Survey of Swarm Intelligence Algorithms

Suganya Selvaraj
Department of Financial Information Security,
Kookmin University,
Seoul, South Korea.
+82-2910-5899
suganya@kookmin.ac.kr

Eunmi Choi*
Department of Financial Information Security / Department of Software,
College of Computer Science,
Kookmin University,
Seoul, South Korea.
+82-2910-4495
emchoi@kookmin.ac.kr

ABSTRACT

Swarm Intelligence (SI) is an AI technique that has the collective behavior of a decentralized, self-organized system. SI has more advantages such as scalability, adaptability, collective robustness and individual simplicity and also has the ability to solve complex problems. Besides, SI algorithms also have few issues in time-critical applications, parameter tuning, and stagnation. SI algorithms need to be studied more to overcome these kinds of issues. In this paper, we studied a few popular algorithms in detail to identify important control parameters and randomized distribution. We also studied and summarized the performance comparison of SI algorithms in different applications.

Keywords

Swarm intelligence; Swarm Intelligence algorithm comparison; Swarm Intelligence Applications.

1. INTRODUCTION

Swarm intelligence (SI) is an artificial intelligence (AI) technique that concerns collective behavior in a decentralized, self-organized manner which includes interacting unintelligent simple agents that follow some simple rules [1]. A few popular SI algorithms are Particle Swarm Optimization (PSO), Artificial Bee Colony Optimization (ABC), Ant Colony Optimization (ACO), Firefly Algorithm (FFA) and Cuckoo Search (CS). SI methods have been very successful in the area of optimization. SI algorithms can be used in many real-world problems such as scheduling, optimization, clustering, and routing. SI algorithms have more advantages like scalability, adaptability, collective robustness and individual simplicity. SI is useful to solve complex problems efficiently than traditional methods. Besides, SI has few limitations in time-critical applications, parameter tuning and stagnation [2]. So there is a need to study SI algorithms to understand the existing system and improve. In this study, we analyze a few popular SI algorithms such as PSO, ABC, ACO, FFA, and CS and summarize with critical control parameters and randomized distribution which is used in SI algorithms.

ICSIM '20, January 12–15, 2020, Sydney, NSW, Australia

ACM ISBN 978-1-4503-7690-7/20/01…$15.00
DOI: https://doi.org/10.1145/3378936.3378977

From the literature review, we have a summary of SI algorithms, especially with major control parameters and random parameters. This analysis will be useful for further study in SI algorithms to find tuning control parameters and avoid stagnation. We also have studied and summarized the performance issues according to the application domain by identifying and classifying those SI algorithms in different applications. Through this study, we may explore a new application domain to apply by using the main characteristics of SI algorithms.

2. Related work

SI-based algorithms are a subset of bio-inspired algorithms (BAs), which itself is a subset of nature-inspired (NAs) algorithms as shown in Figure 1. All NAs generally stochastic and developed by taking inspiration from natural elements such as animals, insects, and plants. NAs includes subclasses like BAs and physics or chemistry based algorithms. The most popular subset of NAs is BAs. Chemistry or physics-based applications include the algorithms like Harmony Search (HS) algorithm, Simulated Annealing (SA), and Black Hole (BH) [3].

