



Evolutionary Factor-Driven Concise Bacterial Foraging Optimization Algorithm for Solving Customer Clustering Problems

Lijing Tan¹, Kuangxuan Qing^{2,3}, Chen Guo^{2(✉)}, and Ben Niu^{2,3}

¹ School of Management, Shenzhen Institute of Information Technology, Shenzhen 518172, China

² College of Management, Shenzhen University, Shenzhen 518060, China
chen.guo@connect.um.edu.mo

³ Greater Bay Area International Institute for Innovation, Shenzhen University, Shenzhen 518060, China

Abstract. To mitigate the disadvantages of the K-means algorithm, an evolutionary factor-driven concise bacterial foraging optimization algorithm is proposed to handle customer data clustering tasks (EFCBFOK). First, to decrease the computing complexity of BFO, a concise BFO with a simplified structure is used. Second, driven by the evolutionary factors, a modified step size strategy is designed. Third, evolutionary factor-driven chemotaxis operation is proposed to make the bacteria select the learning objects from multiple generations of personal historical best and global best; it can expand the search space and enhance the population diversity. To evaluate the performance of the EFCBFOK, EFCBFOK is compared with the other three algorithms on three validity indexes of five customer datasets. Experimental results show that EFCBFOK outperforms the other three clustering algorithms in terms of solution quality, three validity indexes, and computing time.

Keywords: Evolutionary factor · Bacterial foraging optimization · K-means · Customer clustering

1 Introduction

Data clustering is the process of dividing data into multiple clusters or groups based on similarity. Clustering has applications in many fields, including exploratory data analysis, image segmentation, and mathematical programming [1]. Many traditional clustering algorithms have been proposed, and K-means is one of the important algorithms. However, K-means is sensitive to the initialization, which greatly impacts the clustering results; once the quality of initial cluster centers is poor, effective clustering results may not be obtained and easily fall into local optimum [2].

To improve the quality of traditional clustering methods, researchers have started to study the combination of swarm intelligence and clustering algorithms. Swarm intelligence is a computational technique based on the behavioral rules of biological groups. It is inspired by social insects and swarming vertebrates, which have the characteristics of autonomy and robustness and are used to solve distributed problems. Various swarm intelligence algorithms have been proposed, and these algorithms have been used to solve data clustering tasks. For example, cohesive hierarchical clustering is introduced into the brainstorming algorithm [3], particle swarm algorithm (PSO) is efficiently mixed with fuzzy clustering [4], fireworks algorithm is combined with hard clustering technique [5], brainstorming algorithm is combined with K-means for numerical optimization [6], etc.

The bacterial foraging optimization algorithm (BFO), as a new member of the swarm intelligence, mimics the foraging behaviors of bacteria. It has received much attention recently and has been combined with clustering techniques to solve practical problems [7, 8]. To further explore the potential of BFO in solving data clustering problems, in this paper, an evolutionary factor-driven concise bacterial foraging optimization algorithm (EFCBFO) is proposed to solve the customer data clustering problems (EFCBFOK). Due to the original BFO having high computing complexity, a concise BFO (CBFO) [9] with a simplified structure is employed. The main improvements of this paper are as follows. (1) Based on the evolutionary factor [10], a modified step size updating strategy is proposed to make it change with the evolutionary states, which can better balance exploration and exploitation. (2) To guide the bacteria to find global optimum and escape local optimum, an improved chemotaxis operation is designed that integrates the delayed information. Based on this, during each iteration, bacteria can select learning objects from multiple generations of personal historical best and global best individuals, expanding the search space and enhancing the population diversity. (3) Combining the EFCBFO with K-means, EFCBFOK is designed to handle the customer data clustering problems. Comparative experiments verify that EFCBFOK has better performance than its competitors in terms of solution quality, three validity indexes, and computing time.

The remaining parts of this paper are organized as follows. Section 2 briefly introduces the traditional BFO and the K-means algorithm; Sect. 3 presents and discusses the EFCBFOK in detail. Section 4 presents the experiments and analyses. Section 5 concludes the whole paper and provides an outlook for future work.

