods allow to

© Springer Nature Switzerland AG 2020
O. Gervasi et al. (Eds.): ICCSA 2020, LNCS 12254, pp. 923–938, 2020.
https://doi.org/10.1007/978-3-030-58817-5_65

give solution to engineering problems for which exact or approximate methods can be used. Within the approximate methods we find the metaheuristics that are in charge of giving us good solutions, that eventually can be the optimal one in a limited time of process unlike the use of an exact method that can be non-efficient in computer time.

It cannot be ignored that throughout history there is a direct relationship between technological development and human societies. The human being carries out during his daily life a series of activities that he wants to optimize. Similarly, human societies become a source of inspiration for problem optimization algorithms. This inspiration can be based on the behaviour of societies in the face of external agents that affect them, such as natural phenomena or the same interaction between humans that causes changes in behaviour. Similarly, the human being can have psychological changes considering external agents, as it happens for example with pandemics.

Human behavior can be understood under three perspectives: personal, interpersonal and social. But in practice they are linked since to understand a person's behavior we must investigate his interpersonal relationships and social factors, as well as to understand social relationships we must understand the relationships between people and the differences or similarities with others. Therefore, individual and society are two inseparable worlds [24].

It is important to consider that social and human phenomena must be evaluated considering the cultural context of the moment. For example, wars, pandemics (as is COVID-19 today) or important revolutions like the industrial one or the current digital transformation [6,37,38]. There are various human and social behaviours that can be used as inspiration. One of them is the natural ambition to achieve a better life. Success in life can be different for each person and for this reason we work to achieve some specific objectives. The path they use to achieve success may be different, but these paths generally take into account the experience that some have over others, for that reason some have the intention of learning from the most experts with the aim of improving their skills. On the other hand, each person during their life contacts others and can use their experiences to improve their life [1].

An overview of various existing Metaheuristics is found in [36].

Metaheuristics have abilities both of exploration and exploitation, exploration is the process of visiting entirely new regions of a search space (Solutions), whilst exploitation is the process of visiting those regions within the neighborhood of previously visited solutions. In order to be successful, a search algorithm needs to establish a good ratio between exploration and exploitation. Moreover, designing a good metaheuristic is making a proper trade-off between these two "Forces" [39]. Unfortunately, there is no complete answer to this question up to now [40].

A general outline for a metaheuristic is presented in the Fig. 1 [34].

The trade-off between the exploration: A Multidisciplinary Dilemma. The trade-off between the exploration of new possibilities and the exploitation of old certainties constitutes one of the most basic dilemmas that both individuals

and organizations constantly are facing. In [13] Duncan proposed the notion of Organizational Ambidexterity, it is understood as the ability of firms to do two different things at the same time, such as exploration and exploitation, efficiency and flexibility, and matching and adaptation.

```
Create one or several start solutions, randomly
  While termination criterion not satisfied do
    If intensify then
      Create new solution by exploitation;
    Else
      Generate new solution by exploration;
    End
    Update best found solution (if necessary);
  End
Return Best found solution;
```

Fig. 1. General skeleton of the metaheuristic.

In [10] was presented a short paper focusing in a psycological view of metaheuristics in problem resolution. Problem solving from a creative perspective has two key aspects, from the point of view of psychology, as are convergent and divergent thinking. The first one, which corresponds to critical or convergent thinking, is the congenital processing of information around a common point, in other words, it is an attempt to bring thoughts from different conclusions to a common union or conclusion. Creative or divergent thinking, on the other hand, starts from a common point and moves to a variety of perspectives, thus generating a different idea that works as well or better than previous ideas.

Considering the source of inspiration we find an area little explored, such as those based on human and social relations, called socio-cultural inspired [20,21].

In this paper, an implementation of Teaching - Learning - Based Optimization [29] algorithm is presented, solving the Set Covering Problem. Together with this, its results are compared with another social-inspired algorithm, Twitter Optimization [8], in addition to other techniques present in the literature. The results of this type of technique are promising, besides being an interesting line of research, for the implementation of other social-inspired metaheuristics.

The work is structured in the following way, in Sect. 2 the works related to the social-inspired metaheuristics are detailed, in Sect. 3 the algorithm considered are presented, in Sect. 4 the combinatorial problems is presented, in Sect. 5 the results obtained are shown, and finally in Sect. 6 the conclusions and future works are presented.

2 Related Work

In this section the basic concepts to understand the structure and behavior of the metaheuristics will be given, in addition to giving a brief classification of them, to later give a brief review according to what was presented by Kumar and Anand [20], and thus be able to have a better context of the socio-inspired techniques.