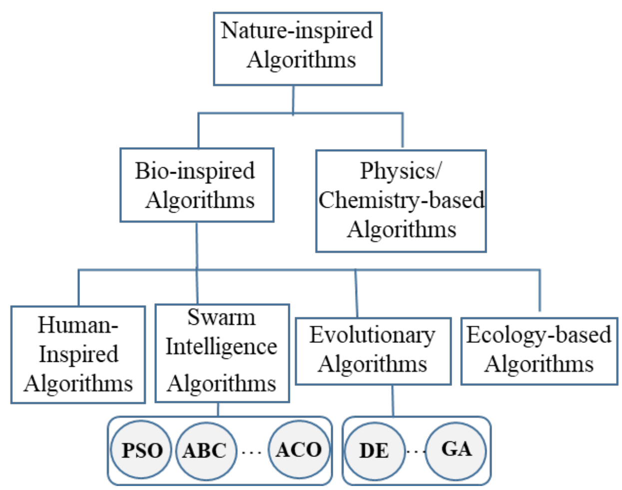

Figure 1. Hierarchical structure of SI Algorithms

Further, BAs are categorized into SI algorithms, Evolutionary Algorithms (EAs), Human-Inspired algorithms and Ecology-based algorithms. EAs includes algorithms like Differential Evolution (DE) and Genetic Algorithm (GA). SI is the collective intelligence of groups of simple agents such as insects, fishes, birds, bacteria, worms, and other animals based on their behavior in real life [4]. SI includes algorithms like PSO, ABC, ACO, BA, CSS, FFA and so on. In this study, our focus is SI algorithms.

3. Analysis of SI Algorithms

In this section, we summarized and analyzed a few popular SI algorithms such as PSO, ABC, ACO, FFA, and CS (Table 1).

We summarized control parameters and randomized distributions that are used in each algorithm. Because identifying the important control parameters and setting the proper value for these parameters are very important for the algorithms to make quality solutions. Furthermore, randomization is more an efficient component for global and local searching. Analyzing these parameters also mandatory to add the modification in algorithms [1]. In general, algorithms are using Lèvy distributions for global searching and uniform distributions (UD) and normal distributions (ND) for the local searching.

3.1 PSO

The PSO algorithm is an optimization algorithm. PSO is developed based on bird and fish flock movement behavior. In PSO algorithm, velocity and position of each particle according to equations 1 and 2 from table 1. $\mathrm{v}_{\mathrm{id}}^{\mathrm{k}}$ and $\mathrm{x}_{\mathrm{id}}^{\mathrm{k}}$ stands for the speed and position of the particle i at iteration k. $\mathrm{pbest}_{\mathrm{id}}^{\mathrm{k}}$ is the personal best position of i. $\mathrm{gbest}_{\mathrm{d}}^{\mathrm{k}}$ is the global best position among all individuals in d-dimension [5, 6].

The parameters ω, c_1, c_2, r_1, r_2 are referring inertia weight, two positive constants and two random parameters within [0,1] respectively. In PSO, ω , c_1, and c_2 are the main control parameters. The inertia (ω) used to balance the global search and local search abilities. c_1 and c_2 are used to control the speed of the particle

Table 1. Summary of control parameters and random distributions in SI algorithms.

