Swarm Intelligence Algorithms and Applications to Real-world Optimization Problems: A Survey

Ibidun C. Obagbuwa

Discipline of Information Systems and Technology
School of Management, Information Technology, and Governance
University of KwaZulu-Natal
Westville, Durban, 4000, South Africa.

Abstract - Exact approaches have been inadequate to provide optimal results to complex optimization problems in acceptable time, and it is a common belief that nature produces an intrinsic result to complicated problems. Nature’s concepts and ideas were investigated by researchers to develop heuristic approaches that offer solutions to difficult problems. The technology of swarm intelligence (SI) that is motivated by the social habit of animals has been found effective to handle complex problems. There are different types of SI methods and their variants in the literature include cockroach swarm optimization, bees colony optimization, particle swarm optimization, and ant colony optimization. The focus of this paper is on SI techniques, and its application to some of the real world optimization problems such as timetabling, scheduling, transportation, and logistics problems. SI methods research trends and applications are highlighted.

Keywords - Swarm Intelligence Algorithms, Real-world Optimization Problems.

I. INTRODUCTION

Recent advancements in scientific research prompted the approach of offering solutions to complex optimization problems by the employment of nature-inspired algorithms (NIA). The traditional algorithms for optimization are not capable of solving complex optimization problems which are largely combinatorial and non-linear optimization problems efficiently. SI which is a branch of NIA centres on animals social habit for the design of techniques that are inspired by problem solving abilities originated from animals social behaviour [1]. The algorithms have the ability to search large solution space. Insects exhibit collective intelligence resulting from their interaction. Adil et al., stated that the communication systems established from insects social behaviours have been tailored as optimization framework for solving problems [1].

The problems that maximize or minimize a function containing some variables with certain constraints to be satisfied are known as optimization problems. The function to be minimized or maximized is called objective function $OF_n$ which is the goal/objective of the problem with respect to the decision variables. The $OF_n$ can either be minimized or maximized by the decision makers. Examples are minimizing overall production cost, or maximizing the whole profit [2, 3]. To create the $OF_n$, the parameters needed in connection with the decision variables are the cost minimization or profit maximization per unit product. Optimization problems can be modelled as constrained or unconstrained problems.

In constrained optimization problems, the search space is usually restricted by its constraints; while in unconstrained problem, constraints are absent or omitted and the whole domain of the $OF_n$ is the feasible set.

Applications with obvious constraints on the variables form constraints problems. The makeup of constraints include linear, nonlinear, convex. In addition the smoothness of the function, for instance differentiable or non-differentiable can be used to further classify constrained optimization problems [3]. Substituting the constraints with a penalty term in the objective function results in unconstrained problems, and this arise directly in some practical applications [3].

Unconstrained optimization problems can be described mathematically as shown in Equation 1:

$$\text{Minimize } f(x)$$

where $x \in \mathbb{R}^n$ is a real vector with $n \geq 1$ components and $f: \mathbb{R}^n \to \mathbb{R}$ being a smooth function.

Constrained optimization problems may include a single variable, and in case it involves more than one objective function, it is being referred to as multi-objective optimization problems [2, 3]. Single objective problem model is given as Equation 2:

$$\text{Minimize } f(x)$$

subject to: $c_i(x) = 0 \forall i \in E$

$c_i(x) \leq 0 \forall i \in I$

where $f$ and $c_i(x)$ are all smooth real-valued functions on a subset of $\mathbb{R}^n$. $E$ and $I$ are index sets for equality and inequality constraints respectively. The feasible set is the set of points $x$ that satisfy the constraints [3]. Multi-objective problem model is given as Equation 3:
Minimize $f(x) := [f_1(x), f_2(x), f_3(x), \ldots, f_k(x)]$  
subject to $g_i(x) \leq 0, \ i = 1, 2, \ldots, m$  
h_j(x) = 0 \ j = 1, 2, \ldots, p$

where $x = [x_1, x_2, \ldots, x_n]^T$ is the vector of decision variables. 
$f_i : R^n \rightarrow R, i = 1, 2, \ldots, k$ are the OFn and $g_i, h_j : R^n \rightarrow R, i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, p$ are the constraint functions of the problem [3].

