VILNIUS TECH
Vilnius Gediminas Technical University

AVIATION
ISSN: 1648-7788 / eISSN: 1822-4180
2021 Volume 25 Issue 3: 211–219
https://doi.org/10.3846/aviation.2021.14216

UNMANNED AIR VEHICLE PATH PLANNING FOR MARITIME SURVEILLANCE USING CLUSTER-BASE METHOD

Prasetyo Ardi Probo SUSENO*, Try Kusuma WARDANA

Indonesian National Institute of Aeronautics and Space (LAPAN), Indonesia

Received 11 May 2020; accepted 27 November 2020

Abstract. This paper discusses a method to determine the operation route for unmanned aerial vehicles for maritime surveillance. It is well known that there are several methods to make an aircraft path planning for ground related missions. On the other hand, path planning for maritime purposes is unnoticeable. The major problem of path planning for maritime is the abundant number of nodes which can make the route becomes quite long. Hence, reducing the number of nodes is necessary to rectify this problem. The main method is to separate the surveillance area into a smaller area of operation using clustering methods and then analyze the vulnerable area using the database to create an optimum flight path in each operation area. Although this paper specifically addresses a maritime-related mission, the path planning procedures can be applied to other missions as well. In this research, the input is given from satellite recorded data. Natuna Sea is chosen as the main discussion as the Natuna Sea currently is one of the most vulnerable regions in Indonesia for illegal fishing activity. The result shows that the aircraft path able to cover most of the vulnerable areas while optimizing the route distance.

Keywords: UAV, path planning, surveillance, maritime, clustering, TSP, K-means, nearest neighbour.

Introduction

Marine resource is a vital element for every maritime country. The Sea border has been an area that always becomes controversial and difficult to monitor due to it is difficult to access. The Natuna Sea is one of the apparent examples where illegal fishing occurs frequently. Natuna Sea poses as an Indonesian boundary with many countries in the South Asian nation. The strategic position and the natural resource its contain leads to international dispute for so many years. Even China claims the Natuna sea as part of its territorial waters.

There are many ways to supervise marine borders. The patrol ship and satellite surveillance is a common way used in many countries. However, both methods have their issue. Patrol ship is costly in both procurement and operational cost but can to confront the suspect directly. Satellite is another option to watch the border. It has a bigger area of supervision although the data may be captured only at a certain time. Both ways have been implemented collaboratively to strengthen border surveillance. However, the amount of illegal, unreported, and unregulated fishing activity suggests that the current method is not enough to prevent illegal fishing.

The utilization of aircraft in maritime surveillance is a trending topic in recent years. Research on Unmanned Aerial Vehicles (UAV) as a real-time monitoring platform has been done in full (Jeon et al., 2019; Kazantsev et al., 2016). There are several advantages for UAV which make it an effective tool for maritime surveillance such as low cost, good flexibility, low risk, and high efficiency (Duan & Zhang, 2014). A study about preventing pirates attacks using UAV is also mentioned in Watanabe et al. (2017). Those are a great support to improve maritime surveillance mission.

Maritime surveillance is generally a harder mission than a land oriented mission. The main major difference is the lack of landing sites, thus the UAV is forced to fly with perfect management. The distance to the ground station also become a problem as it is harder to seek for support, especially for communication and tracking purpose. Geographical location, as well as climate characteristics, are factors that have to be considered to choose the aerial vehicle for maritime missions (Klimkowska et al., 2016).

The purpose of this paper is to create a method to make a flight path for UAV which is suitable for maritime surveillance purposes. The flight path is created based on

*Corresponding author. E-mail: *susenoprasetyo@yahoo.com* the historical position data of the ships that were doing fishing activities. As the main target is to identify the illegal ships, therefore the more the ship it can find the better it is. Depending on the UAV's payload and capability, we can then capture some pictures or videos of the illegal activity happening in the area and give an appropriate response. If the payload system is good enough we can try to capture the hull number, for it can be used as evidence in the international court. Furthermore, at the very least we can get a deterrent effect on the related area.

1. Methodology

The Natuna Sea is very vast. It is not advisable to operate a single UAV over such a wide area, and a more reasonable approach is to separate the Sea into several area operations. The area operation is the area inside the operation boundary which has the potential for illegal fishing activity based on the ship's position data. This operation area geometry is simplified as a circle and defined as a radius with its center is called as operation base. Operation base is the location recommended for ship carrier to standby, because, from this position, the ship will have the best coverage to provide communication. Besides, the ship can react faster to confront the suspect or to recover the UAV in case the UAV runs into failure and fall. For each operation area, a flight mission is created independently, then the flight path is created and the distance for each mission can be estimated to analyze the mission feasibility.

Hereinafter, to simplify the problem, the area of interest is simplified as a rectangular area with geographical limits are given in the equations below. Here Y is the ship's latitude position in degree, and X is the ship's longitude in degree.