2.1 Metaheuristics

The optimization methods can be classified into two large groups, the exact methods, with which we can obtain the optimal value for a given search space, having as main disadvantage that the computation times are increased in large combinatorial problems. While the other group are the approximate methods, which do not evaluate all the search space. Among the approximate methods are the metaheuristics, which have presented high quality results in reasonable computation times for various problems. Metaheuristics can be classified according to their way of operating, that is, based on population or on a single solution. The Fig. 2 below summarizes the classifications of the optimization methods.

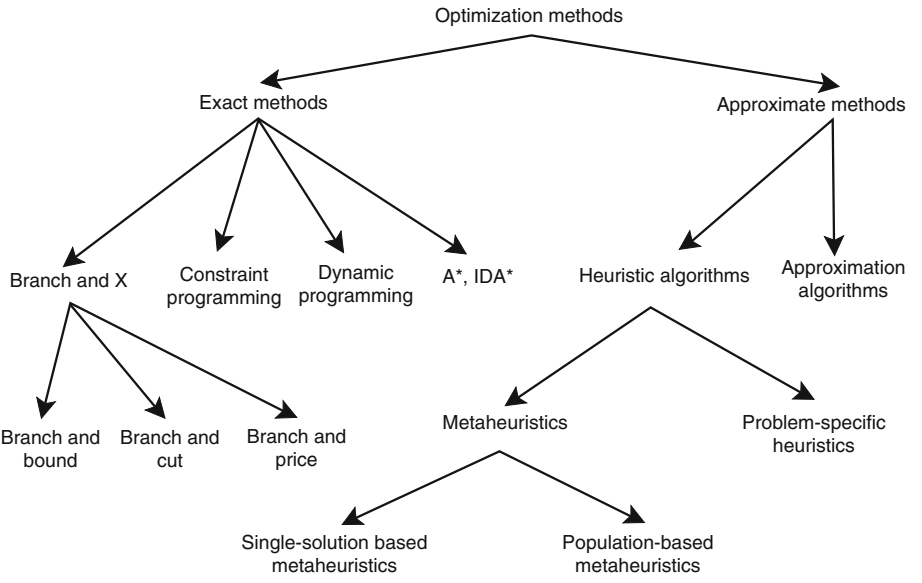

Fig. 2. Optimization methods [35].

Over the years, metaheuristics has become a real alternative solution for everyday problems, for this very reason it becomes an attractive focus for

research using different sources of inspiration such as the processes of biological evolution and natural selection to give rise to Evolution-based, others based on collective behavior or swarming by some animals to give rise to Swarm-based, also others that imitate, there are also some based on physical processes called Physics-based. On the other hand, those inspired by human or social behaviour are called Human-based.

According to Blum and Roli [5], metaheuristics can be defined as an iterative process of generation that guides a subordinate heuristic, generating an efficient combination between exploitation and exploration of the search space, with strategies that structure the information to find almost optimal solutions.

The metaheuristics based on a single solution or also called trajectories, have an approach of exploitation of the search space, where at the level of iteration a solution candidate is evaluated, which was generated by its respective movement or disturbance. While population-based metaheuristics have an exploration approach, which at the iteration level evaluates a set of possible solutions.

2.2 Social-Inspired Algorithms

Social-inspired metaheuristics is a relatively new field, but there are reviews of literature that support its classification. One such review is the Kumar and Anand [20], where they present a classification of four sub-categories of Social-inspired. These categories are detailed in Fig. 3.

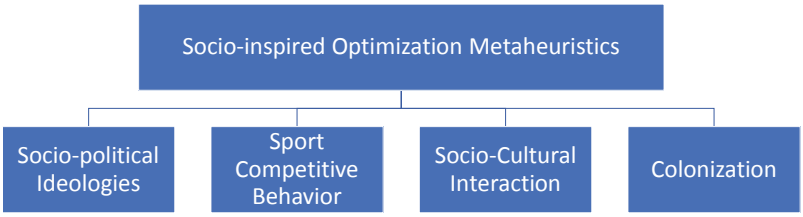

Fig. 3. Classification of socio-inspired algorithms [20].

The first category is Socio-Political Ideologies, among the metaheuristics that we can find this Ideology Algorithm (IA) [16], which is based on the interactions and behavior of individuals belonging to political parties, where each party has a certain political ideology. Election Algorithm (ELA) [14], which is inspired by the functioning of political elections, where this population technique is composed of voters and candidates for public office, these candidates belong to different electoral parties. Election Campaign Algorithm (ECO) [25,26], which bases its behavior on how voters choose the candidate with the best prestige (best quality of objective function).

