Volume 3, Issue 8, August 2013 ISSN: 2277 128X
International Journal of Advanced Research in
Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Taxonomy of Swarm Optimization

Poonam Sehrawat
Research Scholar
Computer Science Department
Banasthali University, Banasthali, India

Harish Rohil
Assistant Professor
Department of Computer Science & Applications
Chaudhary Devi Lal University, Sirsa, India

Abstract*—*Swarm intelligence is a relatively new subfield of artificial intelligence which studies the social behavior of simple agents. It can be seen in ants' colonies, fireflies, flocks of birds, bee hives etc. The goal of this research paper is to describe the taxonomy of swarm optimization. In the beginning two main algorithms of swarm optimization are ant colony optimization and particle swarm optimization presented. In recent years, new swarm intelligence algorithms have come into light that are being presented in this paper.

Keywords*—*Swarm Optimization, ACO, PSO, Bio- Inspired Algorithms

I. Introduction

The growing complexity of real-world problems has motivated computer scientists to search for efficient problem-solving methods. Swarm optimization meta-heuristics are outstanding examples that nature has been an unending source of inspiration.

Swarm intelligence is a relatively new subfield of artificial intelligence which studies the emergent collective intelligence of groups of simple agents. It is based on social behavior that can be observed in nature, such as **ant colonies, flocks of birds, fish schools and bee hives,** where a number of individuals with limited capabilities are able to come to intelligent solutions for complex problems. Swarms inherently use forms of decentralized control and self-organization to achieve their goals. Researchers in computer science have developed swarm-based systems in response to the observed success and efficiency of swarms in nature to solve difficult problems. In such biological swarms, the individuals (ant, bee, termite, bird or fish) are by no means complete engineers, but instead are simple creatures with limited cognitive abilities and limited means to communicate. Yet the complete swarm exhibits intelligent or collective behavior, providing efficient solutions for complex problems such as predator evasion and shortest path finding [5, 22].

The main principles of the collective behavior of swarms as presented in Figure 1 are [39]:

1. *Homogeneity*: every bird in flock has the same behavior model. The flock moves without a leader, even though temporary leaders seem to appear.

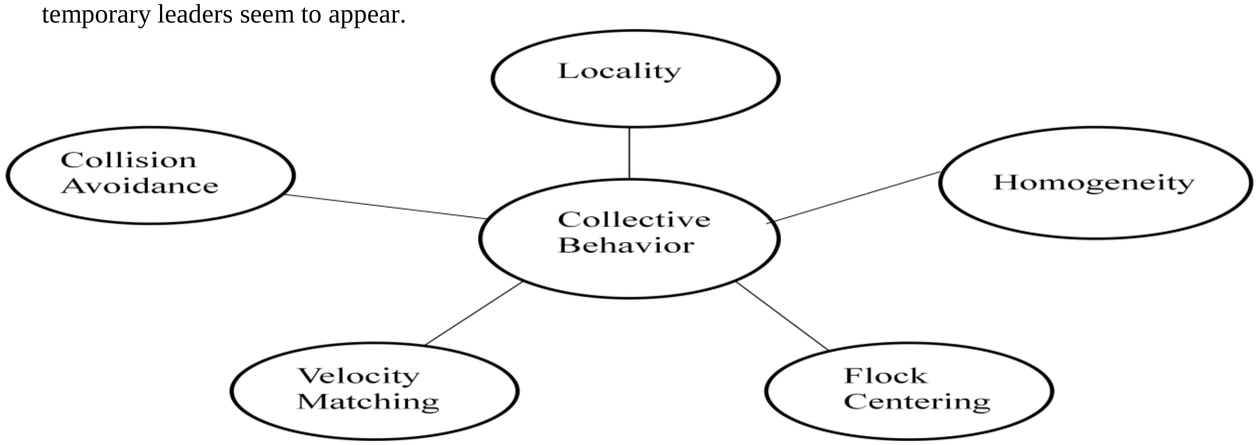

Figure 1 Main Principles of Collective Behavior of Swarms

2. *Locality*: the motion of each bird is only influenced by its nearest flock mates.
3. *Collision Avoidance*: avoid with nearby flock mates.
4. *Velocity Matching:* attempt to match velocity with nearby flock mates.
5. *Flock Centering*: attempt to stay close to nearby flock mates.

