An effective improved co-evolution ant colony optimisation algorithm with multi-strategies and its application

Wu Deng

College of Electronic Information and Automation,
Civil Aviation University of China,
Tianjin 300300, China
and
Traction Power State Key Laboratory,
Southwest Jiaotong University,
Chengdu 610031, China
Email: dw7689@163.com

Junjie Xu

College of Computer Science and Technology,
Civil Aviation University of China,
Tianjin 300300, China
Email: connyadmin@163.com

Yingjie Song

Co-innovation Center of Shandong Colleges and Universities:
Future Intelligent Computing,
Shandong Technology and Business University,
Yantai 264005, China
Email: songyj@sdtbu.edu.cn

Huimin Zhao*

College of Electronic Information and Automation,
Civil Aviation University of China,
Tianjin 300300, China
and
The State Key Laboratory of Mechanical Transmissions,
Chongqing University,
Chongqing 400044, China
Email: hm_zhao1977@126.com
*Corresponding author

Abstract: In this paper, an effective improved co-evolution ant colony optimisation (MSICEAO) algorithm is presented to solve complex optimisation problem. In the MSICEAO, the multi-population co-evolution strategy is used to divide initial population into several sub-populations to interchange and share information. The weighted initial pheromone distribution strategy is used to improve the efficiency and adjust the pheromone factor and distance factor. The elitist retention strategy is used to improve the solution quality. The adaptive dynamic update strategy for pheromone evaporation rate is used to balance the convergence speed and solution quality. The aggregation pheromone diffusion mechanism is used to enhance the cooperative effect and highlight the cooperative idea of swarm intelligence. In order to verify the effectiveness of the MSICEAO, the experiments have been carried out on eight TSPs and one actual gate allocation problem. The MSICEAO is compared with five state-of-the-art algorithms of TS, GA, PSO, ACO and PSACO. The experiment results demonstrate that the MSICEAO is significantly better than the compared methods.

Keywords: ant colony optimisation; ACO; multi-population co-evolution; elitist retention; pheromone control strategy; adaptive dynamic update; gate allocation.

Reference to this paper should be made as follows: Deng, W., Xu, J., Song, Y. and Zhao, H. (2020) 'An effective improved co-evolution ant colony optimisation algorithm with multi-strategies and its application', *Int. J. Bio-Inspired Computation*, Vol. 16, No. 3, pp.158–170.

Biographical notes: Wu Deng received his PhD in Computer Application Technology from Dalian Maritime University, Dalian, China, in 2012. From 2019 to now, he was a Professor with the College of Electronic Information and Automation, Civil Aviation University of China, Tianjin, China. His research interest includes artificial intelligence and optimisation method.

Junjie Xu received her PhD in Computer Application Technology from Dalian Maritime University, Dalian, China, in 2016. From 2016 to now, she was a Lecture with the College of Computer Science and Technology, Civil Aviation University of China, Tianjin, China. Her research interest includes artificial intelligence and information safety.

Yingjie Song received her PhD in Computer Application Technology from Dalian Maritime University, Dalian, China, in 2013. From 2019 to now, she was an Associate Professor with the College of Computer Science and Technology, Shandong Institute of Business and Technology, Yantai China. Her research interest includes artificial intelligence.

Huimin Zhao received her PhD in Mechanical Engineering from Dalian Jiaotong University, Dalian, China, in 2013. From 2019 to now, she was a Professor with the College of Electronic Information and Automation, Civil Aviation University of China, Tianjin China. Her research interest includes artificial intelligence and signal processing.

1 Introduction

At present, some intelligent algorithms have attracted more and more attention, genetic algorithm (GA), particle swarm optimisation (PSO), ant colony optimisation (ACO), bee colony optimisation (BCO), cuckoo search, bacterial foraging optimisation, pigeon-inspired optimisation and so on (Cui et al., 2013; Cui et al., 2019a; Guo et al., 2019; Liu et al., 2019; Saranya et al., 2018; Zhang et al., 2018). These intelligent algorithms cannot only solve the combinatorial optimisation problems and NP-hard problems, but also can achieve better results in solving optimisation problems (Cai et al., 2019a, 2019b; Chen et al., 2019c, 2019a, 2020; He et al., 2019; Liu et al., 2018). However, because each intelligent algorithm has its own advantages and disadvantages, it has become a research hotspot to further study and improve these intelligent algorithms to improve their optimisation ability for solving complex engineering optimisation problems.

ACO algorithm was proposed by Marco Dorigo, which is was inspired by the simulated evolutionary algorithm that ants found path behaviour in the process of searching for food (Ali et al., 2017; Cheng et al., 2019; Dorigo et al., 1996; Goran and Tihana, 2017; Pan, 2016; Xu et al., 2019a). The ACO takes on the characteristics of positive feedback, distributed computing and strong robustness (Cui et al., 2019d; He and Ma, 2014; Lu and Yu, 2013; Meng et al., 2019; Xu et al., 2019b; Xue et al., 2019; Zhao et al., 2019a, 2019b; Cui et al., 2019e; Wu et al., 2020). At present, the ACO has been successfully applied to solve travelling salesmen problem (TSP), vehicle scheduling, job scheduling, and other multi-objective combinatorial optimisation problems (Dell'Orco et al., 2017; Li et al., 2019a, 2019b; Patvardhan et al., 2016; Wang et al., 2019a; Wen et al., 2018; Yu et al., 2016; Zhao et al., 2020). However, the ACO has its inherent drawbacks, such as slow convergence speed, difficultly determining parameters and premature convergence and so on (Cui et al., 2019b; Tong et al., 2018). It cannot efficiently and accurately solve the optimisation problems. To make up for the shortcomings of the ACO, many scholars have improved the ACO from different aspects in order to improve global search ability and convergence speed. Jovanovic and Tuba (2011) presented an ACO with improved pheromone correction strategy for minimum weight vertex cover problem. Xing et al. (2011) presented a hybrid ACO for the extended capacitated arc routing problem. Cheng et al. (2013) presented a novel ACO for scheduling parallel batching machines. Juang et al. (2014) presented a cooperative continuous ACO for accuracy-oriented fuzzy systems. Wang et al. (2015a) proposed a bio-inspired ACO based on clustering algorithm for home automation networks. Wang et al. (2015b) presented a modified ACO for the network coding resource minimisation problem. Gao et al. (2016) presented a clustering ACO with three immigrant schemes. Zhu et al. (2017) presented a multi-objective lazy ACO for polarisation problem. Zhong and Ai (2017) presented a modified ACO for multi-objective line balancing problem. Deng et al. (2017a) presented an improved adaptive PSO for gate assignment. Zheng et al. (2017) presented an adaptive convergence-trajectory controlled ACO for water distribution system design. Huang and Yu (2017) presented an effective ACO for multi-objective job-shop scheduling. Mavrovouniotis et al. (2017) presented a memetic ACO with local search operator for dynamic TSP. Deng et al. (2017b) presented a genetic and ant colony adaptive collaborative optimisation algorithm. Zhang and Zhang (2018) presented an improved ACO based on strengthened pheromone updating mechanism. Zhang and Xiong (2018) presented an immune ACO for grain emergency vehicle scheduling. Sharifipour et al. (2018) presented a memetic ACO for structural test data generation. Yan (2018) presented a parallel ACO based on incorporating (1 + 1)-evolution strategies. Ning et al. (2018) presented a best-path-updating information-guided ACO algorithm. Chen et al. (2019b) presented a balanced whale optimisation algorithm for tension spring, welded beam, pressure vessel design. Li et al. (2019c) presented an improved ACO based on the improved pheromone updating. Luo et al. (2019a) presented an improved grasshopper optimisation algorithm. Deng et al. (2019) presented an improved ACO based on the multi-strategy. Wu et al. (2019) presented a multimodal continuous ACO for multi-sensor remote sensing image registration. Luo et al. (2019b) presented an improved whale optimisation algorithm. Other some optimisation algorithms are proposed in recent years (Cui et al., 2019c; Deivalakshmi et al., 2019; Gao et al., 2013; Li et al., 2019d; Meng et al., 2015; Nalluri et al., 2019; Shao et al., 2019; Wang et al., 2017, 2019b; Zhang et al., 2020).

