ICONTECH INTERNATIONAL JOURNAL OF SURVEYS, ENGINEERING, TECHNOLOGY
ISSN 2717-7270
Journal homepage: http://icontechjournal.com/index.php/iij
Volume 5 (2021) Issue 2

THE NOVEL SOCIAL SPIDER OPTIMIZATION ALGORITHM: OVERVIEW, MODIFICATIONS, AND APPLICATIONS

Saman M. Almufti

College of Computer Science & Information Technology, Nawroz University

Duhok, Kurdistan-Region, Iraq

ABSTRACT

The continues in real-world problems increasing complexity motivated computer scientists and researchers to search for more-efficient problem-solving strategies. Generally natural Inspired, Bio Inspired, Metaheuristics based on evolutionary computation and swarm intelligence algorithms have been frequently used for solving complex, real-world optimization problems because of their ability to adjust to variety of conditions. This paper present a swarm based algorithm that is based on the cooperative behaviors between social spider, it called Social Spider Optimization (SSO) algorithm. In SSO, search agents characterize a set of spiders which together move according to a biological behavior in colony. During the past years after SSO introduction, many modifications has improved the performance of the algorithm and has been applied in several fields. In this paper, the improvements, and applications of the SSO are reviewed.

Keywords: Swarm Intelligence, Social Spider Optimization, Bio-Inspired Algorithm

1-INTRODUCTION:

With the fast growing in the complexity of modern optimization problems. Depending on the nature of phenomenon simulations is becoming increasingly attractive as an efficient tool for optimization, Swarm intelligence (SI) is a field that inspired by the collective behavior of real insects or animals swarms in the natural [1]. In the past, many algorithms developed base on these behaviors such as (ACO, BSC, EHO, BA, HHO, HS, DE, EA, ABC,….etc.) solve a wide range of complex optimization problems [2]. Bonabeau defined swarm intelligence as ‘‘any attempt to design algorithms or distributed problem solving devices inspired by the collective behavior of the social insect colonies and other animal societies” [3]. In this paper, Social Spider Optimization (SSO) which is a swarm based metaheuristics algorithm is reviewed. The Social Spider Optimization algorithm is founded base on the simulation of cooperative behavior in social-spiders colony [4]. In the proposed algorithm, a group of spiders which interact with each other by the biological laws of the cooperative colony. SSO algorithm requires two different search agents (spiders): males and females. Based on gender, each individual is steered by a set of different evolutionary-operators which mimic different cooperative behaviors that are naturally seems in the colony[4,5]. SSO has been widely modified and applied in several fields. This article presents a review of the Social Spider Optimization (SSO) and its modifications.

2-BIO INSPIRED ALGORITHMS:

Real-world problems are often very difficult to solve and involve multi-objective optimization. Most of the Real-world optimization problems are NP-hard problems, that can’t be solved by simple deterministic algorithms [6,7]. Through the past years, it have been approved that Bio-Inspired algorithms are excellent strategies to address complex optimization problems, and have been practically applied to solve various problems which belongs to different fields. Over the past few years, several Bio-Inspired Algorithms been developed by inspirating biological swarms that occur in nature [8].

Generally bio-inspired algorithms are classified over three major categories: Evolutionary, Swarm Intelligence, and Ecological inspired algorithms[6,9], Figure (1) presents a graphical classification of some of well-known Bio-Inspired Algorithms [6].

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

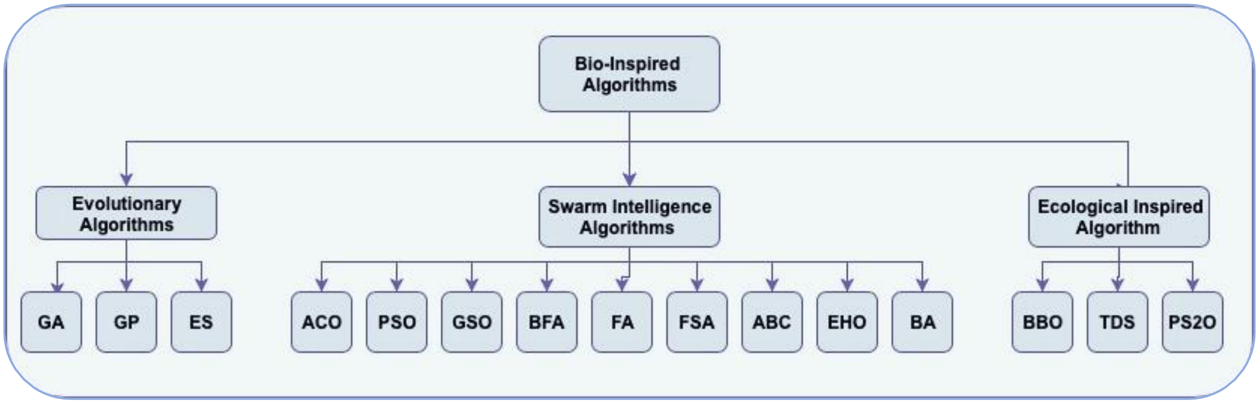

Fig. 1: Bio-Inspired Algorithm Taxonomy

3-SWARM INTELLIGENCE:

Swarm intelligence(SI), is a part of artificial intelligence that is concerned with the designing and developing of intelligent interactive multi-agent systems that cooperate to gather to achieve a specific goal [10,11]. SI defined by Marco Dorigo as "The emergent collective intelligence of groups of simple agents" [1,11,12]. Swarm-based metaheuristics algorithms are inspired from collective behaviors of some social insects, animal, or bacteria's in the nature such as ants, birds, Elephants, bats, bees, Spiders, termites, wolfs, dolphins and fishes [13, 14]. The most amazing characteristics of swarm based systems are "Self-organization" and "decentralized-control" that naturally leads to an emergent behavior in the colony [14,15] as shown in Fig. (2).

- **Self-organization:** This can be characterized by three parameters like structure, multi stability and state transitions. In swarms, interpreted the self-organization through four characteristics: (i) positive feedback,(ii) negative feedback, (iii) fluctuations, and (iv) multiple interactions.
- **Stigmergy:** It means stimulation by work. Stigmergy is based on three principles:(i) work as a behavioral response to the environmental state; (ii) an environment that serves as a work state memory(iii)work that does not depend on specific agents.

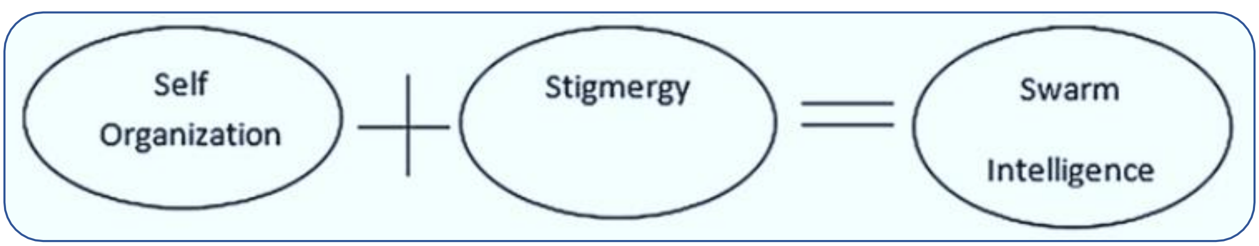

Fig. 2: Swarm intelligence basic characteristics

In computer science there are many algorithms that are designed as an inspiration of real collective behavior systems in the nature, swarm intelligence algorithms includes many Algorithms among these famous algorithms such as:

- *Ant Colony Optimization (ACO):* is a population based optimization algorithm developed by Marco Dorigo as an inspiration of the behavior of ants in finding the optimal way (best path) between their nest and a food source [1,2,15].

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

- *Bat algorithm (BA):* is a natural inspired metaheuristic algorithm for global optimization problems. It was inspired by the echolocation behavior of microbats, with varying pulse rates of emission and loudness [16]. The Bat algorithm (BA) was developed by the scientist Yang in 2010 [2,16,17].
- *Particle Swarm Optimization (PSO):* is a population based optimization algorithm developed by Eberhart and Kennedy in 1997 as an inspired by bird flocks' behavior when searching for food [2,18,19].
- *Elephant herding optimizations (EHO):* is a metaheuristics based algorithm developed by Wang in 2015 [20,21] to solve optimization problems. The algorithm inspired by herding behaviors of elephants in their clan [22].
- *Artificial bee colony algorithm (ABC):* is a natural inspired metaheuristic optimization algorithm based on the intelligent foraging behavior of honey bee swarm, proposed by the scientist Karaboga for solving combinatorial optimization problems [1,7].

According to S. Almufti in [2], there are more than 200 algorithms that inspired by heuristics of natural characteristics in the nature.

In this paper a novel algorithms which belongs to Swarm Intelligence called (Social Spider optimization) is reviewed with its modifications.