Algorithm	Equation	Control parameters	Global Search	Local Search	Ref.
PSO	1. $\mathrm{v}_{id}^{\mathrm{k}+1} = \omega * \mathrm{v}_{id}^{\mathrm{k}} + c_1 r_1^{\mathrm{k}}(pbest_{id}^{\mathrm{k}} - x_{id}^{\mathrm{k}}) + c_2 r_2^{\mathrm{k}}(gbest_{d}^{\mathrm{k}} - x_{id}^{\mathrm{k}})$ 2. $x_{id}^{\mathrm{k}+1} = x_{id}^{\mathrm{k}} + \mathrm{v}_{id}^{\mathrm{k}+1}$	ω, c1, and c2		▪ r_1 and r_2 are drown from UD [0,1].	[5] [6]
ABC	***Scout bees:*** 3. $\vec{\mathrm{x}}_{\mathrm{new}}(v,\mathrm{i}) = \vec{\mathrm{x}}_{\mathrm{old}}(v,\mathrm{i}) + RW(\tau, \vec{\mathrm{x}}_{\mathrm{old}}(v,\mathrm{i}))$ ***Experienced Bees:*** 4. $\vec{\mathrm{x}}_{\mathrm{new}}(\xi,\mathrm{i}) = \vec{\mathrm{x}}_{\mathrm{old}}(\xi,\mathrm{i}) + w_b r_b \left(\vec{b}_{\mathrm{old}}(\xi,\mathrm{i}) - \vec{\mathrm{x}}_{\mathrm{old}}(\xi,\mathrm{i})\right) + w_e r_e\left(\vec{e}(\xi,\mathrm{i}) - \vec{\mathrm{x}}_{\mathrm{old}}(\xi,\mathrm{i})\right)$ ***Onlooker Bees:*** 5. $\vec{\mathrm{x}}_{\mathrm{new}}(k,\mathrm{i}) = \vec{\mathrm{x}}_{\mathrm{old}}(k,\mathrm{i}) + w_e r_e\left(\vec{e}(\xi,\mathrm{i}) - \vec{\mathrm{x}}_{\mathrm{old}}(k,\mathrm{i})\right)$	Parameter that controls the ratio of scout, onlooker and experienced bees.		***Scout Bee:*** ▪ RW– random walk (ND). ***Experienced Bee:*** ▪ r_b *and* r_e are random variable (UD [0,1]). ***Onlooker Bee:*** ▪ w_e and r_e are controlling the attraction of the onlooker bee toward its interesting food source area.	[1]
ACO	6. $P_{ij}^{k} = \frac{\tau_{ij}^{\alpha}.\eta_{ij}^{\beta}}{\sum_{c_{ij} \in N(s)} \tau_{ij}^{\alpha}.\eta_{ij}^{\beta}}$, $\forall c_i \in N(s)$	α and β		Probability for an ant to move from position i to position j depends on a random variable (UD [0, 1]).	[8]
FFA	7. $x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2}(x_j - x_i) + \alpha(rand - 0.5)$	γ, β_0 and α		▪ $rand$ is a Randomly drawn from UD [0,1]. ▪ Modeling the environment randomly (UD and ND (0,1))	[1]
CS	8. $X_i^{(t+1)} = X_i^t + \alpha \oplus L\grave{e}vy(\lambda)$ 9. $X_i^{new} = X_i^{old} + 2 * step * (X_i^{old} - best)$ 10. $X_i^{new} = X_i^{old} + rand1 * (rand2 > p_a) * (X_a - X_b)$	p_a	$L\grave{e}vy(\lambda)$ - Random walk	▪ *Step* is for local random walk. ▪ Selecting agents for local search. ▪ rand1; rand2 - for finding nests to be abandoned. (UD [0,1])	[1]

3.2 ABC

The ABC algorithm is an optimization algorithm based on the intelligent foraging behavior of honey bee swarm. ABC algorithm includes three types of bees such as scout (equation 3), experienced foragers (equation 4), and onlooker bees (equation 5) and which are called worst fitness, mild fitness, and best fitness respectively.

Scout bees use the random flying pattern to discover new food sources and replace the abandoned one. In scout bees (equation 3), x_{old} and x_{new} represents the abandoned and new solution respectively. RW is a random walk using ND and v represents the search radius.

Experience bees remember the best food source positions and their quality from the previous experience. In experienced bees equation, $b(\xi, i)$ represents the best food sources. Onlooker bees used social knowledge provided by the experienced bees and to adjust the moving trajectory for the next time. In equations 3 and 4, $e(\xi, i)$ represents the bee with the best fitness. r_b and r_e are random variables drawn from UD [0,1] which models the stochastic nature of flying patterns.

In ABC, the parameter that controls the ratio of the scout, onlooker and experienced bees is very effective to control the trade-off between the local search and global search. The number of experienced bees should be lower at the beginning of optimization to allow many scout explorative bees, while at the end of optimization the experienced bees should allow more for exploitation and intensive searching [1, 7].

3.3 ACO

ACO used for searching the optimal path in graph-based problems by adopting ant's food searching behavior. Equation 6 shows the probability of ant's movement from node i to node j. τ_{ij} is the amount of pheromone on edge i, j. α is a parameter to control the influence of τ_{ij}. η_{ij} is the desirability of edge i, j. Parameter β is used to control the influence of η_{ij}. Random variable with UD [0, 1] used to calculate the probability for an ant to move from position i to position j [8].