Another category is Sports Competitive, where two techniques stand out, League Championship Algorithm (LCA) [18], which is based on the behavior

formed in the matches of a league, where a team competes with each other to obtain the championship. Another variation is the Soccer League Competition Algorithm (SLC) [28], which is inspired by the relationships between players and teams in a soccer league, where the population represented by each team competes with each other for the best position in the ranking table, while as in real life, the players of each team compete with each other to define which individuals will belong to the starting team or to the substitutes' team.

The third category is those related to Socio-Cultural, where reference is made to techniques such as Teaching Learning Based Optimization (TLBO) [29], which abstracts the synergy that occurs in classroom teaching, where a teacher influences students and how they learn in community. Twitter Optimization (TO) [27], which is inspired by the behavior of the social network of the same name, where users generate tweets (Solutions), which are also followed among users, which generates celebrities and current users, to determine the operators to use. Cohort Intelligence (CI) [19], is based on the interaction of a group of people with similar qualities, who compete with each other to improve and evolve. Cultural Evolution Algorithm (CEA) [22] which seeks to obtain benefits and improve collectively, leaving individual improvement in the background, with the aim of evolving and surviving through social and multicultural influence. Social Learning Optimization (SLO) [23] is based on three co-spaces in how human intelligence evolves and improves, which are the genetic, cultural and belief levels, which are transmitted to future generations, gradually changing according to cultural changes. Socio-evolution and Learning Optimization Algorithm (SELO) [21], is based on the interaction of family nuclei composed of parents and children, which can improve through the interaction of Father and Son, as in the interaction of Sons with Sons, to obtain individual and collective improvements for each family nucleus. Social Group Optimization (SGO) [31] is based on the human need to overcome the problems for which positive and negative qualities are used, where each person has them in different measures, then it becomes necessary the relationship with other individuals to overcome the problems of life. Also used as a source of inspiration is human behavior that has a diverse variability of behaviors and interests, who try to improve and achieve success by their own knowledge and experience or try to learn from others in order to improve their own experience in life and that of their interest [1,32].

The fourth category corresponds to those based on Colonization, where we have the following techniques. Society and Civilization Algorithm (SCO) [30], which classifies individuals into societies that in turn make up civilization, as is typical of each group of individuals, there are leaders (better fitness), which guide that society, so that each individual tries to imitate their leader, in parallel the leaders of societies can migrate to other societies. Imperialist Competitive Algorithm (ICA) [3,15], which behaves imitating the idea of imperialism, where there are powerful nations that are called empires, while the weaker ones are called colonies, which belong to a certain empire. The determination of the empires and colonies are random at the beginning of the algorithm, but iteratively the colonies begin to be conquered by other empires, until all the nations are colonized by one great empire. Anarchic Society Optimization (ASO) [2] based

on the anarchic behavior of individuals in a society, this algorithm randomly initiates the generation of solutions, which iteratively explores the search space based on three movement policies, which are related to the best value in comparison with the other solutions at the present time, in the best global and a third operator that depends on the fitness history of each individual, which is represented in that the movements will be more anarchic, that is random, when the solution is far from the best results.

3 Socio-Cultural: Teaching–Learning-Based Optimization and Twitter Optimization

This section explains the operation of the social-inspired Teaching-Learning-Based Optimization techniques [29] and Twitter Optimization [27], where the operators inspired by social interactions are described.

3.1 Teaching–Learning-Based Optimization

TLBO is a population method based on the knowledge that a teacher in a classroom that is shared with students improves the knowledge level of the class. Moreover, the students are evaluated by the value of the qualification average of the students in the class. Additionally, the results can be improved with learning that occurs with the interaction between students. The population is composed of a group of students, and the variables constitute the subjects offered; finally, the fitness corresponds to the learning results of the students. In the entire population, the best solution is considered the teacher. TLBO is composed of two phases, the Teacher Phase and the Learner Phase.

Teacher Phase. A random sample and orderly points are generated, which are the learners in the search space. A point considers the wisest person to be a teacher who shares his knowledge with students. It is the first part of the algorithm where the mean of a class increases from M_A to M_B depending upon the strength of the teacher. The teacher must guide the students in such a way that they reach their own level of knowledge. In practice, this is not possible, and a teacher can only move the mean of a class up to a certain extent depending on the capability of the class. This method follows a random process depending on many factors. Let M_i be the mean and T_i be the teacher at any iteration i. T_i will try to move M_i towards its own level; thus, the new mean will be T_i, designated M_new. The solution is updated according to the difference between the existing and the new mean given in Eq. (1):

$$DifferenceMean_i = rand(X_{new} - T_F M_i) \tag{1}$$

Where T_F is a teaching factor that decides the value of the mean to be changed, and rand is a random value between 0 and 1. The value of T_F can be either 1 or 2, which is again a heuristic step and is randomly decided with equal probability using the following Eq. (2) [13]:

$$T_F = round[1 + rand\{2 - 1\}] \tag{2}$$

This difference modifies the existing solution by means of Eq. (3) [14]:

$$x_{new} = x_{old,i} + DifferenceMean_i \tag{3}$$

Learner Phase. The students increase their knowledge by interactions among themselves. A solution is randomly interacted to learn something new with other solutions in the population. A solution will learn new information if the other solutions have more knowledge than him or her. The learning phenomenon of this phase is expressed by Eq. (4):

$$\begin{aligned} x^d_{new} &= x^d_i + rand(x^d_j - x^d_i), \;\; If \quad f(x_i) > f(x_j) \\ x^d_{new} &= x^d_i + rand(x^d_i - x^d_j), \;\; If \quad f(x_i) < f(x_j) \end{aligned} \tag{4}$$

At any solution x, considering two different learners x_i and x_j, where $i \neq j$. Consequently, we accept x_new if it gives a better function value. After a number of sequential teaching learning cycles, the teacher passes on knowledge to the learners, and those levels increase towards his or her own level; the randomness distribution within the search space decreases, close to the point of being considered a teacher. The algorithm converges to a solution when the knowledge level of a class shows smoothness. The term criterion can be the number of evaluations or can reach a maximum number of iterations as previously established. The algorithm is shown in Fig. 4.

3.2 Twitter Optimization

Twitter Optimization is inspired by the interaction between people in the social network Twitter, where each tweet is a feasible solution to evaluate, while following corresponds to the relationship between users, where each user generates tweets which are retweetted by other users, which is reflected in the use of operators defined below. With this interaction between users, which correspond to a set of solutions, and the relationship between tweets and retweeting, the technique manages to explore and exploit the search space.

Therefore each Tweet represents a solution composed of a binary vector taking the value 0 and 1, of size n, as indicated in the Eq. 5

$$x = \{x1, x2, ..., xn\}, \forall i \in 1, 2, ..., n \wedge x_i \in \{0, 1\} \tag{5}$$

The algorithm when initialized generates a graph of the relationships between users. The process starts with each user generating a solution, i.e. a tweet, as indicated in the Eq. 6, then the users are followed at random for the first iteration, where each user can only follow F users. Then for each iteration, users update the list of people F that follows, retweeting the tweet with the best fitness and following its author, if it wasn't followed until this moment, after this the user removes the tweet with the worst fitness to keep F.

$$x^k_i = min_i + (max_i - min_i) \cdot rand(0, 1) \tag{6}$$

Algorithm 1: TLBO

Set k=1;
Objective Function $f(x)$, $x_i = (x^1, x^2..., x^d)^t$ *d=no. of design variables*
Generate initial students of the classroom randomly x^i, *i*=1,2...,*n*
n=no. of students
Calculate objective function $f(x)$ for whole students of the classroom
while the termination conditions are not met **do**
  {teacher phase}
  Calculate the mean of each design variable x_{mean}
  Identify the best solution (teacher)
  for $i = 1 \rightarrow n$ **do**
    Calculate teaching factor $T_F^i = round[1 + rand\{2-1\}]$
    Modify solution based on best solution(teacher)
    $x_{new}^i = X^i + DifferenceMean_i^d$
    Calculate objective function for new mapped student $f(x_{new}^i)$
    if x_{new}^i is better than x_i , *i.e* $f(x_{new}^i) < f(x_i)$ **then**
      $x^i = x_{new}^i$
    end if
    {end of teacher phase}
    {student phase}
    {Randomly select another learner (x_j) , such that j≠i}
    if x^i *is better than* x^j , *i.e* $f(x^i) < f(x^j)$ **then**
      $x_{new}^i = x_i + rand(x_i - x_j)$
    else
      $x_{new}^i = x_i + rand(x_j - x_i)$
    end if
    if x_{new}^i *is better than* x_i , *i.e* $f(x_{new}^i) < f(x^i)$ **then**
      $x^i = x_{new}^i$
    end if
    {end of student phase}
  end for
  k=*k*+1
end while

Fig. 4. Classification of socio-inspired algorithms [20].

Where x_i^k is the value of the dimension i for the user k, besides that max_i and min_i represent the maximum and minimum values for the values of the dimension i.

