Journal of Marine Science and Technology

Volume 31 | Issue 3 Article 10

Path Planning of UAV Navigation Mark Inspection Using a K-means Clustering ACA

Jiaqi Li
School of Navigation, Dalian Maritime University, Lingshui Road, Dalian, Liaoning 116031, China

Weifeng Li
School of Navigation, Dalian Maritime University, Lingshui Road, Dalian, Liaoning 116031, China, ljq210a@dlmu.edu.cn

Wenting Zhang
School of Navigation, Dalian Maritime University, Lingshui Road, Dalian, Liaoning 116031, China

Recommended Citation

Li, Jiaqi; Li, Weifeng; and Zhang, Wenting (2023) "Path Planning of UAV Navigation Mark Inspection Using a K-means Clustering ACA," *Journal of Marine Science and Technology*: Vol. 31: Iss. 3, Article 10.
DOI: 10.51400/2709-6998.2705
Available at: https://jmstt.ntou.edu.tw/journal/vol31/iss3/10

Path Planning of UAV Navigation Mark Inspection Using a K-means Clustering ACA

Acknowledgements

Acknowledgments This work was supported by“The Fundamental Research Funds for the CentralUniversities”(3132023153,3132023154). The views expressed here in are those of the authors and do not necessarily reflect the views of the funding agency.

RESEARCH ARTICLE

Path Planning of UAV Navigation Mark Inspection Using a K-means Clustering ACA

Jiaqi Li, Weifeng Li*, Wenting Zhang

School of Navigation, Dalian Maritime University, Lingshui Road, Dalian, Liaoning 116031, China

Abstract

With increasing speeds and application of artificial intelligence in the shipping industry, unmanned aerial vehicle (UAV) technology has been applied to navigation mark inspections to improve the inspection efficiency and safety. Aimed at the UAV path planning problem of navigation mark inspection, this paper proposes an improved K-means clustering ant colony algorithm (KCACA) to design the shortest route for UAV navigation mark inspections. First, the K-means algorithm and the UAV maximum flight distance were used to cluster the navigation marks, which were then split into several secondary clusters. Each cluster was regarded as an independent traveling salesman problem to be evaluated using the ant colony algorithm (ACA). Second, after optimizing the ACA pheromone update formula, the attenuation factor was gradually reduced according to the number of iterations. Experiments showed that the improved KCACA not only optimized the shortest path but also calculated the optimal path in a short time, improving the operation efficiency.

Keywords: Navigation mark inspection, Unmanned aerial vehicle, K-means clustering ant colony algorithm

1. Introduction

The shipping industry is closely related to our daily lives and social and economic development. As the most important means for global economic trade, shipping considerably impacts world economy. It is an important pillar of China's economic and social development. To ensure the safety of shipping, navigation mark inspection is of the utmost importance [24]. In recent years, with the rapid development of the shipping industry, the number of ships has considerably increased, resulting in higher requirements for navigation mark inspection [25].

With the increasing speeds and application of artificial intelligence in the shipping industry, the period and rate of navigation mark inspection need to be improved [26]. Therefore, navigation mark inspection optimization is required. The development of unmanned aerial vehicle (UAV) technology is evolving, and it is now widely applied in many fields, including navigation mark inspection. UAVs have the advantages of low costs and convenience, overcoming the shortcomings of traditional navigation mark inspection methods, which can be costly and slow.

The traveling salesman problem (TSP), whereby a salesman traverses each target point and returns to the origin through the shortest route, is used for the path planning of UAV navigation mark inspections. Traditional methods used to solve the TSP include the enumeration method, A* search algorithm, ant colony algorithm (ACA), genetic algorithm, artificial bee colony algorithm, and simulated annealing method. The ACA is robust and uses parallelism and a positive feedback function. The ACA is applied to classic optimization problems such as the TSP [2]. The ACA can improve search efficiency and exhibits strong adaptability [23]. However, the traditional ACA is relatively simple; therefore, it easily falls into the local optimal solution, and its implementation process is inefficient. Some scholars have improved the ACA. Wei [3] proposed an ACA based on hybrid behavior. Zhang et al. [4] solved this problem using a

Received 28 April 2023; revised 29 August 2023; accepted 5 September 2023.
Available online 6 October 2023

* Corresponding author.
E-mail address: ljq210a@dlmu.edu.cn (W. Li).

https://doi.org/10.51400/2709-6998.2705

generalized ACA. Gao [5] proposed a new, continuous ACA. Dreo [6] proposed an ACA aimed at dynamic continuous optimization.

The traditional ACA [20] does not combine UAV constraints when performing navigation mark inspection path planning. Therefore, this paper proposes a K-means clustering ACA (KCACA), whereby, combined with UAV characteristics, three-dimensional space is simplified into a two-dimensional space. Moreover, the entire inspection path area is divided into several secondary clusters using K-means clustering such that the path planning can be simultaneously and independently undertaken for each cluster.