3.4 FFA

FFA algorithm developed by imitates the mechanism of firefly mating and the exchange of information using light flashes. The movement of firefly i towards the brighter firefly j is shown in equation 7. β_0 is always set to 1 and$\alpha \in$ [0,1]. The parameter γ denotes attractiveness, and its value is crucially important in determining the speed of the convergence and FFA algorithm's behavior.

Parameter α is used to model noise in the environment around the firefly swarm. The higher value of α provides slow convergence and increases the chance of global search. The low value of this parameter provides high convergence speed and gives chance for local search. This parameter should start with high and ends with low for the optimum solution.

Parameter β_0 controls the brighter firefly's attraction. The higher value of this parameter provides a chance for global searching. On the other hand, the lower value provides the local searching capability.

3.5 CS

CS algorithm is imitating the cuckoo bird's breeding behavior and used to solve optimization problems. The main aim of this algorithm is to use new and potentially better solutions to replace bad solutions. Equation 8 is used to find the new solution, equation 9 is used for the local search, and equation 10 is used to abandon the bad solution by generating a new solution.

X_i^{new} is the new nest to be found, X_i^{old} is the new nest to be abandoned, and X_a, X_b are two randomly selected existing nests. Lèvy(λ) and step are a random walk and random number respectively drawn by Lèvy distribution. α is a step size. Best is a current best solution. Parameter rand1 and rand2 are two random numbers drawn from UD [0,1].

The parameter p_a is a discovery rate parameter and it controls the rate of abandoning a given nest. Higher values for this parameter provide the global searching capability but slow down the convergence speed. Lower values for this parameter provide the local searching capability but the threat of falling in local optima is high.

4. Application Domains and Performances of SI Algorithms

In this section, we studied the comparative results of SI algorithms in a few applications. We summarized the results in Table 2 and categorized them into the below applications.

4.1 Travelling Salesman Problem (TSP)

The TSP is the problem faced by a salesman who, starting from a particular town, has the assignment of finding the shortest possible round trip through a given set of customer towns or cities. The salesman has the mandate to visit each city once before finally returning to the starting town/city. The TSP can be represented by a complete weighted graph $G = (V, E)$ with V being the set of nodes (cities) and E being the set of edges fully connecting the nodes in the graph G [4, 9].

In table 2, TSP's implementation with SI algorithms (GA, PSO, ACO and ABC) is compared and the study states that ABC achieved better performance than others. ACO is recommended for less than 80 cities [10].

4.2 Feature Selections (FS)

To learn from data, the dimensionality of the data should be reduced first. FS focuses on the removal of irrelevant and redundant features from the original feature set to reduce the amount of data [11]. As shown in Table 2, two studies suggest that feature selection field needs to be studied more. Because experimental results of FS in SI algorithms obtained on different real-world problems are harder to compare and they almost have the same performance [12].

4.3 Robot Swarm Learning

Table 2 shows the robot swarm learning study which compares SI algorithms include bat algorithm (BA), PSO, grey wolf optimizer (GWO) according to their learning strategies. This comparative study shows that PSO outperforms BA and BA outperforms GWO in general. GWO performs better than PSO and BA under a large number of robots (NR) and long communication range (CR). Increasing NR and CR can significantly improve the performance of GWO [13].

