An Efficient Genetic Fuzzy Approach to UAV Swarm Routing

Article · December 2015

DOI: 10.1142/S2301385016500011

3 authors:

Anoop Sathyan

University of Cincinnati

Nicholas Ernest

Psibernetix Inc.

Kelly Cohen

University of Cincinnati

An Efficient Genetic Fuzzy Approach to UAV Swarm Routing

Anoop Sathyan,* Nicholas D. Ernest,† Kelly Cohen‡

Department of Aerospace Engineering and Engineering Mechanics, University of Cincinnati, OH 45220, USA

Fuzzy logic is used in a variety of applications because of its universal approximator attribute and non-linear characteristics. But, it takes a lot of trial and error to come up with a set of membership functions and rule-base that will effectively work for a specific application. This process could be simplified by using a heuristic search algorithm like Genetic Algorithm (GA). In this paper, genetic fuzzy is applied to the task assignment for cooperating Unmanned Aerial Vehicles (UAVs) classified as the polygon visiting multiple traveling salesman problem (PVMTSP). The PVMTSP finds a lot of applications including UAV swarm routing. We propose a method of genetic fuzzy clustering that would be specific to PVMTSP problems and hence more efficient compared to k-means and c-means clustering. We developed two different algorithms using genetic fuzzy. One evaluates the distance covered by each UAV to cluster the search-space and the other uses a cost function that approximates the distance covered thus resulting in a reduced computational time. We compare these two approaches to each other as well as to an already benchmarked fuzzy clustering algorithm which is the current state-of-the-art. We also discuss how well our algorithm scales for increasing number of targets. The results are compared for small and large polygon sizes.

Keywords: Genetic fuzzy; PVMTSP; UAVs; Clustering; Swarm routing

1. Introduction

Recent technological advancements in both hardware and real-time implementation enable us to push the envelope with regards to introducing intelligence into aerospace systems design for numerous important applications such as propulsion systems [1], satellite attitude control systems [2], and collaborative control of a swarm of UAVs [3, 4]. The mapping of sensor information, collected in real-time, is fused with the dynamic system model and operational and environmental databases onto a set of control actions and/or decisions which need to be computationally efficient, robust in face of uncertainties and noise, scalable and adaptable to dynamic variations to the mission while adhering to all the constraints of the specific application. The basic approach is to make most of what resources are available in order to get the best possible results. We are seeking a high performing, robust and scalable optimal action. Recent experience with the application of genetic fuzzy systems [1–4] has shown a great deal of potential.

Intelligent control techniques are gaining traction and increased focus [1]. Fuzzy logic control is one such intelligent control technique and will be a cornerstone of this study. This non-linear control design technique provides significant benefits in terms of design flexibility, universal approximator attribute and possible coupling with optimization processes. When coupled with the ability to capture expert or heuristic knowledge, and the ability to tune behavior in local envelopes of the operating space, fuzzy logic can be an indispensable control design tool in many applications. Fuzzy logic control also possesses inherent robustness due to having knowledge-based properties, making them good candidates for stochastic systems. One of the main challenges facing control designers is the tuning of the membership functions and the heuristics involved. Fuzzy logic controllers can have a variety of handles to impact performance, from the fuzzy input and output sets to the governing rule base. GA, a branch of evolutionary algorithms, will be utilized in this study to provide an autonomous guided search of the design space to develop a more optimized solution against the design requirements.

In this paper, we propose a method of genetic fuzzy clustering that would be specific to PVMTSP problems and hence more efficient compared to k-means and c-means clustering. We developed two different algorithms using genetic fuzzy. One technique uses the distance covered by each UAV to cluster the search-space by moving targets on the convex hull around to other clusters until all the cluster distances have an almost equal value. The other algorithm uses a cost function that approximates the distance covered which helps in reducing the computational time. We compare these two approaches to each other as well as to an already benchmarked fuzzy clustering algorithm

*sathyaap@mail.uc.edu
†nick.ernest@psibernetix.com
‡cohenky@ucmail.uc.edu

which is the current state-of-the-art. We also discuss how well our algorithm scales for increasing number of targets. The results are compared for small and large polygon sizes.