II. TAXONOMY OF SWARM OPTIMIZATION

Swarm intelligence is a relatively new approach to problem solving that takes inspiration from the social behaviors of insects and of other animals. In literature, a large number of swarm optimization approaches have been discussed at length. Different researchers have presented different taxonomy of these approaches. Common Taxonomy is as under:

A. Ant Colony Optimization

Ant colony optimization (ACO) is a population-based metaheuristic that can be used to find approximate solutions to difficult optimization problems. Ant colony optimization (ACO) takes inspiration from the foraging behavior of some ant species. These ants deposit pheromone on the ground in order to mark some favorable path that should be followed by other members of the colony. Ant colony optimization exploits a similar mechanism for solving optimization problems [1, 2].Fromthe early nineties, when the first ant colony optimization algorithm was proposed, ACO attracted the attention of more researchers and a relatively large amount of successful applications are now available. Moreover, a substantial corpus of theoretical results is becoming available that provides useful guidelines to researchers and practitioners in further applications of ACO.

In the forties and fifties of the twentieth century, the French entomologist Pierre-Paul Grass´e [3] observed that some species of termites react to what he called "significant stimuli". He observed that the effects of these reactions can act as new significant stimuli for both the insect that produced them and for the other insects in the colony. Grass´e [4] used the term *stigmergy*to describe this particular type of communication in which the "workers are stimulated by the performance they have achieved".The two main characteristics of stigmergy that differentiate it from other forms of communication are the following:

- Stigmergy is an indirect, non-symbolic form of communication mediated by the environment: insects exchange information by modifying their environment; and
- Stigmergic information is local: it can only be accessed by those insects that visit the locus in which it was released (or its immediate neighborhood).

Examples of stigmergy can be observed in colonies of ants. In many ant species, ants walking to and from a food source deposit on the ground a substance called *pheromone*. Other ants perceive the presence of pheromone and tend to follow paths where pheromone concentration is higher. Through this mechanism, ants are able to transport food to their nest in a remarkably effective way [1].The principles are illustrated in Figure 2for finding the shortest path between a food source (right) and the nest (left). Two ants start from their nest (left) and look for the shortest path to a food source (right). Initially, no pheromone is present on either trails, so there is a 50–50 chance of choosing either of the two possible paths. Suppose one ant chooses the lower trail and the other one the upper trail. The ant that has chosen the lower (shorter) trail will have returned faster to the nest, resulting in twice as much pheromone on the lower trail as on the upper one [5].

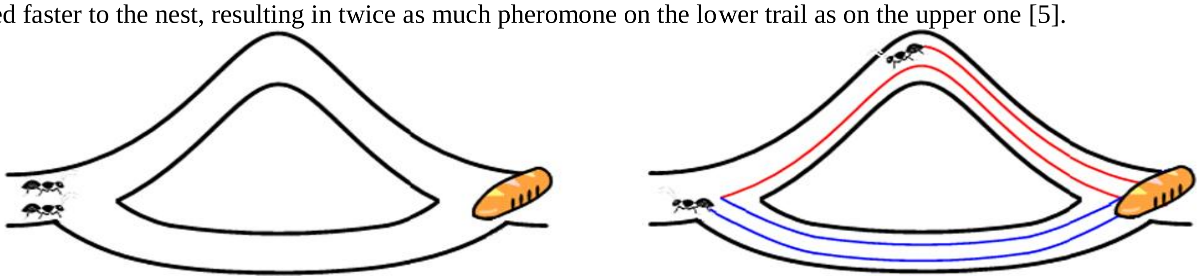

Figure 2 Path Selectionswith Ants Using Pheromone [5]

As a result, the probability that the next ant will choose the lower, shorter trail will be twice as high, resulting in more pheromone; thus more ants will choose this trail, until eventually (almost) all ants will follow the shorter path.

The pseudo code of ACO is:

Set parameters, initialize pheromone trails
Whiletermination condition not met **do**
ConstructAntSolutions
ApplyLocalSearch(optional)
UpdatePheromones
End while

Many variations of the ACO algorithm have been proposed that focus on improving the efficiency of the original algorithm that is following:

Elitist Ant System [6], Ant-Q [9], Ant Colony System [7], ANTS [11], Best-Worst Ant System [14], Population-based ACO [10], Touring ACO [12] and Beam-ACO [13].

