Spider Monkey Optimization Algorithm

Harish Sharma, Garima Hazrati and Jagdish Chand Bansal

Abstract Foraging behavior of social creatures has always been a matter of study for the development of optimization algorithms. Spider Monkey Optimization (SMO) is a global optimization algorithm inspired by Fission-Fusion social (FFS) structure of spider monkeys during their foraging behavior. SMO exquisitely depicts two fundamental concepts of swarm intelligence: self-organization and division of labor. SMO has gained popularity in recent years as a swarm intelligence based algorithm and is being applied to many engineering optimization problems. This chapter presents the Spider Monkey Optimization algorithm in detail. A numerical example of SMO procedure has also been given for a better understanding of its working.

Keywords Spider monkey optimization · Swarm intelligence · Fission-fusion social structure · Numerical optimization

1 Spider Monkey Optimization

Spider monkey optimization (SMO) algorithm is a recent addition to the list of swarm intelligence based optimization algorithms [1, 2]. The update equations are based on Euclidean distances among potential solutions. The algorithm has extensively been applied to solve complex optimization problems. In [3], Dhar and Arora applied

H. Sharma · G. Hazrati
Rajasthan Technical University, Kota, India
e-mail: hsharma@rtu.ac.in

G. Hazrati
e-mail: ghazrati9@gmail.com

J. C. Bansal (✉)
South Asian University, New Delhi, India
e-mail: jcbansal@gmail.com

J. C. Bansal et al. (eds.), *Evolutionary and Swarm Intelligence Algorithms*, Studies in Computational Intelligence 779,
https://doi.org/10.1007/978-3-319-91341-4_4

Spider Monkey Optimization Algorithm (SMO) to design and optimize a fuzzy rule base. SMO is applied to solve optimal capacitor placement and sizing problem in IEEE-14, 30 and 33 test bus systems with the proper allocation of 3 and 5-capacitors by Sharma et al. [4]. In [5] Wu et al. introduced SMO for the synthesis of sparse linear arrays. The amplitudes of all the elements and the locations of elements in the extended sparse subarray are optimized by the SMO algorithm to reduce the side lobe levels of the whole array under a set of practical constraints. Cheruku et al. designed SM-RuleMiner for rule mining task on diabetes data [6]. The SMO has also been used to synthesize the array factor of a linear antenna array and to optimally design an E-shaped patch antenna for wireless applications [7].

The next part of the chapter details the motivation and working of the spider monkey optimization algorithm.

1.1 Motivation

Emergence of Fission-Fusion Society Structure

The concept of fission-fusion society is introduced by the biologist "Hans Kummer" while he was disentangling one of the most complex Mammalian Hamadryas baboons' social organization [8]. The competition for food among the group members of parent group when there is a shortage of food due to seasonal changes lead to fission into many groups and then fusion into a single group. When there is high availability of food then the group is largest whereas, in case of the smallest group, the food scarcity is at its peak. Fission part shows the food foraging behavior of spider monkeys and fusion represents combining of smaller groups to become a larger one.

Foraging Behavior of Spider Monkeys:

Spider monkeys live in the tropical rain forests of Central and South America and exist as far north as Mexico [9]. Spider monkeys are among the most intelligent New World monkeys. They are called spider monkeys because they look like spiders when they are suspended by their tails [10]. Spider monkeys always prefer to live in a unit group called 'parent group'. Based on the food scarcity or availability they split themselves or combine. Communication among them depends on their gestures, positions and whooping. Group composition is a dynamic property in this structure.

Social Organization and Behavior:

The social organization and behavior of spider monkeys can be understood through the following facts:

1. Spider monkeys live in a group of about 40–50 individuals.
2. All individuals in this community forage in small groups by going off in different directions during the day and everybody share the foraging experience in the night at their habitat.