2 Background

2.1 Bacterial Foraging Optimization Algorithm

The BFO is a stochastic search algorithm proposed by Passino in 2002 that mainly simulates the food searching behaviors of *E. coli* in the human intestine [11]. In this paper, three operations of BFO, chemotaxis, reproduction, and elimination-dispersal, are included [12] and described in detail.

Chemotaxis is the essential operation in the BFO, which includes two actions: swimming and tumbling. In this stage, the bacterial swarm moves to high nutrients places or away from low nutrients through these two actions. The chemotaxis operation is shown as Eq. (1),

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i)\phi(i) \quad (1)$$

where $\theta^i(j, k, l)$ represents the position of bacteria i during the j th chemotaxis, k th reproduction, and l th elimination-dispersal operations. $C(i)$ is the step size taken during the chemotaxis process and $\phi(i)$ represents a unit length of the random direction.

The chemotaxis operation is followed by a reproduction operation. At this stage, the bacteria in poor health conditions are deleted, and bacteria in good health split into two bacteria at their current position.

For each elimination-dispersal operation, a fixed probability is used to determine whether a bacterium performs this operation; if the operation is performed, the current bacterium will die, and then a new bacterium is randomly generated in the solution space.

2.2 K-means Algorithm

The K-means is a common and simple clustering technique [13]. K-means randomly initializes a set of k cluster centers, then proceeds by alternating between two steps: assignment and update [14]. Given a dataset X with n data points, $X = \{x_1, x_2, \dots, x_n\} (i = 1, 2, \dots, n)$. M is the set of cluster centers, $M = \{m_1, m_2, \dots, m_k\} (1 \leq p, q \leq k)$. $S_p^{(t)}$ is the set of data points belonging to p th cluster at the t th generation. The assignment and update steps are presented as follows.

Assignment Step: compute the Euclidean distances between the data points and cluster centers, and each data point x_i is assigned to the cluster with the least square Euclidean distance, which is presented in Eq. (2),

$$S_p^{(t)} = \left\{ x_i : \left\| x_i - m_p^{(t)} \right\|^2 \leq \left\| x_i - m_q^{(t)} \right\|^2, \forall p, 1 \leq p \leq k \right\} \quad (2)$$

where $m_p^{(t)}$ and $m_q^{(t)}$ imply the p th and q th cluster centers at the t th generation, respectively.

Update Step: recalculate the average values of the data points assigned to each cluster,

$$m_p^{(t+1)} = \frac{1}{\left| S_p^{(t)} \right|} \sum_{x_i \in S_p^{(t)}} x_{ip} \quad (3)$$

where $\left| S_p^{(t)} \right|$ is the number of data points belonging to the p th cluster at the t th generation. $m_p^{(t+1)}$ is the p th cluster center at the $(t + 1)$ th generation.

Usually, the objective of K-means algorithm is to minimize the sum of squared errors (SSE), which is presented in Eq. (4),

$$SSE^{(t)} = \sum_{p=1}^k \sum_{x_i \in S_p^{(t)}} D\left(x_i, m_p^{(t)}\right)^2 \quad (4)$$

where $D(\cdot)$ is the Euclidean distance, $SSE^{(t)}$ is the SSE at the t th generation.

3 The Proposed Algorithm

To improve the performance of traditional BFO, this paper proposes an evolutionary factor-driven concise bacterial foraging optimization algorithm (EFCBFO). Then, the EFCBFO is combined with K-means (EFCBFOK) to solve customer data clustering tasks. In the EFCBFOK, based on the evolutionary factors proposed in [10], evolutionary factor-driven step size and evolutionary-driven chemotaxis are designed. The details of EFCBFOK are described as follows.

3.1 Evolutionary Factors

Evolutionary factor (E_f) [10] is the indicator of the discovery of the exploration and exploitation states of the population. During the evolution process, the population distribution characteristics change not only with the number of iterations but also according to the E_f [10]. In [10], the E_f can be predicted by the average distance between each individual. Concretely, at the beginning of the iteration, when the population is more dispersed, the average distance between each individual will be relatively large; this is the exploration stage. When the individuals reach the local or global optimal region, the average distance between each individual will be relatively small; this is the exploitation stage.