Moreover, the Twitter Optimization algorithm adds two interaction operators, which are comments and participation, these are applied generating a perturbation at the moment of retweeting, that is, when the user k retweets to another user p, the user k has to choose randomly between the operators comments (Eq. 7) and participation (Eq. 8).

$$x_i^p = x_i^p + Character^k \cdot rand(-1, 1) \tag{7}$$

$$x_i^p = x_i^k \tag{8}$$

Another operator is the so-called publishing new tweet, which eliminates the tweet with the worst fitness after having generated a new tweet. On the

other hand there is the classification of celebrities and current, which make a differentiation for the application of a specific operator for the 1% with better fitness in their tweet. This operator takes into consideration if his solution has not been updated during the last times W, where W corresponds to the number of followers of k in the current round. The behavior for the averages in each round, will be to generate a new tweet according to the Eq. 9. The celebrity will choose the tweet with the best aptitude and own it.

$$x_i^k = x_i^{best} + \delta \cdot rand(-1, 1) \tag{9}$$

Where x^{best} corresponds to the best solution found so far, δ is scanning radius. This operator generates solutions close to the best value, performing a local search.

The algorithm is summarized in the flow chart in Fig. 5.

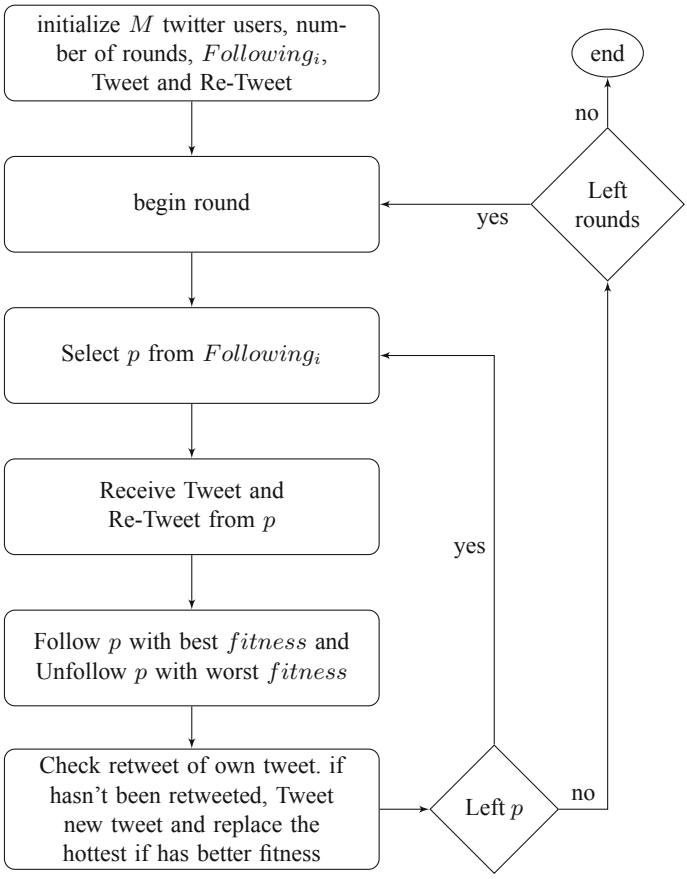

Fig. 5. Social media optimization workflow [8].

4 Combinatorial Problems

Combinatorial problems are in several areas of computer science and their application domains. These problems involve finding a grouping, ordering or assignment of a discrete and finite set of elements, in order to satisfy certain conditions. Candidate solutions do not necessarily have to meet all the conditions, but those that do become feasible, therefore evaluable solutions [5].

The application domains can be discrete, continuous or mixed, therefore they take interest since they can represent diverse problems of the real life, that can be simple to solve when the decision variables are limited, but at the moment of having a relatively great number of variables, the own combinatorial of these problems causes that the possible solutions increase of exponential way, which makes difficult its resolution in time of computation reasons under complete optimization techniques.

Among the classic examples of these problems we can find the Travelling Salesman Problem, Knapsack Problem, in addition to Karp's 21 famous problems [17].

4.1 Set Covering Problem

Set Covering Problem can be defined as follows, considering a A binary array of m-rows and n-columns size, where $a_{i,j} \in \{0, 1\}$ corresponds to the value of the A binary matrix, while i and j are the size of rows m and columns n respectively (Eq. 10).

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \dots & \dots & \dots & \dots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{pmatrix} \tag{10}$$

If a_{ij} takes the value of 1, this means that the column j satisfies the row i, which is related to a cost $c_j \in C$, $C = \{c_1, c_2, ...c_n\}$, besides that $i = 1, 2, ..., m$ and $j = 1, 2, ..., n$.

SCP aims at minimizing the cost associated to the subset $S \subseteq J$, having as a constraints to satisfy all rows $i \in I$ with at least one column $j \in J$. Where $I = \{1, 2, ..., m\}$ and $J = \{1, 2, ..., n\}$ The columns j belonging to the subset S are that have value 1, while those with value 0 do not.