$$3° \text{ N} < \text{Y} < 7° \text{ N}; \tag{1}$$

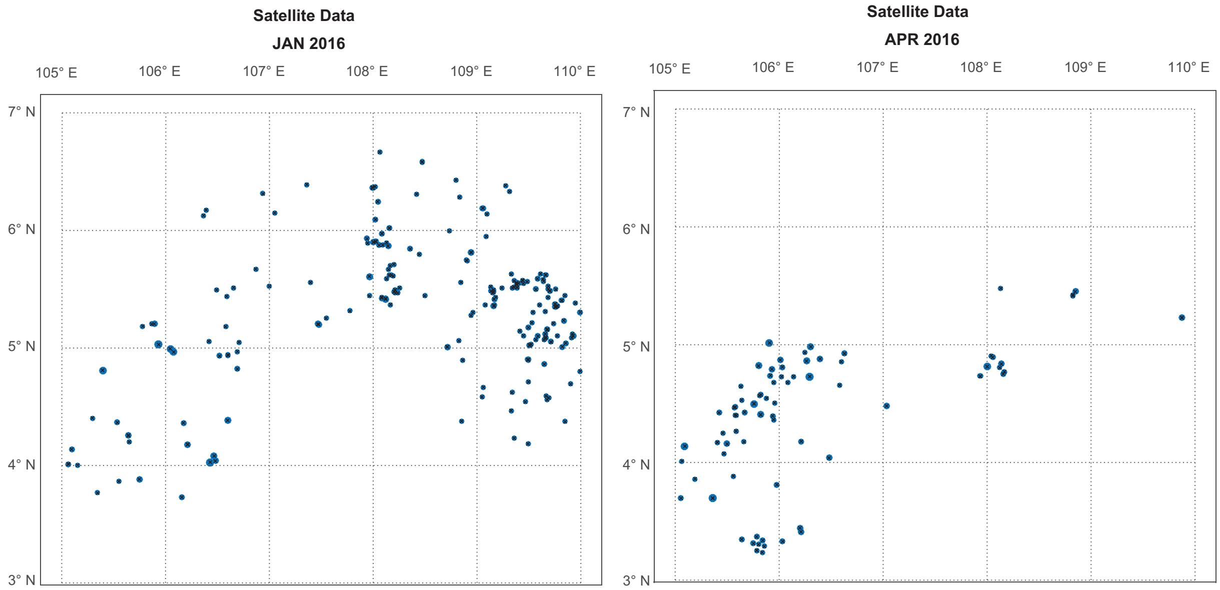
$$105° \text{ E} < \text{X} < 110° \text{ E}. \tag{2}$$

The database is acquired from satellite Radarsat-2 with sample time in the 2016 period. Radarsat-2 satellite is an earth observation satellite for all weathers with full polarimetric imaging capability (Moon et al., 2010). This satellite is a continuation mission from Radarsat-1 which is designed to guarantee the supply continuity of radar data such as ice mapping at sea and monitoring of maritime area (Singhroy & Charbonneau, 2014). Radarsat-2 has orbit at 798 km and encircled the earth 14 times a day with cycle repetition at 24 days (Livingstone et al., 2005).

Considering the typical fishing ships in this area are of a particular size. We can filter the data according to the size of the ships as given in the equation below. According to the Operation Section Head of Control Center I General Directorate of Observation SDKP, Adi Wicaksono, the illegal ship operating in Indonesia, in general, are sized between 15 m to 50 m. The bigger ships usually are either tanker ships or transport ships.

Figure 1 shows the suspect position in several months in the Natuna Sea. The blue dots which denoted the ships' positions are captured several times on a different date following the satellite cycle. It can be seen that the fishing activity is moving over time; therefore, a fixed separation is relatively ineffective to be used.

Now if we take the data from January 2016, for example, we can separate the area into several operation areas and set the mission for each area. Figure 2 shows an illustration of how the Sea is separated into three operation areas with their flight path. The overall mission, however, will depend on the resource such as how many ships or UAV that we have. An example of the operation would be a single cruise ship that goes from the main port while carrying a UAV and moving into each operation base and goes standby for each operation. If we have enough resources, then it is possible to assign the operation area to the corresponding ships and do a parallel mission.