In the case of unsupervised learning, genetic fuzzy logic has advantages over other techniques such as self organizing fuzzy logic which are also used for tuning fuzzy controllers. Self organizing fuzzy controllers learn to control the system in accordance with the desired response. In order to train a self organizing fuzzy controller, a feature set (i.e. input-output pairs) needs to be created. In the case of genetic fuzzy logic, the GA is used to tune the fuzzy parameters by minimizing a fitness function. Thus, in applications where the design requirement can be defined as a mathematical function, genetic fuzzy approach is more straightforward than using self organizing fuzzy control. The PVMTSP is one such application where the requirement is to minimize the maximum distance and hence genetic fuzzy is more appropriate.

The PVMTSP finds applications in UAV swarm routing where a number of UAVs start from a single depot, cover all the targets collaboratively and return back to the depot. As shown in Fig. 1, the polygons in this class of problem designate a visibility area of a UAV. These are created by taking some sphere around the target with the radius of the desired sensor or weapon on board the UAV, removing sections of the sphere blocked by obstacles or terrain, and slicing a plane at some constant altitude. Its simpler variants, namely the TSP and MTSP, are widely researched. The TSP is an NP-hard problem. Hence, if one can find an efficient algorithm for the TSP, then efficient algorithms could be found for all other problems in NP.

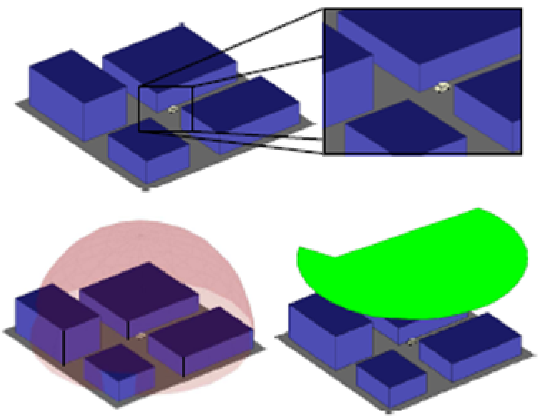


Fig. 1. Creation of visibility polygons [5]

Autonomous routing of UAVs enables the technologies to be employed for numerous aerospace applications. Communications for remote controlled swarms are limited by bandwidth and security constraints as well as the number of trained operators required. The PVMTSP presents a simplified routing problem for real-time control of a UAV swarm, but could also serve as a mission planning tool for groups of remote controlled UAVs.

One of the major limiting factors when it comes to solving TSP is the scalability of the technique; i.e., how well it performs as the number of targets, n, increases. With most algorithms, the computational time increases exponentially as n increases. Lin and Kernighan [6] came up with a heuristic method that produced near-optimum solutions with computational times proportional to n^2. This procedure, popularly known as the Lin-Kernighan (LK) method, is based on a general approach and is currently used successfully in solving a wide range of problems. Other techniques of solving the TSP involve using a new mutation operator with GA [7], parallelized genetic ant colony system [8], edge-assembly crossover [9,10], to name a few.

The solution to the MTSP depends mainly on the effectiveness of the clustering algorithm. Research has been done on the MTSP and its different variants including the Vehicle Routing Problem (VRP). In one of the previous works [11], kmeans was used to cluster the targets and then 2-opt was applied to solve the individual TSPs. This resulted in a high performance algorithm both in terms of maximum distance as well as computational time. It was that the cluster-first approach is definitely a great improvement over directly applying GA to solve the MTSP. Golden et. al proposed a heuristic search approach involving tabu search and an adaptive memory procedure could be used to solve the VRP [12]. Christopher et. al showed another approach involves using tree search algorithms [13] by incorporating lower bounds computed from shortest spanning k-degree centre tree (k-DCT), and q-routes. The results show that the bounds derived from the q-routes are superior to those from k-DCT and that VRPs of up to about 25 customers can be solved exactly. Kivelevitch et. al proposed a method that involves simulating an economic market in which the agents (UAVs) interact to win tasks situated in an environment [14] to solve the MTSP. The agents strive to minimize required costs, defined as either the total distance travelled by all agents or the maximum distance travelled by any agent. The results shows that the Market Based Solution (MBS) is both quick and close to the optimal solution, and robust to changes in the scenario. The problem of scalability is discussed [15]; i.e., how well the MBS performs when applied to larger sized problems, for a min-max variant called the Multiple Depot MTSP (MDMTSP).