4-SOCIAL SPIDER OPTIMIZATION ALGORITHM:

The Social Spider Optimization (SSO) proposed by Erik Cuevas et al., in 2013 [23,24]. It is a population-based Swarm intelligent algorithm that inspired by the natural cooperative behavior of the social spider in the colony. Spider Colonies are formed mainly by two elements: spiders and communal web. Web is represented by the searching field domain while problem solutions are represented by the insects. SSO considers two search agents (spiders): male and female. Each individual is directed by a different set of evolutionary rule depending on gender which mimics different cooperative behaviors typically found in the colony. This individual categorization allows reducing critical flaws present in several SI approaches such as incorrect exploration exploitation balance and premature convergence[25].

In nature, heavier individuals dominate the lighter ones. This behavior is copied to the algorithm and the spider's weight is proportional to the solution evaluation. Spiders are divided by gender and each one have a different behavior in the colony. This difference is implemented using unique evolutionary operators for males and females. Gender balance on the colony is normally around 70% of females. The classical SSO algorithm starts by populating randomly the first generation with uniform distribution in the search space. A gender is addressed to each individual. The first agents (spiders) on the population matrix are addressed feminine and the rest as masculine [25,26]. The cutting point is given by Equ.(1) :

$$N_f = \text{floor}[(0.9 - \text{r} * 0.25)\,\text{N}] \tag{1}$$

where N_f is the number of females, N is the population size, floor rounds each element to the nearest integer, and r is a random number in the unitary range [0,1]. All elements are then evaluated on the objective function and the best solution (spider) and best objective function are recorded. After the initialization process the algorithm starts the searching loop that only ends when the maximum number of function evaluations or the target function value is reached. The first step in the searching loop is to calculate the spiders weight[27]. This calculation by Equ. (2):

$$w_i = 1 - \frac{Fs_i - Fs_{best}}{Fs_{worest} - Fs_{best}} \tag{2}$$

where w_i is the weight for the i_{th} spider, Fs_i is the objective function value for the i_{th} spider, Fs_{best} is the best objective value in the population and Fs_{worest} is the worst objective value reached. The communal web is the natural substrate where all spiders live, exchanging information according to their distance and weight. There are mainly three types of communication among spiders in the web known as Vibci , Vibfi, Vibbi can be shown in Fig (3), and can be calculated by Equ. (3) [23,27,28].

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

$$Spider\ comunication\ Types \begin{cases} Vibci_i = w_c * \exp\left(\dfrac{-d_{i,c}}{d_{coeff}}\right) \\ Vibfi_i = w_{cf} * \exp\left(\dfrac{-d_{i,cf}}{d_{coeff}}\right) \\ Vibbi_i = w_b * \exp\left(\dfrac{-d_{i,b}}{d_{coeff}}\right) \end{cases} \quad (3)$$

Where:

i	Is the i_{th} individual spider
c	is the closest heavier member to the i_{th} element.
cf	is the closest female to the i_{th} individual.
Vibci	it represents the vibration perceived by the i_{th} spider and emitted by the individual c
Vibfi	it represents the vibration perceived by the i_{th} individual, emitted by the member cf
Vibbi	it represents the vibration perceived by the i_{th} spider that is emitted by the best spider in the web.
w_c	is the closest heavier member weight
$d_{i,c}$	is the Euclidian distance between the i_{th} and c individuals
dcoeff	is a adaptive factor.
w_{cf}	is the closest female weight
$d_{i,cf}$	is the distance between the i_{th} spider and its closest female
w_b	is the best solution's weight
$d_{i,b}$	is the distance between the i_{th} and the best individual

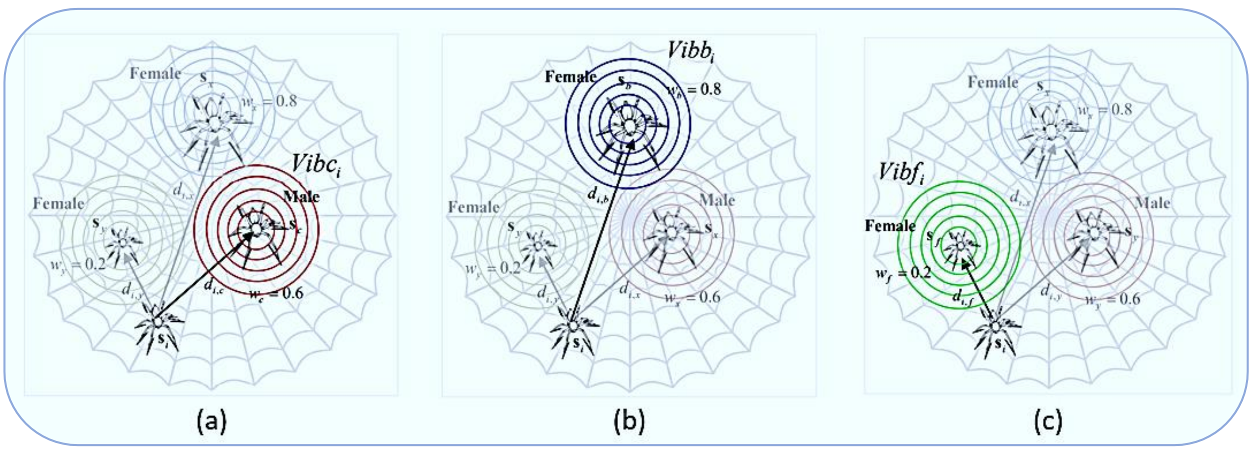

Fig. 3. Configuration of each special relation: a) Vibci , b) Vibbi and c) Vibfi

spiders are moved either on a repulsion movement or on an attraction. This decision is made by attributing a random unitary number, to each female individual with respect to pf, which is a constructive (threshold) parameter [23], it can be calculated by Equ. (4).

$$f_i = \begin{cases} f_i - \alpha.\theta.vibci.(sc - f_i) - \beta.\theta.vibbi.(sb - f_i) - \theta.\left(\gamma - \dfrac{1}{2}\right) & if\ f_i > fp \\ f_i + \alpha.\theta.vibci.(sc - f_i) + \beta.\theta.vibbi.(sb - f_i) + \theta.\left(\gamma - \dfrac{1}{2}\right) & if\ f_i < fp \end{cases} \quad (4)$$

where f_i is the i_{th} female, sc represents the closest heavier spider coordinates, sb is the best spider's position, α, β, γ are unitary-ranged random numbers generated with uniform distribution and θ is an adaptive factor.

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

Male spiders are moved according to dominance. The dominance is given by the males median weight's. Males lighter than the median are consider non-dominant and moved according to Equ. (5):

$$m_i = m_i + \alpha.\theta\left(\frac{\sum_{h=1}^{Nm} m_h \,.\, w_{Nf+h}}{\sum_{h=1}^{Nm} w_{Nf+h}} - m_i\right) \quad (5)$$

where m_i represents the i_{th} male position, m_h is the h_{th} male position, Nm is the total number of males, w_{Nf+h} is the weight for the h_{th} male and α is an unitary-ranged random number[23]. Dominant males are those heavier than the median and are moved according to Equ. (6):

$$m_i = m_i - \alpha.\theta.vibfi.(sf - m_i) + \theta.\left(\gamma - \frac{1}{2}\right) \quad (6)$$

where m_i is the i_{th} male position, sf is the closest female to i_{th} male and α, γ are unitary-ranged random numbers.

Finally, after move all males and females on the web, the last operator is representing the mating behavior where only dominant males will participate. The code will check if there is any female closer than radius of mating to a dominant male. The radius of mating is given by Equ.(7):

$$rm = \frac{\sum_{d-1}^{d}\left(p_d^h - p_d^l\right)}{2D} \quad (7)$$

where rm is the mating radius, p_d^h and p_d^l are respectively the upper and lower bound for a given dimension and D is the problem dimension. Males and females which are under the mating radius generate new candidate spiders according to the roulette method. Each candidate spider is evaluated in the objective function and the result is tested against all the actual population members[23,27,28]. If any member is worse than a new candidate, the new candidate will take the actual individual position assuming actual individual's gender. The classical SSO algorithm can be summarizes by the following flowchart Fig.4.