As can be known from these related works, although these improved ACO can better solve complex optimisation problems and obtain better solutions, there still exist slow convergence speed, difficultly determining parameters and premature convergence. Therefore, it is necessary to further deeply study a new improved ACO. In this paper, an effective improved co-evolution ant colony optimisation (MSICEAO) algorithm based on making use of the advantages of the multi-population co-evolution, weighted initial pheromone distribution, elitist retention, adaptive dynamic update of pheromone evaporation rate and aggregation pheromone diffusion mechanism. Finally, the TSPs and airport gate allocation problem are used to verify the effectiveness of the MSICEAO algorithm.

2 The ACO algorithm

The ACO is a probabilistic algorithm, which is used to find the optimal path (Fadl et al., 2019). It was inspired by the behaviour of ants in finding their way to food. It has the characteristics of distributed computing, positive information feedback and heuristic search. The procedure of the ACO is shown in Figure 1.

2.1 The transition rule

In the route of the ACO, the k^{th} ant starts from city r, the next city s is selected among the unvisited cities in J_r^k.

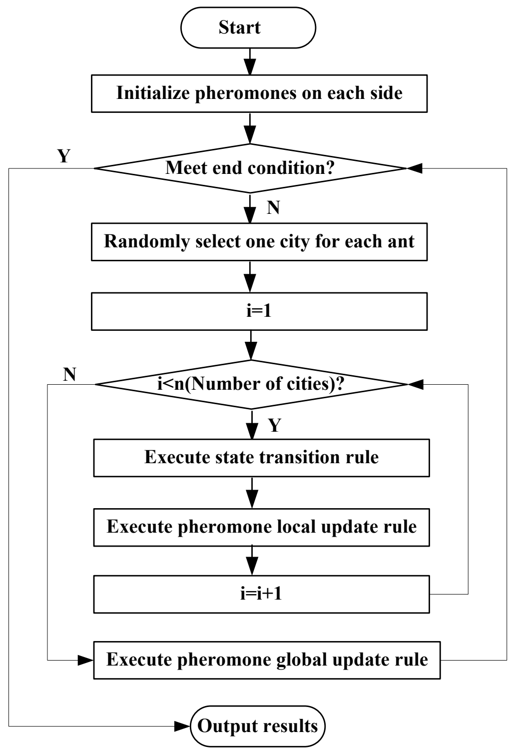
$$s = \arg\max_{u \in J_r^k} \left[\tau_i(r,u)^{\alpha} \cdot \eta(r,u)^{\beta}\right] \quad if\ q \leq q_0 (Exploitation) \tag{1}$$

To visit the next city s with the probability $p_k(r, s)$,

$$p_k(r,s) = \begin{cases} \dfrac{\tau(r,s)^{\alpha} \cdot \eta(r,s)^{\beta}}{\sum_{u \in J_r^k} \tau(r,u)^{\alpha} \cdot \eta(r,u)^{\beta}} & if\ s \in J_r^k \\ 0 & otherwise \end{cases} \tag{2}$$

where $p_k(r, s)$ is transition probability, $\tau(r, s)$ is the pheromone intensity between city r and u, $\eta(r, u)$ is the path length from city r to u, J_r^k is the unvisited cities of the k^{th} ant in the i^{th} population, α and β are the control parameters.

Figure 1 The procedure of the ACO



2.2 Pheromone update rule

To improve the solutions, the pheromone trails need to be updated. The trail updating has the local updating and global updating. The local updating is described.

$$\tau(r,u) = (1-\rho)\tau(r,s) + \sum_{k=1}^{m} \Delta\tau_k(r,s) \tag{3}$$

where ρ ($0 < \rho < 1$) is evaporation rate of pheromone, $\Delta\tau_k(r, s)$ is the added amount of pheromone of the edge between city r and city s, which is described.

$$\Delta\tau_k(r,s) = \begin{cases} \dfrac{Q}{L_k} & (r,s) \in \pi_k \\ 0 & otherwise \end{cases} \tag{4}$$

where Q is the total amount of pheromone, L_k is the distance of the sequence π_k toured by ant in Δt.

3 An MSICEAO algorithm

The ACO has positive feedback, parallelism and better optimisation performance in solving complex optimisation problems, but it exist slow convergence speed, difficultly determining parameters and premature convergence. Therefore, an MSICEAO algorithm is proposed in this paper. In the MSICEAO, the multi-population co-evolution strategy is used to divide initial population into several sub-populations, which are independently and simultaneously evolved to interchange and share information, promote to evolve continuously and obtain the better optimisation ability. The weighted initial pheromone distribution strategy based on nearest neighbour method is used to improve the efficiency and adjust the pheromone influencing factor and distance influencing factor. The elitist retention strategy is used to give additional pheromone enhancement to improve the quality of optimal solutions. The adaptive dynamic update strategy for evaporation rate of pheromone is used to balance between the convergence speed and the optimal solution quality. The aggregation pheromone diffusion mechanism is used to enhance the cooperative effect between individuals and better highlight cooperative idea of swarm intelligence. Therefore, these strategies are fully used to improve the optimisation ability and avoid premature convergence.

3.1 Multi-population co-evolution strategy

The ACO is demonstrated that it takes on effectiveness in solving function optimisation problems. However, when the ACO is used to deal with multimodal functions, it can only find one or two global optimum peaks. The ACO uses one population, it has no ability to find multiple or all optimal solutions of multimodal problems in a search process. Therefore, an isolation niche technology is used to divide the initial population into several sub-populations to independently evolved each other. It can be regarded that multi-single-population are simultaneously evolved. Co-evolution is a new evolutionary algorithm. It considers the coordination between population and environment and between population and population in the evolutionary process. Therefore, the successful experience of co-evolution algorithm based on competition among sub-populations and the competition between population and environment and between population and population is used, the ACO is regarded as subsystem of multi-population competition system. The competition, restriction, coordination and utilisation among sub-populations are used to promote to evolve continuously to obtain the ability for solving multi-modal and multi-modal problems. The multi-population co-evolution strategy emphasises the existing interaction among different sub-populations.

3.2 Weighted initial pheromone distribution strategy

The elements in the initial pheromone intensity matrix of the ACO are equal, which will lead to the lack of purposefulness in the initial iteration. This will cause slow convergence speed and efficiency. Nearest neighbour method is one of the simplest data mining algorithms, because it has the characteristics of intuition and fast solving speed. Therefore, the nearest neighbour method is used to set the initial pheromone intensity in order to improve the efficiency of the algorithm. The nearest neighbour method obtained the initial travelling path to transform the path distance between any two cities, which is regarded as the initial pheromone distribution matrix. The initial pheromone distribution is obtained by the following expression.

$$\tau_{ij}(0) = \begin{cases} \left(\dfrac{1}{d_{ij}} \right) \Big/ \left(\dfrac{l_N}{n} \right), & \text{if } i \neq j. \\ 0, & \text{other} \end{cases} \tag{5}$$

where n is the number of cities. l_N / n is the average value of nearest neighbour between any cities in the initial travelling route, it is the ratio of the length of the initial travelling route, the number of cities by the nearest neighbour method.