Mitchell et. al presented a comparison of fuzzy optimization and genetic fuzzy methods in solving a modified TSP [16]. Targets were randomly placed in a surveillance environment and the objective was to find the shortest path around the environment, where it touched each target area at least once before returning to its starting position. Through fuzzy optimization of a path produced through a GA, this task was completed and it was shown that a shorter path could be found through the fuzzy optimization. The solution was then modified to accommodate Dubins paths.

Ernest and Cohen [17] developed a GA and Fuzzy Inference System (FIS) based approach to the path representation of a variant of the TSP known as the Multi-Depot Polygon

Visiting Dubins Multiple Travelling Salesman Problem (MD-PVDMTSP). Utilizing a hybridization of control techniques, they proved that this approach works effectively and efficiently approximates path planning and visibility problems encountered by a UAV swarm in a constant altitude, constant velocity, two-dimensional case. Ernest et al. [3] provided approximate solutions for complex variants of the TSP, or more precisely named the Multi-Objective Min-Max Multi-Depot Polygon Visiting Dubins Multiple Travelling Salesman Problem (MMM-PVDMTSP). The techniques are utilized in such a way that the problem is examined from a top level view which is then approximated entirely before moving on to the next level. While iterative methods are used at almost every level of the problem, each level is only solved once. Assumptions and generalizations must be made to accommodate this, however the cost of these can be minimized and the pay-off is drastically reduced runtime, even for such a complex problem. Sabo et. al presented a solution to a variation of the VRP that minimized the total time that all the targets have to wait to be picked up and delivered to a communication range [18]. The results indicate that the heuristic gives near-optimal results in real time, thus allowing it to be used for large problem sizes.

Genetic fuzzy can be used to tune different types of membership functions. Triangular membership functions are defined by the x-coordinates of the three vertices. Symmetric triangular membership functions can be defined using the center of the base of the triangle and its width. Gaussian membership functions are defined using the mean (center) and standard deviation. Hosseini et al [19] uses GA to tune an FIS consisting of Gaussian membership functions. GA can be used to tune the parameters of a Gaussian membership function in the same way it is done for triangular membership functions.

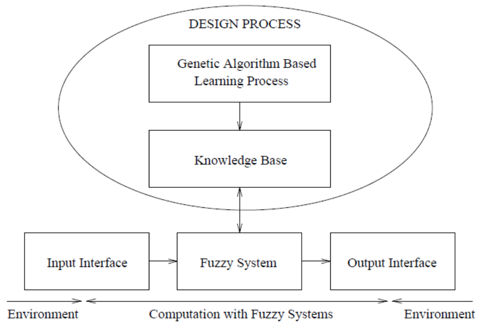


Fig. 2. Schematic of genetic fuzzy system [20]

Cordon and Herrera has presented an overview of the Genetic Fuzzy Systems (GFSs), showing the use of the GAs in the construction of the FLC knowledge bases [20]. The schematic for GFS including the design process is shown in Fig. 2. Surmann et. al has proposed an automatic design method using GAs to train the input and output membership functions [21]. Lee presented three fuzzy system architectures and methods for automatically designing them for high dimensional problems [22]. The results indicate that the real coded algorithms consistently outperformed the binary coded algorithms in both the final performance of the system and the performance of the search algorithm. The Asymmetric-Triangular fuzzy systems consistently improved faster than the hyper-ellipsoidal and shared triangular representations in all cases.