2. UAV navigation mark inspection

2.1. UAV navigation mark inspection mode

With the development of UAV technology, a relatively man−machine interaction research system has been established. UAVs [18] are controlled by a wireless remote control device or program. They do not need to be equipped with any pilot-related equipment, effectively saving space; therefore, they can carry more equipment to complete tasks. UAVs have the advantages of simple operation, low price, safety, and high efficiency. Their precise hovering function and high-altitude perspective can observe navigation marks without a dead angle. In the future, it is expected that UAVs will use artificial intelligence and become multifaceted and cost less, which is considerably suitable for their application in navigation mark inspection.

Currently, two modes of navigation mark inspection are used: “ship-UAV” mode refers to the relevant managers carrying a UAV to the target navigation mark using the inspection ship, and then starting the UAV on the ship and “car-UAV” mode refers to reaching the target navigation mark using a car. UAV navigation mark inspection prefers to use the “car-UAV” mode, but if the navigation mark is relatively far from shore, due to the distance limitation between the operating end and the UAV, the “ship-UAV” mode will be selected.

2.2. Farthest UAV flight distance

Since the UAV needs to be operated by an operator, the farthest flight distance of a UAV ($L_{\max}$) refers to the farthest operating distance that the UAV can receive a signal from the operation end. For UAV navigation mark inspection in Region A, the farthest flight distance of a UAV is determined by Formula (1) as follows:

$$L_A = \max\left\|x_f - x_l\right\|, l \neq f \tag{1}$$

where x_f represents the takeoff point, which usually prioritizes the most accessible beacons in Area A, such as lighthouses and light beacon; x_l represents other navigation marks except the takeoff point in Region A; and L_A represents the distance between the takeoff point in Region A and the navigation mark that is the farthest away from the takeoff point.

Since $L_{\max}$ is limited by the signal acceptance range and radio interference, the actual farthest flight distance of the UAV may be considerably lower than the theoretical UAV farthest flight distance. Therefore, it is necessary to ensure that L_A is $\leq L_{\max}$. If not, Region A must be reclustered.

2.3. Maximum UAV flight range

The maximum flight range (d_{max}) of a UAV for a single flight is limited by the power of the UAV and other factors affecting the UAV. To ensure that the UAV can inspect the navigation mark within the path in a single flight, the optimal path (d_A) calculated by the ACA must be $\leq d_{max}$. If not, Region A must be reclustered.

3. Ant colony algorithm

The ACA, a new simulated evolutionary algorithm, was proposed by Dorigo [1]. It was obtained by simulating the foraging behavior of ants in nature. The ACA [19] uses a positive feedback mechanism to quickly converge on a better solution. It has demonstrated superiority in the field of combinatorial optimization and for planning strategies [21].

As a heuristic algorithm, the ACA [16] is primarily used to solve TSPs. The process of solving the TSP mainly iterates the following two steps until the path does not change.

3.1. Urban transition probability formula

Considering [14] that there are n cities to visit, $d_{ij}(i, j = 1, 2, \ldots, n)$ denotes the distance between city i and city j and $\tau_{ij}(t)$ denotes the number of pheromones between city i and city j in the t cycle. The pheromones denote the actual number of secretions of ants. Considering m ants are present, $p_{ij}^{A}(t)$ denotes the probability of the ant A moving from city i to city j in the t cycle, expressed in Formula (2) as follows:

$$p_{ij}^{A}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}(t)\right]^{\beta}}{\sum_{r \notin T_A}\left[\tau_{ir}(t)\right]^{\alpha}\left[\eta_{ir}(t)\right]^{\beta}}, j \notin T_A \\ 0, \text{ others} \end{cases} \tag{2}$$

where $T_A(A = 1, 2, \ldots, m)$, called the taboo list, records the point reached by ant A; $\tau_{ij}(t)$ represents the pheromone concentration of the distance between city i and city j in the t cycle; and $\eta_{ij}(t)$ represents the expected heuristic for each ant from city i to city j in the t cycle. For Formula (2), $\eta_{ij}(t) = 1/d_{ij}$, where $\tau_{ir}(t)$ represents the pheromone concentration of the distance between city i and city r in the t cycle and $\eta_{ir}(t)$ represents the expected heuristic for each ant from city i to city r in the t cycle. For Formula (2), $\eta_{ir}(t) = 1/d_{ir}$, where d_{ir} represents the distance between city i and city r and α represents the pheromone importance factor. When α is large, subsequent ants are more likely to choose the path passed by previous ants, reducing the randomness of path selection of the ants. When α is small, the randomness of the ant selection path is high, enhancing the randomness of the search. However, this slows the convergence speed of the algorithm. β denotes the importance factor of the heuristic information. When β is large, the randomness of the algorithm is weak, rendering it easier to fall into the local optimal solution. When β is small, the randomness of the algorithm is strong but the difficulty of the optimization is high.