Table 2. Application domain applied with SI algorithms and performances

Application	Algorithms Compared	Remark	Best Performing Algorithm	Ref.
TSP	GA, PSO, ACO and ABC	ABC achieved better performance than others. ACO is recommended for less than 80 cities.	ABC	[10]
FS	Most popular SI algorithms	Experimental results obtained on different real-world problems are harder to compare.	*Need to be studied more.*	[11]
	PSO,ACO, AFSA, ABC, FFA, and BA	More studies are highly needed in the area of feature selections/reductions using meta-heuristic algorithms in the future.	*Need to be studied more.*	[12]
Robot Swarm Learning	BA, PSO, and GWO	SI algorithm's learning strategies are compared. Results: PSO outperforms BA and BA outperforms GWO in general. GWO performs better than PSO and BA under a large number of robots (NR) and long communication range (CR). Increasing NR and CR can significantly improve the performance of GWO.	GWO performs better under large NR and CR	[13]
Clustering	PSO, FFA, CS, and BA	Medical Data analysis: CS clustering is the slowest one. Firefly clustering is slow when the number of agents is large. PSO and BA are relatively faster than the other two approaches. The execution time of these four existing approaches is still not acceptable.	PSO and BA are better than other. *Need to be studied more*.	[15]
	Most popular SI algorithms	Data mining: Use of the PSO algorithm, in a standalone setting or including variations and improvements. It is the primary clustering technique in 66.4% of the selected papers. Other better performing algorithms are Ant Colony Optimization(ACO), Artificial Bee Colony (ABC) and Artificial Fish Swarm Algorithm(AFSA).	Best Performing Algorithm: PSO Second level performers: ACO, ABC, and AFSA	[14]
Scheduling	ABC, PSO, and ACO	Dynamic task scheduling in cloud computing: ABC algorithm is the superior and outperforms other algorithms. The PSO and ACO can be put in the second and third level respectively.	ABC	[16]
	PSO and ACO	Cloud Scheduling: PSO algorithm in cloud scheduling is a better option when compared to ACO.	PSO	[17]

4.4 Clustering

The data clustering process is a core of data mining on real problems. The process aims to divide the data into groups according to their similarities and dissimilarities [14]. There are more algorithms are available for clustering. In here, we study about clustering implemented using SI algorithms.

SI algorithms such as PSO, FFA, CS, and BA are used and compared for clustering in medical data mining [15]. The results of this study show that CS clustering is the slowest one. Firefly clustering is slow when the number of agents is large. PSO and Bat are relatively faster than the other two approaches. The execution time of these four existing approaches is still not acceptable.

Figueiredo et al [14] also compare the most popular swarm intelligence algorithms for clustering in data mining. This study shows that PSO algorithm is the primary clustering technique in 66.4% of the selected papers and performs better than others. The second level performing algorithms are Ant Colony Optimization(ACO), Artificial Bee Colony (ABC) and Artificial Fish Swarm Algorithm(AFSA).

4.5 Scheduling

Cloud dynamic task scheduling is considered as an NP-hard problem, and many meta-heuristic problems are suitable to solve this problem [16]. Here, we summarized a few comparative studies in task scheduling using SI algorithms.

GF Elhady et al [16], comparatively studied the SI algorithms (ABC, PSO, and ACO) in dynamic cloud task scheduling. Their study shows ABC algorithm is superior and outperforms other algorithms. The PSO and ACO can be put in the second level and third level respectively.

S.J. Mohana et al [17], performs a comparative analysis of swarm intelligence optimization techniques (PSO and ACO) for cloud scheduling and summarized their observation as PSO algorithm in cloud scheduling is a better option when compared to ACO.

5. Exploring Application Domain with SI Algorithms

Section 3 shows the control parameter and random parameters to find optimal solution of each SI algorithm, as shown in the Table 1. Based on the parameter setting and tuning schemes, the SI algorithms have the benefit to apply to the appropriate domain of problem solving. In this paper, we provide five classified domains according to the characteristics of SI algorithms. That is to find the optimal route of coverage, feature identification and selection, role-based learning, clustering with aspect of data mining, job scheduling of different tasks, etc. Beyond the groups of classification, we may extend to another domain of distributed intelligent systems as a future work.

6. Summary

To summarize, SI algorithms have more advantages such as scalability, adaptability, collective robustness and individual simplicity and have the ability to solve complex problems such as shortest pathfinding, feature selection, job scheduling, and clustering. Besides, SI algorithms also have few problems in time-critical applications, parameter tuning, and stagnation. To overcome these issues SI algorithms, need to be studied more. In this study, we analyzed a few popular SI algorithms such as PSO, ABC, ACO, FFA, and CS and summarized the identified critical control parameters and randomized distribution which is used in algorithms. This analysis will be useful for further study in SI algorithms to find tuning control parameters and avoid stagnation. We also have studied and summarized the performance issues according to the application domain by identifying and classifying those SI algorithms in different applications. Through this study, we may explore a new application domain to apply by using the main characteristics of SI algorithms.