B. Particle Swarm Optimization

Kennedy and Eberhart proposed the particle swarm optimization (PSO) algorithm in 1995 were essentially aimed at producing computational intelligence by exploiting simple analogues of social interaction, rather than purely individual cognitive abilities. The PSO algorithm has become an evolutionary computation technique and an important heuristic algorithm in recent years. Particle swarm optimization (PSO) is a population based method, where a population is called a swarm. The PSO algorithm simulates the behaviors of bird flocking [15].

Each single solution is a "bird" in the search space and these single solutions considered as a "particle" in the search space. All particles have fitness values, which are evaluated by the fitness function to be optimized. The particles also have velocities which direct the flight of the particles. Each particle has random velocity and adjusts it according to its own flying experience and its neighborhood experience. Each particle is updated by the two best values in every iteration. Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called **pbest.** Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the swarm. This best value is a global best and is called **gbest.** After finding the two best values, each particle updates its velocity and position according to Eqs. (1) and (2) [15]:

$$v_{id}(t+1)= v_{id}(t)+c_1 \text{ rand}(\ )(pbest-x_{id}(t))+c2 \text{ Rand}(\ )(gbest-x_{id}(t)) \quad (1)$$

$$x_{id}(t+1)=x_{id}(t)+ v_{id}(t+1) \quad (2)$$

Where $vi_d(t+1)$ is the velocity of the i^{th} particle at time (t+1) in D-dimensional space and $v_{id}(t)$ is the velocity of the i^{th} particle at t time. x_{id} is the current particle position. rand ( ) and Rand( ) are a random number in (0, 1); c_1 is the cognitive factor; c_2 is the social factor. Usually c1 and c2 are set to be 2.The velocities of particles in each dimension are clamped to amaximum velocity $Vmax \geq V \geq -Vmax$. The PSO is summarized as follows [15]:

i). Initialize a population of particles with random positions and velocities in search space
ii). While not termination condition satisfy
iii). Calculate fitness of each particle's position (P)
iv). if fitness (P) better than pbest then
v). Set pbesti= P
vi). end if
vii). Set best of pbest as gbest
viii). Change velocity and position of particle according to Eqs. (1) & (2)
ix). Loop to step 2 until a criteria is met or maximum iteration.

Kennedy and Eberhart proposed *"discrete binary particle swarm optimization"* extends the capability of the continuous PSO. A Discrete Particle Swarm Optimization (DPSO) proposed in does not consider any velocity since, from the lack of continuity of the movement in a discrete space, the notion of velocity loses sense; however they kept the attraction of the best positions. They interpret the weights of the updating equation as probabilities that, at each iteration, each particle has a random behavior, or acts in a way guided by the effect of an attraction. The moves in a discrete or combinatorial space are jumps from one solution to another [16].

In 1998, Shi and Eberhart et al. proposed another method called the *"Linearly decreasing inertia weight method."* In this method, the particle updates its velocity and position withEqs. (3) and (4) as follows [17]:

$$vi_d(t+1)= \omega v_{id}(t)+c_1 \text{ rand}(\ )(pbest-x_{id}(t))+c2 \text{ Rand}(\ )(gbest-x_{id}(t)) \quad (3)$$

$$xi_d(t+1)=x_{id}(t)+v_{id}(t+1) \quad (4)$$

Clerc and Kennedy et al. proposed "*Constricted PSO*" to improve the convergence rate. They used Constriction factor (K) to prevent explosion and particles to convergence on a local optima. In Clerc's method, the particle updates its velocity and position with Eqs. (5) And (6) [18]:

$$vi_d(t+1)= \chi[v_{id}(t)+c_1 \text{ rand}(\ )(pbest-x_{id}(t))+c2 \text{ Rand}(\ )(gbest-x_{id}(t))] \quad (5)$$

$$xi_d(t+1)=x_{id}(t)+v_{id}(t+1) \quad (6)$$

$$\chi= \frac{2}{|2-\varphi-\sqrt{\varphi 2 - 4\varphi}|} \quad \text{where } \varphi=c_1 + c_2,\ \varphi>4 \quad (7)$$

In the standard version of PSO, the effective sources of influence are in fact only two: self and best neighbor. Information from the remaining neighbors is unused. Mendes has revised the way particles interact with their neighbors. Whereas in the traditional algorithm each particle is affected by its own previous performance and the single best success found in its neighborhood, in Mendes' *"fully informed particle swarm"* (FIPS), the particle is affected by all its neighbors, sometimes with no influence from its own previous success [19].