3.2. Improved pheromone update formula

When the ants traverse all the cities to complete a cycle, the path that each ant takes is a solution to the TSP; however, it may not be the optimal solution for the shortest path. Therefore, the ACA [17] needs to update the number of path pheromones. This research optimizes the pheromone update formula so that the pheromone number increases less during the initial iteration to ensure randomness, and it increases the pheromone amount in the later period to ensure computational efficiency, as shown in Formula (3) as follows:

$$\tau_{ij}(t+1) = \frac{\tau_{ij}(t) + \Delta\tau_{ij}}{\rho^t} \tag{3}$$

where $\tau_{ij}(t+1)$ represents the number of pheromones in the distance between city i and city j in the $t+1$ cycle; $\tau_{ij}(t)$ represents the number of pheromones in the distance between city i and city j in the t cycle; ρ represents the attenuation factor, which indicates the degree of the attenuation of the information, where $\rho \in (0, 1)$; and $\Delta\tau_{ij}$ represents the amount of information increase, expressed by Formula (4) as follows:

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{A} \tag{4}$$

where $\Delta\tau_{ij}^{A}$ represents the amount of information left by the ant A in the city i and city j in this cycle. This is expressed by Formula (5) as follows:

$$\Delta\tau_{ij}^{A} = \begin{cases} \dfrac{Q}{L_A}, \text{If ant } A \text{ passing path } ij \\ 0, \text{ Others} \end{cases} \tag{5}$$

where Q represents a constant and L_A represents the total length of the path that the ant A travels during this cycle, where $\tau_{ij}(0) = C$.

4. K-means clustering

The ACA exhibits good applicability for solving small-scale TSPs [7–11]. However, when solving large-scale TSPs, it is slow and easily falls into the local optimal solution, resulting in unstable results. In addition, due to flight-time limitations, UAVs [12] cannot inspect all navigation marks. Therefore, it is necessary to divide regions according to the position of the navigation marks. To solve large-scale TSPs, the most effective method for dividing a region is K-means clustering.

4.1. Principle of K-means clustering

K-means clustering [13] divides a sample set into K clusters in advance based on the distance between the samples. This clusters the sample points as closely as possible, making the boundaries between the clusters as obvious and clear as possible. K-means clustering [15] is relatively scalable and efficient for large-scale data sets; thus, it is reasonable to use K-means clustering [17] to cluster large-scale TSPs.

4.2. Silhouette coefficient

As an unsupervised knowledge discovery algorithm, K-means clustering [16] needs to determine the number of clusters (K) in advance. Therefore, prior to clustering, it is necessary to analyze the validity of different K values [22]. The clustering effect is usually judged by cohesion between the points and the separation between the points. A good clustering result should show small cohesion and larger separation between the points. The silhouette coefficient index, which combines cohesion within a class and separation between the points, is an effective criterion for determining K.

Assuming that sample x_i gathers to cluster A, the silhouette coefficient is determined by Formulas (6) and (7) as follows:

$$S_i = \frac{b_i - a_i}{\max(a_i, b_i)} \tag{6}$$

$$S_k = \frac{1}{n}\sum_{i=1}^{n} S_i \tag{7}$$

where S_i represents the silhouette coefficient of point i, and for K clusters, it is denoted as $C_k (k = 1, 2, ..., K)$; point i is clustered to C_1 by the clustering calculation; a_i represents the average distance between point i and the other point in C_1; $D(x_i, C_k)$ represents the average distance between point i and the other point in C_k, $b_i = \min_{k \neq 1} D(x_i, C_k)$; x_i represents point i; n represents the number of the point; S_k represents the population mean of S_i, S_k denotes the silhouette coefficient of the clusters, where $S_k \in (-1, 1)$. Thus, the K of maximum of S_k is selected as the final number of clusters.

4.3. K-means clustering model

Considering that initial $K(C_1, C_2, ..., C_k)$ clusters are randomly selected, the centroids of the initial clusters are represented as $m_1^{(0)}, m_2^{(0)}, ..., m_k^{(0)}$. The distance from each point to each centroid is calculated, and each point is divided into the nearest cluster to minimize the sum of squares of the Euclidean distances from each point to the centroid of the cluster, expressed in Formula (8) as follows:

$$\arg\min_{S^{(t)}} = \sum_{k=1}^{K} \sum_{x_i \in C_k^{(t)}} d\left(x_i, m_k^{(t)}\right) \tag{8}$$

where $d(x_i, m_k^{(t)}) = \left\| x_i - m_k^{(t)} \right\|^2$, indicating the sum of the square of the Euclidean distance from point i to the centroid of the cluster that point i belongs to; t represents the number of iterations; $S^{(t)}$ represents S in the t cycle; $\arg\min_{S^{(t)}}$ represents the function of finding the parameter (set) of the function, and it can calculate $m_k^{(t)}$ when $S^{(t)}$ is the minimum; $m_k^{(t)}$ represents the centroid of the clusters in the t cycle; and $C_k^{(t)}$ represents the clusters in the t cycle.