At the same time, if the ratio between α and β is large, the path planning in the later stage of the iteration will still be greatly affected. If the ratio between α and β is small, it will easily lead to blindness of path selection in the initial stage, which makes the low convergence speed of the algorithm. The obtained results by the nearest neighbour method are regarded as the basis for setting the initial pheromone intensity, which will not only make the path selection with more purpose in the initial stage, but also avoid the excessive influence of the distance in the later stage. This method can adjust the pheromone influencing factor α and distance influencing factor β.

3.3 The elitist retention strategy

The elitist retention strategy is to give additional pheromone enhancement to all optimal solutions that have been found at the end of each cycle in order to make the optimal solutions, which have been more attractive to ants in the next cycle. The ants are sorted according to the path length. It is considered that the contribution of ants to pheromone updating is related to the quality of the found path. The path is shorter, the contribution of pheromone updating is greater. After each ant searches the path, the ants are sorted $L_1 < L_2 < \cdots < L_m$ according to the path length. The contribution of ants (μ) to pheromone updating is weighted according to the sorted order of ants. One ant of $\sigma - 1$ ants obtained a certain amount of pheromone, which was proportional to the sorted order of ants.

The optimised pheromone quantity is updated.

$$\tau_{ij}(t) = \rho\tau_{ij}(t) + \Delta\tau_{ij} + \Delta\tau_{ij}^{*} \tag{6}$$

$$\Delta\tau_{ij} = \sum_{\mu=1}^{\sigma-1} \Delta\tau_{ij}^{\mu} \tag{7}$$

$$\Delta\tau_{ij}^{\mu} = \begin{cases} (\sigma-\mu)\dfrac{Q}{L_\mu}, \text{ if the passed path } (i, j) \text{ of the } \mu \text{ elite ant.} \\ 0, \text{ other} \end{cases} \quad (8)$$

$$\Delta\tau_{ij}^{*} = \begin{cases} \sigma\dfrac{Q}{L^*}, \text{ if the path } (i, j) \text{ is part of optimal solutions.} \\ 0, \quad \text{other} \end{cases} \quad (9)$$

where μ is the ordinal number of elite ant, $\Delta\tau_{ij}^{\mu}$ is the increment of pheromones on the path (i, j) by the μ elite ant, L_μ is the path length of the μ elite ant. $\Delta\tau_{ij}^{*}$ is the increment of pheromones on the path (i, j) by the elite ants, σ is the number of elite ants. L^* is the length of optimal path, $\Delta\tau_{ij}$ is updating pheromone quantity of $\sigma - 1$ ants on the path (i, j) according to the sorted order of the ants.

3.4 Adaptive dynamic update strategy for evaporation rate of pheromone

When the space scale of search solution is relatively large, if the evaporation rate of pheromone is larger, the pheromones on the unvisited edges by ants will be reduced to nearly zero, which reduces the global search ability. If the evaporation rate of pheromone is smaller, the global search ability will be improved, but the convergence speed is slower and the convergence time is longer due to the enlarged difference of pheromone intensity for all paths. To solve these problems, an adaptive dynamic update strategy for evaporation rate of pheromone is proposed in this paper. In this strategy, the evaporation rate of pheromone is dynamically updated adaptively instead of setting value. The evaporation rate of pheromone decreases dynamically with the increase of the number of iterations. When the obtained optimal solution doesn't change in the several iterations, the obtained evaporation rate of pheromone will not change. The new evaporation rate of pheromone is described.

$$\rho(t+n) = \begin{cases} e^{-t/T}\varepsilon(t)\rho(t), & \rho(t+n) > \rho_{\min} \\ \rho_{\min}, & (t+n) \le \rho_{\min} \end{cases} \quad (10)$$

where $\varepsilon(t)$ is a step function, t is the current number of iterations and T is the maximum number of iterations. The value of $\rho_{\min}$ has a certain influence on the convergence speed of the algorithm and the quality of the optimal solution. After a lot of experiments, it is shown that when the value of $\rho_{\min}$ is 0.1, which can balance between the convergence speed and the optimal solution quality.

3.5 Aggregation pheromone diffusion mechanism

In the ACO, the ants use single pheromone release mode to update the pheromone. The model can only affect the successor ants passing, and cannot guide the ant search within a certain range. Pheromone aggregation is an important carrier for ant colony to realise swarm intelligence. In general, the number of more nodes in the local environment is aggregated one node, the amount of pheromones will be larger, which can attract more ants to move closer to its neighbours. At the same time, the aggregation pheromone is a chemical substance, which is gradually evaporated with increase of time. Therefore, a pheromone diffusion mechanism is presented. This mechanism considered the influence of pheromones on the close-range positions in solving feasible solutions. There is a coupling effect between adjacent pheromone diffusion. The distance between two positions is closer, the coupling effect is stronger. The pheromone updating requires decoupling compensation for pheromone intensity of adjacent positions.

The pheromone diffusion mechanism is described.

$$\tau_{i,j}(t) = \begin{cases} (1-d_r)\dfrac{\tau_i(t)}{r}, & d_r < r \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

where $\tau_i(t)$ is the pheromone trail on a node i laid by the ant colony. i is the centre dot and j is the adjacent node, r is the radius of diffusion area, d_r is the correlation distance between two objects.

3.6 The steps of MSICEAO algorithm

The steps of the MSICEAO algorithm are described.

Step 1 Initialise the parameters of the MSICEAO, including population size, pheromone amount, pheromone influencing factor α and distance influencing factor β, the initial pheromone evaporation rate ρ, the maximum number of iterations, the current number of iteration, and so on.

Step 2 The isolation niche technology is used to divide the initial population into several sub-populations, which corresponds one optimisation problem.

Step 3 The nearest neighbour method based on nearest neighbour is used to set the initial pheromone distribution according to equation (5).

Step 4 Each ant in sub-populations is randomly initialised the position.

Step 5 The transition rule of the next city is calculated according to the equation(2).

Step 6 Each sub-population is independently and simultaneously evolved to realise information interchanging and sharing.

Step 7 The pheromone intensity of the passed path by ants in each sub-population is updated according to equation (3).

Step 8 The pheromone intensity of the adjacent path in each sub-population is updated according to the aggregation pheromone diffusion mechanism.

Step 9 The elitist retention strategy is used to give additional pheromone enhancement to improve the quality of obtained optimal solutions.

Step 10 Determine whether the obtained solution has met the end condition or the maximum number of iterations is reached.

Step 11 If the end condition is not met, step 5 is re-executed to start a new evolution. Otherwise go to next step.

Step 12 The optimal solution is obtained and output.

4 Numerical experiment and analysis

4.1 Experiment environment

In order to testify the effectiveness of the MSICEAO, several TSPs from TSPLIB standard library (http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/) are used. The MSICEAO is compared with the tabu search, GA, PSO, ACO and PSACO algorithms. The experiment environments are: MATLAB 2018b, the Pentium CPU i7, 8.0 GB RAM with Windows 10.

4.2 Set parameters

The values of parameters of these algorithms could be a complicated problem, their values will seriously affect the obtained results. In the experiment, the alternative values were tested and modified for some functions to obtain the most reasonable initial values. The most reasonable initial values are shown in Table 1.

Table 1 The values of these parameters

Parameter	*GA*	*PSO*	*ACO*	*PSACO*	*MSICEAO*
Population size (N)	120	120	120	120	120
Iterations (T)	1,000	1,000	1,000	1,000	1,000
Pheromone factor (α)	—	—	2.0	2.0	2.0
Distance factor (β)	—	—	2.0	2.0	2.0
Pheromone amount (Q)	—	—	100	100	100
Evaporation rate (ρ)	—	—	0.05	0.05	0.05
Initial intensity ($\tau_{ij}(0)$)	—	—	1.50	1.50	1.50
Chaos search (μ)	—	—	—	3.0	—
Crossover rate (P_c)	0.90	—	—	—	—
Mutation rate (P_m)	0.01	—	—	—	—
Inertia weight (w)	—	0.80	—	—	—
Learning factor (c_1, c_2)	—	$c_1 = c_2 = 2$	—	—	—

4.3 Experiment results and analysis

For each TSP, these algorithms are executed 15 times, respectively. The obtained results are shown in Table 2.