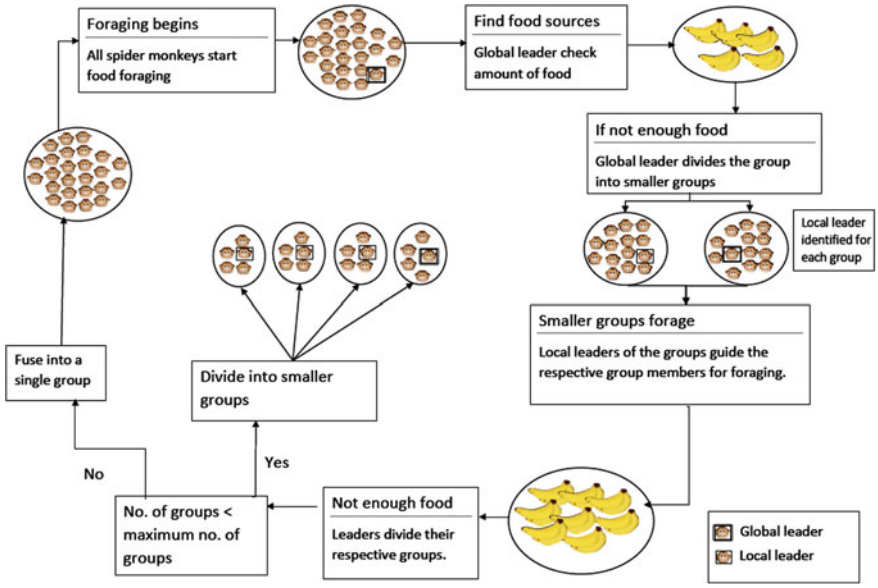

Fig. 1 Foraging behavior of Spider Monkeys

3. The lead female spider monkey decides the forage route.
4. If the leader does not find sufficient food then she divides the group into smaller groups and these groups forage, separately.
5. Individuals of the society might not be noticed closer at one place because of their mutual tolerance among each other. When they come into contact their gestures reflect that they are actually part of a large group.

Communication:

Spider monkeys share their intentions and observations using positions and postures. At long distances they interact with each other by particular sounds such as whooping or chattering. Each monkey has its own discernible sound by which other group members identify that monkey.

The above discussed foraging behavior of spider monkeys is shown in Fig. 1.

1.2 Spider Monkey Optimization Process

SMO is a meta-heuristic technique inspired by the intelligent foraging behavior of spider monkeys. The foraging behavior of spider monkeys is based on the fission-fusion social structure. Features of this algorithm depend on social organization of a group where a female leader takes decision whether to split or combine. The leader of the entire group is named here as the global leader while the leaders of the small groups are named as local leaders. With reference to the SMO algorithm, the phenomenon of food scarcity is defined by no improvement in the solution. Since SMO is a swarm intelligence based algorithm, each small group should have a minimum number of monkeys. Therefore, at any time if a further fission creating at least one group with less than the minimum number of monkeys, we define it as the time for fusion. In SMO algorithm, a Spider Monkey (SM) represents a potential solution. SMO consists of six phases: Local Leader phase, Global Leader phase, Local Leader Learning Phase, Global Leader Learning phase, Local Leader Decision phase and Global Leader Decision phase. All these phases of SMO are explained next:

Initialization:

In the initialization phase, SMO generates a uniformly distributed initial swarm of N spider monkeys, where SM_i represents the ith spider monkey (SM) in the swarm. Each SM_i is initialized as follows:

$$SM_{ij} = SM_{minj} + U(0, 1) \times \left(SM_{maxj} - SM_{minj}\right) \tag{1}$$

where, SM_{minj} and SM_{maxj} are lower and upper bounds of the search space in jth dimension and $U(0, 1)$ is a uniformly distributed random number in the range $(0, 1)$.

Local Leader Phase (LLP):

This is a vital phase of SMO algorithm. Here, all spider monkeys get chance to update themselves. Modification in the position of spider monkey is based on its local leader and local group members' experiences. The fitness value of each spider monkey is calculated at its new position and if fitness is higher than that of its old one, it gets updated otherwise not. Here, position update equation is:

$$SMnew_{ij} = SM_{ij} + U(0, 1) \times \left(LL_{kj} - SM_{ij}\right) + U(-1, 1) \times \left(SM_{rj} - SM_{ij}\right) \tag{2}$$

where, SM_{ij} is the jth dimension of ith SM, LL_{kj} represents the jth dimension of local leader of the kth group and SM_{rj} is the jth dimension of a randomly selected SM from the kth group such that $r \neq i$ and $U(-1, 1)$ is a uniformly distributed random number in the range $(-1, 1)$.

Here, it is clear from Eq. (2) that the spider monkey, which is going to update its position, is attracted towards the local leader while maintaining its self-confidence or persistence. The last component helps to introduce fluctuations in the search process, which helps to maintain the stochastic nature of the algorithm so that a premature stagnation can be avoided. A complete position update process of this phase is well explained in Algorithm 1. In this algorithm pr represents the perturbation rate for the current solution, whose value generally lies in range [0.1, 0.8].