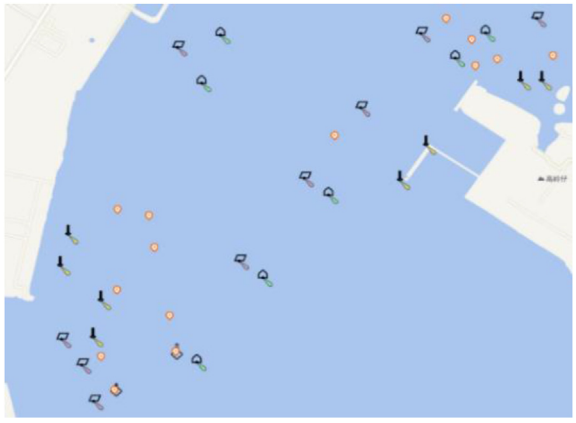


Fig. 1. Experimental region.

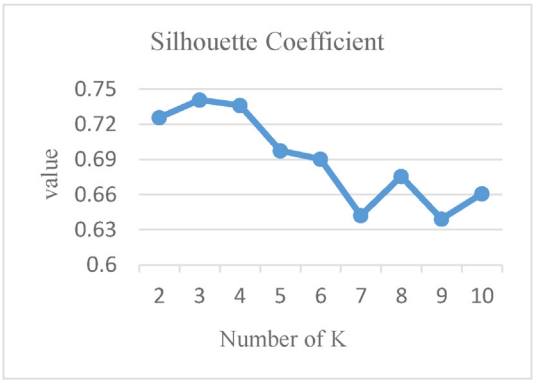


Fig. 2. Silhouette coefficient.

According to the new cluster after division, the average distance between cluster nodes is recalculated as the new cluster particle, expressed in Formula (9) as follows:

$$m_k^{(t+1)} = \frac{1}{\left|C_k^{(t)}\right|}\sum_{x_i \in C_k^{(t)}} x_i \tag{9}$$

where $m_k^{(t+1)}$ represents the new centroid of the clusters and $\left|C_k^{(t)}\right|$ represents the number of the point in $C_k^{(t)}$.

The iteration shown in Formulas (4) and (5) is repeated until the objective function error Z reaches the minimum, as shown in Formula (10) as follows:

$$Z = \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - m_k\|^2 \tag{10}$$

where m_k represents the centroid of the clusters and C_k represents the clusters.

When the error reaches the minimum, clustering is completed, and the centroid of each cluster and

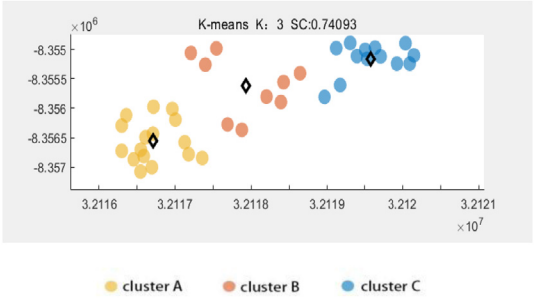
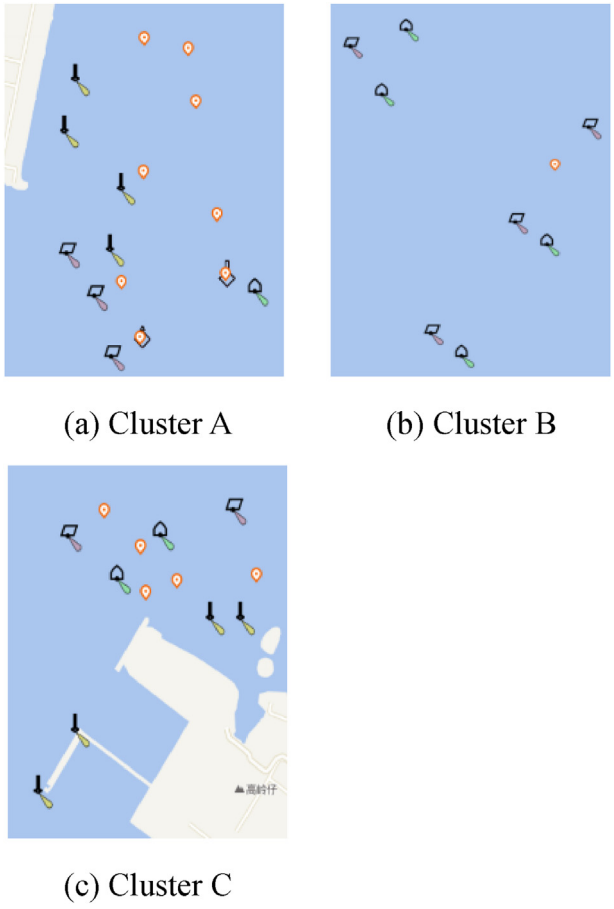


Fig. 3. K-means clustering effect.