### TABLE 1: SWARM INTELLIGENCE ALGORITHMS.

<table>
<thead>
<tr>
<th>NO</th>
<th>Algorithm</th>
<th>Author(s)</th>
<th>Year</th>
<th>Inspiration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Particle swarm optimization (PSO)</td>
<td>Kennedy and Eberhart</td>
<td>1995</td>
<td>Bird flocking and fish schooling [4]</td>
</tr>
<tr>
<td>2</td>
<td>Ant colony optimization (ACO)</td>
<td>Margo Dorigo</td>
<td>1992</td>
<td>Ants food foraging [5]</td>
</tr>
<tr>
<td>3</td>
<td>Bee colony optimization (BCO)</td>
<td>Pham et al</td>
<td>2005</td>
<td>Food foraging [6]</td>
</tr>
<tr>
<td>5</td>
<td>Glowworm swarms optimization (GSO)</td>
<td>Krishnanand and Ghose</td>
<td>2005</td>
<td>Behavioural patterns of glow worms [8]</td>
</tr>
<tr>
<td>6</td>
<td>Bioluminescent swarms optimization (BSO)</td>
<td>Oliveira et al.</td>
<td>2011</td>
<td>Luciferin-based attraction [9]</td>
</tr>
<tr>
<td>7</td>
<td>Firefly Algorithm (FA)</td>
<td>Yang X.</td>
<td>2009</td>
<td>Flashing behaviour of fireflies [10]</td>
</tr>
<tr>
<td>9</td>
<td>Grey wolf optimiser (GWO)</td>
<td>S. Mirjalili, S. M. Mirjalili, and A. Lewis</td>
<td>2014</td>
<td>wolves leadership hierarchy and hunting mechanism [12]</td>
</tr>
<tr>
<td>10</td>
<td>Cuckoo search (CS)</td>
<td>Yang Xin-She and Deb Suash</td>
<td>2009</td>
<td>Brooding behaviour of some cuckoo species [13]</td>
</tr>
<tr>
<td>11</td>
<td>Cat Swarm</td>
<td>Chu et al.</td>
<td>2007</td>
<td>cat swarming [14]</td>
</tr>
<tr>
<td>12</td>
<td>Good lattice swarm optimization</td>
<td>Su et al.</td>
<td>2007</td>
<td>swarming [15]</td>
</tr>
<tr>
<td>13</td>
<td>Fish Swarm/School</td>
<td>Li et al.</td>
<td>2002</td>
<td>swarming/Schooling [16]</td>
</tr>
<tr>
<td>14</td>
<td>Artificial immune system (AIS)</td>
<td>De Castro, Von Zuben and Nicosia and Cutellos</td>
<td>2002</td>
<td>Based on abstract structure and function of immune system [17]</td>
</tr>
<tr>
<td>15</td>
<td>Monkey search</td>
<td>Muecherino and Seref</td>
<td>2007</td>
<td>monkey climbing tree [18]</td>
</tr>
<tr>
<td>16</td>
<td>Bat Algorithm (BA)</td>
<td>X. S. Yang</td>
<td>2010</td>
<td>inspired by the echolocation [19]</td>
</tr>
<tr>
<td>17</td>
<td>Sheep flocks Algorithm (SFA)</td>
<td>Nara, Takeyama and Kim</td>
<td>1999</td>
<td>sheep flocks heredity [20]</td>
</tr>
</tbody>
</table>

Complex optimization problems are usually constrained problems; they have been formulated as constrained problems and have been solved using various SI algorithms. The paper is organised as follows: section presents different SI algorithms, section shows the application of SI algorithms in solving various real-world optimization problems, and equally shows the trends of applications in problems. The paper is summarized in section III.

II. SWARM INTELLIGENCE ALGORITHMS

Optimization techniques inspired by SI have attracted interest and gained popularity and attention because of their efficiency, flexibility, and versatility in solving real world applications. The techniques are described by the working mechanism that imitates the collective behaviour of social animals in a group where there is no central coordination. Since optimization is everywhere, SI algorithms are applicable across all discipline. People are
always optimizing something, either cost/time/resources minimization, or profit/output/performance/efficiency maximization. Examples of SI techniques inspired by social entities, animals and insects’ behaviours are shown in Table I. See the references in Table I for further details about the SI algorithms. SI algorithms and their variants have been successfully tailored to various real-world optimization problems as shown in section III. Table II depicts the trends of applications of SI algorithms to real world optimization problems.

III. APPLICATION OF SWARM INTELLIGENCE ALGORITHMS TO OPTIMIZATION PROBLEMS

Real world optimization problems include diversity of issues in revenue management, scheduling, risk management, transportation, communication operations, inventory planning, financial assets, computer operations, production planning etc. Sections to present the application of some of swarm intelligence algorithms and variants to a number of optimization problems.