C. Firefly Algorithm

Lampyridaeis a family of insects that are capable to produce natural light (bioluminescence) to attract a mate or a prey. They are commonly called fireflies or lightning bugs. In the species Lampyrisnoctilucathe fireflies are also known as glow-worms and, despite of the name, they are not worms. In this species, it is always the female who glows, and only the male has wings. In other species, Luciolalusitanica, both male and female firefly may emit light and both have wings. If a firefly is hungry or looks for a mate its light glows brighter in order to make the attraction of insects or mates more effective.[20, 21, 22].

Firefly Algorithm (FA) is among the most powerful algorithms for optimization. The Firefly Algorithm was developed by Yang is inspired by biochemical and social aspects of real fireflies and it was based on the idealized behaviour of the flashing characteristics of fireflies. Most fireflies produce short and rhythmic flashes are to attract mating partners (communication), and to attract potential prey. In addition, flashing may also serve as a protective warning mechanism. The following three rules are idealized for flashing characteristics of FA [23]:

1. All fireflies are unisex so that one firefly is attracted to other fireflies regardless of their sex.
2. Attractiveness is proportional to their brightness, thus for any two flashing fireflies, the less bright one will move towards the brighter one. The attractiveness is proportional to the brightness and they both decrease as their distance increases. If no one is brighter than a particular firefly, it moves randomly.
3. The brightness or light intensity of a firefly is determined by the landscape of the objective function to be optimized.

In the simplest form, the light intensity I(r) varies with the distance r monotonically and exponentially. That is:

$I = I_0 e^{-r\gamma}$ (8)

Where I_0 is the original light intensity and γ is the light absorption coefficient. As a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, we can now define the attractiveness β of a firefly by:

$\beta = \beta_0 e^{-\gamma r2}$ (9)

Where β_0 is the attractiveness at r = 0.The movement of a firefly i is attracted to another more attractive (brighter) firefly j is determined by:

$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha$ (10)

Where $\beta_0 = 1$, $\alpha \in [0, 1]$ and $\gamma = 1$.The basic FA assumes that a population of n fireflies x_i, i=1,2,…..,n initially positioned randomly in the space and intensity i of each firefly is associated with the objective function. Only firefly with higher light intensity attracts the smaller light intensity i.e, $I_j > I_i$, j=1,2,…n, j≠i. Attractiveness or the brightness of firefly varies with the distance $r_{ij} = \|xi - xj\|_2$ between firefly i and firefly j. In addition the light intensity I decrease with the distance from its source and it is also absorbed in the media, so attractiveness is also vary with the absorption. Thus most of the fireflies are visible only to a limited distance. The FA is summarized in Figure 3 [23]:

1. Objective function f(x), $x = (x_1, \ldots, x_d)^T$
2. Generate initial population of fireflies x_i (i=1,2,...n)
3. Define light absorption coefficient γ
4. While(t <MaxGeneration)
5. for i=1:n all n fireflies
6. for j=1:i all n fireflies
7. Light intensity I_i at xi is determined by $f(x_i)$
8. if (Ij>Ii), Move firefly i towards j in d-dimension;
9. endif
10. Attractiveness varies with distance r via exp[-γr]
11. Evaluate new solution and update light intensity
12. end for j
13. end for i
14. Rank the fireflies and find the current best
15. end while

Bee Based Algorithm

The Bee Colony meta-heuristic belongs to the class of Nature-Inspired Algorithms which are inspired by various biological and natural processes observed in honeybee. Bee-inspired approaches in this narrower sense can be roughly classified into two different main types. The first group is inspired by the foraging behavior of honeybee. The basic idea behind this approach is to create a colony of artificial bees able to efficiently solve hard combinatorial optimization problems. The second group of bee-based algorithm is less directly inspired by mating behavior in honey bee. Each honey-bees colony consists of the queen, drones, workers, and broods. The marriage process starts with a dance performed by the queen who then starts a mating flight during which the drones follow the queen and mate with them in the air. In each mating, sperm reaches the *spermatheca*and accumulates there to form the genetic pool of the colony. Each time a queen lays fertilized eggs, she retrieves at random a mixture of the sperms accumulated in the *spermatheca*to fertilize the egg [38]. Few algorithms appeared during the last decades are:

Bee System (BS) [27], BeeHive [31], Bee Colony Optimization (BCO) [29], Artificial Bee Colony (ABC) [8], Bee Swarm Optimization (BSO) [25], Virtual Bee Algorithm (VBA) [32], Honey Bee Colony Algorithm (HBCA) [24], Bee Algorithm (BA) [28].