Table 2 The obtained experiment results for each TSP

Instances	*Algorithms*	*Optimal value*	*Best value*	*Worst value*	*Average value*	*Variance*
berlin52	Tabu search	7,542	7,976.84	8,286.68	8,014.60	434.84
	GA		8,201.17	8,443.02	8,376.55	659.17
	PSO		8,197.79	8,589.31	8,319.51	655.79
	ACO		7,647.55	7,780.57	7,732.31	105.55
	PSACO		7,568.54	7,618.31	7,586.42	86.43
	MSICEAO		*7,542.00*	7,548.30	*7,543.20*	*11.46*
pr76	Tabu search	108,159	110,941	130,637	122,104	2,782
	GA		115,329	124,851	120,245	7,170
	PSO		118,038	126,583	122,735	9,879
	ACO		110,517	120,922	114,964	2,358
	PSACO		109,244	113,120	110,162	1,085
	MSICEAO		*109,207*	1,126,168	*110,074*	*715*
eil101	Tabu search	629	667.43	709.11	685.49	38.43
	GA		682.37	745.33	706.25	53.37
	PSO		687.32	779.11	731.58	58.32
	ACO		649.87	695.18	664.07	20.87
	PSACO		637.65	674.07	651.36	8.65
	MSICEAO		*631.00*	641.42	*634.78*	*4.90*

Table 2 The obtained experiment results for each TSP (continued)

Instances	*Algorithms*	*Optimal value*	*Best value*	*Worst value*	*Average value*	*Variance*
kroA200	Tabu search	29,368	31,289	33,438	32,219	1,921
	GA		32,261	34,572	33,158	2,893
	PSO		32,350	34,526	33,132	2,982
	ACO		31,669	33,839	32,434	2,301
	PSACO		30,190	33,626	31,927	822
	MSICEAO		*29,582*	31,447	*30,283*	*356*
rat783	Tabu search	8,806	9,185	9,407	9,294	397
	GA		9,217	9,423	9,315	411
	PSO		9,316	9,516	9,423	510
	ACO		9,093	9,403	9,246	287
	PSACO		9,041	9,387	9,177	235
	MSICEAO		*8,988*	9,229	*9,079*	*85.3*
pr1002	Tabu search	259,045	267,617	269,419	268,715	8572
	GA		274,124	278,547	276,812	15,079
	PSO		278,476	282,591	280,738	19,431
	ACO		268,502	269,859	268,934	9,457
	PSACO		266,213	268,549	267,576	7,168
	MSICEAO		*263,679*	267,103	*265,362*	*2,641*

As can be seen from Table 2, it shows the best, worst, and average values of tabu search, GA, PSO, ACO, PSACO and MSICEAO for comparison. The experiment results show that the MSICEAO obtained optimal distances for eight TSPs in terms of best, worst, and average values. The MSICEAO obtained 7,542, 631, 10,9207 and 8,988 for berlin52, eil101, pr76 and rat783, respectively, which are equal or almost close to the best value of 7,542, 629, 108,159 and 8,806. Therefore, the experiment results show that the MSICEAO takes on better optimisation ability in solving TSPs than the tabu search, GA, PSO, ACO and PSACO. From the average value and variance, the MSICEAO has better stability and robustness in solving TSP The proposed algorithm can effectively solve the TSPs and obtain better solutions.

5 Case application and result analysis

5.1 Construct mathematical model of gate assignment

5.1.1 The shortest passenger walking distances

The satisfaction of passengers in airport directly determines the airport service level. Therefore, the shortest walking distances are selected as the optimisation objective.

$$\min F_1 = \min \sum_{i=1}^{n} \sum_{j=1}^{m} q_{ij} f_j y_{ij} \tag{12}$$

where q_{ij} is the number of passengers, f_j is the walking distances of passengers and y_{ij} is a variable of 0–1.

5.1.2 The most balanced idle time of each gate

When gate assignment guarantees a balanced idle time, it can play a buffer role in the small-scale short-term delays. Therefore, the most balanced idle time of each gate is selected as the optimisation objective.

$$\min F_2 = \min \sum_{i=1}^{n} \sum_{k=1}^{m} S_{ik}^2 + \sum_{k=1}^{m} SS_k^2 \tag{13}$$

where n is the total number of flights, m is the number of gates. S_{ik} is the idle time of gate before the i^{th} flight arrives at k^{th} gate.

5.1.3 The best use of large gate

The large gates should be assigned to large aircraft as far as possible to ensure the satisfaction level of passengers Therefore, the best use of large gate is selected as the optimisation objective.

$$\min F_3 = \sum_{i=1}^{n} \sum_{j=1}^{m} G_{ij} \tag{14}$$

where G_{ij} are the parked small and medium-sized aircraft in large gate and the parked small aircrafts in medium gate.

Three optimisation objective functions are considered to construct the gate allocation model. Because these objective functions have their own dimensions, it is necessary to quantitatively deal with these objective functions. In this paper, the linear weighting method is used to set the weight factor $W_i > 0$ for non-quantisation. Therefore, the gate allocation model is obtained.

$$F = \frac{W_1}{F_1^0}\sum_{i=1}^{n}\sum_{j=1}^{m} q_{ij} f_j y_{ij} + \frac{W_2}{F_2^0}\left[\sum_{i=1}^{n}\sum_{k=1}^{m} S_{ik}^2 + \sum_{k=1}^{m} SS_k^2\right] + \frac{W_3}{F_3^0}\sum_{i=1}^{n}\sum_{j=1}^{m} G_{ij} \quad (15)$$

$$F = \sum_{i=1}^{n} \frac{W_i F_i}{F_i^0} \quad (16)$$

The gate allocation problem is subjected to four constraints.

$$\sum_{k=1}^{m}\left(y_{ik} + g_i\right) = 1 \quad (17)$$

$$G_r \geq F_i \quad \left(y_{ir} = 1\right) \quad (18)$$

$$L_{ik} - E_{ik} \geq 5 \quad Z_{ijk} = 1 \quad (19)$$

$$F_{near} > F_{far} > g_i \quad (20)$$

Table 3 The detailed information of 30 gates

Gate	*Distances*	*Gate types*	*Gate*	*Distances*	*Gate types*
G1	190	M	G16	115	L
G2	975	M	G17	215	M
G3	400	L	G18	535	S
G4	333	M	G19	1,050	M
G5	260	L	G20	170	M
G6	135	S	G21	585	L
G7	1,100	M	G22	1,250	M
G8	150	M	G23	500	L
G9	384	L	G24	920	L
G10	960	M	G25	270	L
G11	1,000	S	G26	230	M
G12	235	L	G27	265	L
G13	1,200	S	G28	450	L
G14	580	L	G29	1,300	M
G15	440	L	G30	426	L

5.2 Experiment data

Two hundred fifty flights from Guangzhou Baiyun Airport on July 26, 2015 and 30 gates are used to validate the effectiveness of the MSICEAO. The detailed information of 30 gates is described in Table 3, and the detailed information of 250 flights is described in Table 4. The safe time between two flights is five minutes for the same gate. When the flights are not assigned to the gates, they will spark on the apron.