***Algorithm 1:** Position update process in Local Leader Phase (LLP)*

for each member $SM_i \in k^{th}$ group **do**
  for each $j \in \{1, \ldots, D\}$ **do**
  if $U(0,1) \geq pr$ **then**
  $SMnew_{ij} = SM_{ij} + U(0,1) \times (LL_{kj} - SM_{ij}) + U(-1,1) \times (SM_{rj} - SM_{ij})$
  else
  $SMnew_{ij} = SM_{ij}$
  end if
  end for
end for

Global Leader Phase (GLP):

After completing the local leader phase, the algorithm takes the step towards the global leader phase, Here solutions update is based on a selection probability, which is a function of the fitness. From objective function f_i the fitness fit_i can be calculated by Eq. (3).

$$fitness\ function = fit_i = \begin{cases} \frac{1}{1+f_i}, & if\ f_i \geq 0 \\ 1 + abs(f_i), & if\ f_i < 0 \end{cases} \tag{3}$$

The selection probability $prob_i$ is determined based on the roulette wheel selection. If fit_i is the fitness of ith SM then its probability of being selected in the global leader phase is calculated using either of the following two formulae:

$$prob_i = \frac{fitness_i}{\sum_{i=1}^{N} \text{fitness}_\text{i}}$$

or

$$prob_i = 0.9 \times \frac{fit_i}{\max\_fit} + 0.1$$

To update the position, SM uses knowledge of the global leader, experience of neighboring SM and its own persistence. The position update equation in this phase is as follows:

$$\begin{aligned} SMnew_{ij} = & SM_{ij} + U(0, 1) \times (GL_j - SM_{ij}) \\ & + U(-1, 1) \times (SM_{rj} - SM_{ij}) \end{aligned} \tag{4}$$

where GL_j is the position of global leader in the jth dimension. This position update equation is divided into three components: the first component shows persistence of the parent (current) SM, the second component shows the attraction of the parent SM towards the global leader, and the last component is used to maintain the stochastic behavior of the algorithm. In this equation, the second component is used to enhance the exploitation of already identified search space, while the third component helps the search process to avoid premature convergence or to reduce the chance of being stuck in a local optima. The whole search process of this phase is described in Algorithm 2 below:

Algorithm 2: *Position update process in Global Leader Phase (GLP)*

```
count = 0;
while count < group size do
for each member SM_i ∈ group do
    if U(0,1) < prob_i then
    count = count + 1
    Randomly select j ∈ {1, ... , D}
    Randomly select SM_r ∈ group s.t. r≠ i
    SMnew_ij = SM_ij + U(0,1) × (GL_j − SM_ij) +
                             U(−1,1) × (SM_rj − SM_ij)
    end if
  end for
end while
```

It is clear from Algorithm 2 that the chance of updating a solution depends on $prob_i$. Therefore, the solution of high fitness will get more chance as compared to less fit solutions to update its position. Further, a greedy selection approach is applied to the updated solutions i.e., out of the updated and previous SM, the better fit solution is considered.

Global Leader Learning Phase:

In this phase, the algorithm finds the best solution of the entire swarm. The identified SM is considered as the global leader of the swarm. Further, the position of the global leader is checked and if it is not updated then the counter associated with the global leader, named as Global Limit Count (GLC), is incremented by 1, otherwise it is set to 0.

Global Limit Count is checked for global leader and is compared with Global Leader Limit (GLL).

Local Leader Learning Phase:

In this segment of the algorithm, the position of the local leader gets updated by applying a greedy selection among the group members. If local leader doesn't update its position then a counter associated with local leader, called Local Limit Count (LLC) is incremented by 1; otherwise the counter is set to 0. This process is applied to every group to find its respective local leader.

Local Limit Count is a counter that gets incremented till it reaches a fix threshold called Local Leader Limit (LLL).

Local Leader Decision Phase:

Before this phase, local leaders and the global leader have been identified. If any local leader does not get reorganized to a particular verge, known as Local Leader Limit, then all the members of that group update their positions either by random initialization or by using global leader's experience via Eq. (5). Equation (5) is applied with a probability pr called the perturbation rate.

$$SMnew_{ij} = SM_{ij} + U(0, 1) \times \left(GL_j - SM_{ij}\right) + U(0, 1) \times \left(SM_{rj} - LL_{kj}\right) \quad (5)$$

From this equation, it can be understood that the solutions of this group are repelled from the existing local leader as it is exhausted (not updated up to LLL number of iterations) and solutions are attracted towards the global leader to change the existing search directions and positions. Further, based on *pr,* some dimensions of the solutions are randomly initialized to introduce some disturbance in the existing positions of the solutions. Here, Local Leader Limit is the parameter which checks that local leader does not get stuck in local minima, and normally, it is calculated as D × N, where D is dimension and N is total number of SM. If LLC is more than LLL then LLC is set to zero and SM gets initialized as described above to improve the exploration of the search space.