The major factor contributing to the computational time is the clustering algorithm which requires the distance to be evaluated during each iteration [23]. A cost function that is proportional to the distance covered could be used instead to reduce the computational time. Beardwood et. al proposed the following approximation for the shortest path through n points bounded within an area A [24].

$$VRP \approx k\sqrt{An} \tag{1}$$

The constant of proportionality k depend only upon the dimensionality of the space, and are independent of the shape of the region. Figliozzi studied the approximations to the average length of VRPs [25]. The following six approximations were studied where A is the area of the map and r is the average distance between the depot and the targets. The parameters k_l, k_b, and k_m are estimated by linear regression.

$$VRP \approx k_1\sqrt{An} + 2rm \tag{2}$$

$$VRP \approx k_1\frac{n-m}{n}\sqrt{An} + 2rm \tag{3}$$

$$VRP \approx k_1\sqrt{An} + 2k_m m \tag{4}$$

$$VRP \approx k_1\frac{n-m}{n}\sqrt{An} + k_m m \tag{5}$$

$$VRP \approx k_1\sqrt{An} + k_b\sqrt{\frac{A}{n}} + 2k_m m \tag{6}$$

$$VRP \approx k_1\frac{n-m}{n}\sqrt{An} + k_b\sqrt{\frac{A}{n}} + k_m m \tag{7}$$

The results indicate that as the distance between the depot and delivery region increases, the accuracy of the approximation increases.

2. Problem Formulation

In this paper, we propose to apply genetic fuzzy logic to solving the PVMTSP. Rather than targets being represented by points in the case of classical TSP, in the PVMTSP problem, each target is an area defined by a polygon. Here, the figures of merit include the computational time as well as the minimum time to complete the mission.

Let $N = \{1, 2, 3, , n-1, n\}$ be the set of indices defining the targets, m be the number of UAVs and d_{ij} be the distance between the i^{th} and j^{th} targets. Each of these n targets are defined by a polygon defined by k points that on its boundary, as shown below, where $p_{ij} = (x_{ij}, y_{ij})$.

$$\begin{aligned} &P_1 : \{p_{11}, p_{12}, p_{13}, \ldots, p_{1k}\} \\ &P_2 : \{p_{21}, p_{22}, p_{23}, \ldots, p_{2k}\} \\ &P_3 : \{p_{31}, p_{32}, p_{33}, \ldots, p_{3k}\} \\ &\vdots \\ &P_n : \{p_{n1}, p_{n2}, p_{n3}, \ldots, p_{nk}\} \end{aligned} \tag{8}$$

Let $n_1, n_2, n_3, \ldots, n_m$ be the number of targets assigned to respective UAVs. This implies that

$$n_1 + n_2 + n_3 + \cdots + n_m = n \tag{9}$$

Let $T_1, T_2, T_3, \ldots, T_m$ each define the tours of the m UAVs without including the depot. The actual tour starts with the depot and ends at the depot. t_{ij}s refer to the unique indices assigned to the targets.

$$\begin{aligned} &T_1 : \{t_{11}, t_{12}, t_{13}, \ldots, t_{1n1}\} \\ &T_2 : \{t_{21}, t_{22}, t_{23}, \ldots, t_{2n2}\} \\ &T_3 : \{t_{31}, t_{32}, t_{33}, \ldots, t_{3n3}\} \\ &\vdots \\ &T_m : \{t_{m1}, t_{m2}, t_{m3}, \ldots, t_{mnm}\} \end{aligned} \tag{10}$$

The union of all the tours should be equal to the set of targets, N, and there shouldn't be any target common to any of the tours.

$$T_1 \cup T_2 \cup T_3 \cup \cdots \cup T_m = N \tag{11}$$

$$T_1 \cap T_2 \cap T_3 \cap \cdots \cap T_m = \{\phi\} \tag{12}$$

Each of the indices in the tour should be assigned to a point on the corresponding polygon such that the total distance is minimized. The objective is to minimize the mission time t_M.

$$minimize \quad t_M \tag{13}$$

The mission time is proportional to the minimum of the maximum distance amongst the UAVs.

$$D_{max} = \max_q (D_q) \tag{14}$$

D_q is the distance covered by the q^{th} UAV. Hence, the objective can be rewritten as [11]

$$minimize \quad D_{max} = max_q(D_q) \tag{15}$$

This shows that the PVMTSP is a min-max optimization problem. This paper also discusses how the algorithm scales with increasing number of targets. Four UAVs start from a common depot, cover 200 targets that are spread over a 1000 units x 1000 units space, and return to the depot in the shortest time. The polygon radius is assumed to be 10 units. A genetic fuzzy approach is used to cluster the targets among the four UAVs such that almost equal distance is covered by each UAV. Fig. 3 shows a PVMTSP solution.