Marriage Bee Optimization (MBO)is the first search algorithm inspired by thisbehaviour developed by [33]. In the artificial analogue model, the mating flight can be visualized as a set of transitions in a state space where the queen moves between the

different states in the space and mate with the drone encountered at each state probabilistically. The probability of mating is high when either the queen is still in the start of her 2 mating flight and therefore her speed is high, or when the fitness of the drone is as good as the queen's one. The algorithm starts with initializing the queen's genotype at random. After that, a heuristic is used to improve the queen's genotype realized by workers. In [30] the marriage of honey bee is analyzed as the continuation of the work presented in [33].
Artificial Bee Colony (ABC) algorithm, the colony of artificial bees contains three groups of bees: employed bees, onlookers and scouts. A bee waiting on the dance area for making decision to choose a food source is called an onlooker and a bee going to the food source visited by it previously is named an employed bee. A bee carrying out random search is called a scout. In the ABC algorithm, first half of the colony consists of employed artificial bees and the second half constitutes the onlookers. For every food source, there is only one employed bee. In other words, the number of employed bees is equal to the number of food sources around the hive. The employed bee whose food source is exhausted by the employed and onlooker bees becomes a scout [26].
The main steps of the algorithm are given below:

- i). Initialize the population of each bee
- ii). Repeat condition not met
  - a) Place the employed bees on the food sources
  - b) Place the onlooker bees on the food sources
  - c) Send the scouts to the search area for discovering new food sources
- iii). Until termination condition are not met.

D. Bat Algorithm

Bats are fascinating animals. They are the only mammals with wings and they also have advanced capability of echolocation. It is estimated that there are about 996 different species which account for up to 20% of all mammal species. Their size ranges from the tiny bumblebee bat (of about 1.5 to 2g) to the giant bats with wingspan of about 2m and weight up to about 1 kg. Microbats typically have forearm length of about 2.2 to 11cm [34].
For simplicity, use the following approximate or idealized rules [34]:

1. All bats use echolocation to sense distance, and they also 'know' the difference between food/prey and background barriers in some magical way;
2. Bats fly randomly with velocity v_i at position x_i with a fixed frequency fmin, varying wavelength λand loudness A_0 to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission r ∈ [0, 1], depending on the proximity of their target;
3. Although the loudness can vary in many ways, we assume that the loudness varies from a large (positive) A_0 to a minimum constant value A_{min}.

E. Roach Infestation Optimization

In RIO algorithm, cockroaches' agents are defined using three simple behaviors: cockroaches search for the darkest location in the search space and the fitness value is directly proportional to the level of darkness (find darkness phase); cockroaches socialize with nearby cockroaches (find friend phase) and third behavior is cockroaches periodically become hungry and leave the friendship to search for food (find food phase) [35].

F. Shuffled Frog Algorithm

Recently, two researchers, MuzaffarEusuff and Kevin Lansey brought up a new meta-heuristic algorithm—Shuffled Frog-Leaping Algorithm (SFLA) through observing, imitating and modeling the behavior of frogs searching for food laid on discrete stones randomly located in a pond [37]. Shuffled frog leaping algorithm (SFLA) is a memetic meta-heuristic that is based on evolution of memes carried by interactive individuals and a global exchange of information among the frog population. It combines the advantages of the genetic-based memeticalgorithm (MA) and the social behavior-based PSO algorithm with such characteristics as simple concept, fewer parameters adjustment, prompt formation, great capability in global search and easy implementation.In the SFLA, there are a number of frogs with the same structure but different adaptabilities. Each frog represents a feasible solution to an optimization problem. The entire population of frogs is divided into a number of frog memeplexes according to specific principles and each memeplex represents a type of meme. Frogs in the memeplex conduct local exploration of solution space according to specific strategies which allow the transference of meme among local individuals. To prevent local optima, a submemeplex is constructed in each memeplex, which consists of frogs chosen on the basis of their respective fitness. The better the fitness, the easier it is chosen [36].