Based on this concept, the E_f is calculated as follows. The first step is to calculate the average distance between the i th individual and the other individuals in the population by using the Euclidean distance. The equation is as follows.

$$d_i = \frac{1}{S-1} \sum_{j=1, j \neq i}^S \sqrt{\sum_{d=1}^D (\theta^{id} - \theta^{jd})^2} \quad (5)$$

where d_i is the average distance of the i th individual. S and D are the number of population and dimensions, respectively. θ^{id} and θ^{jd} are the position vectors of i th and j th individual in the d th dimension.

Based on the average distances of all the individuals, three important distances, d_g , d_{min} , and d_{max} , are defined. Specifically, d_g is the average distance of the global best individual. d_{min} and d_{max} are the minimal and maximal average distances in all the average distances, respectively. After getting these distances, the E_f is calculated as,

$$E_f = \frac{d_g - d_{min}}{d_{max} - d_{min}} \quad (6)$$

It can be seen that the E_f is located in the range [0,1]. It will be relatively small when the average distance between bacteria is relatively close and relatively large when the average distance between bacteria is relatively far.

According to the E_f , evolutionary states can be obtained [10, 15]. In [15], four types of evolutionary states are exploration state, exploitation state, convergence state, and jump-out state. These states denoted $\xi(k)$ can be acquired by dividing the E_f into four

equal intervals, which is presented in Eq. (7),

$$\xi(k) = \begin{cases} 1, & 0 \leq E_f < 0.25 \\ 2, & 0.25 \leq E_f < 0.5 \\ 3, & 0.5 \leq E_f < 0.75 \\ 4, & 0.75 \leq E_f \leq 1 \end{cases} \quad (7)$$

When $\xi(k)$ is equal to 1, 2, 3, and 4, it is the convergence, exploitation, exploration, and jumping-out states, respectively.

3.2 Evolutionary Factors-Driven Step Size

In the original BFO, the step size $C(i)$ is the length of each step during the swimming action, which is a constant. However, if $C(i)$ is too small, the bacteria focus on local search/exploitation, and it may take a long time to find the optimal value; if $C(i)$ is too larger, the bacteria focus on global search/exploration, and the optimal value may be missed. Based on these analyses, it can be observed that E_f shares some characteristics with the $C(i)$, i.e., E_f is relatively large in the exploration and jump-out states and relatively small in the convergence state [10]. Therefore, $C(i)$ can be defined based on the E_f , which is presented in the following equation,

$$C(i) = (C_{max} - C_{min})E_f + C_{min} \quad (8)$$

where C_{max} and C_{min} are the maximal and minimal step sizes, respectively. This paper sets C_{max} as 0.1 and C_{min} as 0.01. The step size varies with the E_f , and a larger $C(i)$ will be more favorable for global search in the jump-out and exploration states; the smaller $C(i)$ in the convergence state favors the local search.

3.3 Evolutionary Factors-Driven Chemotaxis Operation

To make better use of the historical information, delayed information of bacterial swarm is used to guide the bacteria to move to the optimal directions. Concretely, two indicators denoted $\varepsilon_i(k)$ and $\varepsilon_g(k)$ are employed. Among them, k is the information delay interval, which implies that the personal historical best and global best of recent k generations should be recorded and used. $\varepsilon_i(k)$ and $\varepsilon_g(k)$ are two uniformly generated integers in the range of $[1, k]$. i and g represent the indexes of personal best and global best, respectively.

Additionally, another two indicators denoted as $s_i(k)$ and $s_g(k)$ are used in this paper. Combining the evolutionary states, the values of $s_i(k)$ and $s_g(k)$ are shown in Table 1. In the convergence state, the bacteria are expected to reach the region near the global optimum, so the value of $s_i(k)$ and $s_g(k)$ is taken as 0. In the exploitation state, as much local information as possible needs to be used, so the value of $s_i(k)$ is taken as $E_f(k)$. In the exploration state, more global information needs to be used, so the value of $s_g(k)$ is taken as $E_f(k)$. In the jump-out state, the bacterial subsets desire to jump out from the region near the local optimum, so the value of $E_f(k)$ needs to be taken at the same time to provide more information for the bacteria to jump out from the local optimum.

Table 1. Values of indicators.