The SCP can be defined as follows:

$$\text{Min} \quad \sum_{j=1}^{n} c_j x_j \tag{11}$$

Subject to

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1 \quad \forall i \in I \tag{12}$$

$$x_j \in \{0, 1\} \quad \forall j \in J \tag{13}$$

Table 1. Instances group 4.

Instances	Zopt	TLBO	TO	BCSO	BFO	BSFLA	BELA	BABC
4.1	429	430	451	459	429	430	447	430
4.2	512	524	611	570	517	516	559	513
4.3	516	526	522	590	519	520	537	519
4.4	494	501	514	547	495	501	527	495
4.5	512	518	520	545	514	514	527	514
4.6	560	566	566	637	563	563	607	561
4.7	430	433	446	462	430	431	448	431
4.8	492	507	499	546	497	497	509	493
4.9	641	660	644	711	655	656	682	649
4.10	514	524	543	537	519	518	571	517
Avg	510	518.9	531.6	560.4	513.8	514.6	541.4	512.2

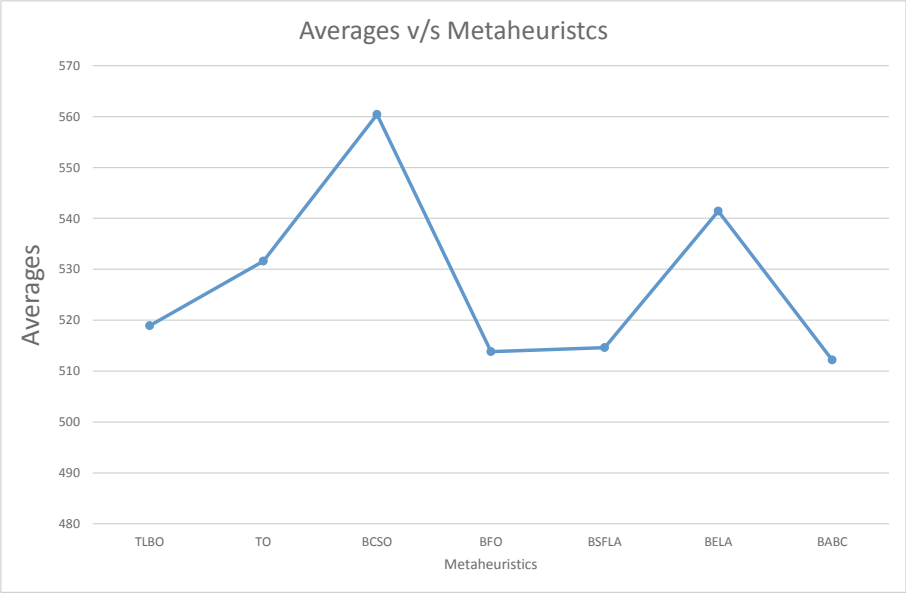

Fig. 6. Averages for metaheuristics.

5 Experiments and Results

The experiments were performed using a computer with a Windows 10 operating system, 16 GB of RAM, and an i7-6700 processor are used. The instances from Beasley's OR-Library are used in the experiments in this study [4]. For the instance group, the algorithm was executed 30 times to use the average of the results obtained. Each instance was pre-processed. In addition to being compared

with different literature optimization techniques, such as Cat Swarm Optimization (BCSO) [7], Binary Firefly Optimization (BFO) [12], Binary Shuffled Frog Leaping Algorithm (BSFLA) [11], Binary Artificial Bee Colony (BABC) [9] and Binary Electromagnetism-Like Algorithm (BELA) [33].

The results are shown in Table 1 and Fig. 6.

6 Conclusion

In this paper we have made a brief review of the literature related to socio-inspired metaheuristic techniques, where we highlight the work in this area, specifically those related to socio-culture, which is inspired by the behavior of societies and groups of people, such as TLBO and TO techniques, which imitate the behavior of students and teachers in the classroom, as well as interactions in social networks such as Twitter.

Under the current and recurrent scenario, of having scarce resources to manage, it is necessary to use this type of techniques to solve the diverse combinatorial problems present in real life, where the socio-inspired techniques have proven to be participants in diverse implementations, besides being an attractive alternative at present.

In this work, after comparing the performance of TLBO and TO, in benchmark instances of OR-library, a clear superiority of TLBO solving Set Covering Problem is noticed, but at the same time comparing with other techniques of the literature, its results are not the best quality making a comparison of the averages obtained, however this does not mean that this kind of techniques are not a competitive alternative for the resolution of combinatorial problems.

Among the future works for this kind of socio-inspired problems, the use of hybridization techniques to improve the performance of these are denoted, in addition Autonomous Search techniques could be implemented, along with the use of techniques from Machine Learning to improve some of the operators.