(a) Cluster A (b) Cluster B

(c) Cluster C

Fig. 4. Clusters determined by K-means clustering.

the cluster will not change. The whole region can then be divided into *K* regions for path planning.

5. Experimental results and analysis

MATLAB R2022a was used as the simulation test platform on a computer configured with a Windows 10 system, an Intel Core i7-10750H processor, a 2.60 GHz frequency, and a 16 G running memory.

In the simulation experiment, the UAV path planning navigation mark inspection used a region with 38 navigation marks positioned in a port, as shown in Fig. 1.

This experiment used a consumer-grade, civil, multi-rotor UAV (Inspire 2; DJI) with a pan-tilt camera (ZENMUSE X5 S; DJI) that was relatively simple to operate. It had a theoretical flight time in a windless environment of 25 min, a farthest flight distance of 7 km, and a maximum flight range of 30 km.

The latitude and longitude of the navigation marks were collected and converted into Cartesian coordinates and subsequently processed using K-means clustering. K was iteratively calculated for100 times from 2 to 10. The K with the maximum overall average contour coefficient was selected as the number of clusters. The silhouette coefficient is shown in Fig. 2.

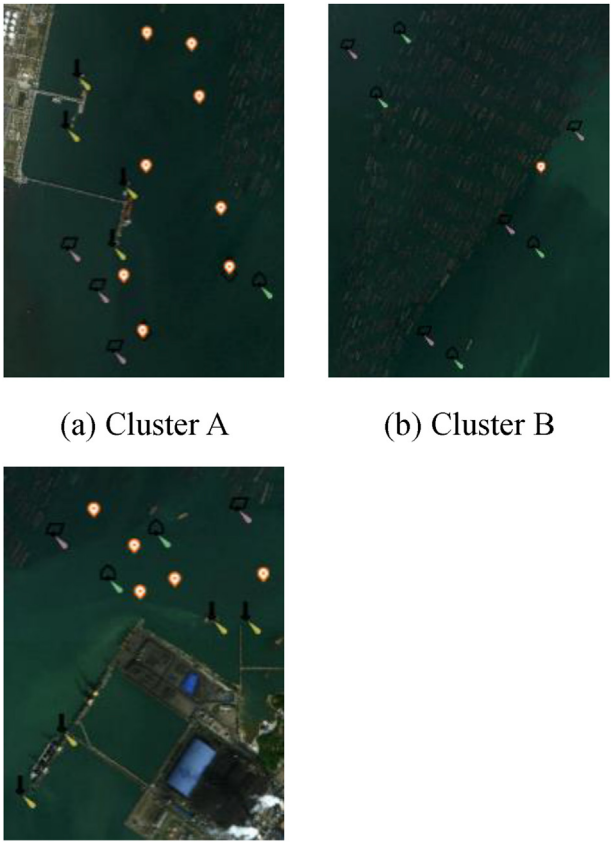
As shown in Fig. 2, the maximum silhouette coefficient (0.74093), indicating the best clustering effect, was obtained for K = 3. Therefore, K = 3 was selected for the K-means clustering. The K-means clustering effect is shown in Fig. 3.

Using K-means clustering, the navigation marks in the region were divided into three clusters: Cluster A, Cluster B, and Cluster C (Fig. 4).

Since the three areas were close to the shore, "car-UAV" mode was selected for the UAV navigation mark inspection. The point closest to shore, determined using satellite images (Fig. 5), was selected as the UAV takeoff point.

Cluster A was selected as the UAV takeoff point due to the presence of an onshore light pile (21°35′17.20″N, 108°21′33.60″E) that could be reached by car.

Since the navigation marks in Cluster B could not be reached by car, it was necessary to select an onshore landing point near the navigation marks as



(a) Cluster A (b) Cluster B

(c) Cluster C

Fig. 5. Cluster satellite maps.

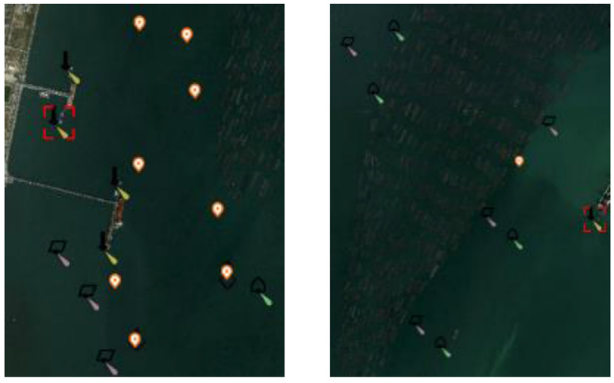
the UAV takeoff point. The light pile (21°35′39.20″N, 108°23′0.08″E) from Cluster C was chosen as the takeoff point because it was the closest.