The process of local leader decision phase is described in Algorithm 3.

Algorithm 3: *Local Leader Decision Phase (LLD):*

If $LocalLimitCount > LocalLeaderLimit$ **then**
    $LocalLimitCount = 0$
for each $j \in \{1, \ldots, D\}$ **do**
    if $U(0,1) \geq pr$ **then**
        $SMnew_{ij} = SM_{minj} + U(0,1) \times \left(SM_{maxj} - SM_{minj}\right)$
    else
        $SMnew_{ij} = SM_{ij} + U(0,1) \times \left(GL_j - SM_{ij}\right) + U(0,1) \times \left(SM_{rj} - LL_{kj}\right)$
    end if
end for
end if

Global Leader Decision Phase:

Similar to the local leader decision phase, if the global leader does not get reorganized to a particular verge known as Global leader limit, then the global leader divides the swarm into smaller groups or fuse groups into one unit group. Here, GLL is the parameter, which check whether there is any premature convergence, and varies in the range of N/2 to 2 × N. If GLC is more than GLL then GLC is set to zero and number of groups are compared to maximum groups. If existing number of groups is less than the pre-defined maximum number of groups then the global leader further divides the groups otherwise combines to form a single or parent group. This fission-fusion process is described in Algorithm 4.

***Algorithm 4** Global Leader Decision Phase (GLD):*

if $GlobalLimitCount > GlobalLeaderLimit$ **then**
    $GlobalLimitCount = 0$
if $Number\ of\ groups < MG$ **then**
        Divide the swarms into groups
else
        Combine all the groups to make a single group
end if
    update Local Leaders position
end if

Following algorithm 5 provides the complete working mechanism of SMO to solve an optimization problem.

***Algorithm 5:** Spider Monkey Optimization*

Step 1. **Initialize** population, local leader limit, global leader limit and perturbation rate pr;
Step 2. **Evaluate** the population;
Step 3. **Identify** global and local leaders;
Step 4. **Position update** by local leader phase (Algorithm 1);
Step 5. **Position update** by global leader phase (Algorithm 2);
Step 6. **Learning** through global leader learning phase;
Step 7. **Learning** through local leader learning phase;
Step 8. **Position update** by local leader decision phase (Algorithm 3);
Step 9. **Decide fission or fusion** using global leader decision phase (Algorithm 4)**;**
Step 10. If termination condition is satisfied stop and declare the global leader position as the optimal solution else go to step 4.

2 Analyzing SMO

SMO better balances between exploitation and exploration while search for the optima. Local Leader phase is used to explore the search region as in this phase all the members of the groups update their positions with high perturbation in the dimensions. While the global leader phase promotes the exploitation as in this phase, better candidates get more chance for updating their positions. This property makes SMO a better candidate among the search based optimization algorithms. SMO also possesses an inbuilt mechanism for stagnation check. Local leader learning phase and global leader learning phase, are used to check if the search process is stagnated. In case of stagnation (at local or global level) local leader and global leader decision phases work. The local leader decision phase creates an additional exploration while in the global leader decision phase, a decision about fission or fusion is taken. Therefore, in SMO exploration and exploitation are better balanced while maintaining the convergence speed.

3 Parameters of SMO

SMO has mainly four new control parameters: Local leader limit, Global leader limit, the maximum number of groups (MG), and perturbation rate pr. The suggested parameter setting are given as follows [2]:

- $MG = \frac{N}{10}$, i.e., it is chosen in such a way such that the minimum number of SM's in a group should be 10,
- Global leader limit $\in \left[\frac{N}{2}, 2 \times N\right]$,
- Local leader limit is set to $D \times N$,
- $pr \in [0.1, 0.8]$.

4 Performance Analysis of SMO

Performance of SMO has been analyzed against three well-known meta-heuristics, Artificial Bee Colony (ABC), Differential Evolution (DE), and Particle Swarm Optimization (PSO) in [2]. After testing on 25 benchmark problems and performing various statistical tests, it is concluded that the SMO is a competitive meta-heuristic for optimization. It has been shown that the SMO performed well for unimodal, multimodal, separable and non-separable optimization problems [2]. It was found that for continuous optimization problems SMO should be preferred over PSO, ABC or DE for better reliability.