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

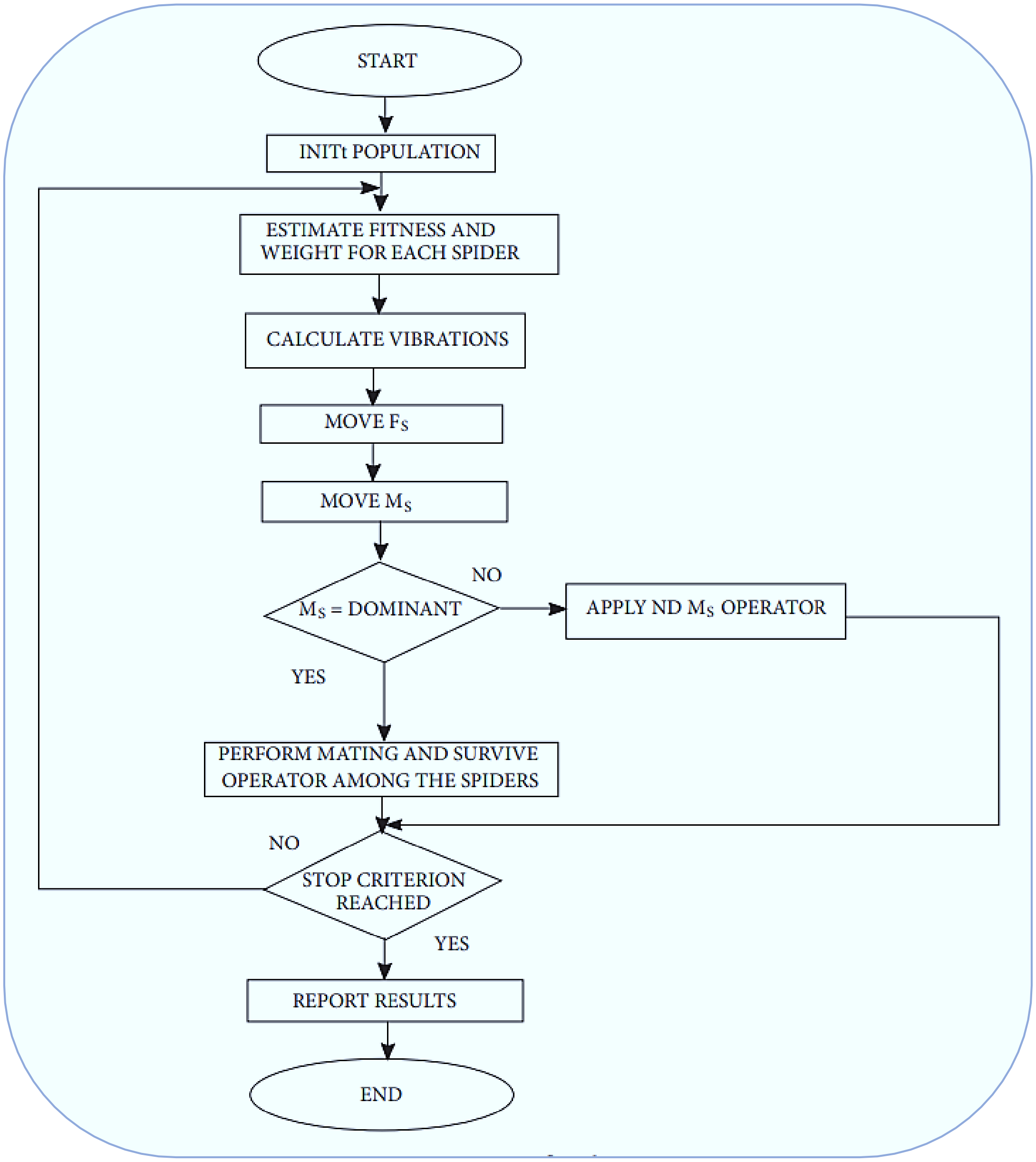

Fig. 4. SSO flowchart [25].

5-SOCIAL SPIDER OPTIMIZATION ALGORITHM MODIFICATIONS:

After the appearance of Social Spider Optimization (SSO) algorithm in 2013 [23], many modifications has been applied to the classical SSO to improve the performances of the proposed algorithm.

In this section, the modifications and improvements of the SSO algorithm are reviewed.

5.1 Social Spider Optimization Based on Rough Sets (SSORS):

At the end of 2016, Mohamed Abd El Aziz et al. [29] proposed a modification in Social Spider Optimization by adding the rough-set model for evaluating the fitness function to improve the performance of SSO, the proposed method called Social Spider Optimization Based on Rough Sets (SSORS) and has been adapt to solve the minimum attribute reduction problem [25, 29].

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

In the algorithm, the fitness function depends on the rough sets dependency degree and takes into consideration the number of selected features. The algorithm starts by randomly initializing a population of spiders. Next, each individual is converted into a binary vector of length *N* by using the following Equ.(8 and 9) [25]:

$$FP = \frac{1}{1 + e^{-x_i^j(t)}} \tag{8}$$

$$x_i^j(t+1) = \begin{cases} 0 & , \quad if\ x_i^j(t) > € \\ 1 & , \quad other\ case \end{cases} \tag{9}$$

where $x_i^j(t)$ is the spider value at the iteration t and € ∈ [0, 1]. The algorithm uses the dependency degree given by Equ.(10):

$$\gamma_c(D) = \frac{|POS_c(D)|}{|U|} \quad , \qquad A = C \cup D \tag{10}$$

Where C and D are called condition and decision features and $POS_C(D)$ is the positive region that contains all the objects of U that can be classified in classes of U/D using information of C, which is used to evaluate the performance of the solutions. Furthermore, the fitness function is defend by Equ.(11) [25].

$$F(R) = p\gamma_R(D) + (1-p)\left(1 - \frac{|R|}{|C|}\right) \tag{11}$$

where $P \in [0, 1]$ is a parameter that gives balance between number of selected features and the classification quality and |·| is the length of the feature set. The fitness of each spider is compared with the global best (F_{best}), and if it has a better fitness value, then F_{best} is replaced with the current spider and its position becomes the redact set R. The process is repeated until a stop criterion is satisfied. Fig. (5) shows the SSORS computational procedure in form of a flowchart [25, 29].

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

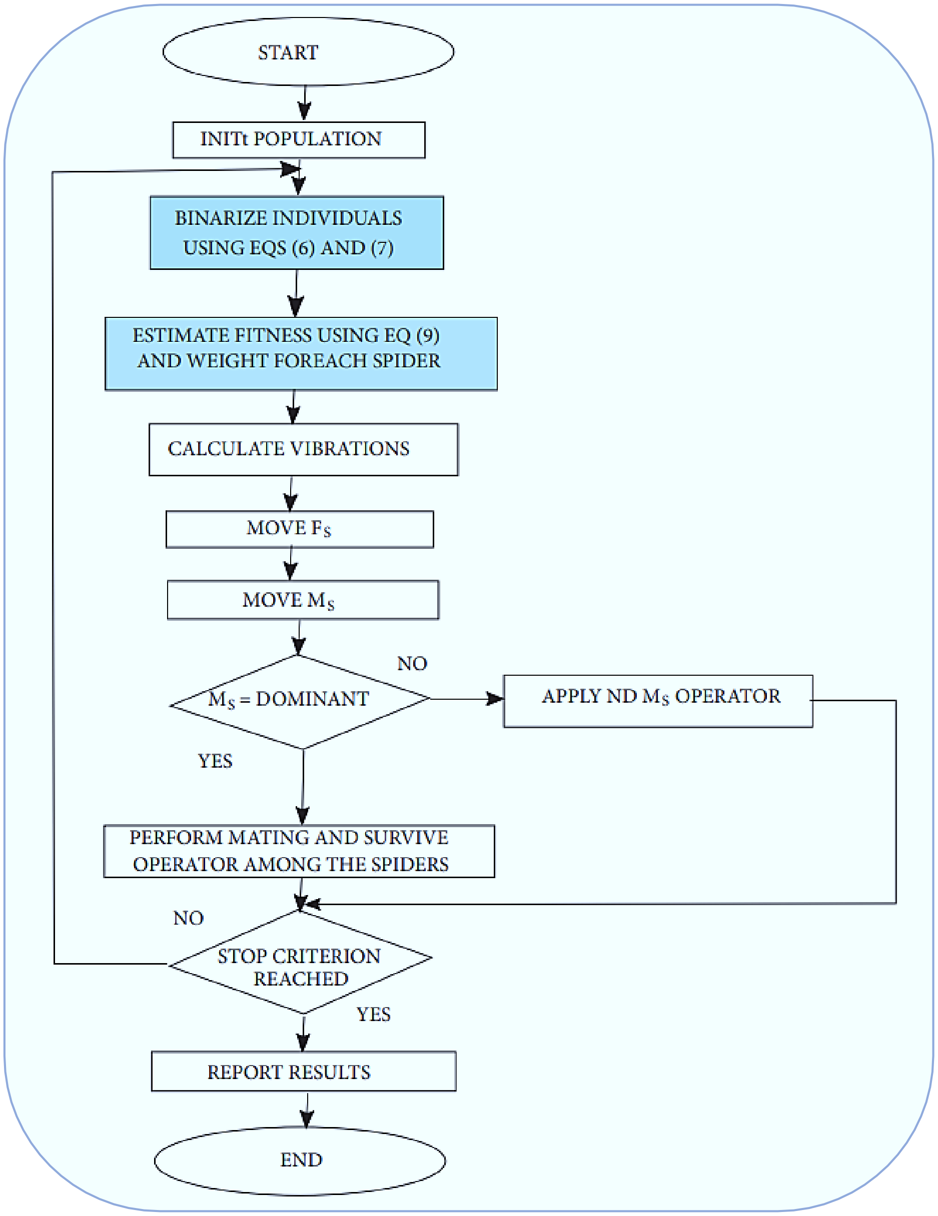

Fig. 5. SSORS flowchart [25].