Table 4 The detailed information of 250 flights

Flights	*Arrival time*	*Departure time*	*Passengers*	*Types*
F1	2015-7-26 0:05:00	2015-7-26 5:15:00	482	L
F2	2015-7-26 0:05:00	2015-7-26 5:45:00	273	M
F3	2015-7-26 0:10:00	2015-7-26 5:30:00	261	M
F4	2015-7-26 0:15:00	2015-7-26 5:30:00	116	M
F5	2015-7-26 0:15:00	2015-7-26 5:15:00	244	M
F6	2015-7-26 0:20:00	2015-7-26 5:30:00	312	L
F7	2015-7-26 0:25:00	2015-7-26 5:20:00	340	L
F8	2015-7-26 0:30:00	2015-7-26 6:00:00	198	M
F9	2015-7-26 0:35:00	2015-7-26 6:10:00	184	M
F10	2015-7-26 0:35:00	2015-7-26 6:55:00	494	L
⋮	⋮	⋮	⋮	⋮
F249	2015-7-26 23:50:00	2015-7-27 1:50:00	252	S
F250	2015-7-26 23:55:00	2015-7-27 9:10:00	378	M

5.3 Experiment results and analysis

The constructed airport gate allocation model is solved by the MSICEAO for continuous 15 times. The obtained best allocation result is shown Table 5.

As can be seen from Table 5, Figure 2 and Figure 3, 242 flights are allocated to the 30 gates, eight flights are allocated to the apron. The allocated rate for gates is 96.8%, which is an ideal allocation result for airport gates. There are G13 and G21, which are allocated five flights. There are G6, G11, G16 and G27, which are allocated seven flights. There are 12 gates, which are allocated eight flights. There are 12 gates, which are allocated 12 flights. From the number of allocated result for each gate, except for G13 and G21, the rest 28 gates are allocated seven flights to nine flights, respectively. Therefore, the number of allocated flights for each gate is more balanced, and the idle time of each gate is more balanced, which enables staff to have sufficient time to schedule. The proposed MSICEAO can effectively allocate 242 flights to 30 gates, and obtain ideal allocation result. Therefore, the MSICEAO has better optimisation ability to solve gate allocation problem.

Figure 2 The Gantt chart of gate allocation (see online version for colours)

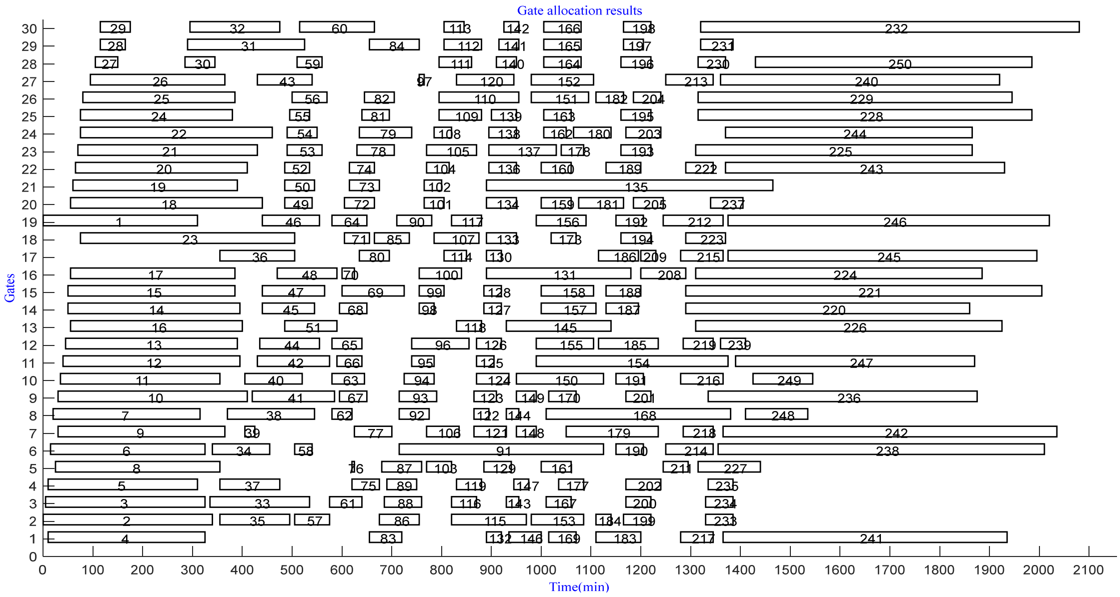


Table 5 The obtained gate assignment results

Gate	*Flights*									*Total*
G1	F4	F83	F132	F146	F169	F183	F217	F241		8
G2	F2	F35	F57	F86	F115	F153	F184	F199	F233	9
G3	F3	F33	F61	F88	F116	F143	F167	F200	F234	9
G4	F5	F37	F75	F89	F119	F147	F177	F202	F235	9
G5	F8	F76	F87	F103	F129	F161	F211	F227		8
G6	F6	F34	F58	F91	F190	F214	F238			7
G7	F9	F39	F77	F106	F121	F148	F179	F218	F242	9
G8	F7	F38	F62	F92	F122	F144	F168	F248		8
G9	F10	F41	F67	F93	F123	F149	F170	F201	F236	9
G10	F11	F40	F63	F94	F124	F150	F191	F216	F249	9
G11	F12	F42	F66	F95	F125	F154	247			7
G12	F13	F44	F65	F96	F126	F155	F185	F219	F239	9
G13	F16	F51	F118	F145	F226					5
G14	F14	F45	F68	F98	F127	F157	F187	F220		8
G15	F15	F47	F69	F99	F128	F158	F188	F221		8
G16	F17	F48	F70	F100	F131	F208	F224			7
G17	F36	F80	F114	F130	F186	F209	F215	F245		8
G18	F23	F71	F85	F107	F133	F173	F194	F223		8
G19	F1	F46	F64	F90	F117	F156	F192	F212	F246	9
G20	F18	F49	F72	F101	F134	F159	F181	F205	F237	9
G21	F19	F50	F73	F102	F135					5
G22	F20	F52	F74	F104	F136	F160	F189	F222	F243	9
G23	F21	F53	F78	F105	F137	F178	F193	F225		8
G24	F22	F54	F79	F108	F138	F162	F180	F203	F244	9
G25	F24	F55	F81	F109	F139	F163	F195	F228		8
G26	F25	F56	F82	F110	F151	F182	F204	F229		8
G27	F26	F43	F97	F120	F152	F213	F240			7
G28	F27	F30	F59	F111	F140	F164	F196	F230	F250	9
G29	F28	F31	F84	F112	F141	F165	F197	F231		8
G30	F29	F32	F60	F113	F142	F166	F198	F232		8
Total										242

Figure 3 The changing curve of the optimal values (see online version for colours)