Satellite Data JAN 2016

Satellite Data APR 2016

Figure 1. *To be continued*

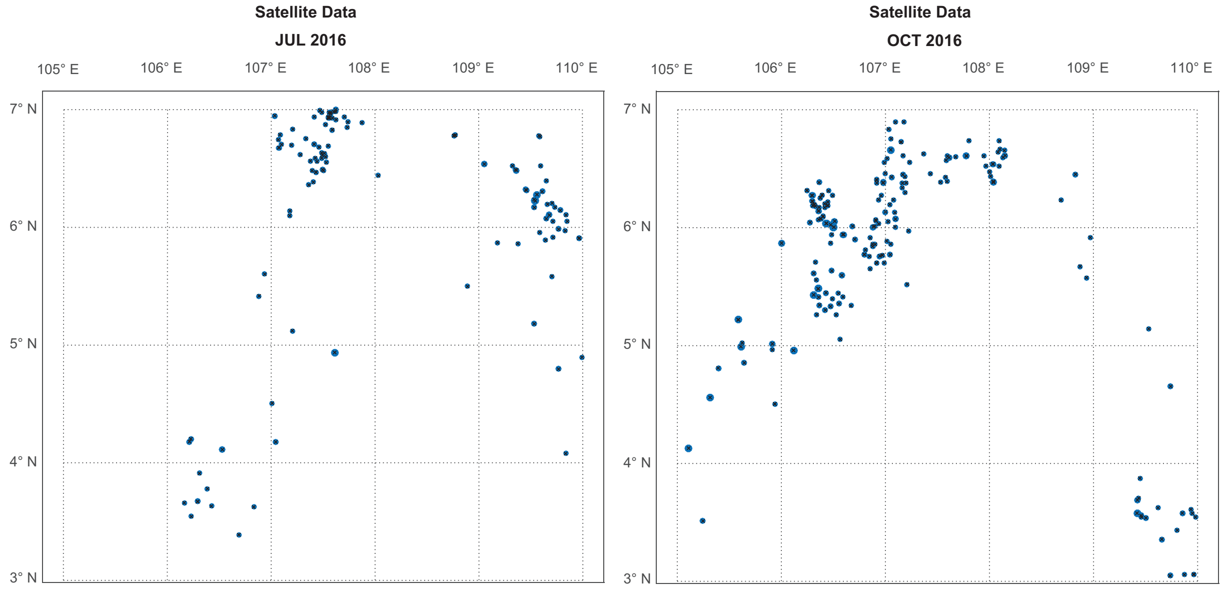


Figure 1. Ships doing fishing activities in Natuna Sea over years

1.1. Determining operation area

The method to create the flight path for the UAVs, in this paper including the clustering technique and determining the base position, hence the name "cluster-base" methods. Clustering is a grouping process of a set object so that the object in the same group tend to have similar identity or parameters than the object in another group. In this paper, clustering is used to solve two problems. One is to separate the mission area into several operation areas. The second is to determine the UAV's waypoint by calculating the weight point of several ships position adjacent to each other. Several clustering techniques are shown in Table 1 along with its traits and characteristics.

There are several things to consider which clustering method should be used. In a maritime surveillance case using UAV's, communication and data transfers are a necessary element to be considered. If we assume the UAV's used are using Line of Sight (LOS) communication, then, a close ground control station is needed. Usually, the carrier ships will make up for this. Nevertheless, now we have to make sure the UAV's to be always in the communication range. It will be challenging to determine the operation area based on the carrier ship's movement. However, if we set the carrier ships to standby on one position and set the take-off and landing position here, then we only need to make sure the communication radius capability bigger than the radius of the operation area. This way, the clusters that we make preferably are to have a similar size.

Among the general clustering techniques shown in Table 1, the centroid-based clustering is the only one that capable to consistently produce clusters with similar size. That is why the clustering method used in this paper is based on the K-clustering model, which is one of the commonly used centroid-based clustering techniques. However, some modifications are made to suit our use.

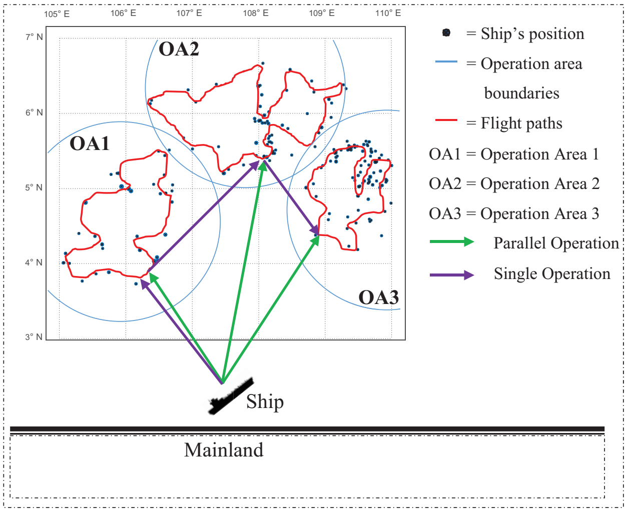


Figure 2. Illustration of the overall mission

Table 1. General technique on clustering

Hierarchical clustering	Centroid-based clustering	Distribution-based clustering
Connect objects to form clusters based on their distance	A central vector represents clusters	Clusters are defined as objects belonging most likely to the same distribution
Not very robust towards outliers	Most algorithms require the number of clusters to be specified in advance	Produces clusters which assume concisely defined mathematical models underlying the data
Slow computing time for a large data set	The algorithms prefer clusters of approximately similar size	May have multiple clusters per data point.

Instead of specifying the number of clusters, we specify the radius of the cluster. This method has been explored in previous research (Suseno & Wirawan, 2019).

1.2. Clustering

K-means progressed clustering

K-means progressed clustering is a means to produce an indefinite number of clusters. The terms progressed comes from its unique traits which specify the initial number of the cluster as one, but increased for every loop, compared to the standard K-means method where the number of clusters usually is set as an input parameter. Another notable characteristic of the K-means progressed clustering is the cluster number becomes the output and cannot be specified. In return, the radius parameter will be used to limit the cluster size. Every member outside of the set radius will be taken out and forced to create a new cluster. This will prevent the member that has been released to re-enter the previously visited cluster and make an endless looping. This algorithm is repeated until there is no member which does not belong to any cluster. Like other centroid based clusterings, this method will make the radius cluster the same for every cluster. The clustering algorithms are given below.

Table 2 shows an example of how the results from K-means progressed clustering in comparison with other methods. The detailed algorithm for hierarchical clustering and K-medoids method can be found in Murtagh and Contreras (2011) and Park and Jun (2009).

The results from all methods seem similar, but in both hierarchical and K-medoids, we need to specify the number of clusters. Furthermore, the K-cluster progressed produce better results to be grouped into a circular geometry because each member simply belongs to the closest centroid.

1.3. Generating waypoints

After all operation area has been determined, the next step is to create an appropriate flight path for the UAVs. The flight path is created by setting the waypoints related to the satellite data. We should notice that the data used is cumulative historical data. Visiting the exact location of the ships will be pointless. This holds, even if the data is taken briefly before the mission starts, as the ships would keep moving. As shown in Figure 3, rather than visiting the blue dot (exact position of the ships when captured by satellite), it is better to set the waypoints into the vulnerable point that frequently visited (which not necessarily passed through the dot). The weight points of adjacent data defined this point, and we will call this vulnerable point as *VP*. To do this, we need to use the clustering