5.2- Simplex Method Social Spider Optimization (SMSSO):

An obstacle for the SSO to be applied in complex problems is its high computational cost. Therefore, as the dimensionality of a search space and the data amount increases, a problem of local optima entrapment and poor convergence rates is present. To overcome these problems, Yongquan Zhou et al. proposed to apply the simplex method to the original SSO (SMSSO) in 2017 [30], enhancing the algorithm global and local search abilities and avoiding local optima entrapment while increasing the convergence rate.

The simplex method was introduced by Spendley et al. [31] and is defined by some points equal to the number of dimensions in the search space plus one. The process of the simplex method is described as follows:

(1) Evaluate the solutions of the entire population and select the global best s_{best1} and the second best s_{best2}, assuming that s_{best1} is the spider that must be replaced and that $f(s_{best1})$, $f(s_{best2})$, and $f(s_r)$ are the corresponding fitness values.

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

(2) Calculate the middle position (s_m) of s_{best1} and s_{best2} defined by:

$$s_m = \left(\frac{s_{best1} + s_{best2}}{2}\right) \tag{12}$$

(3) Determine the refection point s_{ref} given by

$$s_{ref} = s_m + \alpha(s_m - s_r) \tag{13}$$

The refection coefficient α is typically set to one.

(4) Compare the fitness value between s_{ref} and sbest1. If $f(s_{ref}) < f(s_{best1})$, the extension operation was performed based on the next equation

$$s_e = s_m + \gamma(s_{ref} - s_m) \tag{14}$$

where γ is the extension coefficient, and this parameter is usually set in two.Ten compare the ftness value of the extension point and the global best. If $s_e < s_{best1}$, s_r should be replaced by s_e and, in another case, s_{ref} will be substituted for s_r.

(5) Compare the fitness values of s_{ref} and s_r. If $f(s_{ref}) < f(s_r)$, the compression operation should be performed using the following equation:

$$s_{comp} = s_m - \beta(s_r - s_m) \tag{15}$$

β is the condense coefficient; it is typically set to 0.5. Then, it is compared to the fitness values between s_{comp} and s_r, and if $f(s_{comp}) < f(s_r)$, s_r should be replaced with s_{comp}; otherwise, s_{ref} will be substituted with s_r.

(6) If $f(s_{best1}) < f(s_{ref}) < f(s_r)$, the condense point s_{cond} must identify performing shrink operations where the shrink coefficient is described by σ:

$$s_{cond} = s_m - \sigma(s_r - s_m) \tag{16}$$

If $f(s_{cond}) < f(s_r)$, s_r will be replaced with s_{cond}. In other case, s_{ref} will be replaced with s_r.

The proposed SMSSO algorithm is shown in Figure 6 as a flowchart [25,29,30,31].

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

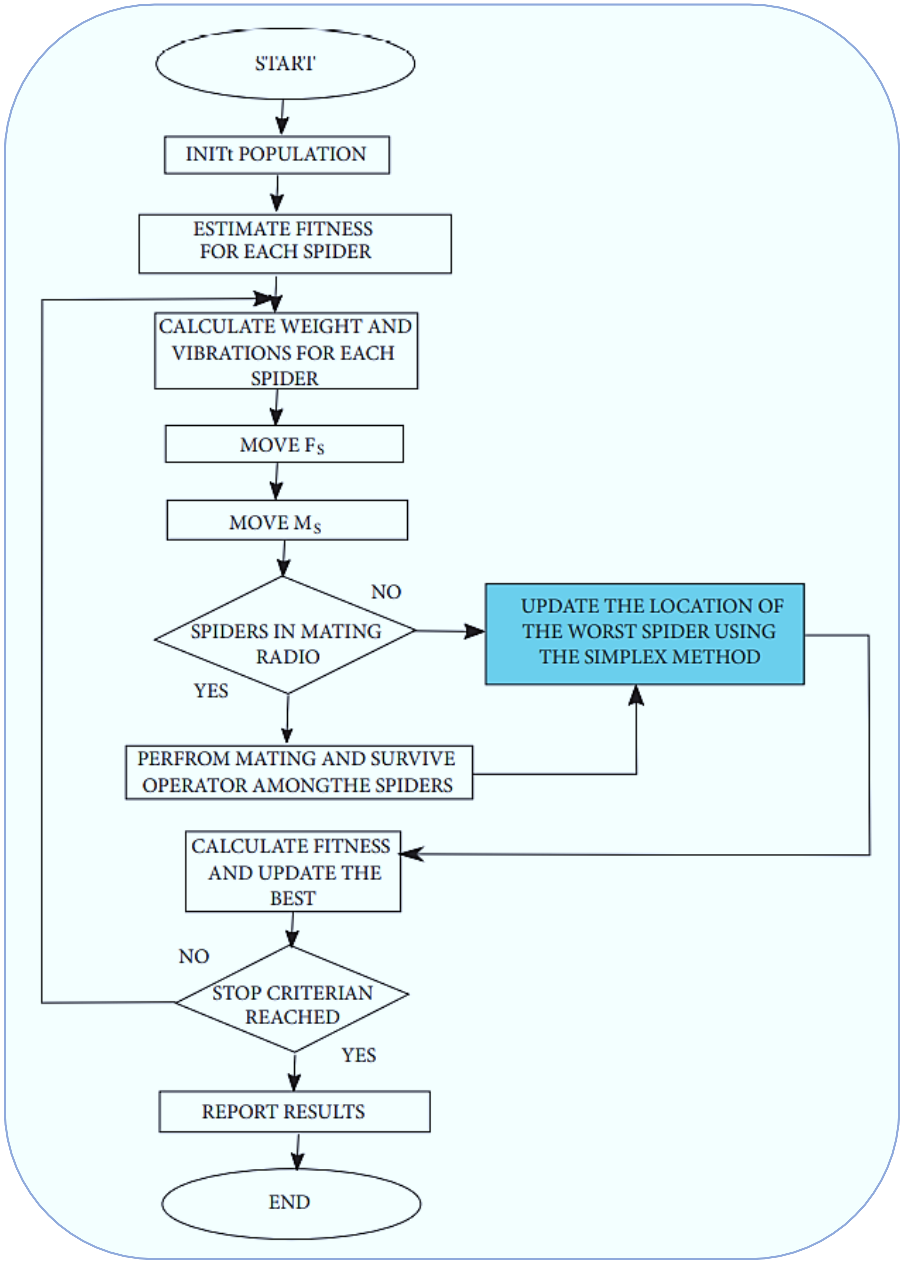

Fig. 6. SMSSO flowchart

5.3-Modifed SSO Approach Based on Beta Distribution and Natural Gradient Local Search (MSSO):

Metaheuristic algorithms need a right balance between exploration and exploitation to give successful results in optimization problems. The classical SSO requires the random selection of parameters α, β, and θ in Equ.((4) and (5)) to control the movement of the spiders, which can afect the mentioned balance leading the algorithm to a premature convergence. With the aim of improving this balance, Carlos E. Klein et al. proposed a modifcation for the SSO (MSSO) in 2016 [32], where the mentioned parameters are selected from a Beta distribution in the range [0-1] [32,33] instead of the use of random numbers. The use of this distribution helps to preserve diversity and avoid premature convergence, improving the algorithm exploration. Furthermore, to improve the exploitation the author proposed the use of natural gradient (NG) with a rank one covariance matrix approximation to local search in each generation. In this method, the best spider realizes local search using NG after performing the operator [32,33]. The

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

NG parameters are the learning rate for the mean value and for the scale factor. The complete computational process is summarized in Figure 7 as a flowchart. To prove the performance of the MSSO algorithm, authors considered the solution of two engineering problems such as the solenoid and brushless motor design.

