A Review on Hybridization of Particle Swarm Optimization with Artificial Bee Colony

Bin Xin[1,2,3(✉)], Yipeng Wang[1], Lu Chen[1], Tao Cai[1,2], and Wenjie Chen[1,2]

[1] School of Automation, Beijing Institute of Technology,
Beijing 100081, China
brucebin@bit.edu.cn
[2] Key Laboratory of Intelligent Control and Decision of Complex Systems,
Beijing Institute of Technology,
Beijing 100081, China
[3] Beijing Advanced Innovation Center for Intelligent Robots and Systems,
Beijing 100081, China

Abstract. Particle swarm optimization (PSO) and artificial bee colony (ABC) are two formidable population-based optimizers inspired by swarm intelligence(SI). They follow different philosophies and paradigms, and both are successfully and widely applied in scientific and engineering research. The hybridization of PSO and ABC represents a promising way to create more powerful SI-based hybrid optimizers, especially for specific problem solving. In the past decade, numerous hybrids of ABC and PSO have emerged with diverse design ideas from many researchers. This paper is aimed at reviewing the existing hybrids based on PSO and ABC and giving a classification and an analysis of them.

Keywords: Particle swarm optimization (PSO) · Artificial bee colony (ABC) · Hybridization · Review · Swarm intelligence

1 Introduction

The concept of swarm intelligence (SI) derives from the observation of swarms of insects in nature. The swarm refers to a group of individuals that can communicate with each other directly or indirectly, and such swarm can solve complicated problems using decentralized control and self-organization of relatively simple individuals [1,2]. Optimization techniques inspired by SI (i.e. swarm intelligence optimization algorithms, SIOAs), which inherits the features of SI, are population-based stochastic methods mainly for solving combinatorial optimization problems. Besides, SIOAs have become increasingly popular during the last

decade, and plenty of optimizers have been proposed. Particle swarm optimization (PSO), ant colony optimization (ACO), and artificial bee colony (ABC) are among the most representative SIOAs.

In the past decade, hybrid optimizers have attracted persistent attention from scholars that are interested in design of optimizers and their applications [3–8]. As Raidl claimed in his unified view of hybrid meta-heuristics [8], it seems that choosing an adequate hybrid approach is determinant to achieve top performance in solving most difficult problems. A common template for hybridization is provided by memetic algorithms (MAs), which combine the respective advantages of global search and local search (LS) [9]. Due to excellent performance, MAs have been favored by many scholars in their research on different optimization problems [9]. However, MAs only represent a special class in the family of hybrid optimizers. There are manifold possibilities of hybridizing different optimizers, which follow diverse philosophies and paradigms. This paper is aimed at giving a classification and an analysis of various hybrid optimizers based on PSO and ABC by the systematic taxonomy we proposed in a recent work [10].

The rest of this paper is structured as follows. In Sect. 2, we present the simple introduction of PSO and ABC. A systematic taxonomy of hybridization strategies is described briefly in Sect. 3. Section 4 presents different hybrids based on ABC and PSO. Finally, the conclusion is drawn in Sect. 5.

2   Particle Swarm Optimization and Artificial Bee Colony

2.1   Particle Swarm Optimization

The particle swarm optimization algorithm is proposed by Kennedy and Eberhart in 1995, where a swarm stands for the population and the swarm consists of a certain amount of individuals called particles. Originally, PSO was inspired by the social and cognitive behavior of animal groups, such as bird flocks, fish schools and so on. During the past decade, PSO has been successfully and widely applied in the practice of science and engineering, which demonstrates the superiority of this algorithm. In the PSO model, each particle has its own current position $\mathbf{X}_i$ (which represents a solution), current velocity $\mathbf{V}_i$ and the precious best position $\mathbf{pbest}_i$. Particles accumulate their own experiences about the problem space, and learn from each other according to their fitness values as well.