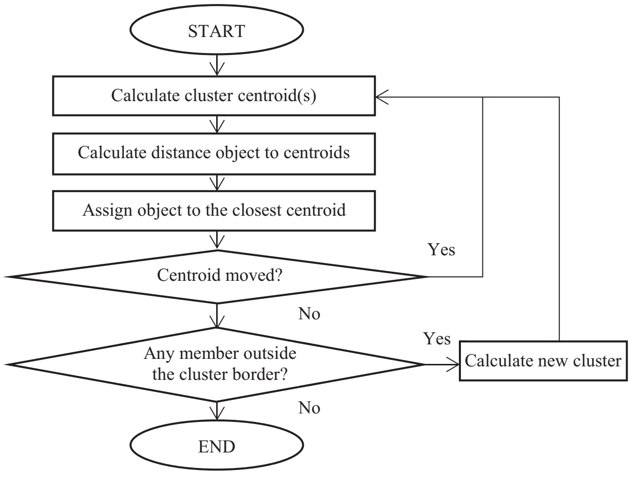
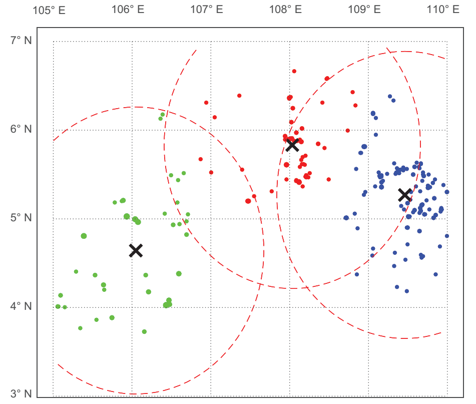
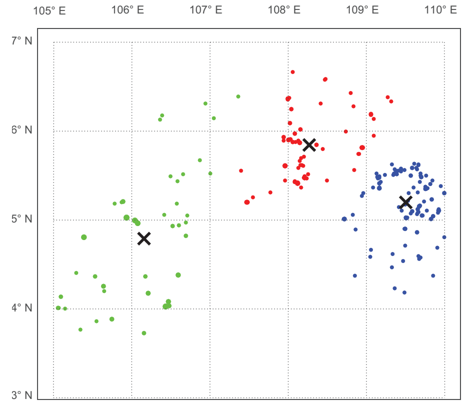
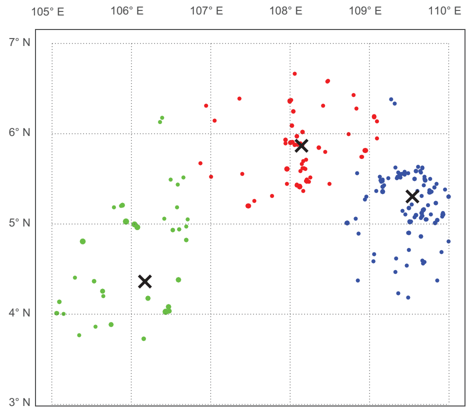

START
Calculate cluster centroid(s)
Calculate distance object to centroids
Assign object to the closest centroid
Centroid moved?
Yes
No
Any member outside the cluster border?
Yes
Calculate new cluster
No
END

Figure 3. K-Means progressed clustering flowchart

Table 2. Comparison of clustering methods

Clustering methods	Results
K-Means Progressed	
Hierarchical Clustering	
K-Medoids	

technique once again but on a much smaller scale. The adjacent points would only be visited once. This way, the flight path will be shorter and the fuel can be reduced. We also need to consider the take-off/landing location. As we mentioned before, the ships will be set to standby at the operation base (the center of the operation area); this is shown as a cross mark in Figure 4.

For an arbitrary operation area K with a total number of ships l, let S be the ship's position and n is the number of mini-cluster in this operation area. H is a set of mini-cluster with h_i contains an m_i amount of ships. We can write the mathematical formulation as follows:

$$S=\left\{s_1,s_2,\dots,s_l\right\} \quad l\in N\,; \tag{3}$$

$$H=\left\{h_1,h_2,\dots,h_n\right\} \quad n\in N\,; \tag{4}$$

$$\sum_{i=1}^{n} m_i = l \quad m_i\in N\,. \tag{5}$$

From the clustering process we can map each ship s_i into mini-cluster h_i as written:

$$h_i=\left\{s_{ij}\right\} \quad 1\le i\le n;\ \ 1\le j\le m_i\,. \tag{6}$$

Label the VP with the numbers 1, …, n. VP_i is the centroid for mini-cluster h_i which can be calculated as follows:

$$VP_i=\frac{\sum_{j=1}^{m_i} s_{ij}}{m_i}\,. \tag{7}$$

The problem now is how to find the shortest possible route that visits each VP and returns to the operation base. This problem is similar to the traveling salesman problem (TSP) and has been formulated in full (Miller et al., 1960). Now define:

$$x_{ij}=\begin{cases}1 & \text{the path goes from VP}i\text{ to VP}j\\ 0 & \text{otherwise}\end{cases}\,. \tag{8}$$

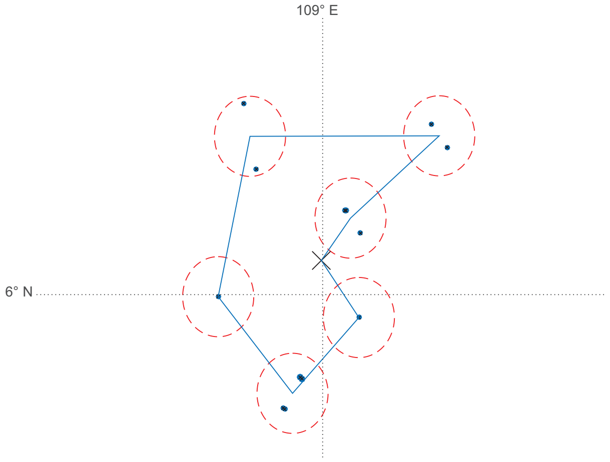


Figure 4. Grouping adjacent points to reduce waypoints and produce shorter flight path