A. Transportation Problems

Transportation problem’s objective is to minimize the transportation cost for transporting various quantities of a single homogeneous commodity from the origin points where it was initially stored to the desired destinations with the specified requirements at each destination [2, 21]. Transportation problem model is shown in Equation 4:

\[
\text{Minimize } Z = \sum_i \sum_j C_{ij} X_{ij}
\]

Subject to:

\[
\sum_j X_{ij} = S_i, \quad \forall i \\
\sum_i X_{ij} \leq D_j, \quad \forall i \\
X_{ij} \geq 0, \quad \forall i, j
\]

Where:

- \(C_{ij}\) is the cost of shipping a unit from source \(i\) to destination \(j\).
- \(S_i\) is the supply (in units) at source \(i\),
- \(D_j\) is the demand (in units) at destination \(j\) [2, 21].

Lucic and Teodorovic were inspired by the bees foraging behaviour and investigated bees behavior. They modelled and developed bee systems based on bee colonies foraging behaviour to provide solutions to complex combinatorial optimization problems (COP). The concept of swarm intelligence was employed by Lucic for modelling transportation problems [21]. Lucic and Teodorovic also integrated bees system with Fuzzy system and applied this to some complex transportation problems [22, 23].

Han and Zhifeng [24] proposed PSO for non-linear transportation problems (PSO-NLTP) to deal with non-linear transportation problems (NLTP) which cannot be solved by the methods that were used for linear-TP. PSO-NLTP was also successfully applied to both linear and non-linear transportation problems.

B. Travelling Salesman Problems

Travelling salesman problem (TSP) is a combinatorial optimization problem (COP) that is considered to be an NP-hard problem. Finding the possible shortest route that visits each city exactly once and returns to the original city in a given list of cities and pairwise distance is the task of a TSP. A situation where the distance between two cities does not depend on the direction is referred to as a symmetric TSP and if otherwise asymmetric. TSP model is shown in Equation 5 [2].

\[
\text{Minimize } Z = \sum_{(i,j) \in A} C_{ij} X_{ij}
\]

Subject to:

\[\sum_{i=0}^{n} i \neq j X_{ij} = 1, \quad j = 0, ..., n\]

After visiting \(j\), he must leave for another city \(i\)

\[\sum_{i=0}^{n} i \neq j X_{ij} = 1, \quad j = 0, ..., n\]

Where:

- \(n\) is the number of cities/locations(customers to visit),
- \(C_{ij}\) is cost/distance of travelling from city \(i\) to city \(j\),
- \(A\) is a set of arcs (i, j) that exist.

\[X_{ij} = \begin{cases} 1 & \text{if the salesman travels from city } i \text{ to city } j \\ 0 & \text{otherwise} \end{cases}\]

Yang et al., introduced algorithm of marriage in honey bees based on Nelder-Mead method optimization (NMFMBO) and algorithm of marriage in honey bees based on wolf park search optimization (WPS-MBO) [25, 26]. NMFMBO and WPS-MBO are successfully applied to TSP.

Sabine et al., demonstrated how communities of autonomous units can be used as a formal structure for modelling ant colony systems [28]. An ant colony system was modelled as TSP solely for a community.

C. Vehicle Routing Problem

The vehicle routing problem (VRP) is a COP and integer programming problem designed to provide solutions to fleet management, it is an important problem.
in the transportation, distribution and logistics fields. VRP was proposed by Dantzig and Ramser [29]. The goal of VRP is to minimize the distribution cost while delivering goods from central depot to customers [2]. Listed below are variations of VRP:

1. **VRP with pickup and delivery (VRPPD):** The goal of VRPPD is to find the optimal routes for a fleet of vehicles to visit the pickup and drop-off locations, goods are moved from certain pickup locations to delivery locations.

2. **VRP with LIFO (last item added must be the first item to be removed):** VRPLIFO is similar to the VRPPD, but with an additional restriction at the delivery point, the most recently picked up item must be the first to be delivered.

3. **VRP with time windows (VRPTW):** There is time windows at the delivery points within which the deliveries (or visits) must be made.

4. **Capacitated VRP with or without time windows (CVRP or CVRPTW):** The vehicles have limited carrying capacity of the goods that must be delivered.