Suppose that the search space of the problem is D-dimensional, then the iteration equations for the velocity and position in standard PSO are given as follows:

$$v_{i,d}^{k+1} = \omega \cdot v_{i,d}^k + c_1 \cdot \text{rand1} \cdot (pbest_{i,d}^k - x_{i,d}^k) + c_2 \cdot \text{rand2} \cdot (gbest_d^k - x_{i,d}^k), \quad (1)$$
$$x_{i,d}^{k+1} = x_{i,d}^k + v_{i,d}^{k+1}, \quad (2)$$

where $\mathbf{X}_i^k = [x_{i,1}^k, x_{i,2}^k, \ldots, x_{i,D}^k]$ and $\mathbf{V}_i^k = [v_{i,1}^k, v_{i,2}^k, \ldots, v_{i,D}^k]$ represent the position and velocity of the ith particle at the kth generation ($i = 1, 2, ..., PS$; PS

is the population size), respectively; $\mathbf{pbest}_i^k = [pbest_{i,1}^k, pbest_{i,2}^k, \ldots, pbest_{i,D}^k]^T$ is the best position that is found so far by the ith particle; $\mathbf{gbest}^k = [gbest_1^k, gbest_2^k, \ldots, gbest_D^k]^T$ is the global best position that is found by particles in the swarm; and $rand1$ and $rand2$ are the random numbers that are uniformly distributed in $[0, 1]$; ω is the so-called inertia weight; c_1 and c_2 are acceleration coefficients, which are also termed as the cognitive factor and the social factor, respectively. The velocity of the particles on each dimension is clamped to the range $[-V_{max}, V_{max}]$.

2.2   Artificial Bee Colony

The artificial bee colony algorithm simulating the foraging behavior of honey bees was developed by Karaboga to solve numerical optimization problems in 2005. The bee colony is a complicated natural society with specialized social divisions, and the ABC just assumes a simplified model composed by three groups of bees: *employed bees*, *onlooker bees* and *scout bees*. Each employed bee is assigned to a food source, and each onlooker bee waits in the hive and chooses a food source depending on the information shared by the employed bees. Besides, the scout bees will search for new food sources surrounding the hive. The food sources are the solutions of the optimization problem and the bees are the variation operators. The exchange of information among bees is the most important occurrence in the formation of the collective knowledge.

Suppose that the solution space of the problem is D-dimensional, then ABC will start with producing food sources randomly, and each food source stands for a candidate solution $\mathbf{X}_i = [x_{i1}, x_{i2}, \ldots, x_{iD}], i \in \{1, 2, \ldots, N_s\}$. N_s is equal to the number of food sources and half the population size.

Employed Bee Phase. Each employed bee generate a new food source v_i by a modification on the position of the old food source, which is shown as follows:

$$v_{i,d} = x_{i,d} + \psi \cdot (x_{i,d} - x_{k,d}), \tag{3}$$

where $k \in \{1, 2, \ldots, N_s, k \neq i$, x_k is a food source randomly selected in the neighbor of the ith food source; and ψ is is a random number in the range $[-1, 1]$.

Onlooker Bee Phase. An artificial onlooker bee determines which food source to forage according to the probability value P_i of those food sources shared by the employed bees. The probability value can be calculated by the following equation:

$$P_i = \frac{f_i}{\sum_{k=1}^{N_s} f_k}, \tag{4}$$

where f_i is the fitness value of the ith food source. After the selection, the onlooker bees also generate new food sources as described in Eq. 3.

Scout Bee Phase. If a food source position cannot be further improved within a finite steps, it will be abandoned and the corresponding employed bee will become a scout bee. Then the scout bee will search randomly for new food source position which is obtained as follows:

$$x_{i,d} = x_d^{min} + \text{rand} \cdot (x_d^{max} - x_d^{min}), \tag{5}$$

where $rand$ is random number selected in the range [0.1]; x_d^{max} and x_d^{min} are the upper and lower borders of the dth dimension of the solution space.

3   Taxonomy on Hybridization Strategies

A favorable taxonomy of hybridization strategies should be capable of differentiating different strategies as well as providing designers a convenient and efficient means of determining a hybridization scheme. In this section, we will introduce the elements of a hybridization strategy (i.e., hybridization factors) and the taxonomy, which we proposed in recent work [10].

Different hybrids can be differentiated by the relationship between parent optimizers (PR), hybridization level (HL), operation order (OO), type of information transfer (TIT), and type of transferred information (TTI). And we will introduce some concise notations to express the candidate classes with respect to each factor as follows.