For i = 1, …, n, let u_i be a dummy variable, and finally take c_{ij} to be the distance from VP i to VP j. Then the problem can be written as the following integer linear programming problem:

$$\min \sum_{i=1}^{n} \sum_{j\ne i,j=1}^{n} c_{ij}x_{ij}: \tag{9}$$

$$x_{ij}\in\left\{0.1\right\} \quad i,j=1,\dots,n; \tag{10}$$

$$u_i\in Z \quad i=2,\dots,n; \tag{11}$$

$$\sum\nolimits_{i=1,i\ne j}^{n} x_{ij}=1 \quad j=1,\dots,n; \tag{12}$$

$$\sum\nolimits_{j=1,j\ne i}^{n} x_{ij}=1 \quad i=1,\dots,n; \tag{13}$$

$$u_i-u_j+nx_{ij}\le n-1 \quad 2\le i\ne j\le n; \tag{14}$$

$$0\le u_i\le n-1 \quad 2\le i\le n. \tag{15}$$

The first set of equalities requires that each VP is arrived at from exactly one other VP, and the second set of equalities requires that from each VP there is a departure to exactly one other spot. The last constraints enforce that there is only a single tour covering all the VP, and not two or more disjointed tours that only collectively cover all VP. There are many known heuristics and exact algorithm to solve this problem. In this paper, we use the Nearest Neighbour (NN) heuristic algorithm.

NN algorithm chooses the nearest unvisited *VP* until all *VP* are visited. This algorithm is capable of producing effectively short routes with a short amount of time. On average, for N cities randomly distributed on a plane, this algorithm yields a path 25% longer than the shortest possible path (Johnson & McGeoch, 1997). Given a set S of points in a space M, the NN algorithm has a running time of $O(dN)$, where N is the cardinality of S, and d is the dimensionality of M. Although the *NN* algorithm only yields a suboptimal route, it is however much faster than the exact algorithm where the solution is merely trying all the possible routes, leading the computing times to lies within a polynomial factor of $O(n!)$ (Abdulkarim & Alshammari, 2015). That means for 10 numbers of *VP*, the computational times would be 3628800, which is impractical. The *NN* algorithm is given in Figure 5.

A simple simulation is provided in Table 3 to make a performance comparison for the *NN* algorithm and the other alternative methods using identical data. The detailed algorithm for ant colony optimization and Held & Karp methods (dynamic programming) can be found in Ibrahim (2020) and Kivelevitch (2020).

Each of the methods mentioned above has its advantages and disadvantages. While in this simulation, the *NN* algorithm produces a considerably longer route than the optimal solution (16% longer), it still can perform well in a large number of data. The Ant optimization colony provides a different result in each run. This algorithm, however, is not efficient for dealing with large-scale

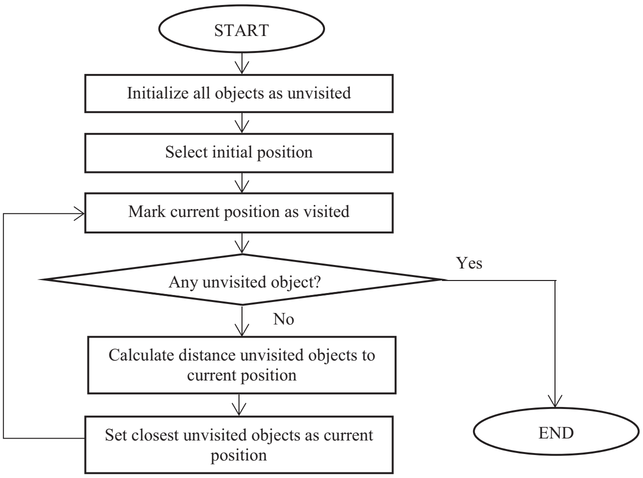

START
Initialize all objects as unvisited
Select initial position
Mark current position as visited
Any unvisited object?
Yes
No
Calculate distance unvisited objects to current position
Set closest unvisited objects as current position
END

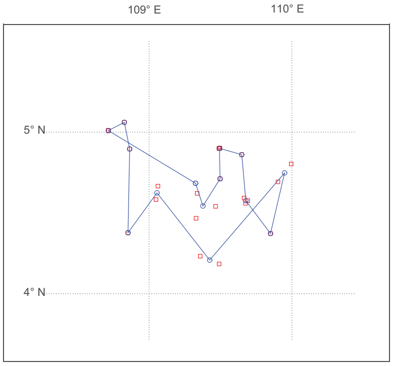
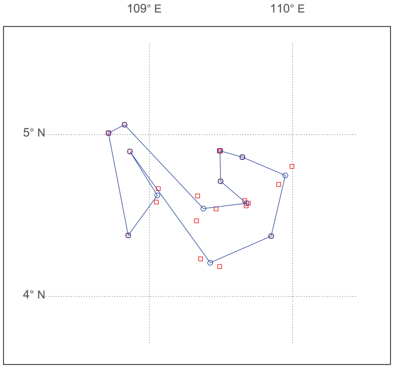
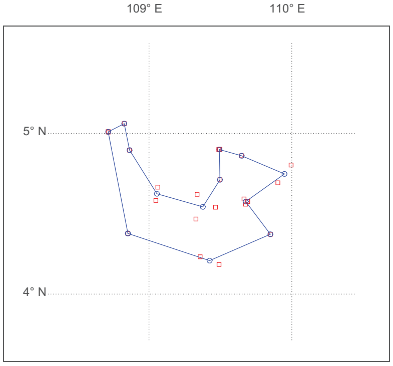
Figure 5. Nearest "Neighbour" (United Kingdom) algorithm flowchart

combinatorial problems because the time complexity of ACO is $O\left(n\cdot\left(n-1\right)\cdot m\cdot T/2\right)$ (Wang, 2018). The Held & Karp method is guaranteed to provide the accurate (optimal) result to the TSP, but the time complexity of this dynamic programming algorithm is $O\left(2^{n}\ n^{2}\right)$, which limits the use of this algorithm to 15 cities or less (Kivelevitch, 2020).