Modes	States	$s_i(k)$	$s_g(k)$
$\xi(k)=1$	Convergence	0	0
$\xi(k)=2$	Exploitation	$E_f(k)$	0
$\xi(k)=3$	Exploration	0	$E_f(k)$
$\xi(k)=4$	Jumping-out	$E_f(k)$	$E_f(k)$

Based on the aforementioned analysis, an improved chemotaxis operation is designed, which is shown as follows,

$$\begin{aligned}
 \theta^i(j+1, k, l) &= \theta^i(j, k, l) + C(i)\phi(i) \\
 &+ s_i(k)C(i)r_1(p_i(k - \varepsilon_i(k)) - \theta^i(j, k, l)) \\
 &+ s_g(k)C(i)r_2(p_g(k - \varepsilon_g(k)) - \theta^i(j, k, l))
 \end{aligned}
 \tag{9}$$

where r_1 and r_2 are the uniformly generated numbers in $[0,1]$. $p_i(k - \varepsilon_i(k))$ And $p_g(k - \varepsilon_g(k))$ are the selected personal historical best and global best individuals, respectively. It can be seen that the designed chemotaxis operation includes four parts. The first and second parts are the same as the original BFO. The third and fourth parts are the self-learning and global learning parts with delayed information. Based on the evolutionary states, the bacteria can learn from different individuals.

3.4 The Framework of EFCBFOK

Combining EFCBFO and K-means, EFCBFOK is designed to handle customer clustering tasks. In EFCBFOK, SSE is the objective function. The framework of EFCBFOK is described as follows (Fig. 1).

- Step 1. Initialize the position of the population and the parameters of the algorithm.
- Step 2. Evaluate the fitness values of the population, and store their personal historical best and global best.
- Step 3. Iteration loop.
 - Step 3.1. Obtain the evolutionary factors according to Eq. (6), and obtain Table 1 according to Eq. (7).
 - Step 3.2. Update the positions of the population by implementing evolutionary factors-driven chemotaxis operation.
 - Step 3.3. If the iteration number is a multiple of reproduction frequency (F_{re}), implement the reproduction operation.
 - Step 3.4 If the iteration number is a multiple of elimination-dispersal frequency (F_{ed}), implement the elimination-dispersal operation.
- Step 4. Repeat step 3 until the conditions are met.

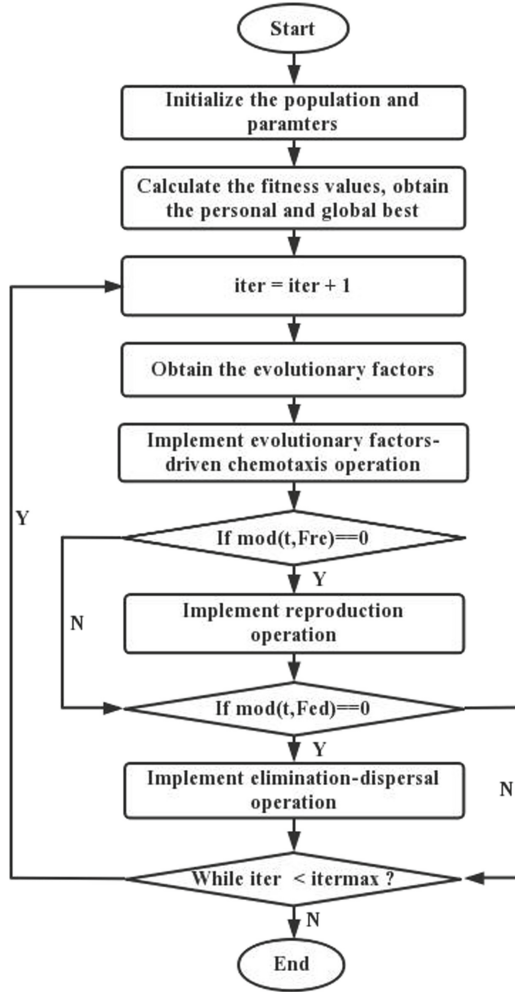


Fig. 1. The framework of EFCBFOK.