(1) Parent relationship (PR)
   Collaboration: $\langle C \rangle$, Embedding: $\langle E \rangle$, Assistance: $\langle A \rangle$.
(2) hybridization level (HL)
   Population level: $\langle P \rangle$, Subpopulation level: $\langle S \rangle$, Individual level: $\langle I \rangle$, Component level: $\langle C \rangle$.
(3) Operating order (OO)
   Sequential order: $\langle S \rangle$, Parallel order: $\langle P \rangle$, No order: $\langle N \rangle$.
(4) Type of information transfer (TIT)
   Simplex TIT: $\langle S \rangle$, Duplex TIT: $\langle D \rangle$.
(5) Type of transferred information (TTI).
(6) Solutions: $\langle S \rangle$, Fitness information: $\langle F \rangle$, Solution components: $\langle S_c \rangle$, Auxiliaries: $\langle A \rangle$, Control parameters: $\langle C_p \rangle$, Algorithm-induced in-betweens: $\langle A_i \rangle$.

4   Previous PSOABCs

In this section, we will introduce some typical hybrids based on PSO and ABC in the literature, and give a classification and a simple analysis according to the taxonomy aforementioned. Since the **t**radeoff between **e**xplo**r**ation and **e**xplo**i**tation (Tr:Er&Ei) is a core of all kinds of optimizers [11], we primarily divide the hybrids into two parts according to the combination patterns that involve global search (GS) and local search (LS). Besides, it is imperative to implement certain type of global search for solving complex problems, and SIOAs can be used for

both GS and LS in solution space. Therefore, the hybrids considered here include two main patterns: "GS⊕GS" and "GS⊕LS".

In order to build a unified nomenclature and differentiate various hybrids with the same parents conveniently, we will name a given hybrid by combining the initial of the last name of each inventor of the hybrid into a suffix that follows the parent algorithms.

4.1   GS⊕GS

Although the tradeoff between exploration and exploitation has been considered in SIOAs, the stagnation phenomenon caused by the weakness of a sole algorithm still could not be prevented. Therefore, the methods with GS ability can be combined with any SIOA to design a new optimizer. A rational motivation behind this hybridization is that the two or more optimizers correspond to different landscapes, which may give birth to some shortcuts to escape from the local optima [12], even though both of them are implemented as GS.

El-Abd proposed a hybrid optimizer combining PSO and ABC in order to gain benefits from their respective strengths [13]. This optimizer is denoted by PSOABC-E, and it incorporates an ABC component into a standard PSO which updates the **pbest** information of the particles in every iteration using the ABC update equation. This component is added to the standard PSO after the main loop. For every particle i in the swarm, the ABC update equation is applied to its personal best $\mathbf{pbest}_i = \{pbest_{i,1}, pbest_{i,2}, \ldots, pbest_{i,D}\}$, which is given as follows:

$$pbest_{ij} = pbest_{ij} + \phi_{ij} \times (pbest_{ij} - pbest_{kj}), \tag{6}$$

where j is a randomly selected number in $[1, D]$ and D is the number of dimensions, and ϕ_{ij} is a random number uniformly distributed in the range $[-1, 1]$, and k is the index of a randomly chosen solution. The new $\mathbf{pbest}_i$ replaces the previous one if it has a better fitness. According to our taxonomy, the type of the hybridization strategy in PSOABC-E is $\langle C, C, S, D \rangle$.

Besides, Sharma *et al.* proposed another PSOABC hybrid [14], and it is termed as PSOABC-SPB. In this PSOABC-SPB, a modified method derived from the velocity update equation of PSO is presented for solution update of the employed as well as onlooker bees, respectively. According to our taxonomy, the type of the hybridization strategy in PSOABC-SPB is $\langle C, P, S, D \rangle$.

Moreover, Shi *et al.* developed a PSOABC hybrid [15], and it is named PSOABC-SLLGWL. The approach is initialized by two sub-systems of PSO and ABC, and then they are executed in parallel. During the two sub-systems are executing, two information exchanging processes are introduced into the system. These processes are called by Information Exchanging Process1 and Information Exchanging Process2, respectively. The first one forwards "better information" from particle swarm to bee colony, and the second is reversed. According to our taxonomy, the type of the hybridization strategy in PSOABC-SLLGWL is $\langle C, P, P, D \rangle$. The hybridization strategies of the hybrids based on 'GS⊕GS' are listed in Table 1.

In addition, there are also various other versions based on PSO and ABC in the literature, and hybridization strategies are similar with those aforementioned. As space is limited, these optimizers will not be covered here.