2. Results and discussions

To test the method that has been designed, some simulations are conducted. The data needed are the ship's positions in longitude and latitude and the size of the ships. Several results are presented in this paper using the data from January 2016. The parameters for the simulation are given in Table 4. First, we applied the clustering method to create operation areas. Some simulations were also

Table 3. Comparison of clustering methods

Algorithm	Flight path	Flight distance, nmi	
		Single run	Average in 10 run
Nearest "Neighbor" (United Kingdom)		287	287
Ant Colony		307	317
Held & Karp		247	247

conducted by varying the operation radius to find out the number of operation areas needed for a particular operation area radius. The results are shown in Figure 6.

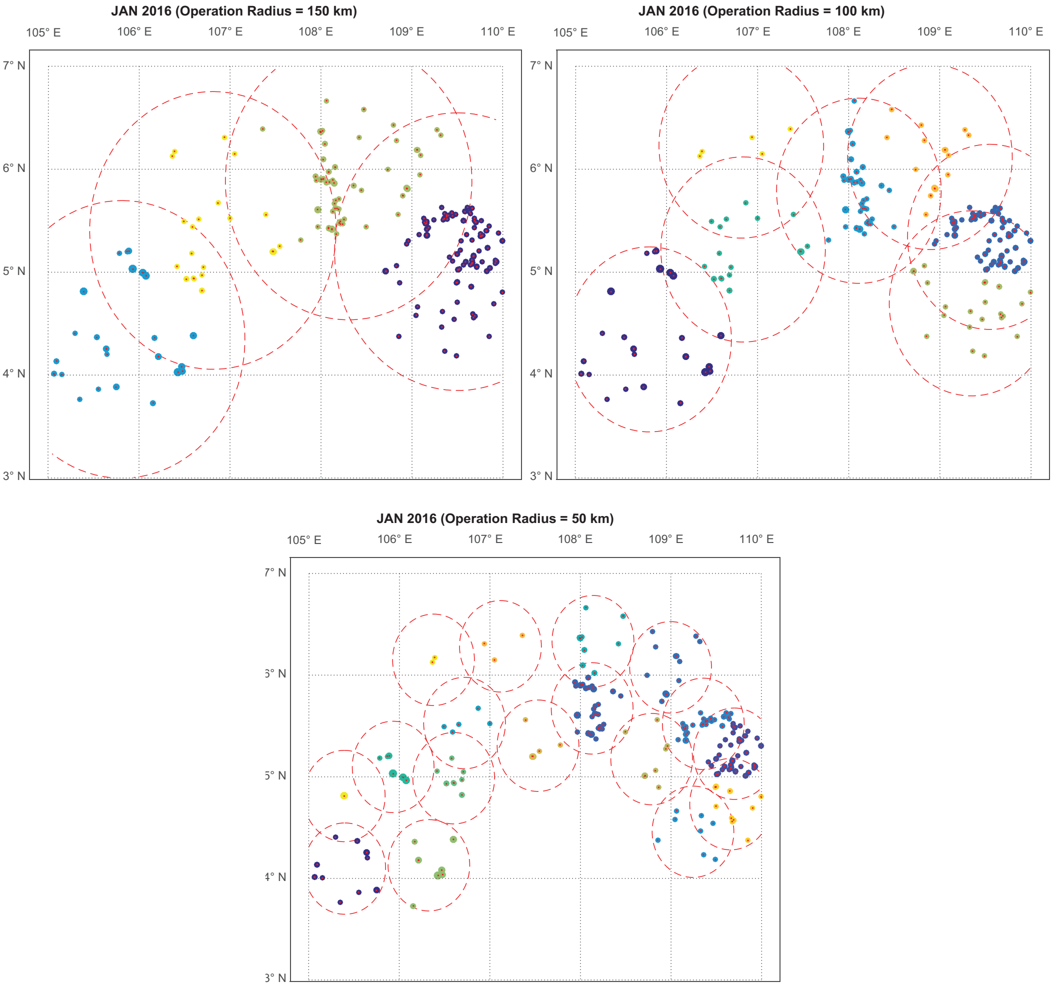
The dots in the intersection area belong to set with the closest center point as indicated by the colour (United Kingdom). The smaller the radius of the operation area, the more number of operation area is needed to cover the whole area, as shown in Figure 7. In some cases, especially for small radius areas, sometimes will be found some operation area with a small number of dots. This is caused by the outlier that too far from other members. These outliers may be ignored for efficiency purposes.

Table 4. Simulation parameters

Parameters	Value
Minimum ship's size, m	15
Maximum ship's size, m	50
Minimum latitude, deg	3
Maximum latitude, deg	7
Minimum longitude, deg	105
Maximum longitude, deg	110
Maximum mission area radius, km	150, 100, 50

More simulations are conducted to determine the flight path for each operation area. The operation area for an operation radius of 150 km is selected for example. In this case, there are four operation areas (see Figure 6). The flight paths are shown in Figure 8.