Because there was an onshore light pile that could be reached by car in Cluster C, and the takeoff point of Cluster B was also in Cluster C the same takeoff point was chosen (21°35′39.20″N, 108°23′0.08″E).

The takeoff points are shown in Fig. 6.

The maximum theoretical signal acceptance distance of the Inspire2 UAV was 7 km; however, because the actual operation was affected by radio interference from electronic equipment at the terminal, during the flight experiment, the signal became poor at a distance greater than 2 km. The UAV automatically returned to avoid the consequences of poor operation, such as crashing. To ensure that the UAV could safely complete the navigation mark inspection task, the navigation mark with the farthest clustering area was inspected, as shown by the red lines in Fig. 7.

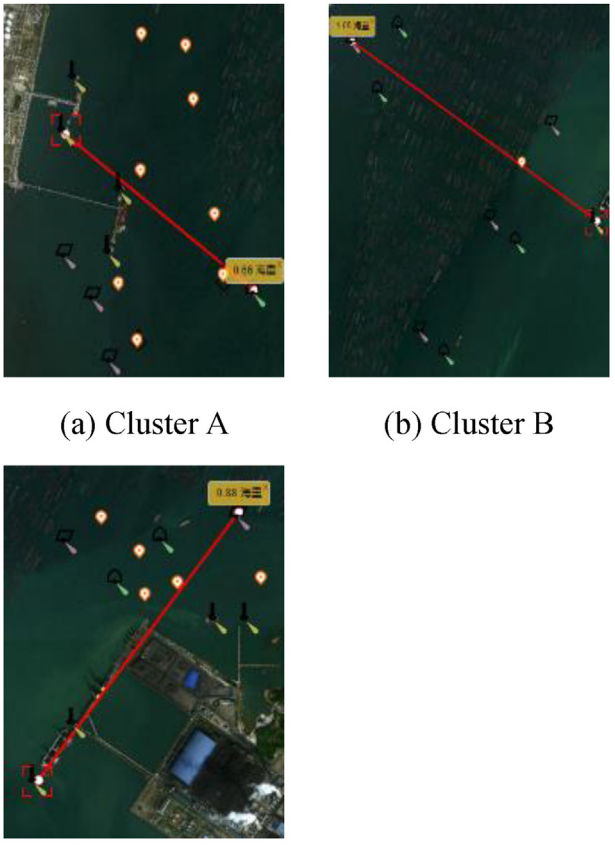
The farthest flight distances were calculated using the international unified mile conversion formula (1 mile = 1852 m), as shown in Table 1.



(a) Cluster A (b) Cluster B

(c) Cluster C

Fig. 6. Cluster takeoff points (shown in red).



(a) Cluster A (b) Cluster B

(c) Cluster C

Fig. 7. Farthest clustering points.

Table 1. UAV farthest flight distances.

Cluster	Farthest flight distance (mile)	Farthest flight distance(m)
A	0.66	1222.32
B	1.05	1944.60
C	0.88	1629.76

As shown in Table 1, the farthest flight distance for each cluster was <2 km, ensuring good UAV signal acceptance.

The path planning was then calculated using the ACA. The algorithm parameter settings are shown in Table 2, where *NC_MAX* denotes the maximum number of iterations.

Table 2. Ant colony algorithm parameter.