Table 1. The hybridization strategies of the hybrids based on 'GS$\oplus GS$'

Name	$Parent_A$	$Parent_B$	PR	OL_A	OL_B	HL	OO	TIT	TTI
PSOABC-E	PSO	ABC	C	P	C	C	S	D	S, F, S_c
PSOABC-SPB	PSO	ABC	C	P	P	P	S	D	S, F
PSOABC-SLLGWL	PSO	ABC	C	P	P	P	P	D	S, F
PSOABC-AK [16]	PSO	ABC	C	P	P	P	S	D	S, F
PSOABC-MRR [17]	PSO	ABC	C	I	P	I	S	D	S, F
PSOABC-BM [18]	PSO	ABC	C	P	P	P	S	D	S, F
PSOABC-AKS [19]	PSO	ABC	C	S	S	S	P	D	S, F
PSOABC-VRB [20]	PSO	ABC	C	P	P	P	S	D	S, F

4.2 GS$\oplus$LS

Since the ABC can also be implemented as local search, some PSOABC hybrids are designed follow the concept of MA as well. Li *et al.* proposed a hybrid algorithm denoted by PSOABC-LWYL, which combined the local search phase in PSO with two global search phases in ABC for the global optimum [21]. In the iteration process, the algorithm examines the aging degree of **pbest** for each individual to decide which type of search phase (PSO phase, onlooker bee phase, and modified scout bee phase) to adopt. According to our taxonomy, the type of the hybridization strategy in PSOABC-LWYL is $\langle C, I, S, D \rangle$.

Similarly, Alqattan and Abdullah proposed a PSOABC hybrid optimizer named PSOABC-AA [22]. In this PSOABC-AA, the employed bees are eliminated in the process of ABC. Instead, the particle movement process of PSO is applied to the local search. According to our taxonomy, the type of the hybridization strategy in PSOABC-AA is $\langle C, P, S, D \rangle$. The hybridization strategies of the hybrids based on 'GS$\oplus LS$' are listed in Table 2.

Table 2. The hybridization strategies of hybrids based on 'GS$\oplus LS$'

Name	Parent A	Parent B	PR	OL A	OL B	HL	OO	TIT	TTI
PSOABC-LWYL	ABC	PSO	C	I	I	I	S	D	S, F
PSOABC-AA	ABC	PSO	C	P	P	P	S	D	S, F
PSOABC-WLS [23]	ABC	PSO	E	P	P	P	N	D	S
PSOABC-BDAA [24]	ABC	PSO	C	P	P	P	P	D	S, F
PSOABC-XPZCLZ [25]	ABC	PSO	C	P	P	P	S	D	S, F

5 Conclusion

As can be seen from the existing hybrids based on PSO and ABC, designers have manifold choices to design a hybrid optimizer, and most of the scholars either update onlooker phase or employed bees phase by single update equation of PSO or with other features. Besides, the "GS⊕GS" pattern seems to be more common in the hybridization of PSO and ABC. Nevertheless, more hybridization strategies should be taken into account for the design of a new PSOABC hybrid.

The tradeoff between exploration and exploitation is a core of all kinds of optimizers. Generally, a hybrid follows the hybridization pattern of the MAs can generate a feasible solution to the considered problem. Thus, more PSOABCs which follows the hybridization pattern of the MAs deserve further research.

Besides, the "GS⊕GS" hybridization pattern also contributes to seeking for a more suitable optimizer for a specific problem. Even though the validity of a hybrid optimizer that follows the "GS⊕GS" hybridization pattern cannot be ensured, a rational motivation behind such hybridization is that the two or more optimizers correspond to different landscapes, which can give birth to a new search paradigm and may suppress the stagnation during the iteration process that arises from the limitation of a sole search method [12].

References

1. Dorigo, M., Birattari, M.: Swarm intelligence. Scholarpedia **2**(9), 1462 (2007)
2. Hinchey, M.G., Sterritt, R., Rouff, C.: Swarms and swarm intelligence. Computer **40**(4), 111–113 (2007)
3. Andrea, R., Blesa, M., Blum, C., Michael, S.: Hybrid Metaheuristics-an Emerging Approach to Optimization. Springer, Heidelberg (2008)
4. Voß, S.: Hybridizing metaheuristics: the road to success in problem solving. In: Proceedings of the 8th EU/MEeting on Metaheuristics in the Service Industry, Stuttgart (2007)
5. Talbi, E.-G.: A taxonomy of hybrid metaheuristics. J. Heuristics **8**(5), 541–564 (2002)
6. Gendreau, M., Potvin, J.-Y.: Metaheuristics in combinatorial optimization. Ann. Oper. Res. **140**(1), 189–213 (2005)
7. Blum, C., Roli, A.: Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput. Surv. (CSUR) **35**(3), 268–308 (2003)
8. Raidl, G.R.: A unified view on hybrid metaheuristics. In: Almeida, F., Blesa Aguilera, M.J., Blum, C., Moreno Vega, J.M., Pérez Pérez, M., Roli, A., Sampels, M. (eds.) HM 2006. LNCS, vol. 4030, pp. 1–12. Springer, Heidelberg (2006). doi:10. 1007/11890584_1
9. Krasnogor, N., Smith, J.: A tutorial for competent memetic algorithms: model, taxonomy, and design issues. IEEE Trans. Evol. Comput. **9**(5), 474–488 (2005)
10. Xin, B., Chen, J., Zhang, J., Fang, H., Peng, Z.-H.: Hybridizing differential evolution and particle swarm optimization to design powerful optimisers: a review and taxonomy. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **42**(5), 744–767 (2012)