5 A Worked-Out Example

This section describes a numerical example of SMO. In this step-by-step procedure of SMO, a simple optimization problem $f(x) = x_1^2 + x_2^2$ is solved using the SMO algorithm.

Consider an optimization problem:

$$\text{Minimize}\,(x) = x_1^2 + x_2^2; \quad -5 \leq x_1, x_2 \leq 5$$

Control parameters of SMO:

Swarm or population size, N = 20.
Dimension of problem, D = 2.
If we consider minimum number of individuals in a group are 10 then the maximum number of groups (MG) $= \frac{N}{10} = 2$.
Global Leader Limit, $GLL \in \left\{\frac{N}{2}, 2N\right\} \Rightarrow GLL \in \{10, 40\}$.
Let GLL = 30.
Local Leader Limit, $LLL = D \times N = 2 \times 20 = 40$.

Perturbation rate, $pr \in [0.1, 0.8]$.
Let $pr = 0.7$.

Initialization:

Now, we randomly initialize positions (SMs) of 20 food resources in the range of $[-5, 5]$.

SM number	x_1	x_2	SM number	x_1	x_2
1	1.4	1.2	11	0.1	−0.9
2	−2.4	−2.5	12	0.3	0.3
3	0.6	−0.4	13	−0.4	0.6
4	0.3	1.5	14	0.5	0.7
5	−0.7	1.9	15	1.3	−1.5
6	2.9	3.2	16	−1.1	0.8
7	1.6	−0.9	17	0.8	−0.9
8	0.8	0.2	18	0.4	−0.2
9	−0.5	0.1	19	−0.6	0.3
10	0.3	0.2	20	0.8	1.6

Corresponding function values are

SM number	f_i(x)	SM number	f_i(x)
1	3.4	11	0.82
2	12.01	12	0.18
3	0.52	13	0.52
4	2.34	14	0.74
5	4.1	15	3.94
6	18.65	16	1.85
7	3.37	17	1.45
8	0.68	18	0.2
9	0.26	19	0.45
10	0.13	20	3.2

Fitness values are calculated by using the formula of fitness function, i.e.

$$fitness\ function = fit_i = \begin{cases} \frac{1}{1+f_i(x)}, & if\ f_i(x) \geq 0 \\ 1 + abs(f_i(x)), & if\ f_i(x) < 0 \end{cases}$$

Here since the maximum fitness value is 0.8850, which corresponds to 10th SM, 10th SM becomes the global leader. At this stage there is only single group so 10th SM is the global as well as local leader.

SM number	fit_i	SM number	fit_i
1	0.227	11	0.549
2	0.077	12	0.847
3	0.658	13	0.658
4	0.299	14	0.575
5	0.196	15	0.202
6	0.051	16	0.351
7	0.229	17	0.408
8	0.595	18	0.833
9	0.794	19	0.690
10	0.885	20	0.238

Position Updated Phases:
Local Leader Phase

In this phase, all the SMs will get a chance to update their positions. For updating the position, following position update equation is used.

$$SMnew_{ij} = SM_{ij} + U(0, 1) \times \left(LL_{kj} - SM_{ij}\right) + U(-1, 1) \times \left(SM_{rj} - SM_{ij}\right)$$

where,

$SMnew_{ij}$ = new position of ith spider monkey in the jth direction.
SM_{ij} = old position of ith spider monkey in the jth direction.
SM_{rj} = position of randomly selected spider monkey in the jth direction.
LL_{kj} = position of the local leader. Since there is only one group, k = 1.

Updating 1st spider monkey (*i*=1)

For first dimension j = 1, generate a random number U(0,1). Let U(0, 1) = 0.3. Since *pr* (=0.7) ≤ U(0, 1) is false, therefore $SMnew_{11} = SM_{11}$.
For j = 2, let U(0, 1) = 0.8. Since *pr* ≤ 0.8, SM_{12} is going to update.
If randomly selected neighboring solution index *r* is 6 and U(−1, 1) = −0.7 then

$$\mathrm{SMnew}_{12} = 1.2 + 0.8 \times (0.2 - 1.2) + (-0.7) \times (3.2 - 1.2)$$
$$\mathrm{SMnew}_{12} = -1$$

So the new solution $x_1 = (1.4, -1)$.
Calculating function value and fitness of $SMnew_1$
$f_1(SMnew_1) = 2.96$; fit $(SMnew_1) = 0.252$.
Applying greedy selection between $SMnew_1$ and SM_1 based on fitness
0.227 < 0.252 so it is improved, i.e. $SM_1 = (1.4, -1)$.
Similarly other solutions are also updated and listed next.