The flight paths that have been produced able to cover the vulnerable area. The K-means progressed clustering able to produce a similar size of operation area. This method, however, has a significant drawback which is


JAN 2016 (Operation Radius = 150 km)

JAN 2016 (Operation Radius = 100 km)

JAN 2016 (Operation Radius = 50 km)

Figure 6. Operation Area by varying the operation radius

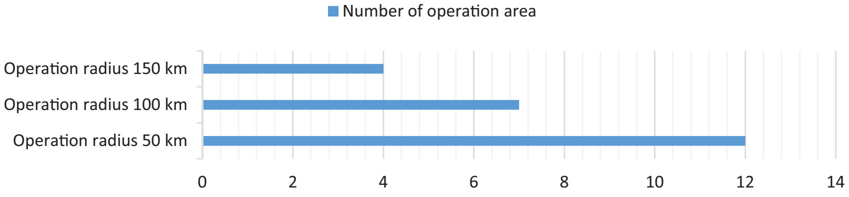


Figure 7. Number of operation area vs operation radius

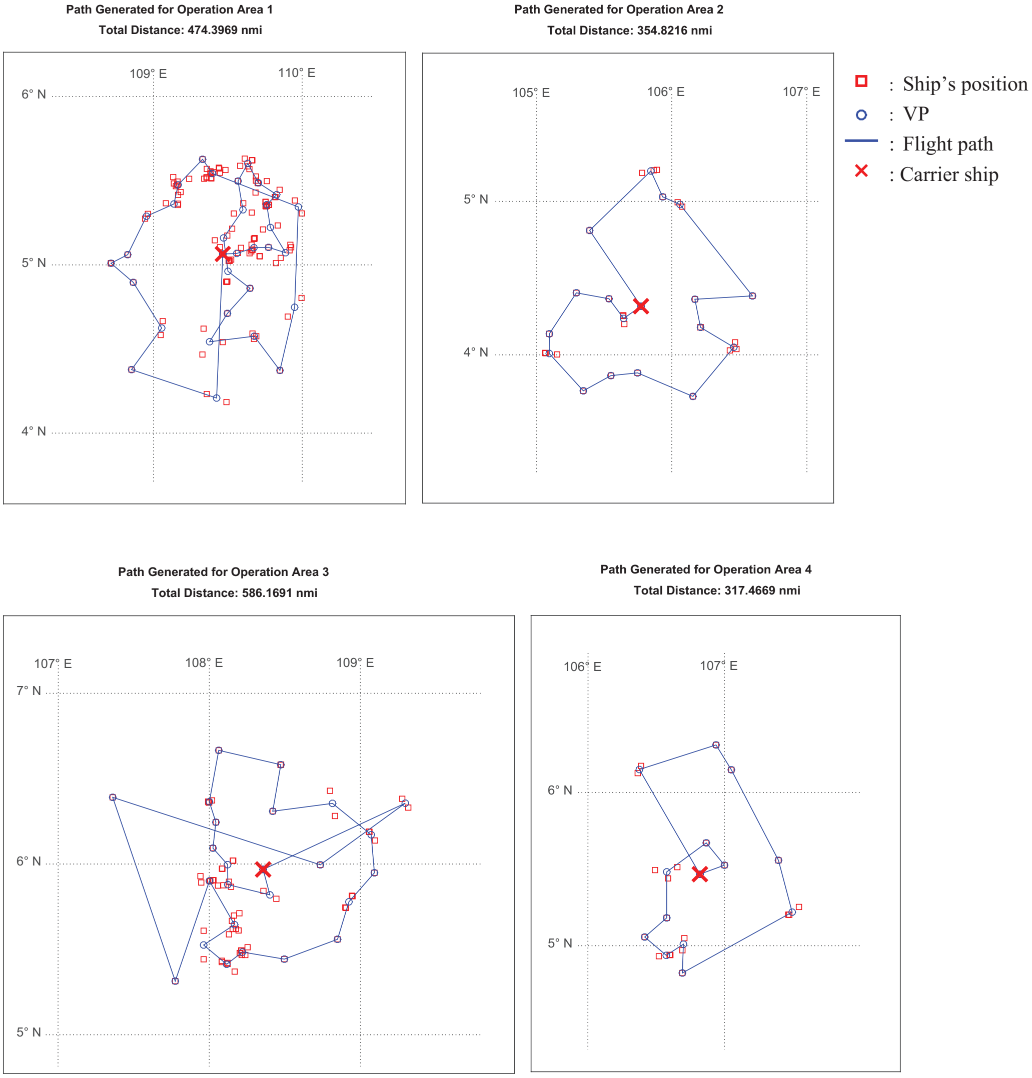


Figure 8. UAV routes for all courses

may produce clusters with very few members, especially when outliers exist. The mini-clustering (second clustering) able to reduce the number of nodes significantly, therefore shorten the flight distance. In Figure 8 we can see that there are path intersections which indicating that the resulted flight paths are not the exact shortest route. The flight distance may be shortened by improving the path planning method, and this could become material for future research.

Here, although the radius of the operation area is the same, the total distances vary from one another depending on the number of the suspects and its distribution. The total distance calculated will help to verify whether the actual UAV could do the mission or not.

Note that the *VP's* are the weight points of the adjacent ships. The more ships contained within a single *VP* the more vulnerable those spots are. In some cases, we can find a ship which located far away from other ships and formed its *VP*. We may try to omit this data because *VP's* with small members means it is less vulnerable. Omitting these outliers may reduce the total flight distance significantly. Overall, the method used to produce the flight path is relatively flexible and can be used to solve other problems with similar nature.