11. Chen, J., Xin, B., Peng, Z., Dou, L., Zhang, J.: Optimal contraction theorem for exploration-exploitation tradeoff in search and optimization. IEEE Trans. Syst. Man Cybern. Part A: Syst. Hum. **39**(3), 680–691 (2009)

12. Jones, T.: One operator, one landscape, Santa Fe Institute Technical report, 95-02-025 (1995)

13. El-Abd, M.: A hybrid ABC-SPSO algorithm for continuous function optimization. In: IEEE Symposium on Swarm Intelligence (SIS), pp. 1–6. IEEE (2011)

14. Sharma, T.K., Pant, M., Bhardwaj, T.: PSO ingrained artificial bee colony algorithm for solving continuous optimization problems. In: 2011 IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE) (2011)

15. Shi, X., Li, Y., Li, H., Guan, R., Wang, L., Liang, Y.: An integrated algorithm based on artificial bee colony and particle swarm optimization. In: 2010 Sixth International Conference on Natural Computation (ICNC), vol. 5, pp. 2586–2590. IEEE (2010)

16. Altun, O., Korkmaz, T.: Particle swarm optimization-artificial bee colony chain (PSOABCC): a hybrid meteahuristic algorithm. In: Scientific Cooperations International Workshops on Electrical and Computer Engineering Subfields, pp. 22–23, August 2014

17. Muthiah, A., Rajkumar, A., Rajkumar, R.: Hybridization of artificial bee colony algorithm with particle swarm optimization algorithm for flexible job shop scheduling. In: Proceedings of 2016 International Conference on Energy Efficient Technologies for Sustainability (ICEETS), pp. 896–903. IEEE (2016)

18. Baktash, N., Meybodi, M.: A new hybridized approach of PSO and ABC for optimization. In: Proceedings of the 2011 International Conference on Measurement and Control Engineering, pp. 69–80 (2011)

19. Amudha, P., Karthik, S., Sivakumari, S.: A hybrid swarm intelligence algorithm for intrusion detection using significant features. Sci. World J. **2015**, 15 (2015). doi:10.1155/2015/574589. Article ID 574589

20. Vitorino, L., Ribeiro, S., Bastos-Filho, C.J.: A mechanism based on artificial bee colony to generate diversity in particle swarm optimization. Neurocomputing **148**, 39–45 (2015)

21. Li, Z., Wang, W., Yan, Y., Li, Z.: PS-ABC: a hybrid algorithm based on particle swarm and artificial bee colony for high-dimensional optimization problems. Expert Syst. Appl. **42**(22), 8881–8895 (2015)

22. Alqattan, Z.N., Abdullah, R.: A hybrid artificial bee colony algorithm for numerical function optimization. Int. J. Mod. Phys. C **26**(10), 1550109 (2015)

23. Chun-Feng, W., Kui, L., Pei-Ping, S.: Hybrid artificial bee colony algorithm and particle swarm search for global optimization. Math. Probl. Eng. **2014**, 8 (2014). doi:10.1155/2014/832949. Article ID 832949

24. Bouaziz, S., Dhahri, H., Alimi, A.M., Abraham, A.: Evolving flexible beta basis function neural tree using extended genetic programming & hybrid artificial bee colony. Appl. Soft Comput. **47**, 653–668 (2016)

25. Xiang, Y., Peng, Y., Zhong, Y., Chen, Z., Lu, X., Zhong, X.: A particle swarm inspired multi-elitist artificial bee colony algorithm for real-parameter optimization. Comput. Optim. Appl. **57**(2), 493–516 (2014)