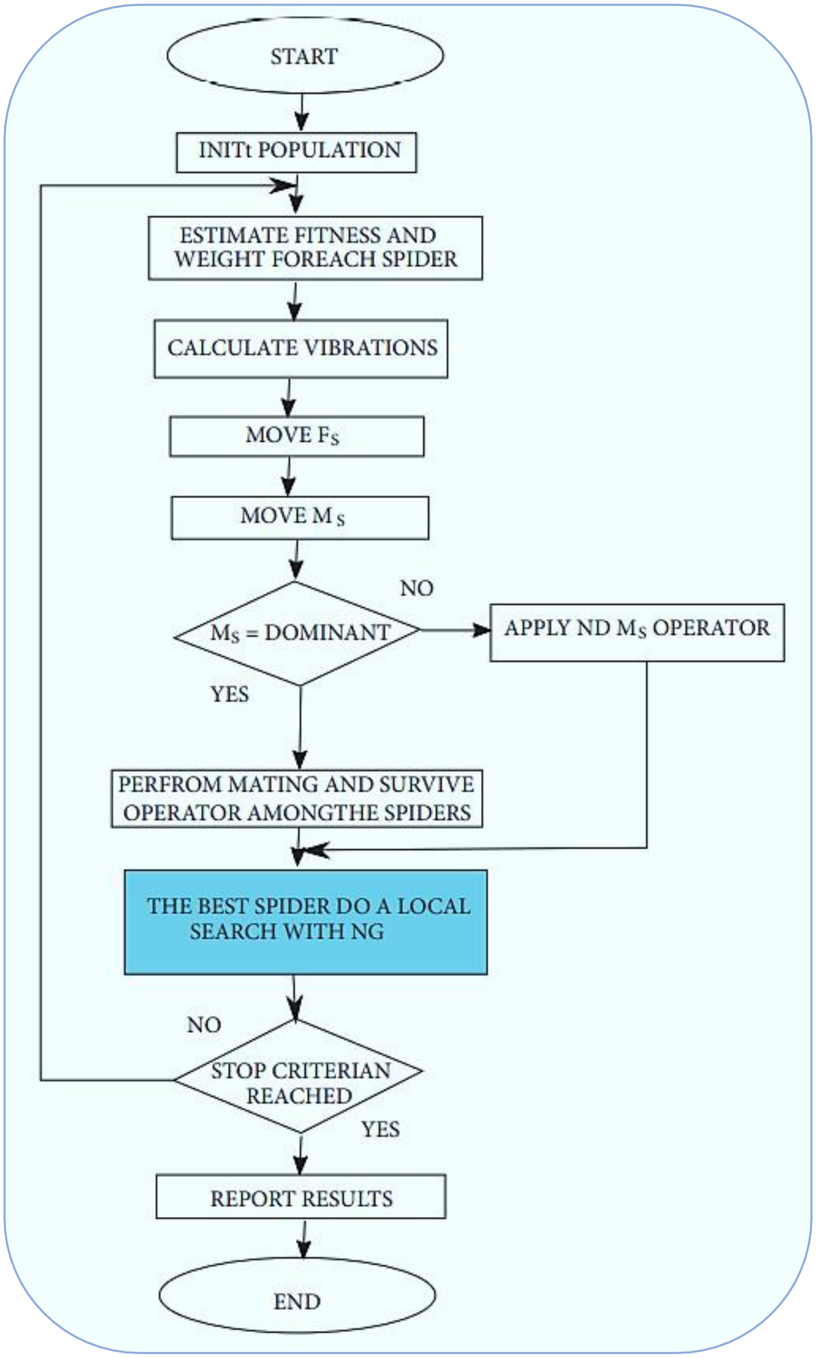

Fig. 7. MSSO flowchart [25].

5.4- Elite Opposition-Based Social Spider Optimization (EOSSO):

The SSO emulates the cooperative behavior of a spider colony by stochastic position changes; with this method the probability of getting a good solution is relatively low. With the purpose to increase the probability of getting a better solution Ruxin Zhao et al. proposes the use of Elite Opposition-Based Learning Strategy EOLS in the SSO and creates

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

ICONTECH INTERNATIONAL JOURNAL OF SURVEYS, ENGINEERING, TECHNOLOGY
ISSN 2717-7270
Journal homepage: http://icontechjournal.com/index.php/iij
Volume 5 (2021) Issue 2

the EOSSO in 2017 [34]. Opposition-Based Learning (OBL) is a machine intelligence strategy that considers the current individual and its opposite particle at the same time to get a better approximation for the current candidate solution. It has been proved that an opposite candidate has a greater chance to be closer to the global optimal than a random candidate solution. Some of the concepts of OBL are defined as follows. The EOSSO main idea is to calculate and evaluate the opposite solution of each particle at the same time and select the better one as an individual for the next generation. The best fitness valued individual is seen as an elite individual. At the global optimization process, the strategy expands the search space of the algorithm and strengths the diversity of the population . Tus the global searching ability can be enhanced and help to get a better optimal solution [25,34]. The complete computational process is summarized in Figure 8.

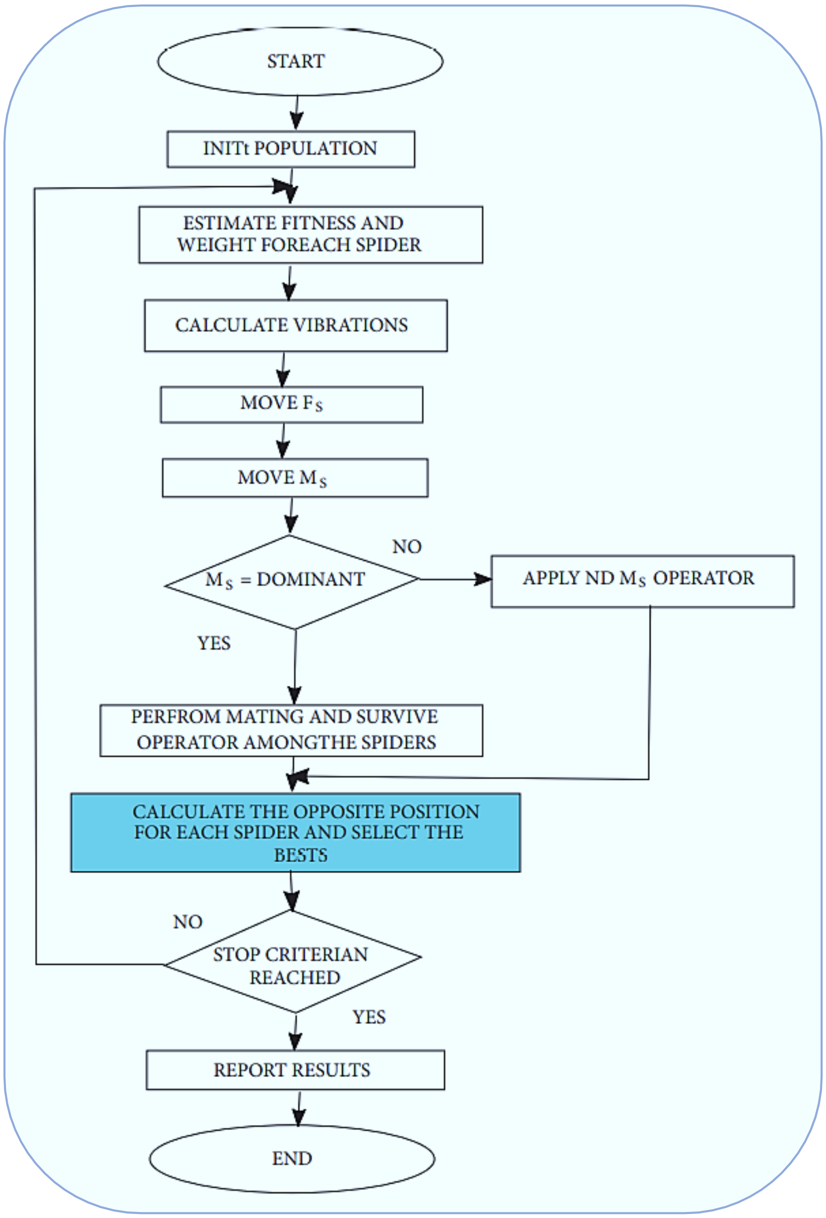

Fig. 8. EOSSO flowchart

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

ICONTECH INTERNATIONAL JOURNAL OF SURVEYS, ENGINEERING, TECHNOLOGY
ISSN 2717-7270
Journal homepage: http://icontechjournal.com/index.php/iij
Volume 5 (2021) Issue 2

5.5- Parallel Social Spider Clustering Algorithm for High Dimensional Data sets (P-SSO):

As the dimensionality in specific problems increases, it is desirable that the execution time of SI algorithms would be reduced. With this aim, Urvashi Prakash Shukla et al. suggested a parallel version of the SSO (P-SSO) in 2016 [35]. In this version, the position of each spider (female, dominant and nondominant male) is updated simultaneously; this increases the computational speed, giving the algorithm the ability to work in high dimensional problems. The computational procedure of P-SSO is illustrated in Figure 9 as a flowchart [25,35]. Tis algorithm was applied to solve clustering problems with high dimensional data. According to the authors, the approach performs ten times faster than the original SSO algorithm. The P-SSO outperforms several clustering schemes even in other real-life applications as multispectral image segmentation as a clustering problem. In terms of accuracy, the P-SSO provides two times better precision than one of its competitors [35].