4 Experiments and Analyses

4.1 Datasets and Experimental Parameters

To demonstrate the superiority of the EFCBFOK, five datasets, Taiwan, German, Australian¹, Marketing, and Hotel², are selected as the testing datasets. The missing and invalid data are deleted before clustering. The description of the five testing datasets is shown in Table 2.

¹ Data source: <https://archive.ics.uci.edu/ml/datasets.php>.

² Data source: <https://www.kaggle.com/>

Table 2. The description of the testing datasets.

Dataset	Samples	Features	Clusters
German	1000	24	2
Marketing	2216	25	2
Australian	690	14	2
Taiwan	30000	23	2
Hotel	6665	9	4

Additionally, three algorithms are selected as the competitors, which are K-means, PSO-based clustering technique (PSOK) [16], and CBFO-based clustering algorithm (CBFOK) [9]. The parameters of EFCBFOK, PSOK, and CBFOK are listed as follows. The population size is 100; the number of independent runs and iterations are 10 and 100, respectively. For EFCBFOK and CBFOK, the reproduction frequency is 5, and the elimination-dispersal frequency is 2. For PSOK, the C_1 and C_2 are 2. These algorithms are coded using PyCharm Community Edition 2021. To evaluate the clustering quality of all the algorithms, inter-cluster distance, Silhouette [17], and F-measure [18] are selected as validity indexes.

4.2 Experimental Results and Analysis

Table 3 gives the average optimal solutions of three validity indexes over 10 runs. Table 3 also gives the computation times for the four algorithms. This paper uses boldface with underline and boldface to highlight the best and second-best values of the four algorithms on different metrics. Figure 2 shows the SSE convergence curves for all algorithm traversals in the five datasets, respectively. From Table 3 and Fig. 2, three observations can be concluded.

- The EFCBFOK algorithm performs well than its competitors regarding to the three validity indexes on all five datasets, especially on German and marketing datasets. As for the F-measure, EFCBFOK obtains overwhelming advantages over its peers. This implies that the EFCBFOK algorithm effectively improves the clustering quality of customer datasets. Conversely, PSOK has the worse performance among these four algorithms, which only gets several second ranking on some datasets regarding one validity index.
- In terms of computing time, although K-means performs optimally, the EFCBFOK uses less time on the five data sets compared to the swarm intelligence-based clustering algorithms. This implies that the proposed EFCBFOK has a faster convergence speed than that of CBFOK and PSOK.
- From the iterative curves, it can be seen that the iterative curve of EFCBFOK is below the other algorithms. This means that the EFCBFOK outperforms the other three algorithms in terms of global optimality regardless of the dataset.

Table 3. The experimental results of EFCBFOK and its competitors on five datasets.

Datasets	Metrics	EFCBFOK	CBFOK	PSOK	K-means
German	Inter-cluster distance	<u>5.00E-01</u>	2.55E-01	2.53E-01	3.60E-01
	Silhouette	4.76E-02	4.74E-02	4.14E-02	<u>1.18E-01</u>
	F-measure	<u>5.80E-01</u>	5.76E-01	5.80E-01	5.38E-01
	Time(s)	8.30E+01	8.90E+01	8.88E+01	<u>8.22E+01</u>
Marketing	Inter-cluster distance	<u>4.20E-01</u>	3.23E-01	2.78E-01	3.08E-01
	Silhouette	<u>1.62E-01</u>	7.67E-02	7.86E-02	8.18E-02
	F-measure	<u>6.28E-01</u>	6.04E-01	6.03E-01	6.12E-01
	Time(s)	2.26E+02	2.37E+02	2.29E+02	<u>2.23E+02</u>
Australian	Inter-cluster distance	4.01E-01	4.08E-01	3.54E-01	<u>4.96E-01</u>
	Silhouette	<u>1.87E-01</u>	1.38E-01	1.76E-01	1.70E-01
	F-measure	<u>6.65E-01</u>	6.26E-01	6.43E-01	6.49E-01
	Time(s)	3.93E+01	4.02E+01	4.03E+01	<u>3.40E+01</u>
Taiwan	Inter-cluster distance	4.55E-01	<u>5.03E-01</u>	5.00E-01	3.38E-01
	Silhouette	3.34E-01	<u>3.39E-01</u>	2.55E-01	2.67E-01
	F-measure	<u>6.01E-01</u>	5.79E-01	5.61E-01	5.77E-01
	Time(s)	2.27E+02	2.30E+02	2.30E+02	<u>2.06E+02</u>
Hotel	Inter-cluster distance	1.67E+00	1.62E+00	1.69E+00	<u>1.72E+00</u>
	Silhouette	1.64E-01	1.52E-01	1.60E-01	<u>2.33E-01</u>
	F-measure	<u>3.36E-01</u>	3.26E-01	3.35E-01	3.16E-01
	Time(s)	9.71E+01	9.72E+01	9.85E+01	<u>8.46E+01</u>