Parameter	Value
NC_MAX	2* navigation mark
m	50
α	1
β	5
ρ	0.1
Q	20

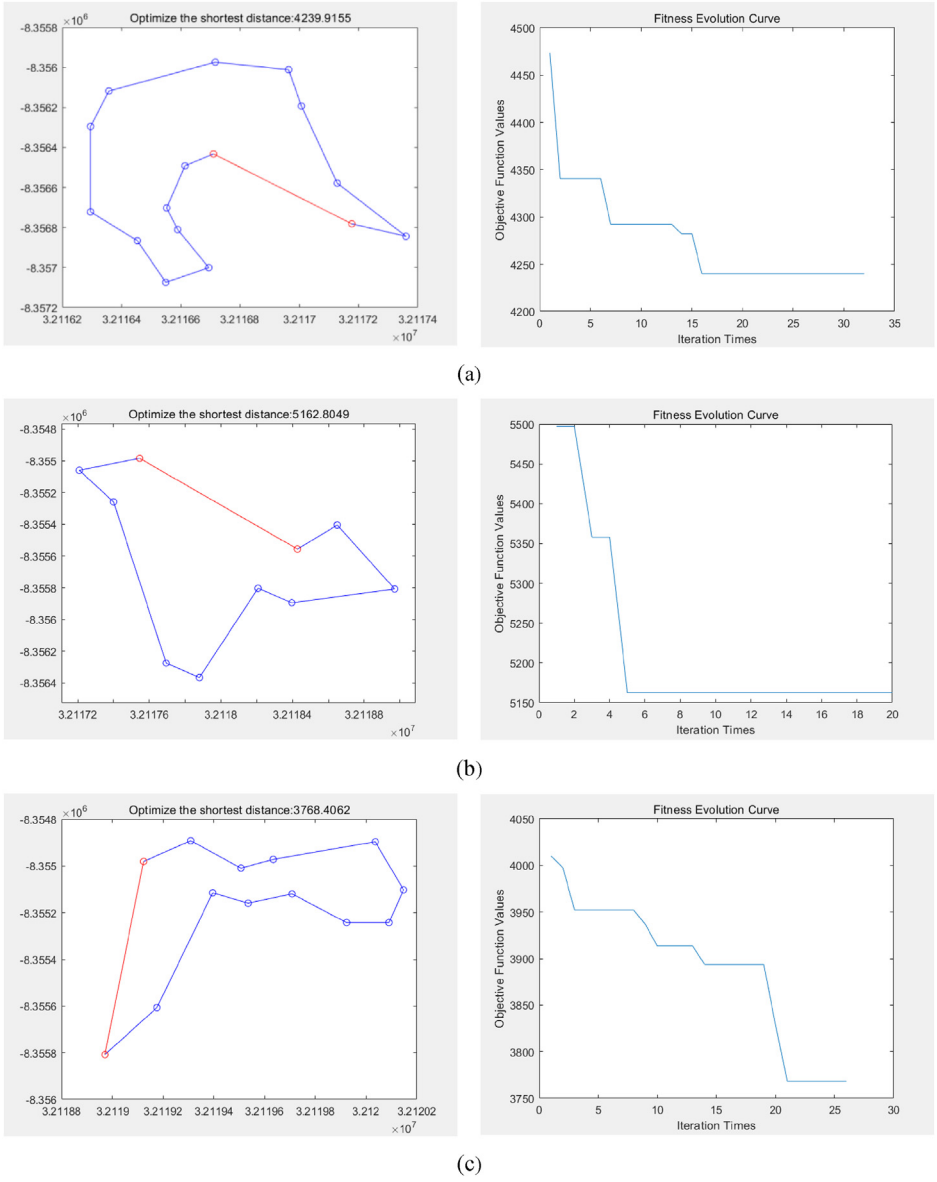


Fig. 8. Shortest paths and fitness evolution curves.

For the experiment, twice the number of navigation marks in the clusters was selected; i.e., 32 times the Cluster A iteration, 20 times the Cluster B iteration, and 26 times the Cluster C iteration. α denotes the pheromone importance factor, β denotes the heuristic factor importance factor, ρ denotes the attenuation factor, and Q denotes the constant coefficient.

The experiment was programmed according to the parameters shown in Table 2, and the three clusters were calculated following the K-means clustering. The experimental results are shown in Fig. 8.

As shown in Fig. 8, the use of K enabled optimal path planning for the UAV navigation mark inspection. The experiment used the standard ACA, the standard KCACA, and the changed KCACA to plan the UAV navigation mark inspection 100 times. The average range and times for the UAV navigation mark inspection are shown in Table 3.

The optimized KCACA had a fast evolution speed was fast and the number of iterations used was small. Therefore, it has good effect and practicability in path planning.

Table 3. Comparison of the three algorithm times and ranges.

Algorithm	Time(s)	Range(km)
ACA	5.635	16.495
KCACA	3.298	13.171
CKCACA	2.422	13.171

6. Conclusion

Aimed at solving the path planning problem encountered by UAV navigation mark inspections, this paper proposes a changed KCACA. The algorithm is based on the idea of K-means clustering, whereby the path planning region containing the navigation marks is clustered to avoid long total planned-path distances. The ACA was optimized to improve the pheromone update formula. The amount of pheromone in the previous iteration was reduced, and the amount of pheromone in later iterations was increased to avoid the algorithm falling into local optimal solutions. The positive feedback mechanism of the ACA was then used to carry out the UAV navigation mark inspection path planning. Experiments showed that this method quickly obtained optimal paths, proving that it is feasible and has a good effect. Therefore, this combination of theory and practice can scientifically and reasonably plan the paths of UAV navigation mark inspections.

However, this study has some limitations. The batteries of consumer-grade, civil UAVs can only maintain short flight times, and they consume additional power when operating in high winds and for low-temperature battery heating. They may also have to hover when navigation marks are in doubt. Therefore, actual flight distances need to be verified. These limitations will be experimentally studied in depth in the future.

Conflict of interest

All authors disclosed no relevant relationships.

Acknowledgments

This work was supported by "The Fundamental Research Funds for the Central Universities"(3132023153,3132023154). The views expressed herein are those of the authors and do not necessarily reflect the views of the funding agency.