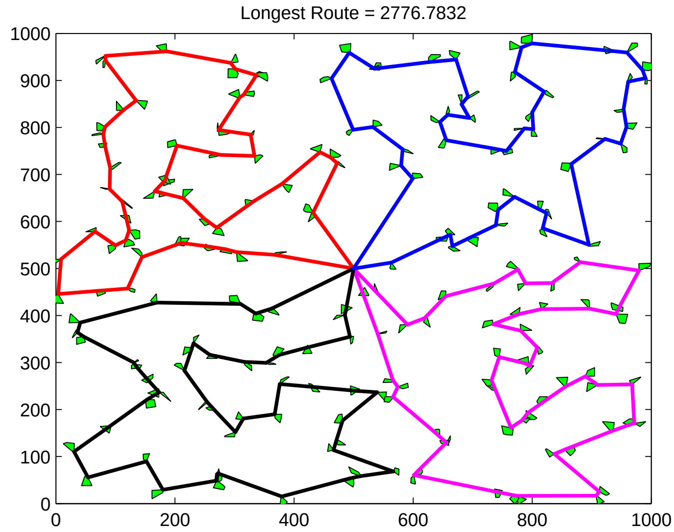


Fig. 3. Benchmark 2: PVMTSP

Assumptions

- The problem is assumed to be symmetric i.e the distance from A to B is the same as the distance from B to A.

$$d_{ij} = d_{ji} \, \forall \, i, j \in N \tag{16}$$

- It is assumed that the triangle inequality holds.

$$d_{ij} + d_{jk} \geq d_{ik} \, \forall \, i, j, k \in N \tag{17}$$

- Only two dimensional motion is considered i.e each UAV flies at a constant altitude.
- All UAVs travel at the same speed. This makes the time taken by the UAVs to complete the tour proportional to the distance covered.

$$t_q \propto D_q \tag{18}$$

- The number of targets is greater than the number of UAVs.

$$n > m \tag{19}$$

- There is no need for loitering.
- The route between two targets is the line connecting them.
- Collision avoidance is not considered. UAVs could work at different altitudes.
- The UAVs do not need a turn radius thus ignoring Dubins paths.
- The UAVs have to just touch the polygons without the need to necessarily enter them.
- The UAVs do not have any constraints on fuel. Thus, the UAVs have infinite range.

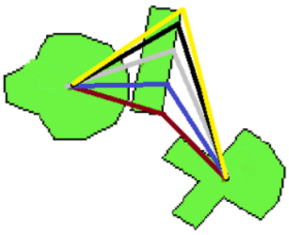
3. Methodology

3.1. *Fuzzy Clustering Method*

The fuzzy clustering method is a subset of the methods from previous work [3] that similarly utilizes a cluster-first approach to solve the PVMTSP. Here the angle between the depot and the targets is considered for clustering rather than the Cartesian coordinates. The radius measurement is ignored for this problem. A clustering FIS develops an initial guess at the proper clustering of the targets.

The convex hulls of this initial estimation is calculated and then the solution is refined through additional FISs which analyze a single target at a time. Each convex hull is analyzed, and its relative number of targets and target densities are calculated. FISs then swap points between each of the clusters in an effort to achieve dense clusters of equal and minimal size. This process ends once the clusters are within some threshold of each other in these two statistics.

To determine which point on the polygons the UAV will visit, a simple algorithm is utilized which iterates around a number of points on each side of the polygon. This optimizes the combined length of the two lines connecting the point on the polygon in question with the selected points on the polygon before and after it, as shown in Fig. 4. This iterates over the route three times, at which point the solution converges. LK algorithm then solves the individual TSPs of these points for each UAV.