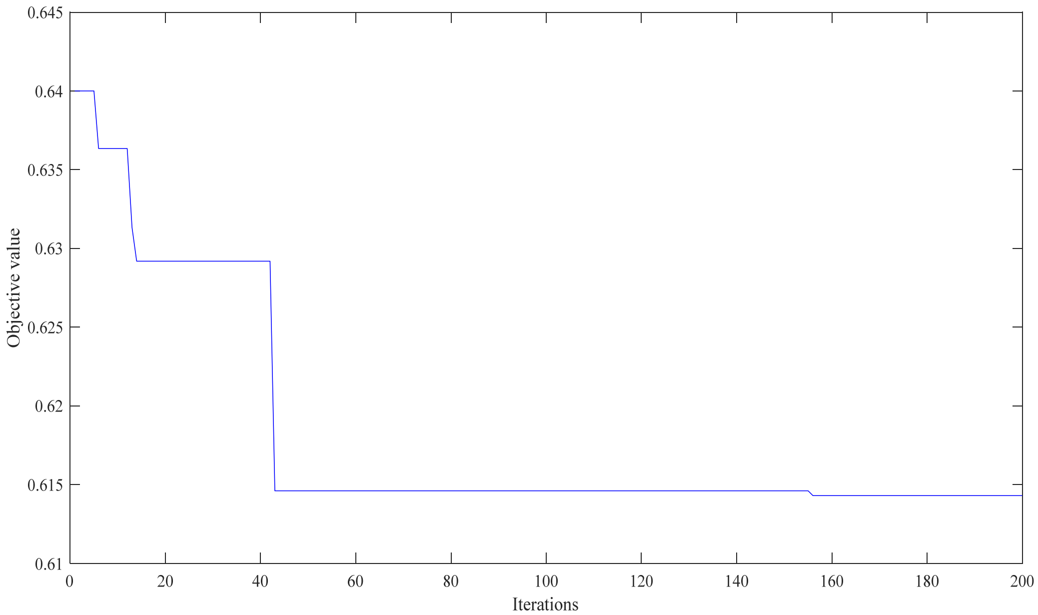


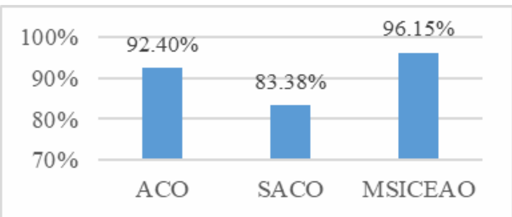
Table 6 The calculation and comparison results

Algorithms	*ACO*		*SACO*		*MSICEAO*	
Indexes	*Allocated rate (%)*	*Iterations*	*Allocated rate (%)*	*Iterations*	*Allocated rate (%)*	*Iterations*
1	90.80	154	87.50	17	*96.80*	156
2	93.20	172	76.00	174	96.80	184
3	92.40	100	79.00	5	96.00	17
4	92.00	96	84.00	68	96.40	172
5	*93.20*	132	*90.50*	139	96.00	52
6	92.80	22	78.50	91	94.80	163
7	92.40	148	85.50	113	96.40	196
8	92.40	159	86.00	147	96.00	12
Average	92.40	122.9	83.38	94.3	*96.15*	119

5.4 Comparison and analysis of results

The MSICEAO is compared with the ACO and SACO to further investigate the optimisation ability for solving gate allocation problem. The experiments were carried out for eight simulations. The experiment results are shown in Table 6.

Figure 4 The comparison result of gate allocation (see online version for colours)



As can be seen from Table 6 and Figure 4, the best and average allocation rates of gates using the ACO are 93.20% and 92.40%, respectively. The best and average allocation rates of gates using the SACO are 90.50% and 83.38%, respectively. The best and average allocation rates of gates using the MSICEAO are 96.80% and 96.15%, respectively. Therefore, the best allocation result and the average allocation result using the MSICEAO are better than those of the ACO and SACO. The stability in eight consecutive simulations for the gate assignment problem is best. Therefore, the MSICEAO can effectively improve the comprehensive optimisation ability for gate assignment. It takes on better optimisation ability and stronger robustness.

6 Conclusions and future work

In this paper, an MSICEAO with multi-strategies is proposed to solve large-scale optimisation problem. Multi-strategies of multi-population co-evolution, weighted initial pheromone distribution, elitist retention, adaptive dynamic update for evaporation rate of pheromone and aggregation pheromone diffusion are introduced into the

ACO to realise information interchanging and sharing, improve the obtained optimal solution quality and the efficiency, adjust the pheromone factor and distance factor, balance between the convergence speed and the solution quality and enhance the cooperative effect. The TSPs and airport gate allocation problem are used to verify the effectiveness of the MSICEAO. The optimisation performance of the MSICEAO is compared with tabu search, GA, PSO, ACO algorithm and PSACO. The obtained results show that the MSICEAO can effectively solve these optimisation problems and obtain better optimisation results. It can allocate 242 flights to 30 gates and the allocated rate is 96.8%. Therefore, the proposed MSICEAO takes on better optimisation ability and stability.

Due to the longer computation time for solving complex optimisation problem, the MSICEAO will be further deeply studied to decrease the time complexity in the future works.

Acknowledgments

This research was funded by the National Natural Science Foundation of China (61771087, 51605068; 51475065), the Open Project Program of State Key Laboratory of Mechanical Transmissions of Chongqing University (SKLMT-KFKT-201803) and Traction Power State Key Laboratory of Southwest Jiaotong University (TPL2002), the Open Subject Project of the Key Laboratory of Air Traffic Control Operation Planning and Safety Technology of CAUC (600001010932), Liaoning Provincial Natural Science Foundation Guidance Project (2019-ZD-0099, 20170540145), the Science and Technology Project of Liaoning Provincial Department of Education (JDL2019025).

References

Ali, M.Z., Awad, N.H., Suganthan, P.N. et al. (2017) 'An adaptive multipopulation differential evolution with dynamic population reduction', *IEEE Transactions on Cybernetics*, Vol. 47, No. 9, pp.2768–2779.

Cai, X.J., Wang, P.H., Du, L., Cui, Z.H., Zhang W.S. and Chen J.J. (2019a) 'Multi-objective three-dimensional DV-hop localization algorithm with NSGA-II', *IEEE Sensors Journal*, Vol. 19, No. 21, pp.10003–10015.

Cai, X.J., Zhang, J.J., Liang, H., Wang, L. and Wu, Q.D. (2019b) 'An ensemble bat algorithm for large-scale optimization', *International Journal of Machine Learning and Cybernetics*, Vol. 11, No. 10, pp.3099–3113.

Chen, H., Jiao, S., Heidari, A.A., Wang, M., Chen, X. and Zhao, X. (2019a) 'An opposition-based sine cosine approach with local search for parameter estimation of photovoltaic models', *Energy Conversion and Management*, Vol. 195, pp.927–942.

Chen, H.L., Xu, Y.T., Wang, M.J. et al. (2019b) 'A balanced whale optimization algorithm for constrained engineering design problems', *Applied Mathematical Modelling*, Vol. 71, pp.45–59.

Chen, R., Guo, S.K., Wang, X.Z. and Zhang, T.L. (2019c) 'Fusion of multi-RSMOTE with fuzzy integral to classify bug reports with an imbalanced distribution', *IEEE Transactions on Fuzzy Systems*, DOI: 10.1109/TFUZZ. 2019.2899809.

Chen, H., Jiao, S., Wang, M., Heidari, A.A. and Zhao, X. (2020) 'Parameters identification of photovoltaic cells and modules using diversification-enriched Harris hawks optimization with chaotic drifts', *Journal of Cleaner Production*, Vol. 244, p.118778.

Cheng, B.Y., Wang, Q., Yang, S.L. et al. (2013) 'An improved ant colony optimization for scheduling identical parallel batching machines with arbitrary job sizes', *Applied Soft Computing*, Vol. 13, No. 2, pp.765–772.

Cheng, Y.W., Zhu, H.P., Wu, J. and Shao, X.Y. (2019) 'Machine health monitoring using adaptive kernel spectral clustering and deep long short-term memory recurrent neural networks', *IEEE Trans. Ind. Inform.*, Vol. 15, No. 2, pp.987–997.

Cui, Z.H., Gao, X.Z. and Deb, S. (2019a) 'Theory and applications of soft computing methods', *Neural Computing & Applications*, Vol. 31, No. 7, pp.1983–1985.

Cui, Z.H., Cao Y., Cai X.J. et al. (2019b) 'Optimal LEACH protocol with modified bat algorithm for big data sensing systems in internet of things', *Journal of Parallel and Distributed Computing*, Vol. 132, pp.217–229.