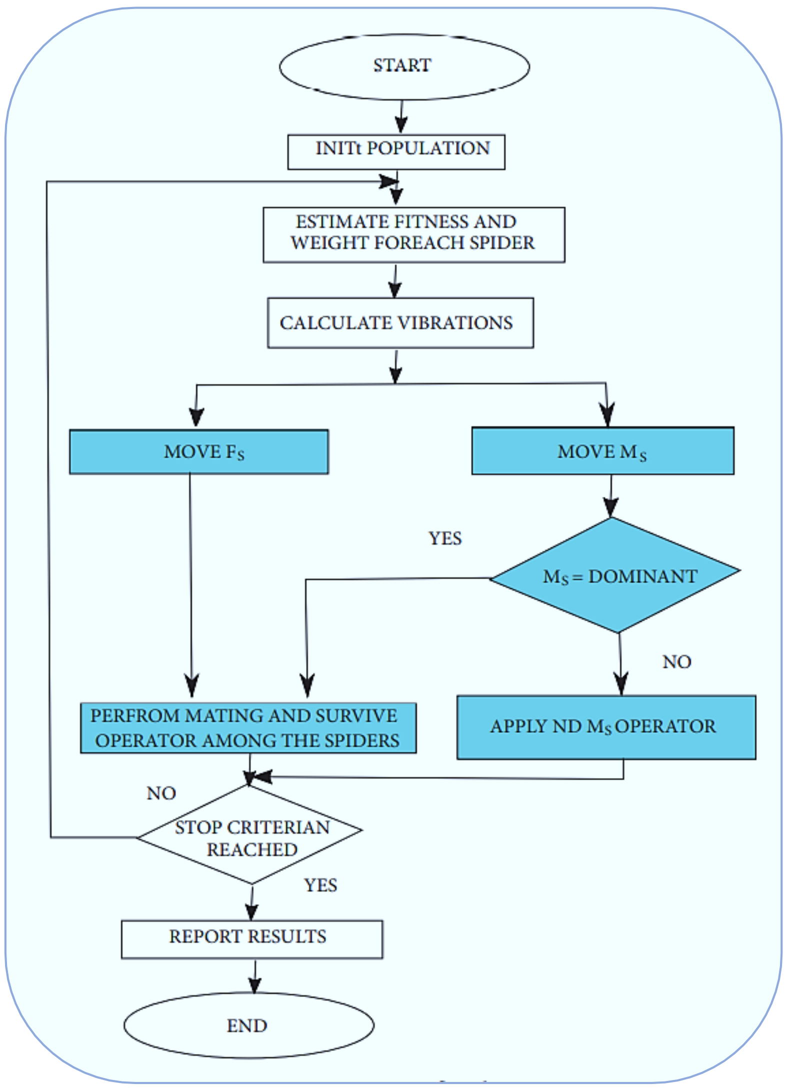

Fig. 9. P-SSO flowchart

5.6- Hybrid Social Spider Optimization and Genetic Algorithm (HSSOGA):

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

A desirable property for any population-based optimization technique is to avoid premature convergence and entrapment into local optima; with this aim, Tawhid and Ali proposed a hybrid optimization approach based on both SSO and GA algorithms in 2016 [36]. In that approach, the authors combine the properties of exploration and exploitation of the SSO. Ten, it applies the arithmetical crossover and mutation operators of the GA, calling the algorithm Hybrid Social Spider Optimization and Genetic Algorithm (HSSOGA). With this combination, the search process is accelerated and helps to find a near optimal solution in a reasonable time [25,36]. The HSSOGA algorithm consists of mainly three steps:

(1) Apply the social spider cooperative operators to take advantage of its balance between exploration and exploitation.

(2) Subdivide the population and apply the arithmetical crossover operation on the populations with the aim of improving the search diversity of the proposed algorithm.

(3) Apply the GA mutation operator in order to avoid premature convergence.

Figure 10 presents the computational procedure of HSSOGA as a flowchart.

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

ICONTECH INTERNATIONAL JOURNAL OF SURVEYS, ENGINEERING, TECHNOLOGY
ISSN 2717-7270
Journal homepage: http://icontechjournal.com/index.php/iij
Volume 5 (2021) Issue 2

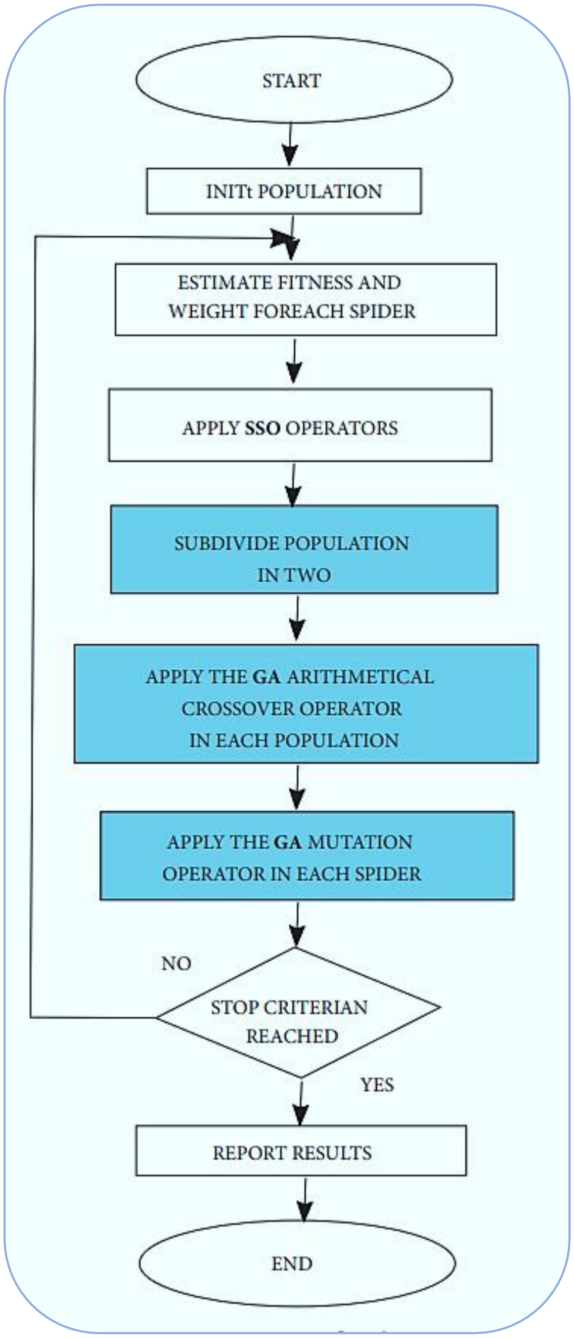

Fig. 10. HSSOGA flowchart

5.7- Improved Social Spider Optimization (ISSO):

With the objective of accelerating the convergence efficiency and search ability of the SSO, Shuang-Cheng Sun et al. [37] proposed five improved SSO algorithms in 2017. The first algorithm ISSO1 takes advantage of the historical process adding the historical best position to operate the movement of the spiders in the search area through a vibration perceived by the i-th spider, leading to an increment of the convergence speed. The second improvement ISSO2 tries to overcome the chaotic search on the earlier stages of the search process due to the low influence of the spider's vibrations on a high distance between them, adding acceleration coefficients to the half of the female and dominant spiders. Furthermore, a random search term that decreases with each iteration and is controlled by an inertia weight is

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

added. A small step size causes a careful search among the spiders but decreases the convergence speed. On the other hand, a big step size increases the convergence speed but decreases the search accuracy, which may lead to losing the optimal position [25,37]. To solve these troubles in the ISSO3 algorithm add a linearly decreased step size leading this to a reasonable tradeoff between computational time and search accuracy. In the ISSO4 algorithm, a mutation operator for the nondominant spiders is added with the objective to improve the solutions diversity. Finally, the ISSO5 algorithm makes use of the improvements of the other four algorithms at the same time. Figure 11 presents the computational process of each version ISSO1, ISSO2, ISSO3, and ISSO4 in form of a flowchart [25].