III. Conclusion

This work presents taxonomy of the most recent development in swarm optimization. In the beginning two main algorithms of swarm optimization were in picture ant colony optimization and particle swarm optimization. In recent years, new swarm intelligence algorithms have come into light that are reviewed in this paper. Swarm intelligence is a new domain of Artificial Intelligence. A few evolutionary algorithms have been used with multi-objective functions. Swarm optimization algorithms such as ant colony, particle swarm optimization, bee based algorithms and firefly algorithm have been used in wide variety of engineering problems.

References

[1] Dorigo, M., and Stützle, T. “Ant Colony Optimization”, MIT Press, 2004.
[2] Dorigo, M., Birattari, M., and Stützle, T., “Ant Colony Optimization”, IEEE Computational Intelligence Magazine, November 2006, 28-39.
[3] Grass´e,P.P. “Les InsectesDansLeurUnivers”, Paris, France: Ed. du Palais de la d´ecouverte,1946.
[4] Grass´e, “La reconstruction du nidet les coordinationsinterindividuelles chez BellicositermesNatalensis et Cubitermes sp. La th´eorie de la stigmergie: Essaid’interpr´etation du comportementdes termites constructeurs,” InsectesSociaux, 1959, Vol. 6, 41–81.
[5] Martens, D., Baesens, B., and Fawcett, T, “Editorial survey: swarm intelligence for data mining”, Mach Learn (2011) 82, 1–42.
[6] Dorigo, M.,Maniezzo, V., &Colorni, A. “Ant System: optimization by a colony of cooperating agents”, IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, 26(1),1996, 29–41.
[7] Dorigo, M., & Gambardella, L. “Ant colony system: a cooperative learning approach to the traveling salesman problem”, IEEE Transactions on Evolutionary Computation, 1(1), 1997, 53–66.
[8] Drias, H., Sadeg, S., Yahi, S.”Cooperative bees swarm for solving the maximum weighted satisfiability problem”, IWAAN International Work Conference on Artificial and Natural Neural Networks, Barcelona, Spain, 2005, 318-325.
[9] Gambardella, L. M., &Dorigo, M. “Ant-Q: a reinforcement learning approach to the traveling salesman problem. In A. Prieditis& S. Russell (Eds.), Proceedings of the twelfth international conference on machine learning, 1995, 252–260, Morgan Kaufmann: Palo Alto.
[10] Guntsch, M., &Middendorf, M. (2002). A population based approach for ACO. In Proceedings of the applications of evolutionary computing on EvoWorkshops 2002,72–81, London: Springer.
[11] Maniezzo, V.” Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem”, INFORMS Journal on Computing, 11(4), 1999, 358–369.
[12] Hiroyasu, T., Miki, M., Ono, Y., & Minami, Y. “Ant colony for continuous functions”, Tech. rep., The Science and Engineering Doshisha University, 2000.
[13] Blum, C. “Beam-ACO: hybridizing ant colony optimization with beam search: an application to open shop scheduling”, Computers and Operations Research, 32(6), 2005b, 1565–1591.
[14] Cordon, O., de Viana, I. F., and Herrera, F. “Analysis of the best-worst ant system and its variants on the QAP”, In M. Dorigo, G. Di Caro, & M. Sampels (Eds.), Lecture notes in computer science: Vol. 2463. Proceedings of the third international workshop on ant algorithms (ANTS’2002), 2002, 228–234, Brussels: Springer.
[15] Kennedy, J., and Eberhart, R.C. “Particle swarm optimization”, Proceedings of IEEE International Conference on Neural Networks, Piscataway, NJ., 1995, 1942-1948.
[16] Kennedy, J. and Eberhart, R.C “A Discrete binary version of the particle swarm algorithm”, Proceedings of the Conference on Systems, Man, and Cybernetics, Piscataway: IEEE, 1997, 4104–4109.
[17] Shi, Y., and Eberhart, R.C. “A modified particle swarm optimizer”, Proceedings of the IEEE International Conference on Evolutionary Computation, Piscataway, 1998, pp. 69–73.
[18] Clerc, M., “The particle swarm—explosion, stability, and convergence in a multidimensional complex space”, IEEE Transactions on Evolutionary Computation, Vol. 6, No. 1, 2002, 58-73.
[19] Mendes, R., Kennedy, J., and Neves, J. “The Fully Informed Particle Swarm: Simpler, Maybe Better”, IEEE Transactions on Evolutionary Computation, Vol. 8, No. 3, 2004, 204-210.
[20] H. Fraga, “Firefly luminescence: a historical perspective and recent developments”, Journal of Photochemical &Photobiological Sciences, 2008, Vol. 7, 146–158.
[21] O. Shimomura, “Bioluminescense: Chemical Principles and Methods”, World Scientific Publishing, 2006.
[22] R.S. Parpinelli, and H.S. Lopes, “New inspirations in swarm intelligence: a survey”, International Journal of Bio-Inspired Computation, 2011, Vol. 3, No. 1, 1-16.
[23] X.S. Yang, “Firefly algorithms for multimodal optimization”, SAGA 2009, Lecture Notes in Computer Science, 5792, 2009, 169-178.
[24] Chong, C.S., Low, M.Y.H, Sivakumar, A.I.and Gay, K.L. “A bee colony optimization algorithm to job shop scheduling”, Proceedings of the 37th Winter Simulation, Monterey, California,2006, 1954-1961.
[25] Karaboga, D. “An idea based on honey bee swarm for numerical optimization’, Technical report, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
[26] Karaboga,D. and Basturk,B. “A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm”,J Glob Optim (2007) 39:459–471.