VRP model is given as Equation 6:

\[
\text{Minimize } Z = \sum_{k=1}^{K} \sum_{(i,j) \in A} C_{ij}x_{kj}
\]

Subject to:

\[
\begin{align*}
\sum_{j=1}^{n} y_{ij} & = 1, \quad j = 2, 3, \ldots, n, \\
\sum_{i=1}^{n} y_{ij} & = 1, \quad j = 2, 3, \ldots, n, \\
\sum_{k=1}^{K} x_{kj} & = K, \\
\sum_{j=1}^{n} y_{ij} & = K, \\
\sum_{i=1}^{n} \sum_{j=2}^{n} D_{ij}x_{kj} & \leq U, \quad k = 1, 2, \ldots, K, \\
\sum_{k=1}^{K} x_{kj} & = y_{ij} \forall i,j, \\
\sum_{(i,j) \in S} x_{kj} & \leq |S| - 1, \forall \text{ subsets } S \text{ of } \{2, 3, \ldots, n\}, \\
y_{ij} & = 0 \text{ or } 1 \text{ } \forall (i,j) \in A \text{ and } \forall k, \\
y_{ij} & = 0 \text{ or } 1 \text{ } \forall (i,j) \in A
\end{align*}
\]

where a fleet of $M$ capacitated vehicles located in a depot $(i = 1)$ A set of customer sites (of size $N - 1$), each having a demand $D - j = 2, \ldots, N)$.

A cost $C_{ij}$ of travelling from location $i$ to location $j$.

The problem is to find a set of routes for delivering/picking up goods to/from the customer sites at minimum possible cost. It is assumed that the vehicle fleet is homogeneous and that each vehicle has a capacity of $U$ units.

\[
x_{kj} = \begin{cases} 1 & \text{if the vehicle } k \text{ travels on the arc } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}
\]

\[
y_{ij} = \begin{cases} 1 & \text{if any vehicle travels on the arc } (i,j) \\ 0 & \text{otherwise} \end{cases}
\]

Teodorovic and Dellaorco [30, 31] proposed bees-based models for weekly routing and scheduling of vehicles, and their goal is minimizing the total distance travelled by all partakers. The model was tested on ride sharing demand from Trani (a small city in southern Italy) using a data of 97 travellers with the estimate of the capacity of four passengers in their cars.

### D. Assignment Problems

Assignment problem can be described as the allocation of any task to any agent for execution at a cost that is determined by the agent-task assignment, given a number of agents and tasks. The goal of assignment problem is to minimize the overall cost of the assignment. It is expedient that all tasks must be assigned to agents, and this can be achieved by assigning precisely only one agent to one task, and only one task to one agent in such a way that the overall assignment cost is minimized [2].

Assignment problem in simplest form can be model as shown in Equation 7:

\[
\text{Minimize } Z = \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij}X_{ij}
\]

Subject to:

\[
\begin{align*}
\sum_{j=1}^{m} X_{ij} & = 1, \quad i = 1, \ldots, n, \\
\sum_{i=1}^{n} a_{ij}X_{ij} & \leq b_{j}, \quad j = 1, \ldots, m, \\
X_{ij} & \in \{0, 1\}, \quad i = 1, \ldots, n, \quad j = 1, \ldots, m.
\end{align*}
\]

where $n$ is the number of tasks $m$ is the number of servers $C_{ij}$ is the cost of assigning task $i$ to server $j$ $b_{j}$ is the units of resource available to server $j$ $a_{ij}$ is the units of resource required to perform task $i$ by server $j$.

\[
X_{ij} = \begin{cases} 1 & \text{if task } i \text{ is assigned to server } j \\ 0 & \text{Otherwise} \end{cases}
\]

By the inspiration of the recent success of SI algorithms, Bees-based and PSO-based system were applied to assignment problem. Adil et al., discussed application of ABC algorithm to generalized assignment problem which was found effective in solving small to medium sized generalized assignment problems [1]. PSO-based system such as Hybrid PSO (HPSO) was applied to assignment problems [32].