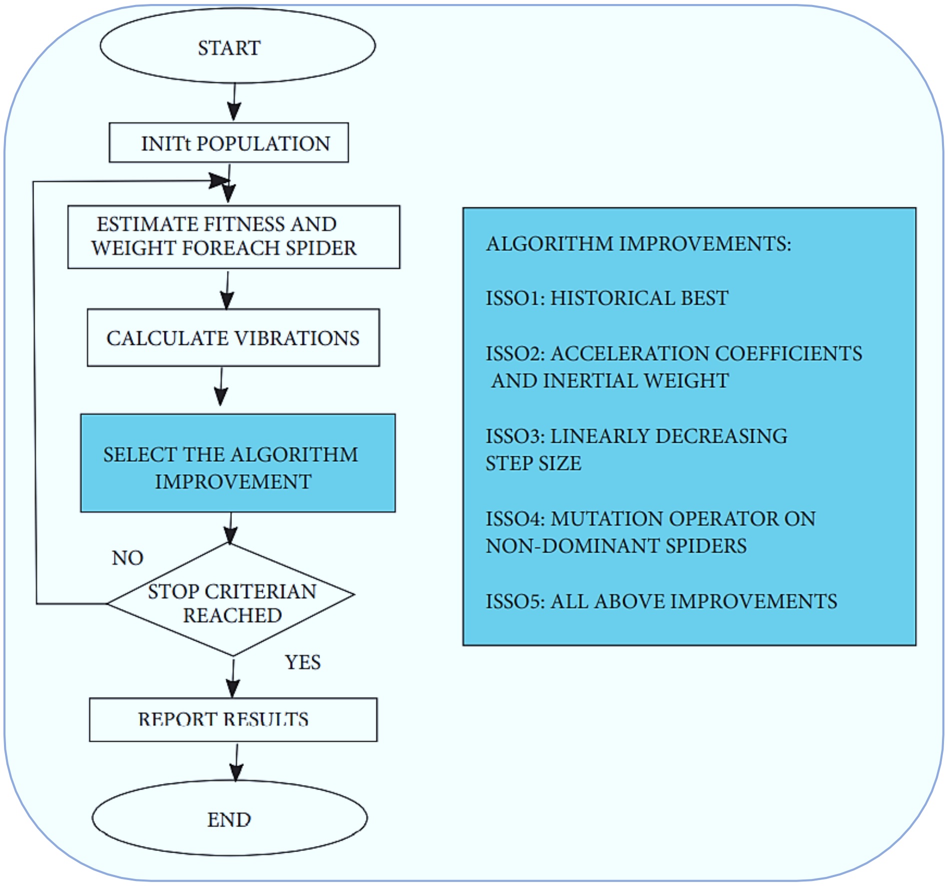

Fig. 11. ISSO flowchart

APPLICATIONS:

Since its introduction, SSO and its modified algorithms has potential applied in a wide to solve various problems in several fields. This section presents a review of some of the most important problems.

- binary optimization problems (BOPs)[38].
- Wireless sensor networks (WSN)[39].
- Multiple-Pursuer Multiple-Evader (MPME) [40].
- Increasing Performance of Multiple Pursuer Drones in Neutralizing Attacks From Multiple Evader Drones [40].
- Optimal reactive power dispatch problem [41].
- Clustering for High Dimensional Data Sets [35].

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

- Feedforward Neural Networks(FNN)[25].
- Vector Machines Parameters Tuning[25].
- Parameter Improvement in the Lukasiewicz Structure[25].
- Image Fusion Approach for Contrast Enhancement and Brightness Preservation [25].
- Image Enhancements [42].
- select data features for the optimal parameters in the Lukasiewicz structure[43].
- image segmentation problems via Multilevel Image Thresholding Approach[44].
- Template matching (TM) is a key procedure for many different image processing applications[45].
- Unscented Transform for Wind Turbine Uncertainty for Economic Dispatch Approach [46].
- Sensor Deployment Scheme[47].
- Optimal Congestion Management Approach[48].
- Efficient Frequency Controllers for Autonomous Two-Area[49].
- tuning the gains of a PID controller[49].
- Optimal Design of Fractional Controller for LFC in an Interconnected Multisource Power System[50].

And many other applications and fields.

DISCUSSION:

Social Spider Optimization (SSO) is a population-based algorithm that inspired by the cooperative behavior of the social spider proposed by Erik Cuevas et al. in 2013,. SSO considers two search agents male and female which are steered by a different set of evolutionary operators depending on its gender to verify a cooperative behaviors typically found in the spider colony. After its introduction, the SSO has been used to solve a wide variety of problem in computer and engineering fields.

REFERENCES:

[1]. S. Almufti, "Using Swarm Intelligence for solving NPHard Problems", *Academic Journal of Nawroz University*, vol. 6, no. 3, pp. 46-50, 2017. Available: https://doi.org/10.25007/ajnu.v6n3a78.

[2]. S. M. Almufti, "Historical survey on metaheuristics algorithms", *International Journal of Scientific World*, vol. 7, no. 1, p. 1, 2019. Available: https://doi.org/10.14419/ijsw.v7i1.29497 [Accessed 30 March 2021].

[3]. E. Bonabeau, M. Dorigo, G. Theraulaz, Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, Inc., New York, NY, USA, 1999.

[4]. M. Jain, V. Singh and A. Rani, "A novel nature-inspired algorithm for optimization: Squirrel search algorithm", *Swarm and Evolutionary Computation*, vol. 44, pp. 148-175, 2019. Available: https://doi.org/10.1016/j.swevo.2018.02.013 [Accessed 30 March 2021].

[5]. E. Cuevas, M. Cienfuegos, D. Zaldívar and M. Pérez-Cisneros, "A swarm optimization algorithm inspired in the behavior of the social-spider", *Expert Systems with Applications*, vol. 40, no. 16, pp. 6374-6384, 2013. Available: https://doi.org/10.1016/j.eswa.2013.05.041.

[6]. S. Almufti, “Using Swarm Intelligence for solving NPHard Problems,” Academic Journal of Nawroz University, vol. 6, no. 3, pp. 46–50, 2017. https://doi.org/10.25007/ajnu.v6n3a78.

[7]. S. Almufti, R. Marqas, and V. Ashqi, "Taxonomy of bio-inspired optimization algorithms. Journal of Advanced Computer Science & Technology", 8(2), 23. 2019. https://doi.org/10.14419/jacst.v8i2.29402.

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

[8]. D. Rai and K. Tyagi, "Bio-inspired optimization techniques", *ACM SIGSOFT Software Engineering Notes*, vol. 38, no. 4, pp. 1-7, 2013. Available: https://doi.org/10.1145/2492248.2492271 [Accessed 31 March 2021].

[9]. Binitha, S., SATHYA, S.S., (2012), A Survey of Bio inspired Optimization Algorithms. International Journal of Soft Computing and Engineering, Vol. 2, No. 2, pp 137-151.

[10]. S. Almufti, "U-Turning Ant Colony Algorithm powered by Great Deluge Algorithm for the solution of TSP Problem", Hdl.handle.net, 2018. [Online].

[11]. Renas R. Assad, Abdulnabi, N. (2018). Using Local Searches Algorithms with Ant Colony Optimization for the Solution of TSP Problems. Academic Journal of Nawroz University, 7(3), 1-6. https://doi.org/10.25007/ajnu.v7n3a193.

[12]. S. Almufti and A. Shaban, "U-Turning Ant Colony Algorithm for Solving Symmetric Traveling Salesman Problem", Academic Journal of Nawroz University, vol. 7, no. 4, pp. 45-49, 2018. https://doi.org/10.25007/ajnu.v6n4a270.

[13]. X. Yang, "Metaheuristic Optimization" Scholarpedia, 6(8), p.11472, 2011. https://doi.org/10.4249/scholarpedia.11472.

[14]. J. Rajpurohit, T. Sharma and A. Abraham, "Glossary of MetaheuristicAlgorithms", International Journal of Computer Information Systems and Industrial Management Applications, vol. 9, pp. 181-205, 2017.

[15]. M. Dorigo," Optimization, Learning and Natural Algorithms", PhD thesis, Politecnico di Milano, Italy, 1992.

[16]. A. Yahya Zebari, S. Almufti and C. Abdulrahman, "Bat algorithm (BA): review, applications and modifications", *International Journal of Scientific World*, vol. 8, no. 1, p. 1, 2020. Available: https://doi.org/10.14419/ijsw.v8i1.30120 [Accessed 30 March 2021].

[17]. X. Yang, "A New Metaheuristic Bat-Inspired Algorithm", *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, pp. 65-74, 2010. Available: https://doi.org/10.1007/978-3-642-12538-6_6 [Accessed 31 March 2021].

[18]. S. Almufti, A. Yahya Zebari and H. Khalid Omer, "A comparative study of particle swarm optimization and genetic algorithm", *Journal of Advanced Computer Science & Technology*, vol. 8, no. 2, p. 40, 2019. Available: https://doi.org/10.14419/jacst.v8i2.29401 [Accessed 30 March 2021].

[19]. S. Hochbaum, "Approximation Algorithms for NP-Hard Problems", *ACM SIGACT News*, vol. 28, no. 2, pp. 40-52, 1997. Available: https://doi.org/10.1145/261342.571216 [Accessed 31 March 2021].