For Global Leader Phase, we need to calculate probability function using fitness vector, i.e.

SM number	Updated dimension j	SM_{new}		f_i(x)	Fit_i
1	2	1.4	−1	2.96	0.252
2	1	−1.56	−2.5	8.6836	0.1032
3	2	0.6	0.12	0.3744	0.727
4	–	0.3	1.5	2.34	0.299
5	1	−0.34	1.5	2.366	0.2971
6	–	0.69	3.2	10.716	0.0854
7	1	1.6	−0.9	3.370	0.2288
8	–	0.4	0.2	0.200	0.8333
9	–	−0.5	0.1	0.260	0.7937
10	–	0.3	−0.2	0.130	0.8850
11	2	0.1	0.31	0.106	0.9041
12	1	0.42	0.3	0.266	0.7896
13	–	−0.4	0.6	0.520	0.6579
14	2	0.5	−0.26	0.318	0.7590
15	–	1.3	−1.5	3.940	0.2024
16	2	−1.1	−0.165	1.237	0.4470
17	2	0.8	−0.33	0.749	0.5718
18	2	0.4	0.14	0.180	0.8477
19	–	−0.6	0.3	0.450	0.6897
20	2	0.8	−0.142	0.660	0.6024

$$prob_i = 0.9 \times \frac{fit_i}{\max\_fit} + 0.1\ ;\ \text{Here}, \max\_fit = \max\{fit_i;\ i = 1, 2, \ldots, 20\}$$

Clearly, $\max\_fit = 0.9041$, which corresponds to the 11th solution. Following Table lists the fitness probabilities.

SM	$prob_i$	SM	$prob_i$
1	0.38968	11	1
2	0.202732	12	0.886019
3	0.823703	13	0.754916
4	0.397644	14	0.855558
5	0.395753	15	0.301482
6	0.185013	16	0.544973
7	0.327762	17	0.669207
8	0.929521	18	0.943856
9	0.890101	19	0.786572
10	0.980987	20	0.699668

Global Leader Phase

In this phase, SMs are updated based on the probability $prob_i$. As $prob_i$ is the function of fitness, high fit SMs will get more chance for the update.

In this phase, the position update equation is

$$SMnew_{ij} = SM_{ij} + \mathrm{U}(0, 1) \times \left(GL_j - SM_{ij}\right) + \mathrm{U}(-1, 1) \times \left(SM_{rj} - SM_{ij}\right)$$

where, GL_j =Global leader position in jth direction.

The total number of updates in Global Leader Phase depends on population size in the group. In this example, we are showing position update process of two spider monkeys (8th and 17th) only. It should be noted that in this phase also, only one dimension of every selected solution is updated.

Updating 8th spider monkey (*i* = 8)

$prob_8 = 0.929521$. Let $U(0, 1) = 0.6 < prob_8$. Therefore 8th SM is to be updated.

Let j = 2 selected randomly.

Apply the position update equation of global leader phase to $SM_8 = (0.4, 0.2)$.

We obtain $SMnew_8 = (0.4, -0.75)$.

f_8(x) = 0.7225 and fit$_8$ = 0.5805.

Apply the greedy selection between $SMnew_8$ and SM_8 based on fitness.

Since 0.8333 > 0.5805, SM_8 is not updated.

Updating 17th spider monkey (*i* = 17)

$prob_{17} = 0.6692. LetU(0, 1) = 0.52 < prob_{17}$. Therefore 17th SM is to be updated.

Let j = 1 is selected randomly.

Apply the position update equation of global leader phase to $SM_{17} = (0.8, -0.33)$.

We obtain $SMnew_{17} = (-0.264, -0.33)$.

f_{17}(x) = 0.1785 and fit$_{17}$ = 0.8484.

Apply the greedy selection between $SMnew_{17}$ and SM_{17} based on fitness.

Since 0.571 < 0.8484, SM_{17} is improved.

Updated $SM_{17} = (-0.264, -0.33)$.