### E. Time Tabling Problems

A lot of organizations and businesses including public transport and schools (examination and courses) are face with timetabling problem [2]. Periodically, at every educational institution arises course-timetabling problems. Usually the problem involve the assignment of a set of events including lectures, classes, and tutorials into time slots number that are limited, given a set of constraints that must be met. Constraints such as student cannot attend two classes at a time [2].
Course tabling problem model is given in Equation 8:

\[
\text{Minimize } Z = \sum_{i} \sum_{j} C_{ij} X_{ij}
\]  

Subject to:
\[
\begin{align*}
\sum_{j \in J} X_{ij} &= S_i, \quad \forall i \in I \\
\sum_{i \in R_l} X_{ij} &\leq A_l, \quad \forall j \in J \forall i \in L \\
\sum_{i \in T_m} X_{ij} &\leq 1, \quad \forall j \in J \forall m \in M
\end{align*}
\]
\[X_{ij} \in \{0, 1\} \quad \forall i \in I \forall j \in J
\]

Where:
- \(I\) is set of all subject groups (index \(i\))
- \(J\) is set of time groups (index \(j\))
- \(L\) is set of classroom groups (index \(l\))
- \(M\) is set of subject groups in conflict (index \(m\))
- \(R_l\) is a subset of subject groups that can be allocated to classroom group \(l\)
- \(T_m\) is a subset of subject groups in conflict; the \(mth\) row of the conflict matrix
- \(A_l\) is the number of classrooms of type \(l\)
- \(S_i\) is the number of courses in subject group \(i\)
- \(C_{ij}\) is the desirability coefficient of assigning subject groups \(i\) to time groups \(j\)

Ant-based and PSO-based techniques were modelled and used to solve time tabling problems. Krzysztof et al. modeled Ant-based systems [33], and Daniel proposed a PSO-based model called PSO-NoConflicts for university examination time tabling problem [34].

\[\text{F. Network Routing Problems}\]

Network Routing is a means of selecting paths in a network through which data can be sent. Network Routing Problem Model is shown in Equation 9:

\[
\text{Minimize } Z = \sum_{(i,j) \in A} C_{ij} X_{ij}
\]

Subject to:
\[
\sum_{(j,i) \in A} X_{ji} - \sum_{(j,i) \in A} X_{ij} \geq b_i, \quad \forall i \in N \\
X_{ij} \geq 0, \quad \forall i, j \in A
\]

Where:
- \(N\) is the number of nodes \(A\) is a set of existing arcs \((i,j)\)
- \(C_{ij}\) is the arc length (or arc cost) associated with each arc \((i,j)\)
- \(X_{ij}\) is the flow from node \(i\) to node \(j\)

Ant-based and PSO-based techniques were modelled and used to solve time tabling problems. Krzysztof et al. modeled Ant-based systems [33], and Daniel proposed a PSO-based model called PSO-NoConflicts for university examination time tabling problem [34].

\[\text{G. Scheduling Problems}\]

Scheduling refers to allocation of resources to tasks. The goal of scheduling is the assignment of jobs to a single or to many machines so that several operational criteria are met. These criteria are usually modelled as the minimization of one or more objective functions. Given an \(n\) jobs and \(m\) machines such that a job can only be processed on a machine at a time, and each job must be processed on all machines in a given order, and each job must be completely processed [2]. The objective is to minimize total completion times of all the jobs.

Scheduling model is given in Equation 10:

\[
\begin{align*}
\text{Minimize } Z &= \sum_{(i,j) \in A} t_{j(m),j} \\
\text{Subject to:}
\end{align*}
\]
\[
\begin{align*}
t_{j(r+1),j} &\geq t_{j(r),j} + P_{j(r),j} \quad \forall r = 1, 2, ..., m - 1 \quad \text{and} \quad \forall j \\
t_{ij} - t_{ik} &\leq U(1 - X_{ijk}), \quad \forall i, j, k \\
t_{ik} - t_{ij} &\leq Pa + UX_{ijk}, \quad \forall i, j, k \\
t_{ij} &\geq 0, \quad \forall i, j
\end{align*}
\]
\[X_{ijk} \in \{0, 1\}, \quad \forall i, j, k
\]

Where:
- \(n\) is the number of jobs \(m\) is the number of machines,
- \(P_{j}\) is the processing time of job \(j\) on machine \(i\),
- \(j(r)\) is the order of machines/operations for job \(j\)
- \(t_{ij}\) is the start time of job \(j\) on machine \(i\)

Quite a number of SI techniques were investigated, modelled and used to solve various types of scheduling problems such as job-shop, flow-shop, and MT10 Job, machine and power control etc. PSO variants were investigated and applied in [38]. Chong et al and Chin et al., presented BCO variants for scheduling problems [39, 40]. Chandramouli modelled “an improved sheep flock heredity algorithm for job shop scheduling and flow shop scheduling problems” [41].