[20]. S. Almufti, R. Asaad and B. Salim, "Review on Elephant Herding Optimization Algorithm Performance in Solving Optimization Problems", Sciencepubco.com, 2019. [Online]. Available: https://www.sciencepubco.com/index.php/ijet/article/view/28473. [Accessed: 26- May- 2019].

[21]. G. Wang, L. Dos Santos Coelho, X. Gao and S. Deb, "A new metaheuristic optimisation algorithm motivated by elephant herding behaviour", International Journal of Bio-Inspired Computation, vol. 8, no. 6, p. 394, 2016. https://doi.org/10.1504/IJBIC.2016.10002274.

[22]. S. Almufti, R. Marqas, and R. Asaad. "Comparative study between elephant herding optimization (EHO) and U-turning ant colony optimization (U-TACO) in solving symmetric traveling salesman problem (STSP)". Journal of Advanced Computer Science & Technology, 8(2), 32, 2019. https://doi.org/10.14419/jacst.v8i2.29403.

[23]. E. Cuevas, M. Cienfuegos, D. Zaldívar and M. Pérez-Cisneros, "A swarm optimization algorithm inspired in the behavior of the social-spider", *Expert Systems with Applications*, vol. 40, no. 16, pp. 6374-6384, 2013. Available: 10.1016/j.eswa.2013.05.041.

[24]. E. Cuevas and M. Cienfuegos, "A new algorithm inspired in the behavior of the social-spider for constrained optimization", *Expert Systems with Applications*, vol. 41, no. 2, pp. 412-425, 2014. Available: 10.1016/j.eswa.2013.07.067.

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

[25]. A. Luque-Chang, E. Cuevas, F. Fausto, D. Zaldívar and M. Pérez, "Social Spider Optimization Algorithm: Modifications, Applications, and Perspectives", *Mathematical Problems in Engineering*, vol. 2018, pp. 1-29, 2018. Available: 10.1155/2018/6843923 [Accessed 1 April 2021].

[26]. D. Singh, "A New Bio-Inspired Social Spider Algorithm", *International Journal of Applied Metaheuristic Computing*, vol. 12, no. 1, pp. 79-93, 2021. Available: 10.4018/ijamc.2021010105.

[27]. J. Yu and V. Li, "A social spider algorithm for global optimization", *Applied Soft Computing*, vol. 30, pp. 614-627, 2015. Available: 10.1016/j.asoc.2015.02.014.

[28]. D. Mahato and R. Singh, "On maximizing reliability of grid transaction processing system considering balanced task allocation using social spider optimization", *Swarm and Evolutionary Computation*, vol. 38, pp. 202-217, 2018. Available: 10.1016/j.swevo.2017.07.011.

[29]. M. Abd El Aziz and A. Hassanien, "An improved social spider optimization algorithm based on rough sets for solving minimum number attribute reduction problem", *Neural Computing and Applications*, vol. 30, no. 8, pp. 2441-2452, 2017. Available: 10.1007/s00521-016-2804-8.

[30]. Y. Zhou, Y. Zhou, Q. Luo, and M. Abdel-Basset, "A simplex method-based social spider optimization algorithm for clustering analysis," Engineering Applications of Artificial Intelligence, vol. 64, pp. 67–82, 2017.

[31]. W. Spendley, G. R. Hext, and F. R. Himsworth, "Sequential application of simplex designs in optimisation and evolutionary operation," Technimetrics. A Journal of Statistics for the Physical, Chemical and Engineering Sciences, vol. 4, pp. 441–461, 1962.

[32]. C. E. Klein, E. H. Segundo, V. C. Mariani, and L. dos S. Coelho, "Modifed Social-Spider Optimization Algorithm Applied to Electromagnetic Optimization," IEEE Transactions on Magnetics, vol. 52, no. 3, pp. 28–31, 2016.

[33]. M. M. Ali, "Synthesis of the β-distribution as an aid to stochastic global optimization," Computational Statistics & Data Analysis, vol. 52, no. 1, pp. 133–149, 2007.

[34]. R. Zhao, Q. Luo, and Y. Zhou, "Elite opposition-based social spider optimization algorithm for global function optimization," Algorithms, vol. 10, no. 1, Paper No. 9, 21 pages, 2017.

[35]. U. P. Shukla and S. J. Nanda, "Parallel social spider clustering algorithm for high dimensional datasets," Engineering Applications of Artifcial Intelligence, vol. 56, pp. 75–90, 2016.

[36]. M. A. Tawhid and A. F. Ali, "A hybrid social spider optimization and genetic algorithm for minimizing molecular potential energy function," Sof Computing, vol. 21, no. 21, pp. 6499–6514, 2017.

[37]. S.-C. Sun, H. Qi, Y.-T. Ren, X.-Y. Yu, and L.-M. Ruan, "Improved social spider optimization algorithms for solving inverse radiation and coupled radiation–conduction heat transfer problems," International Communications in Heat and Mass Transfer, vol. 87, pp. 132–146, 2017.

[38]. E. Baş and E. Ülker, "A binary social spider algorithm for continuous optimization task", *Soft Computing*, vol. 24, no. 17, pp. 12953-12979, 2020. Available: 10.1007/s00500-020-04718-w [Accessed 2 April 2021].

[39]. F. Fausto, E. Cuevas, O. Maciel-Castillo and B. Morales-Castañeda, "A Real-Coded Optimal Sensor Deployment Scheme for Wireless Sensor Networks Based on the Social Spider Optimization Algorithm", *International Journal of Computational Intelligence Systems*, vol. 12, no. 2, p. 676, 2019. Available: 10.2991/ijcis.d.190614.001 [Accessed 2 April 2021].

[40]. A. Husodo, G. Jati, A. Octavian and W. Jatmiko, "Enhanced Social Spider Optimization Algorithm for Increasing Performance of Multiple Pursuer Drones in Neutralizing Attacks From Multiple Evader Drones", *IEEE Access*, vol. 8, pp. 22145-22161, 2020. Available: 10.1109/access.2020.2969021 [Accessed 2 April 2021].

[41]. T. Nguyen and D. Vo, "Improved social spider optimization algorithm for optimal reactive power dispatch problem with different objectives", *Neural Computing and Applications*, vol. 32, no. 10, pp. 5919-5950, 2019. Available: 10.1007/s00521-019-04073-4 [Accessed 2 April 2021].

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51

[42]. L. Maurya, P. K. Mahapatra, and A. Kumar, "A social spider optimized image fusion approach for contrast enhancement and brightness preservation," Applied Sof Computing, vol. 52, pp. 575–592, 2017.

[43]. J. Dollaor, S. Chiewchanwattana, K. Sunat, and N. Muangkote, "Te application of social-spider optimization for parameter improvement in the Lukasiewicz structure," in Proceedings of the 8th International Conference on Knowledge and Smart Technology, KST 2016, pp. 27–32, Tailand, February 2016.

[44]. S. Ouadfel and A. Taleb-Ahmed, "Social spiders optimization and fower pollination algorithm for multilevel image thresholding: a performance study," Expert Systems with Applications, vol. 55, pp. 566–584, 2016.

[45]. E. Cuevas, V. Osuna, and D. Oliva, Evolutionary Computation Techniques: A Comparative Perspective, vol. 686, Springer, 2017.

[46]. R. Khorramnia, M.-R. Akbarizadeh, M. K. Jahromi, S. K. Khorrami, and F. Kavusifard, "A new unscented transform for considering wind turbine uncertainty in ED problem based on SSO algorithm," Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology, vol. 29, no. 4, pp. 1479–1491, 2015.

[47]. Y. Zhou, R. Zhao, Q. Luo, and C. Wen, "Sensor Deployment Scheme Based on Social Spider Optimization Algorithm for Wireless Sensor Networks," Neural Processing Letters, pp. 1–24, 2017.

[48]. Z. Hejrati, S. Fattahi, and I. Faraji, "Optimal congestion management using the Social Spider Optimization algorithm," in 29th International Power System Conference, Iran, 2014.

[49]. A. A. El-Fergany and M. A. El-Hameed, "Efcient frequency controllers for autonomous two-area hybrid microgrid system using social-spider optimiser," IET Generation, Transmission & Distribution, vol. 11, no. 3, pp. 637–648, 2017.

[50]. H. Shayeghi, A. Molaee, and A. Ghasemi, "Optimal design of fopid controller for LFC in an interconnected multi-source power system," International Journal on "Technical and Physical Problems of Engineering", pp. 36–44, 2016.

Received 02 April 2021; Received in revised form 06 April 2021; Accepted 06 May 2021;
Available online 28June 2021;
doi: 10.46291/ICONTECHvol5iss2pp32-51