Acknowledgements. Broderick Crawford is supported by Grant CONICYT/FONDECYT/REGULAR/1171243, Ricardo Soto is supported by Grant CONICYT/FONDECYT/REGULAR/1190129. José Lemus-Romani is supported by National Agency for Research and Development (ANID)/Scholarship Program/DOCTORADO NACIONAL/2019 - 21191692.

References

1. Ahmadi, S.-A.: Human behavior-based optimization: a novel metaheuristic approach to solve complex optimization problems. Neural Comput. Appl. **28**(1), 233–244 (2017)
2. Ahmadi-Javid, A.: Anarchic society optimization: a human-inspired method. In: 2011 IEEE Congress of Evolutionary Computation (CEC), pp. 2586–2592. IEEE (2011)

3. Atashpaz-Gargari, E., Lucas, C.: Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: 2007 IEEE Congress on Evolutionary Computation, pp. 4661–4667. IEEE (2007)
4. Beasley, J.E., Chu, P.C.: A genetic algorithm for the set covering problem. Eur. J. Oper. Res. **94**(2), 392–404 (1996)
5. Blum, C., Roli, A.: Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput. Surv. (CSUR) **35**(3), 268–308 (2003)
6. Crawford, B., Soto, R., Astorga, G., Lemus-Romani, J., Misra, S., Rubio, J.-M.: An adaptive intelligent water drops algorithm for set covering problem. In: 2019 19th International Conference on Computational Science and Its Applications (ICCSA), pp. 39–45. IEEE (2019)
7. Crawford, B., et al.: A binary cat swarm optimization algorithm for the non-unicost set covering problem. Math. Probl. Eng. (2015)
8. Crawford, B., Soto, R., Cabrera, G., Salas-Fernández, A., Paredes, F.: Using a social media inspired optimization algorithm to solve the set covering problem. In: Meiselwitz, G. (ed.) HCII 2019. LNCS, vol. 11578, pp. 43–52. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21902-4_4
9. Crawford, B., Soto, R., Cuesta, R., Paredes, F.: Using the bee colony optimization method to solve the weighted set covering problem. In: Stephanidis, C. (ed.) HCI 2014. CCIS, vol. 434, pp. 493–497. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07857-1_86
10. Crawford, B., Soto, R., de la Barra, C.L., Crawford, K., Paredes, F., Johnson, F.: A better understanding of the behaviour of metaheuristics: a psychological view. In: Stephanidis, C. (ed.) HCI 2014. CCIS, vol. 434, pp. 515–518. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07857-1_90
11. Crawford, B., Soto, R., Peña, C., Palma, W., Johnson, F., Paredes, F.: Solving the set covering problem with a shuffled frog leaping algorithm. In: Nguyen, N.T., Trawiński, B., Kosala, R. (eds.) ACIIDS 2015. LNCS (LNAI), vol. 9012, pp. 41–50. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-15705-4_5
12. Crawford, B., Soto, R., Suárez, M.O., Paredes, F., Johnson, F.: Binary firefly algorithm for the set covering problem. In: 2014 9th Iberian Conference on Information Systems and Technologies (CISTI), pp. 1–5. IEEE (2014)
13. Duncan, R.B.: The ambidextrous organization: designing dual structures for innovation. Manag. Organ. **1**(1), 167–188 (1976)
14. Emami, H., Derakhshan, F.: Election algorithm: a new socio-politically inspired strategy. AI Commun. **28**(3), 591–603 (2015)
15. Hosseini, S., Al Khaled, A.: A survey on the imperialist competitive algorithm metaheuristic: implementation in engineering domain and directions for future research. Appl. Soft Comput. **24**, 1078–1094 (2014)
16. Huan, T.T., Kulkarni, A.J., Kanesan, J., Huang, C.J., Abraham, A.: Ideology algorithm: a socio-inspired optimization methodology. Neural Comput. Appl. **28**(1), 845–876 (2017)
17. Karp, R.M.: Reducibility among combinatorial problems (1972). https://people.eecs.berkeley.edu/~luca/cs172/karp.pdf
18. Kashan, A.H.: League championship algorithm: a new algorithm for numerical function optimization. In: 2009 International Conference of Soft Computing and Pattern Recognition, pp. 43–48. IEEE (2009)
19. Kulkarni, A.J., Durugkar, I.P., Kumar, M.: Cohort intelligence: a self supervised learning behavior. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics, pp. 1396–1400. IEEE (2013)