After first round (all the solutions should get chance to update their positions), the new positions of the SMs are as follows:

As the total 12 solutions have been updated in first round of global leader phase and as per the termination condition of this phase, the total number of modifications should be equal to the number of SMs in the population. Hence, a next round will be initiated to update the SMs. After the second round, the updated positions of the SMs are as follows:

Now, in this phase, the total numbers of SM modifications are 20, so the phase is terminated now. Hence, it is clear from this phase that the solutions having high fitness value get more chance to update their positions, which improves the exploitation capability of the algorithm.

SM number	j	SM_{newij}		f_i(x)	Fit_i
1	–	1.4	−1	2.96	0.252525
2	–	−1.56	−2.5	8.6836	0.103267
3	2	0.6	0.4	0.52	0.657895
4	–	0.3	1.5	2.34	0.299401
5	1	0.1	1.5	2.26	0.306748
6	–	0.69	3.2	10.7161	0.085353
7	2	1.6	−0.3	2.65	0.273973
8	1	1.1	0.2	1.25	0.444444
9	1	−0.8	0.1	0.65	0.606061
10	2	0.3	−0.9	0.9	0.526316
11	2	0.1	−0.4	0.17	0.854701
12	1	0.3	0.3	0.18	0.847458
13	1	−0.8	0.6	1	0.5
14	1	−0.2	−0.26	0.1076	0.902853
15	–	1.3	−1.5	3.94	0.202429
16	–	−1.1	−0.165	1.237225	0.446982
17	1	−0.264	−0.33	0.178596	0.848467
18	1	0.45	0.14	0.2221	0.818264
19	–	−0.6	0.3	0.45	0.689655
20	–	0.8	−0.142	0.660164	0.60235

Global Leader Learning Phase

Global leader learning phase decides the global leader of the swarm. All solutions' fitness will be compared with each other. If the global leader attains the better position than global limit count is set to 0 otherwise it is incremented by 1. Since the 9th SM's fitness is the best in the updated swarm, it becomes the global leader. Also the global limit count will be set to 0 as the global leader has been updated.

Local Leader Learning Phase

Local leader learning phase decides the local leaders of the groups. Similar to global leader learning phase all solutions' fitness will be compared with each other. If the local leader attains the better position than local limit count is set to 0 otherwise it is incremented by 1. Here we have only one group so 9th SM is the global leader as well as the local leader. The local limit count is set to 0 as the local leader has been updated.

Local Leader Decision Phase

As per our parameter settings local leader limit is 40. Since local leader limit count = 0 < local leader limit = 40, this phase will not be implemented.

Global Leader Decision Phase

SM number	j	SM_{newij}		f_i(x)	Fit_i
1	–	1.4	−1	2.96	0.252525
2	–	−1.56	−2.5	8.6836	0.103267
3	–	0.6	0.4	0.52	0.657895
4	–	0.3	1.5	2.34	0.299401
5	–	0.1	1.5	2.26	0.306748
6	–	0.69	3.2	10.7161	0.085353
7	–	1.6	−0.3	2.65	0.273973
8	–	1.1	0.2	1.25	0.444444
9	1	−0.4	0.1	0.17	0.854701
10	1	−0.8	−0.9	1.45	0.408163
11	1	1.2	−0.4	1.6	0.384615
12	1	1.8	0.3	3.33	0.230947
13	–	−0.8	0.6	1	0.5
14	2	−0.2	0.7	0.53	0.653595
15	–	1.3	−1.5	3.94	0.202429
16	–	−1.1	−0.165	1.237225	0.446982
17	2	−0.264	−0.8	0.709696	0.584899
18	2	0.45	−0.3	0.2925	0.773694
19	1	−0.4	0.3	0.25	0.8
20	–	0.8	−0.142	0.660164	0.60235

In this phase, the position of the global leader is monitored and if it is not updated up to the global leader limit (=30, for this example), then the population is divided into smaller groups. If the numbers of sub-groups are reached to its maximum count (=2, for this example) then all sub-groups are combined to form a single group. After taking the decision, the global limit count is set to 0 and the positions of the local leaders are updated.

In order to explain the role of global leader decision phase, consider a case when for some iteration, global limit count is 31, then the group is divided into two sub-groups. The solutions SM_1–SM_{10} belong to the first group while SM_{11}–SM_{20} belong to the second group. Let the swarm is represented by the following Table:

As the fitness of the 9th SM is highest among all the SMs of the first group, so it is designated as the local leader of the first group, i.e. $LL_1 = (-0.4, 0.1)$. Further, for the second group 19th SM has the highest fitness, so it is considered the local leader of the second group, i.e. $LL_2 = (-0.4, 0.3)$. The local limit counts for both the local leaders are set to 0.