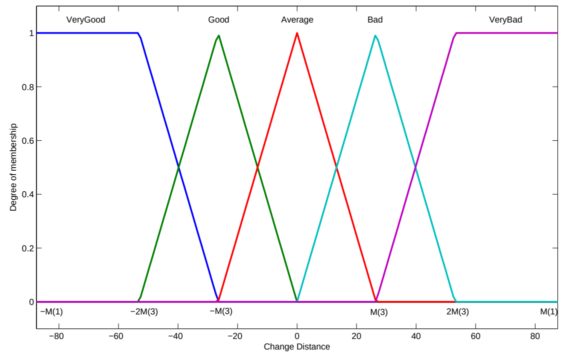
$$Distance = \sqrt{\left((y_{i,n} - y_{i-1})^2 + (x_{i,n} - x_{i-1})^2\right)} + \sqrt{\left((y_{i+1} - y_{i,n})^2 + (x_{i+1} - x_{i,n})^2\right)}$$

Fig. 4. Point selection on each polygon

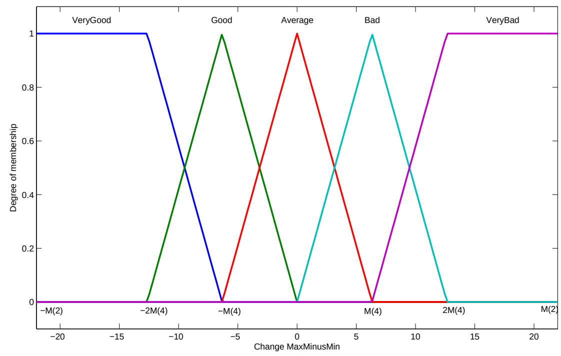
3.2. *Genetic Fuzzy Clustering Method*

In one of our previous works [11], we used a cluster first approach using K-means and then applied 2-opt to solve the individual clusters, which achieved high levels of performance. But the K-means and even fuzzy C-means are purely distance optimizing clustering algorithms and do not optimize appropriately for MTSP problems.

In the genetic fuzzy clustering method, an FIS [26] divides the targets into four clusters and each cluster is solved using a TSP solver like the LK method. The technique shown in Fig. 4 is used to find the optimal point of contact on each polygon. The FIS used for clustering [26] has two inputs, ChangeMaxMinusMin and ChangeDistance, which are respectively the change in maximum minus minimum distance and the change in maximum distance as a result of shifting points, measured as a percentage. Minimizing the longest distance is the overall goal of the algorithm, but ”maximum minus minimum” distance is an important measure of how close the solution is to being optimal. Maximum minus minimum distance would be zero in a perfect clustering. The FIS gives the membership grade as the output which describe how likely a particular point is to move. The membership functions are assumed to be triangular and symmetric with equal width.



(a)ChangeDistance



(b)ChangeMaxMinusMin

Fig. 5. Membership function tuning for both inputs

GA is used to determine the width of the membership functions for both the inputs. GA also tunes the rule-base to obtain the optimum combination of rules which minimizes the cost function. In order to obtain the membership functions, GA tunes a vector M which has four elements. M(1) and M(2) are the range of the two inputs ChangeDistance and ChangeMaxMinusMin, respectively. M(3) and M(4) represent the width of the membership functions for the two respective inputs as shown in Fig. 5. Rulebase is obtained by tuning a vector R which consists of 10 elements which includes the 5 rules and their corresponding weights. R(1) through R(5) represent the rules and their value should be integers ranging from 1 to 5. R(6) through R(10) represent the weights for each rule and hence they range between 0 and 1. Thus, the rulebase is set as follows with weights for each rule shown in square brackets.

- If ChangeDistance is VeryGood OR ChangeMaxMinusMin is VeryGood THEN MemGrade is R(1) [R(6)]
- If ChangeDistance is Good OR ChangeMaxMinusMin is Good THEN MemGrade is R(2) [R(7)]
- If ChangeDistance is Average OR ChangeMaxMinusMin is Average THEN MemGrade is R(3) [R(8)]
- If ChangeDistance is Bad OR ChangeMaxMinusMin is Bad THEN MemGrade is R(4) [R(9)]
- If ChangeDistance is VeryBad OR ChangeMaxMinusMin is VeryBad THEN MemGrade is R(5) [R(10)]

The flowchart for this genetic fuzzy approach is shown in Fig. 6. The expanded form of the fuzzy logic clustering block of this flowchart is shown in Fig. 7.