7. ACKNOWLEDGMENTS

This work was supported by “the Ministry of Education BK21 program (2011-0011507)”, in Korea.

8. REFERENCES

[1] Hassanien A.E., Emary E., “Swarm Intelligence: Principles, Advances, and Applications”, *CRC Press, Taylor & Francis Group*, 2015.

[2] H. Ahmed, J. Glasgow, *Swarm intelligence: concepts, models and applications.* Technical Report. Queen's University, Canada, School of Computing (2012).

[3] S. Binitha and S. S. Sathya, “A survey of bio inspired optimization algorithms,” *International Journal of Soft Computing and Engineering*, vol. 2, pp. 137–151, 2012.

[4] Odili JB, Kahar MNM (2016) Solving the traveling salesman’s problem using the african buffalo optimization. *Comput Intell Neurosci* 2016:3.

[5] Q. Bai, "Analysis of Particle Swarm Optimization Algorithm," *Computer and Information Science*, vol. volume 3 No1, Pebruari 2010.

[6] G. Xu, “An adaptive parameter tuning of particle swarm optimization algorithm,” *Applied Mathematics and Computation*, vol. 219, no. 9, pp. 4560–4569, 2013.

[7] C. Worasucheep. “A Hybrid Artificial Bee Colony with Differential Evolution.” *Int. J. Mach. Learn. Comput.*, 5(3), 179-186, 2015.

[8] Dorigo, Marco & Birattari, Mauro & Stützle, Thomas. (2006). Ant Colony Optimization. *Computational Intelligence Magazine, IEEE*. 1. 28-39. 10.1109/MCI.2006.329691.

[9] G. Dong et al., “Solving Traveling Salesman Problems with Ant Colony Optimization Algorithms in Sequential and Parallel Computing Environments: A Normalized Comparison.” *Int. J. Mach. Learn. Comput.*, 8(2), 98-103, 2018.

[10] Shima Sabet, Mohammad Shokouhifar, and Fardad Farokhi; "A Comparison Between Swarm Intelligence Algorithms For Routing Problems"; *Electrical & Computer Engineering: An International Journal (ECIJ)* Volume 5, Number 1, March 2016.

[11] Brezočnik, Lucija, et al., "Swarm Intelligence Algorithms for Feature Selection: A Review." *Applied Sciences* 8.9 (2018): 1521.

[12] Basir, M.A.; Ahmad, F. Comparison on Swarm Algorithms for Feature Selections Reductions*. Int. J. Sci. Eng. Res*. 2014, 5, 479–486.

[13] Fan, J.; Hu, M.; Chu, X.; Yang, D. A comparison analysis of swarm intelligence algorithms for robot swarm learning. In *Proceedings of the 2017 Winter Simulation Conference (WSC)*, Las Vegas, NV, USA, 3–6 December 2017; pp. 1–6.

[14] Figueiredo et al., “Swarm intelligence for clustering—A systematic review with new perspectives on data mining” *Engineering Applications of Artificial Intelligence*, 82 (2019), pp. 313-329.

[15] X. Gong, L. Liu, S. Fong, Q. Xu, T. Wen and Z. Liu, "Comparative Research of Swam Intelligence Clustering Algorithms for Analyzing Medical Data," in *IEEE Access*, vol. 7, pp. 137560-137569, 2019. doi:10.1109/ACCESS.2018.2881020

[16] GF Elhady, M A. Tawfeek, "A comparative study into swarm intelligence algorithms for dynamic tasks scheduling in cloud computing", *Intelligent Computing and Information Systems (ICICIS) 2015 IEEE Seventh International Conference* on, pp. 362-369, 2015.

[17] S.J. Mohana, M. Dr, Dr Saroja, M. Venkatachalam Comparative analysis of swarm intelligence optimization techniques for cloud scheduling, *IJISET*, 1 (10) (2014), pp. 15-19.