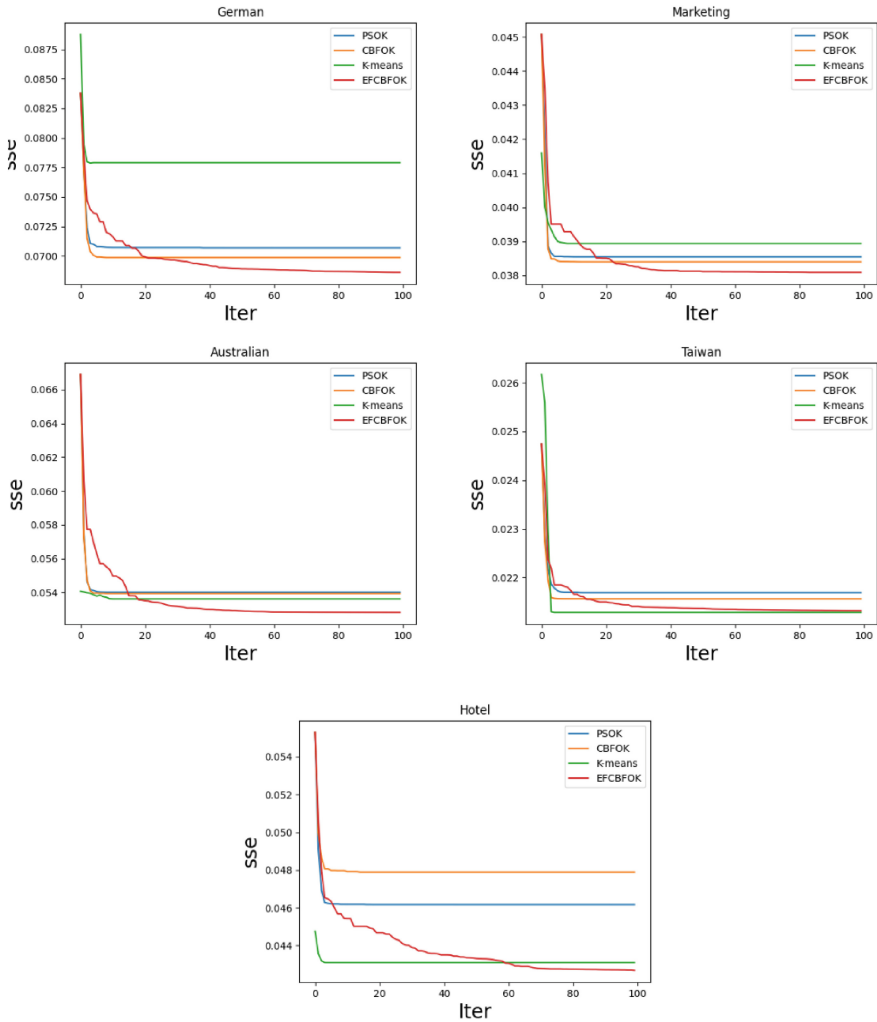


Fig. 2. SSE iterative curves of four algorithms on five datasets

5 Conclusion

This paper proposes a concise evolutionary factor-driven bacterial foraging optimization algorithm to solve the customer clustering problem (EFCBFOK). First, the concise BFO with a simplified structure is used to decrease the computing complexity of BFO. Then, a modified step size strategy is proposed according to the evolutionary factors. Additionally, driven by the evolutionary factor, an improved chemotaxis operation is proposed to let the bacteria select the learning individuals from multiple generations of personal historical best and global best; it can expand the search space and enhance the diversity. To validate the effectiveness of the EFCBFOK, EFCBFOK is compared with

the other three algorithms on three validity indexes of five customer datasets. Experimental results demonstrate that EFCBFOK has better performance than its competitors regarding solution quality, three validity indexes, and computing time.