Conclusions

This paper presents a method to create a flight path for a maritime surveillance mission using UAV. The main difference between the method used in this paper and other known methods is that in this paper, the number of nodes is significantly reduced by introducing vulnerable points to shorten the flight path. Firstly we separate the mission into several operation areas using the K-cluster progressed method correspond to the UAV communication capabilities. Next, we used the clustering method again but on a much smaller scale to incorporate the adjacent data into one group, then, we can determine the vulnerable point in the operation area which is defined as the weight points of the data on each group. Finally, we can set the flight path using the Nearest Neighbour algorithm starting from the take-off site then to every vulnerable point, and back to the landing site.

The cluster results from the K-means progressed method is similar to both hierarchical and K-medoids, however, among the above-mentioned methods, the K-Means progressed method is the only one that capable to consistently produce clusters with similar size. The NN algorithm gives a relatively longer distance (16%) than the shortest possible route, however, this algorithm can still perform well in a large number of nodes compared to the ant colony and the Held & Karp methods.

From the simulations, we found out that despite having a similar size of operation area, the resulted flight distance may vary greatly. For example in this paper using an operating radius of 150 km (81 nmi) resulted in the shortest flight distance of 587 km (317 nmi) while the longest distance reaches 1085 km (586 nmi). The ship's distribution greatly influences this flight distance.

We can reduce the number of data beforehand by filtering the ships according to the real condition in the Sea. For example, in this paper, we only choose the ships with size between 15 to 50 m, which is typically used for fishing in the Natuna Sea. In some cases, it is also a good idea to omit the outliers to reduce the flight distance as sometimes it is not worth the cost of surveillance. Further improvements also can be made to the flight planning algorithm to achieve a shorter flight distance.

References

Abdulkarim, H., & Alshammari, I. F. (2015). Comparison of algorithms for solving traveling salesman problem. *International Journal of Engineering and Advanced Technology*, *4*(6), 76–79.

Duan, G. J., & Zhang, P. F. (2014). Research on application of UAV for maritime supervision. *Journal of Shipping and Ocean Engineering, 4*(2014), 322–326.

Ibrahim, S. (2020). *Ant Colony Optimization (ACO) to solve traveling salesman problem (TSP)*. https://www.mathworks.com/matlabcentral/fileexchange/51113-ant-colony-optimization-aco-to-solve-traveling-salesman-problem-tsp

Jeon, I., Ham, S., Cheon, J., Klimkowska, A., Kim, H., Choi, K., & Lee, I. (2019). A Real-time drone mapping platform for marine surveillance. *ISPRS – International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII-2/W13*, 385–391. https://doi.org/10.5194/isprs-archives-XLII-2-W13-385-2019

Johnson, D. S., & McGeoch, L. A. (1997). The traveling salesman problem: a case study in local optimization. In E. H. L. Aarts & J. K. Lenstra (Eds.), *Local search in combinatorial optimization* (pp. 215–310). John Wiley and Sons Ltd.

Kazantsev, P., Sadakov, V., & Chupakov, M. (2016). Maritime vessels real-time tracking-by-detection in UAV Videos. *Indian Journal of Science and Technology*, *9*(48), 1–9. https://doi.org/10.17485/ijst/2016/v9i48/107490

Kivelevitch, E. (2020). *Dynamic programming solution to the TSP*. https://www.mathworks.com/matlabcentral/fileexchange/31454-dynamic-programming-solution-to-the-tsp

Klimkowska, A., Lee, I., & Choi, K. (2016). Possibilities of UAS for maritime monitoring. *ISPRS – International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. XLI-B1,* 885–891. https://doi.org/10.5194/isprsarchives-XLI-B1-885-2016

Livingstone, C. E., Sikaneta, I., Gierull, C., Chiu, S., & Beaulne, P. (2005). RADARSAT-2 system and mode description. In *Meeting Proceedings RTO-MP-SCI-150*, Paper 15 (pp. 1–22).

Miller, C. E., Tucker, A. W., & Zemlin R. A. (1960). Integer programming formulation of traveling salesman problems. *Journal of the ACM, 7*(4(October 1960), 326–329. https://doi.org/10.1145/321043.321046

Moon, W. M., Staples, G., Kim, D. J., Park, S. E., & Park K. A. (2010). RADARSAT-2 and coastal applications: surface wind, waterline, and intertidal flat roughness. In *Proceedings of the IEEE*, *98*(5), May 2010 (pp. 800–815). https://doi.org/10.1109/JPROC.2010.2043331

Murtagh, F., & Contreras, P. (2011). Methods of hierarchical clustering. *Computing Research Repository – CORR*. https://doi.org/10.1007/978-3-642-04898-2_288

Park, H.-S., & Jun, C.-H. (2009). A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications*, *36*, 3336–3341. https://doi.org/10.1016/j.eswa.2008.01.039

Singhroy, V., & Charbonneau, F. J. (2014). RADARSAT: Science and applications. *La Physique Au Canada, 70*(4), 212–217.

Suseno, P. A. P., & Wirawan, A. (2019). Penentuan Basis Operasi pada Sistem Pemantauan Maritim Berbasis Wahana Terbang tak Berawak. *Jurnal Teknologi Dirgantara*, *16*(2), 149. https://doi.org/10.30536/j.jtd.2018.v16.a3034

Wang, G. (2018). A comparative study of Cuckoo algorithm and Ant Colony algorithm in optimal path problems. In *MATEC Web of Conferences, 232,* 03003. https://doi.org/10.1051/matecconf/201823203003

Watanabe, K., Takashima, K., Mitsumura, K., Utsunomiya, K., & Takasaki, S. (2017). Experimental study on the application of UAV drone to prevent maritime pirates attacks. *TransNav, International Journal on Marine Navigation and Safety of Sea Transportation*, *11*, 705–710. https://doi.org/10.12716/1001.11.04.18