Cui, Z.H., Zhang, J.J., Wang, Y.C. et al. (2019c) 'A pigeon-inspired optimization algorithm for many-objective optimization problems', *Science China-Information Sciences*, Vol. 62, No. 7, p.070212.

Cui, Z.H., Zhang, M.Q., Wang, H. et al. (2019d) 'A hybrid many-objective cuckoo search algorithm', *Soft Computing*, Vol. 23, No. 21, pp.10681–10697.

Cui, Z.H., Chang, Y., Zhang J.J. et al. (2019e) 'Improved NSGA-III with selection-and-elimination operator', *Swarm and Evolutionary Computation*, Vol. 49, pp.23–33.

Cui, Z.H., Fan, S.J., Zeng, J.C. et al. (2013) 'APOA with parabola model for directing orbits of chaotic systems', *International Journal of Bio-Inspired Computation*, Vol. 5, No. 1, pp.67–72.

Deivalakshmi, S., Palanisamy, P. and Gao, X.Z. (2019) 'Balanced GHM mutiwavelet transform based contrast enhancement technique for dark images using dynamic stochastic resonance', *Intelligent Automation and Soft Computing*, Vol. 25, No. 3, pp.459–471.

Dell'Orco, M., Marinelli, M. and Altieri, M.G. (2017) 'Solving the gate assignment problem through the fuzzy bee colony optimization', *Transportation Research Part C – Emerging Technologies*, Vol. 80, pp.424–438.

Deng, W., Xu, J.J. and Zhao, H.M. (2019) 'An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem', *IEEE ACCESS*, Vol. 7, pp.20281–20292.

Deng, W., Zhao, H.M., Yang, X.H. et al. (2017a) 'Study on an improved adaptive PSO algorithm for solving multi-objective gate assignment', *Applied Soft Computing*, Vol. 59, pp.288–302.

Deng, W., Zhao, H.M., Zou, L. et al. (2017b) 'A novel collaborative optimization algorithm in solving complex optimization problems', *Soft Computing*, Vol. 21, No. 15, pp.4387–4398.

Dorigo, M., Maniezzo, V. and Colorni, A. (1996) 'Ant system: optimization by a colony of cooperating agents', *IEEE Transactions on Systems Man and Cybernetics Part B – Cybernetics*, Vol. 26, No. 1, pp.29–41.

Fadl, D., Khalil, E.H., Hassan, M. and Hussien, A. (2019) 'Dynamic flying ant colony optimization (DFACO) for solving the traveling salesman problem', *Sensors*, Vol. 19, No. 8, p.1837.

Gao, S., Wang, Y.R., Cheng, J.J. et al. (2016) 'Ant colony optimization with clustering for solving the dynamic location routing problem', *Applied Mathematics and Computation*, Vol. 285, pp.149–173.

Gao, X.Z., Wang, X. and Zenger, K. (2013) 'A modified harmony search method for wind generator design', *International Journal of Bio-Inspired Computation*, Vol. 5, No. 6, pp.336–349

Goran, M. and Tihana, G.G. (2017) 'Co-evolutionary multi-population genetic programming for classification in software defect prediction: an empirical case study', *Applied Soft Computing Journal*, Vol. 55, pp.331–351.

Guo, S.K., Chen, R., Li, H., Zhang, T.L. and Liu, Y.Q. (2019) 'Identify severity bug report with distribution imbalance by CR-SMOTE and ELM', *International Journal of Software Engineering and Knowledge Engineering*, Vol. 29, No. 6, pp.139–175.

He, Y.J. and Ma, Z.F. (2014) 'Optimal design of linear sensor networks for process plants: a multi-objective ant colony optimization approach', *Chemometrics and Intelligent Laboratory Systems*, Vol. 135, pp.37–47.

He, Z., Shao, H., Zhang, X., Cheng, J. and Yang, Y. (2019) 'Improved deep transfer auto-encoder for fault diagnosis of gearbox under variable working conditions with small training samples', *IEEE Access*, Vol. 7, pp.115368–115377.

Huang, R.H. and Yu, T.H. (2017) 'An effective ant colony optimization algorithm for multi-objective job-shop scheduling with equal-size lot-splitting', *Applied Soft Computing*, Vol. 57, pp.642–656.

Jovanovic, R. and Tuba, M. (2011) 'An ant colony optimization algorithm with improved pheromone correction strategy for the minimum weight vertex cover problem', *Applied Soft Computing*, Vol. 11, No. 8, pp.5360–5366.

Juang, C.F., Hung, C.W. and Hsu, CH. (2014) 'Rule-based cooperative continuous ant colony optimization to improve the accuracy of fuzzy system design', *IEEE Transactions on Fuzzy Systems*, Vol. 22, No. 4, pp.723–735.

Li, T., Zhou, Y., Li, X., Wu, J. and He, T. (2019a) 'Forecasting daily crude oil prices using improved CEEMDAN and ridge regression-based predictors', *Energies*, Vol. 12, No. 19, p.3603.

Li, T., Shi, J., Li, X., Wu, J. and Pan, F. (2019b) 'Image encryption based on pixel-level diffusion with dynamic filtering and DNA-level permutation with 3D Latin cubes', *Entropy*, Vol. 21, No. 3, p.319.

Li, Y.B., Soleimani, H. and Zohal, M. (2019c) 'An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives', *Journal of Cleaner Production*, Vol. 227, pp.1161–1172.

Li, X., Xie, Z., Wu, J. and Li, T. (2019d) 'Image encryption based on dynamic filtering and bit cuboid operations', *Complexity*, p.7485621.

Liu, Y.Q., Wang, X.X., Zhai, Z.G,. Chen, R., Zhang, B. and Jiang, Y. (2019) 'Timely daily activity recognition from headmost sensor events', *ISA Transactions*, DOI: 10.1016/j.isatra.2019.04.026.

Liu, Y.Q., Yi, X.K., Chen, R., Zhai, Z.G. and Gu, J.X. (2018) 'Feature extraction based on information gain and sequential pattern for English question classification', *IET Software*, Vol. 12, No. 6, pp.520–526.

Lu, T.C. and Yu, G.R. (2013) 'An adaptive population multi-objective quantum-inspired evolutionary algorithm for multi-objective 0/1 knapsack problems', *Information Sciences*, Vol. 243, pp.39–56.

Luo, J., Chen, H.L., Heidari, A.A. et al. (2019a) 'Multi-strategy boosted mutative whale-inspired optimization approaches', *Applied Mathematical Modelling*, Vol. 73, pp.109–123.

Luo, J., Chen, H.L., Zhang, Q. et al. (2019b) 'An improved grasshopper optimization algorithm with application to financial stress prediction', *Applied Mathematical Modelling*, Vol. 64, pp.654–668.

Mavrovouniotis, M., Muller, F.M. and Yang, S.X. (2017) 'Ant colony optimization with local search for dynamic traveling salesman problems', *IEEE Transactions on Cybernetics*, Vol. 47, No. 7, pp.1743–1756.

Meng, X.B., Gao, X.Z., Liu, Y. and Zhang, H-Z. (2015) 'A novel bat algorithm with habitat selection and effect in echoes for optimization', *Expert Systems with Applications*, Vol. 42, Nos. 17–18, pp.6350–6364.

Meng, X.B., Li, H.X. and Gao, X.Z. (2019) 'An adaptive reinforcement learning-based bat algorithm for structural design problems', *Int. J. Bio-Inspired Computation*, Vol. 14, No. 2, pp.14–124

Nalluri, M.S.R., Kannan K., Gao X.Z. et al. (2019) 'An efficient hybrid meta-heuristic approach for cell formation problem', *Soft Computing*, Vol. 23, No. 19, pp.9189–9213.