[27] Lučić, P., and Teodorović, D. "Bee system: modelling combinatorial optimization transportation engineering problems by swarm intelligence", In: Preprints of the TRISTAN IV Triennial Symposium on Transportation Analysis, Sao Miguel, Azores Islands, Portugal, 2001, 441-445.
[28] Pham, D.T., Kog, E., Ghanbarzadeh, A., Otri, S., Rahim, S., and Zaidi, M. "The Bees algorithm : A novel tool for complex optimisation problems", IPROMS 2006 Proceeding 2nd International Virtual Conference on Intelligent Production Machines and Systems, Oxford, Elsevier,2006.
[29] Teodorovic, D., and Dell'Orco, M."Bee Colony Optimization - A cooperative learning approach to complex transportation problems", Advanced OR and AI Methods in Transportation, 2005, 51-60.
[30] Teo, J., Abbass, H.A. "A true annealing approach to the marriage in honey-bees optimization algorithm", International Journal of Computational Intelligence and Applications, 3(2), 2003, 199-211.
[31] Wedde, H.F., Farooq, M., and Zhang, Y. "BeeHive: An efficient fault-tolerant routing algorithm inspired by honey bee behaviour",Ant Colony, Optimization and Swarm Intelligence, Eds. M. Dorigo, Lecture Notes in Computer Science 3172, Springer Berlin, 2004, 83-94.
[32] Yang, X.S., "Engineering optimizations via nature-inspired virtual bee algorithms", IWINAC 2005, LNCS 3562, Yang, J. M. and J.R. Alvarez (Eds.), Springer-Verlag, Berlin Heidelberg, 2005, 317-323.
[33] Abbass, H.A."MBO: Marriage in honey bees optimization a haplometrosispolygynous swarming approach", CEC2001 Proceedings of the Congress on Evolutionary Computation, Seoul, Korea, 2001, 207-214
[34] Yang, X. "A new metaheuristic bat-inspired algorithm", Nature Inspired Cooerative Strategies for Optimization (NISCO 2010), Studies in Computational Intelligence, 2010 Vol. 284, 65–74, Springer Berlin.
[35] Havens, T., Spain, C., Salmon, N. and Keller, J. "Roach infestation optimization", IEEE Swarm Intelligence Symposium, September, 2008, 1–7.
[36] Eusuff, M., Lansey,K., and Pasha,F. "Shuffled frog-leaping algorithm: a memetic meta- heuristicfor discrete optimization", Engineering Optimization, 2006, 38 (2), pp. 129-154.
[37] Eusuff, M.M. and Lansey, K.E. "Optimization of water distribution network design using the shuffled frog leaping algorithm", Journal of Water Resource Planning Management, 2003, 210 – 225.
[38] Bitam, S., Batouche, M., and Talbi, E., "A survey on bee colony algorithms",IEEE,2010.
[39] Grosan, C., Abraham, A., and Chis, M., "Swarm intelligence in data mining", Studies in Computational Intelligence (SCI) 34, 1–20 (2006),