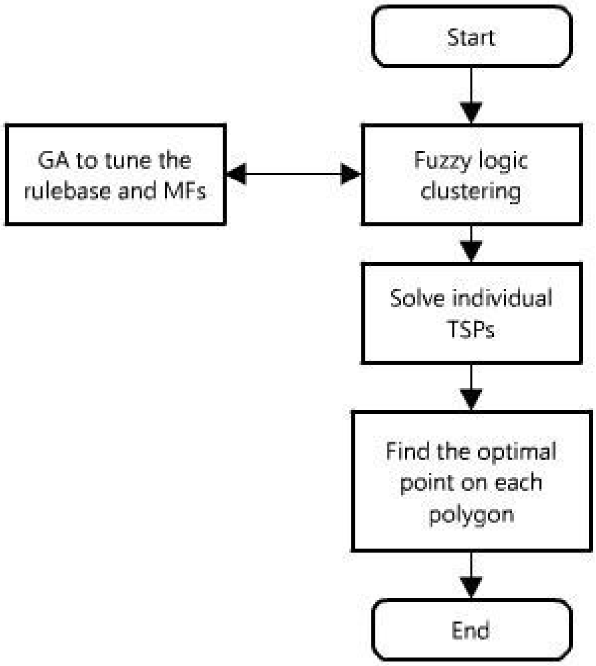


Fig. 6. Flowchart of Genetic Fuzzy Clustering Method for PVMTSP

3.3. *Genetic fuzzy clustering using an approximate cost function*

Although the genetic fuzzy clustering method provides good results in terms of maximum distance, it requires a higher computational time [23]. The clustering algorithm evaluates the distance covered by each UAV during each iteration and tries to equalize them by shifting targets between the UAVs. Thus, the LK algorithm m times during each iteration. This results in the higher computational time. Hence, it could be reduced if a cost function that is proportional to distance covered by each UAV, and easier to evaluate, is used. The cost function used by Beardwood [24] is modified to account for the polygon area.

$$C_q = \sqrt{n_q(S-P)} \tag{20}$$

n_q, S and P are the number of targets, area of the convex hull and total area covered by polygons respectively of the k^{th} cluster.

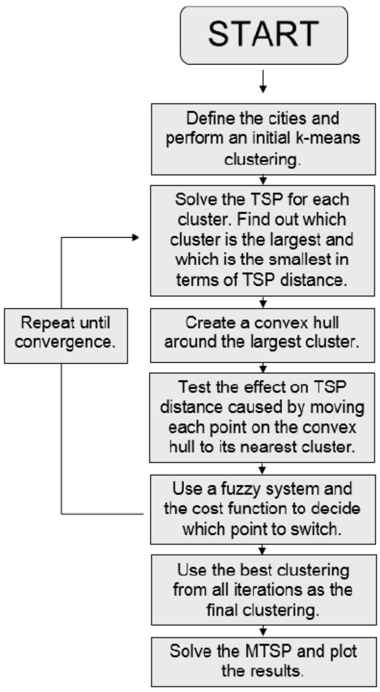


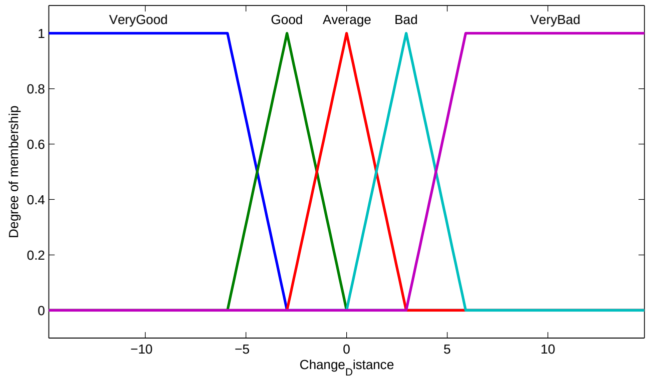
Fig. 7. Flowchart of fuzzy logic clustering used in genetic fuzzy clustering method [26]