Also it can be seen that the fitness of the 9th SM is the best among all the members of the swarm, hence the 9th SM is considered as the global leader of the swarm, i.e. $GL = (0.1, 0.31)$. As the global leader decision has been implemented, the global limit count becomes 0.

Group number, SM number	SM_{newij}		f_i(x)	Fit_i
K = 1, SM = 1	1.4	−1	2.96	0.252525
K = 1, SM = 2	−1.56	−2.5	8.6836	0.103267
K = 1, SM = 3	0.6	0.4	0.52	0.657895
K = 1, SM = 4	0.3	1.5	2.34	0.299401
K = 1, SM = 5	0.1	1.5	2.26	0.306748
K = 1, SM = 6	0.69	3.2	10.7161	0.085353
K = 1, SM = 7	1.6	−0.3	2.65	0.273973
K = 1, SM = 8	1.1	0.2	1.25	0.444444
K = 1, SM = 9	−0.4	0.1	0.17	0.854701
K = 1, SM = 10	−0.8	−0.9	1.45	0.408163
K = 2, SM = 11	1.2	−0.4	1.6	0.384615
K = 2, SM = 12	1.8	0.3	3.33	0.230947
K = 2, SM = 13	−0.8	0.6	1	0.5
K = 2, SM = 14	−0.2	0.7	0.53	0.653595
K = 2, SM = 15	1.3	−1.5	3.94	0.202429
K = 2, SM = 16	−1.1	−0.165	1.237225	0.446982
K = 2, SM = 17	−0.264	−0.8	0.709696	0.584899
K = 2, SM = 18	0.45	−0.3	0.2925	0.773694
K = 2, SM = 19	−0.4	0.3	0.25	0.8
K = 2, SM = 20	0.8	−0.142	0.660164	0.60235

After global leader decision phase, the swarm is updated by local leader phase and other phases in the similar way. This process is continued iteratively, until the termination criteria is reached.

6 Conclusion

In this chapter, a recent swam intelligence based algorithm, namely Spider Monkey Optimization is discussed which is developed by taking inspiration from the social behavior of spider monkeys. In SMO, the local leader phase and the global leader phase help in exploitation of the search space, while exploration is done through the local leader decision phase and global leader decision phase. SMO performance analysis shows that SMO outpaced ABC, DE and PSO, in terms of dependability, effectiveness and precision. However, the presence of a large number of user dependent parameters in SMO is a matter of concern for further research. Self-adaptive parameter tuning may help to improve the robustness and reliability of the algorithm. Just in 4 years, a large number of publications on development and applications of SMO show that it has a great potential to be an efficient optimizer.

Note: SMO codes in C++, Python and Matlab can be downloaded from http://smo.scrs.in/.

References

1. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York, NY (1999)
2. Bansal, J.C., Sharma, H., Jadon, S.S., Clerc, M.: Spider monkey optimization algorithm for numerical optimization. Memetic Comput. **6**(1), 31–47 (2014)
3. Dhar, J., Arora, S.: Designing fuzzy rule base using spider monkey optimization algorithm in cooperative framework. Future Comput. Info. J. **2**(1), 31–38 (2017). ISSN 2314-7288
4. Sharma, A., Sharma, H., Bhargava, A., et al.: Memetic Comp. **9**, 311 (2017). https://doi.org/10.1007/s12293-016-0208-z
5. Wu, H., Yan, Y., Liu, C., Zhang, J.: Pattern synthesis of sparse linear arrays using spider monkey optimization. In: IEICE Transactions on Communications, Released 01 March 2017
6. Cheruku, R., Edla, D.R., Kuppili, V.: SM-RuleMiner: Spider monkey based rule miner using novel fitness function for diabetes classification. Comput. Biol. Med. **81**, 79–92 (2017)
7. Al-Azza, A.A., Al-Jodah, A.A., Harackiewicz, F.J.: Spider monkey optimization: a novel technique for antenna optimization. IEEE Antennas Wirel. Propag. Lett. **15**, 1016–1019 (2016)
8. Couzin, I.D, Laidre, M.E: Fission–fusion populations. Curr. Biol. **19**(15), R633–R635
9. https://www.nationalgeographic.com/animals/mammals/group/spider-monkeys/
10. https://animalcorner.co.uk/animals/spider-monkey/