In future work, EFCBFOK will be used to solve multi-objective data clustering tasks. Furthermore, more strategies should be designed to enhance the performance of BFO.

Acknowledgments. The work described in this paper was supported by The Natural Science Foundation of Guangdong Province (Grant No. 2020A1515010752, 2020A1515010749), Key Research Foundation of Higher Education of Guangdong Provincial Education Bureau (Grant No. 2019KZDXM030), University Innovation Team Project of Guangdong Province (Grant No. 2021WCXTD002).

References

1. Madhulatha, T.S.: An overview on clustering methods. arXiv preprint arXiv (2012)
2. Atabay, H.A., Sheikhzadeh, M.J., Torshizi, M.: A clustering algorithm based on integration of K-Means and PSO. In: 2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC), pp. 59–63. IEEE (2016)
3. Chen, J., Wang, J., Cheng, S., Shi, Y.: Brain storm optimization with agglomerative hierarchical clustering analysis. In: Tan, Y., Shi, Y., Li, L. (eds.) ICSI 2016. LNCS, vol. 9713, pp. 115–122. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-41009-8_12
4. Mehdizadeh, E.: A fuzzy clustering PSO algorithm for supplier base management. *Int. J. Manage. Sci. Eng. Manage.* **4**, 311–320 (2009)
5. Misra, P.R., Si, T.: Image segmentation using clustering with fireworks algorithm. In: 2017 11th International Conference on Intelligent Systems and Control (ISCO), pp. 97–102. IEEE (2017)
6. Zhu, H., Shi, Y.: Brain storm optimization algorithms with k-medians clustering algorithms. In: 2015 Seventh International Conference on Advanced Computational Intelligence (ICACI), pp. 107–110. IEEE (2015)
7. Guo, C., Tang, H., Niu, B.: Evolutionary state-based novel multi-objective periodic bacterial foraging optimization algorithm for data clustering. *Expert. Syst.* **39**, e12812 (2022)
8. Bhaladhare, P.R., Jinwala, D.C.: A clustering approach for the-diversity model in privacy preserving data mining using fractional calculus-bacterial foraging optimization algorithm. In: *Advances in Computer Engineering 2014* (2014)
9. Ben, N., Qiqi, D., Hong, W., Jing, L.: Simplified bacterial foraging optimization with quorum sensing for global optimization. *Int. J. Intell. Syst.* **36**, 2639–2679 (2021)
10. Zhan, Z.-H., Zhang, J., Li, Y., Chung, H.S.-H.: Adaptive particle swarm optimization. *IEEE Trans. Syst. Man Cybern. Part B* **39**, 1362–1381 (2009)
11. Passino, K.M.: Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Syst. Mag.* **22**, 52–67 (2002)
12. Liu, Y., Passino, K.: Biomimicry of social foraging bacteria for distributed optimization: models, principles, and emergent behaviors. *J. Optim. Theory Appl.* **115**, 603–628 (2002)
13. Phillips, S.J.: Acceleration of k-means and related clustering algorithms. In: Mount, D.M., Stein, C. (eds.) *ALENEX 2002*. LNCS, vol. 2409, pp. 166–177. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45643-0_13
14. MacKay, D.J., Mac Kay, D.J.: *Information theory, inference and learning algorithms*. Cambridge university press (2003)

15. Song, B., Wang, Z., Zou, L.: On global smooth path planning for mobile robots using a novel multimodal delayed PSO algorithm. *Cogn. Comput.* **9**, 5–17 (2017). <https://doi.org/10.1007/s12559-016-9442-4>
16. Van der Merwe, D., Engelbrecht, A.P.: Data clustering using particle swarm optimization. In: *The 2003 Congress on Evolutionary Computation, 2003. CEC 2003*, pp. 215–220. IEEE (2003)
17. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **20**, 53–65 (1987)
18. Taha, A.A., Hanbury, A.: Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. *BMC Med. Imaging* **15**, 1–28 (2015)