20. Kumar, M., Kulkarni, A.J.: Socio-inspired optimization metaheuristics: a review. In: Kulkarni, A.J., Singh, P.K., Satapathy, S.C., Husseinzadeh Kashan, A., Tai, K. (eds.) Socio-cultural Inspired Metaheuristics. SCI, vol. 828, pp. 241–265. Springer, Singapore (2019). https://doi.org/10.1007/978-981-13-6569-0_12
21. Kumar, M., Kulkarni, A.J., Satapathy, S.C.: Socio evolution & learning optimization algorithm: a socio-inspired optimization methodology. Future Gen. Comput. Syst. **81**, 252–272 (2018)
22. Kuo, H., Lin, C.: Cultural evolution algorithm for global optimizations and its applications. J. Appl. Res. Technol. **11**(4), 510–522 (2013)
23. Liu, Z.-Z., Chu, D.-H., Song, C., Xue, X., Lu, B.-Y.: Social learning optimization (SLO) algorithm paradigm and its application in QoS-aware cloud service composition. Inf. Sci. **326**, 315–333 (2016)
24. Luque, A.G., Dorado, S.R., de Fátima Vieira Severiano, M., Burillo, F.J.: Fundamentos sociales del comportamiento humano. Editorial UOC (2013)
25. Lv, W., He, C., Li, D., Cheng, S., Luo, S., Zhang, X.: Election campaign optimization algorithm. Procedia Comput. Sci. **1**(1), 1377–1386 (2010)
26. Lv, W., et al.: Verifying election campaign optimization algorithm by several benchmarking functions. In: Tan, Y., Shi, Y., Tan, K.C. (eds.) ICSI 2010. LNCS, vol. 6146, pp. 582–587. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13498-2_76
27. Lv, Z., Shen, F., Zhao, J., Zhu, T.: A swarm intelligence algorithm inspired by Twitter. In: Hirose, A., Ozawa, S., Doya, K., Ikeda, K., Lee, M., Liu, D. (eds.) ICONIP 2016. LNCS, vol. 9949, pp. 344–351. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46675-0_38
28. Moosavian, N., Roodsari, B.K., et al.: Soccer league competition algorithm, a new method for solving systems of nonlinear equations. Int. J. Intell. Sci. **4**(01), 7 (2013)
29. Rao, R.V., Savsani, V.J., Vakharia, D.: Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. Comput.-Aided Des. **43**(3), 303–315 (2011)
30. Ray, T., Liew, K.-M.: Society and civilization: an optimization algorithm based on the simulation of social behavior. IEEE Trans. Evol. Comput. **7**(4), 386–396 (2003)
31. Satapathy, S., Naik, A.: Social group optimization (SGO): a new population evolutionary optimization technique. Complex Intell. Syst. **2**(3), 173–203 (2016)
32. Soto, R., Crawford, B., González, F., Vega, E., Castro, C., Paredes, F.: Solving the manufacturing cell design problem using human behavior-based algorithm supported by autonomous search. IEEE Access **7**, 132228–132239 (2019)
33. Soto, R., Crawford, B., Muñoz, A., Johnson, F., Paredes, F.: Pre-processing, repairing and transfer functions can help binary electromagnetism-like algorithms. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Prokopova, Z., Silhavy, P. (eds.) Artificial Intelligence Perspectives and Applications. AISC, vol. 347, pp. 89–97. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-18476-0_10
34. Sotoudeh-Anvari, A., Hafezalkotob, A.: A bibliography of metaheuristics-review from 2009 to 2015. Int. J. Knowl.-Based Intell. Eng. Syst. **22**(1), 83–95 (2018)
35. Talbi, E.-G.: Metaheuristics: from Design to Implementation, Chapter 1.3, vol. 74. Wiley, Hoboken (2009)
36. Tzanetos, A., Fister Jr., I., Dounias, G.: A comprehensive database of nature-inspired algorithms. In: Data in Brief, p. 105792 (2020)
37. Valdivia, S., et al.: Bridges reinforcement through conversion of tied-arch using crow search algorithm. In: Misra, S., et al. (eds.) ICCSA 2019. LNCS, vol. 11623, pp. 525–535. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24308-1_42

38. Vásquez, C., et al.: Galactic swarm optimization applied to reinforcement of bridges by conversion in cable-stayed arch. In: Misra, S., et al. (eds.) ICCSA 2019. LNCS, vol. 11623, pp. 108–119. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24308-1_10
39. Xu, J., Zhang, J.: Exploration-exploitation tradeoffs in metaheuristics: survey and analysis. In: Proceedings of the 33rd Chinese Control Conference, pp. 8633–8638. IEEE (2014)
40. Yang, X.-S.: Metaheuristic optimization: algorithm analysis and open problems. In: Pardalos, P.M., Rebennack, S. (eds.) SEA 2011. LNCS, vol. 6630, pp. 21–32. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20662-7_2