Ning, J.X., Zhang, Q., Zhang, C.S. et al. (2018) 'A best-path-updating information-guided ant colony optimization algorithm', *Information Sciences*, Vol. 433, pp.142–162.

Pan, Q.K. (2016) 'An effective co-evolutionary artificial bee colony algorithm for steelmaking-continuous casting scheduling', *European Journal of Operational Research*, Vol. 250, No. 3, pp.702–714.

Patvardhan, C., Bansal, S. and Srivastav, A. (2016) 'A parallel improved quantum inspired evolutionary algorithm to solve large size quadratic knapsack problems', *Swarm and Evolutionary Computation*, Vol. 26, pp.175–190.

Saranya, G., Nehemiah, H.K. and Kannan, A. (2018) 'Hybrid particle swarm optimisation with mutation for code smell detection', *International Journal of Bio-Inspired Computation*, Vol. 12, No. 3, pp.186–195.

Shao, H., Cheng, J., Jiang, H., Yang, Y. and Wu, Z. (2019) 'Enhanced deep gated recurrent unit and complex wavelet packet energy moment entropy for early fault prognosis of bearing', *Knowledge-Based Systems*, DOI: 10.1016/j.knosys.2019.105022.

Sharifipour, H., Shakeri, M. and Haghighi, H. (2018) 'Structural test data generation using a memetic ant colony optimization based on evolution strategies', *Swarm and Evolutionary Computation*, Vol. 40, pp.76–91.

Tong, Y.F., Zhong, M., Li, J.W. et al. (2018) 'Research on intelligent welding robot path optimization based on GA and PSO algorithms', *IEEE Access*, Vol. 6, pp.65397–65404.

Wang, J., Cao, J.Y., Li, B. et al. (2015a) 'Bio-inspired ant colony optimization based clustering algorithm with mobile sinks for applications in consumer home automation networks', *IEEE Transactions on Consumer Electronics*, Vol. 61, No. 4, pp.438–444.

Wang, Z.Y., Xing, H.L., Li, T.R. et al. (2015b) 'A modified ant colony optimization algorithm for network coding resource minimization', *IEEE Transactions on Evolutionary Computation*, Vol. 20, No. 3, pp.325–342.

Wang, Z., Ji, Z., Wang, X. et al. (2017) 'A new parallel DNA algorithm to solve the task scheduling problem based on inspired computational model', *BioSystems*, Vol. 162, pp.59–65.

Wang, J.H., Weng, T.Y. and Zhang, Q.F. (2019a) 'A two-stage multiobjective evolutionary algorithm for multiobjective multidepot vehicle routing problem with time windows', *IEEE Transactions on Cybernetics*, Vol. 49, No. 7, pp.2467–2478.

Wang, Z., Ren, X., Ji, Z. et al. (2019b) 'A novel bio-heuristic computing algorithm to solve the capacitated vehicle routing problem based on Adleman–Lipton model', *BioSystems*, Vol. 184, p.103997.

Wen, J., Zhong, Z.F., Zhang, Z., Fei, L.K., Lai, Z.H., Chen, R.Z. (2018) 'Adaptive locality preserving regression', *IEEE Transactions on Circuits and Systems for Video Technology*, DOI: 10.1109/TCSVT.2018.2889727.

Wu, J., Shi, J. and Li, T. (2020) 'A novel image encryption approach based on a hyperchaotic system, pixel-level filtering with variable kernels, and DNA-level diffusion', *Entropy*, Vol. 22, No. 1, p.5.

Wu, Y., Ma, W.P., Miao, Q.G. et al. (2019) 'Multimodal continuous ant colony optimization for multisensor remote sensing image registration with local search', *Swarm and Evolutionary Computation*, Vol. 47, pp.89–95.

Xing, L.N., Rohlfshagen, P., Chen, Y.W. et al. (2011) 'A hybrid ant colony optimization algorithm for the extended capacitated arc routing problem', *IEEE Transactions on Systems Man and Cybernetics Part B – Cybernetics*, Vol. 41, No. 4, pp.1110–1123.

Xu, Y., Chen, H., Heidari, A.A., Luo, J., Zhang, Q., Zhao, X. and Li, C. (2019a) 'An efficient chaotic mutative moth-flame-inspired optimizer for global optimization tasks', *Expert Systems with Applications*, Vol. 129, pp.135–155.

Xu, Y., Chen, H., Luo, J., Zhang, Q., Jiao, S. and Zhang, X. (2019b) 'Enhanced moth-flame optimizer with mutation strategy for global optimization', *Information Sciences*, Vol. 492, pp.181–203.

Xue, Y., Xue, B. and Zhang, M.J. (2019) 'Self-adaptive particle swarm optimization for large-scale feature selection in classification', *ACM Transactions on Knowledge Discovery from Data*, Vol. 13, No. 5, p.50.

Yan, F.L. (2018) 'Autonomous vehicle routing problem solution based on artificial potential field with parallel ant colony optimization (ACO) algorithm', *Pattern Recognition Letters*, Vol. 116, pp.195–199.

Yu, C.H., Zhang, D. and Lau, H.Y.K. (2016) 'MIP-based heuristics for solving robust gate assignment problems', *Computers & Industrial Engineering*, Vol. 93, pp.171–191.

Zhang, M.Q., Wang, H., Cui, Z.H. and Chen, J.J. (2018) 'Hybrid multi-objective cuckoo search with dynamical local search', *Memetic Computing*, Vol. 10, No. 2, pp.199–208.

Zhang, Q. and Xiong, S.W. (2018) 'Routing optimization of emergency grain distribution vehicles using the immune ant colony optimization algorithm', *Applied Soft Computing*, Vol. 71, pp.917–925.

Zhang, Q. and Zhang, C.S. (2018) 'An improved ant colony optimization algorithm with strengthened pheromone updating mechanism for constraint satisfaction problem', *Neural Computing & Applications*, Vol. 30, No. 10, pp.3209–3220.

Zhang, Y., Gong D.W., Gao, X.Z. et al. (2020) 'Binary differential evolution with self-learning for multi-objective feature selection', *Information Sciences*, Vol. 507, pp.67–85.

Zhao, H.M., Liu, H.D., Deng, W. and Xu, J.J. (2019a) 'Performance prediction using high-order differential mathematical morphology gradient spectrum entropy and extreme learning machine', *IEEE Transactions on Instrumentation & Measurement*, DOI: 10.1109/TIM.2019. 2948414.

Zhao, H.M., Zheng, J.J., Xu, J.J. and Deng, W. (2019b) 'Fault diagnosis method based on principal component analysis and broad learning system', *IEEE ACCESS*, Vol. 7, pp.99263–99272.

Zhao, H.M., Zheng, J.J., Deng, W. and Song, Y.J. (2020) 'Semi-supervised broad learning system based on manifold regularization and broad network', *IEEE Transactions on Circuits and Systems – I: Regular Papers*, DOI: 10.1109/ TCSI.2019.2959886.

Zheng, F.F., Zecchin, A.C., Newman, J.P. et al. (2017) 'An adaptive convergence-trajectory controlled ant colony optimization algorithm with application to water distribution system design problems', *IEEE Transactions on Evolutionary Computation*, Vol. 21, No. 5, pp.773–791.

Zhong, Y.G. and Ai, B. (2017) 'A modified ant colony optimization algorithm for multi-objective assembly line balancing', *Soft Computing*, Vol. 21, No. 22, pp.6881–6894.

Zhu, D.Z., Werner, P.L. and Werner, D.H. (2017) 'Design and optimization of 3-D frequency-selective surfaces based on a multiobjective lazy ant colony optimization algorithm', *IEEE Transactions on Antennas and Propagation*, Vol. 65, No. 12, pp.7137–7149.