References

[1] Dorigo M, Maniezzo V, Colorni A. Ant system: optimization by a colony of cooperating agents. IEEE Trans Syst Man Cybern B Cybern 1996;26(1). 49-41.
[2] Cooper J, Nicolescu R. The Hamiltonian cycle and travelling salesman problems in cP systems. Fundam Inf 2019;164(2−3): 157−80.
[3] Wei XM. Hybrid behavior ant colony algorithm. Adv Mater Res 2012;433−440:4496−9.
[4] Zhang DY, Fu P. Robot path planning by generalized ant colony algorithm. Appl Mech Mater 2014;494−495:1229−32.
[5] Gao W. New continuous cnt colony clgorithm. 7th World Congress on Intelligent Control and Automation 2008;1−23: 1280−4.
[6] Dreo J, Siarry P. An ant colony algorithm aimed at dynamic continuous optimization. Appl Math Comput 2006;181(1): 457−67.
[7] Wang XY, Bai YP. The global Minmax k-means algorithmvol. 5. Springerplus; 2016.
[8] Li M, Xu DC, Zhang DM, Zou J. The seeding algorithms for spherical K-means Clustering. J Global Optim 2020;76(4): 695−708.
[9] Laporte G, Semet F. Computational evaluation of a transformation procedure for the symmetric generalized traveling salesman problem. Inform Rev 1999;37(2):114−20.
[10] Longani V. Another approach for the traveling salesman problem. Appl Math Comput 2000;114(2−3):249−53.
[11] Takashima Y, Nakamura Y. Theoretical and experimental analysis of traveling salesman walk problem. In: IEEE Asia Pacific Conference on Circuits and Systems (APCCAS 2021) & 2021 IEEE Conference on Postgraduate Research in Microelectronics and Electronics (PRIMEASIA 2021), 2021; 2021. p. 241−4.
[12] Cao ZY. Optimization Design of multi-UAV communication network based on reinforcement learning. Wireless Commun Mobile Comput (Online) 2022. https://doi.org/10.1155/2022/7726338.
[13] Zhao B, Zhu ZX, Mao ER, Song ZH. Image segmentation based on ant colony optimization and K-means Clustering. In: IEEE International Conference on Automation and Logistics, 1−6; 2007. p. 459−63.
[14] Wang Y, Zhang W. A novel clustering method with ants. Prog Intell Comput Appl 2007:524−6.
[15] Wang XM, Wang JB. Improved artificial bee colony clustering algorithm based on K-means. Appl Mech Mater 2014; 556−562:3852−5.
[16] Ju ZW, Chen JZ, Zhou JL. Image segmentation based on edge detection using K-means and an improved ant colony optimization. Proceedings of 2013 International Conference on Machine Learning and Cybernetics (ICMLC) 2013;1−4:297−303.
[17] Reddy TN, Supreethi KP. Optimization of K-Means algorithm: ant colony optimization. In: International Conference on Computing Methodologies and Communication (ICCMC), 2017; 2017. p. 530−5.
[18] Cai Y, Juang JG. Path planning and obstacle avoidance of UAV for cage culture inspection. J Mar Sci Technol 2020; 28(5):444−5.
[19] Tsou MC, Hsueh CK. The Study of ship collision avoidance route planning by ant colony algorithm. J Mar Sci Technol 2010;18(5):746−56.
[20] Gao S, Zhang ZY, Zhang XR, Cao CG. A new hybrid ant colony algorithm for clustering problem. In: International workshop on education technology and training and 2008 international workshop on geoscience and remote sensing, 2008; 2008. p. 1−645.
[21] Hsu CC, Wang WY, Chien YH, Hou RY. FPGA Implementation of improved ant colony optimization algorithm based on pheromone diffusion mechanism for path planning. J Mar Sci Technol Taiwan 2018;26(2):170−9.
[22] Sha M, Yang HX. Speaker recognition based on APSO-K-means clustering algorithm. In: International Conference on Artificial Intelligence and Computational Intelligence, 2009; 2009. p. 440.
[23] Saatchi S, Hung CC. Hybridization of the ant colony optimization with the K-means algorithm for clustering. Lecture Notes in Artificial Intelligence 2005;3540:511−20.
[24] Wang H, Liu Z, Wang X, Graham T, Wang J. An analysis of factors affecting the severity of marine accidents. Reliab Eng Syst Saf 2021:210.
[25] Wang X, Xia G, Zhao J, Wang J, Yang Z, Loughney S, et al. A novel method for the risk assessment of human evacuation from cruise ships in maritime transportation. Reliab Eng Syst Saf 2023:230.
[26] Liu J, Schonfeld PM, Zhan SG, Du B, He MW, Wang KCP, et al. The economic value of reserve capacity considering the reliability and robustness of a rail transit network. J Transp Eng A Syst 2023:149.