A novel PSO-based approach for optimal schedule of refrigerators using experimental models was presented [42] where two strategies were proposed to reduce the annual electricity consumption cost for a refrigerator using PSO. The strategies control the temperature of the refrigerator at the start and end instants of the pick-load-span. In addition the strategies determine the range at which the temperature...
is working to minimise the annual cost of electricity. The result of their simulation studies show that an annual cost reduction of more than 28 percent, and also the annual consumption and peak hour load of the refrigerator decreased by 20.46 percent.

The trends of applications of SI algorithms in solving real world optimization problems are shown in Table II. PSO and variants are shown to be more widely used for solving real world optimization problems, followed by BCO and variant, and then ACO and variants. PSO, BCO and ACO algorithms and respective variants were widely used when compared to other swarm intelligence algorithms.

IV. CONCLUSIONS

This paper presented swarm intelligence (SI) techniques and application to real world problems. The trends of application of different SI methods and their variants on optimization problems are shown. Some of the optimization problems solved by SI techniques in the literature are also described in this paper.

REFERENCES


<table>
<thead>
<tr>
<th>Authors</th>
<th>Algorithm</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PSO FAMILY</strong> 2005</td>
<td>Daniel, R. [34]</td>
<td>PSO-NoConflicts</td>
</tr>
<tr>
<td>2009</td>
<td>Ponnambalam et al. [38]</td>
<td>MODPSO</td>
</tr>
<tr>
<td>2009</td>
<td>Han, H.and Zhifeng,H. [24]</td>
<td>PSO-NLTP</td>
</tr>
<tr>
<td>2011</td>
<td>Li et al. [35]</td>
<td>APSO-AR</td>
</tr>
<tr>
<td>2015</td>
<td>Ze-Dong et al. [44]</td>
<td>PSO</td>
</tr>
<tr>
<td>2015</td>
<td>Naglaa et al. [46]</td>
<td>PSO</td>
</tr>
<tr>
<td>2016</td>
<td>Saleh, et al [42]</td>
<td>PSO</td>
</tr>
<tr>
<td>2016</td>
<td>Quy and Pham [45]</td>
<td>PSO</td>
</tr>
<tr>
<td>2016</td>
<td>Tanweer et al. [43]</td>
<td>PSO</td>
</tr>
<tr>
<td>2017</td>
<td>Khairuzzaman et al. [49]</td>
<td>PSO</td>
</tr>
<tr>
<td>2017</td>
<td>Kumawat et al. [50]</td>
<td>PSO</td>
</tr>
<tr>
<td><strong>BEE FAMILY</strong> 2002</td>
<td>Lucic P, Teodorovic D [22]</td>
<td>Bee and fuzzy system</td>
</tr>
<tr>
<td>2002</td>
<td>Lucic P. [21]</td>
<td>Bee and fuzzy system</td>
</tr>
<tr>
<td>2003</td>
<td>Lucic P, Teodorovic D [23]</td>
<td>Bee and fuzzy system</td>
</tr>
<tr>
<td>2005</td>
<td>Teodorovic and Dell [31]</td>
<td>BCO</td>
</tr>
<tr>
<td>2006</td>
<td>Chong CS, et. al. [39]</td>
<td>BCO</td>
</tr>
<tr>
<td>2007</td>
<td>Chin, S. et. al. [40]</td>
<td>Bee colony</td>
</tr>
<tr>
<td>2007</td>
<td>Yang C, Jie-Chen J , Tu X [26]</td>
<td>WPS-MBO</td>
</tr>
<tr>
<td>2008</td>
<td>Teodorovic [30]</td>
<td>BCO</td>
</tr>
<tr>
<td><strong>ANT FAMILY</strong> 2002</td>
<td>Krzysztof Socha et al [33]</td>
<td>ACO-MAX-MIN Ant System</td>
</tr>
<tr>
<td>2007</td>
<td>Rizzoli A. et al [27]</td>
<td>ACO</td>
</tr>
<tr>
<td>2009</td>
<td>Sabine Kuske et al [28]</td>
<td>ACO</td>
</tr>
<tr>
<td><strong>OTHERS</strong> 2002</td>
<td>Passino,K. [7]</td>
<td>BFOA</td>
</tr>
<tr>
<td>2005</td>
<td>Krishmanand,K. etal [8]</td>
<td>GSO</td>
</tr>
<tr>
<td>2011</td>
<td>Chandramouli,A. [41]</td>
<td>Sheep flock heredity</td>
</tr>
</tbody>
</table>