4. Results

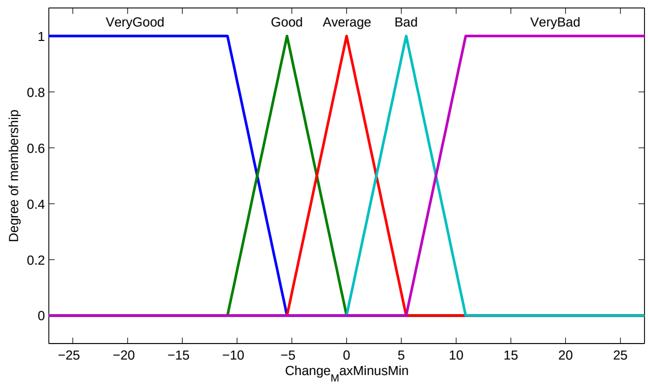
All the results were obtained with a laptop utilizing MATLAB with an Intel i3 2.3GHz processor and 4GB of RAM.

The rulebase obtained after tuning using GA is as follows:

- If ChangeDistance is VeryGood OR ChangeMaxMinusMin is VeryGood THEN MemGrade is VeryHigh [0.9920]
- If ChangeDistance is Good OR ChangeMaxMinusMin is Good THEN MemGrade is Good [0.9938]
- If ChangeDistance is Average OR ChangeMaxMinusMin is Average THEN MemGrade is Average [0.9925]
- If ChangeDistance is Bad OR ChangeMaxMinusMin is Bad THEN MemGrade is Bad [0.9953]
- If ChangeDistance is VeryBad OR ChangeMaxMinusMin is VeryBad THEN VeryBad is R(5) [0.9961]



(a)ChangeDistance



(b)ChangeMaxMinusMin

Fig. 8. Tuned membership functions for PVMTSP

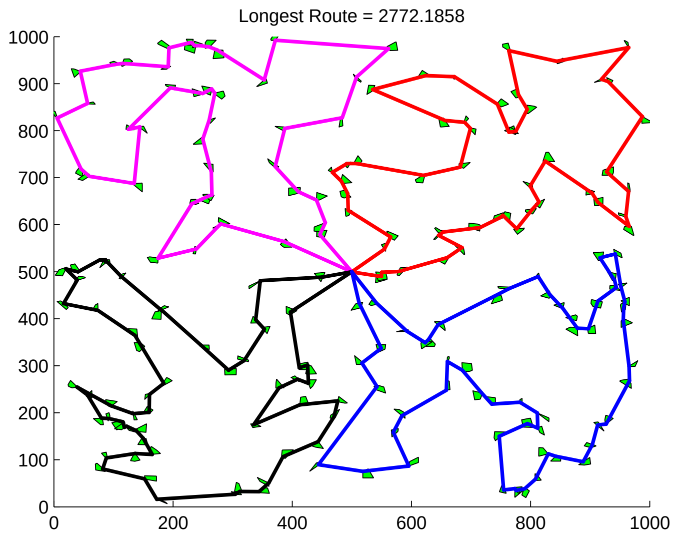
The membership functions obtained after tuning are shown in Fig. 8. The results obtained are compared with fuzzy clustering method [3] and is shown in table 1. Table 1 also shows the results obtained by using the approximate cost function of Eq. 20. The values shown in the table are the averages obtained over 100 runs of the code for two different polygon radii. The tests were conducted for ”small” as well as ”large” polygons. Small and large polygons refer to polygons of radius 10 units and 30 units, respectively. A larger polygon represents a larger area of visibility. It can be observed that both GFCM and the approximate cost function approach gives 6% better result compared to that obtained using fuzzy clustering method for small polygons. But, the computational time for the GFCM is significantly higher. in the case of large polygons, GFCM gives 5% better distances while the approximate cost function method gives just 2% improvement over the fuzzy clustering method.

The computational time for the fuzzy clustering method is quite low due to the fact that only simple algorithms, FISs, and the LK solver are utilized, allowing the algorithm to scale well. Quality results are obtained, however there are weaknesses present in the clustering method. The strength of the initial guess has a significant effect on the quality of the final result. Additionally, the refinement process occasionally performs suboptimal swaps of clusters and nearest-neighbor points. Increasing the points utilized for this check could help reduce suboptimal swaps, though computation time would increase to some degree.

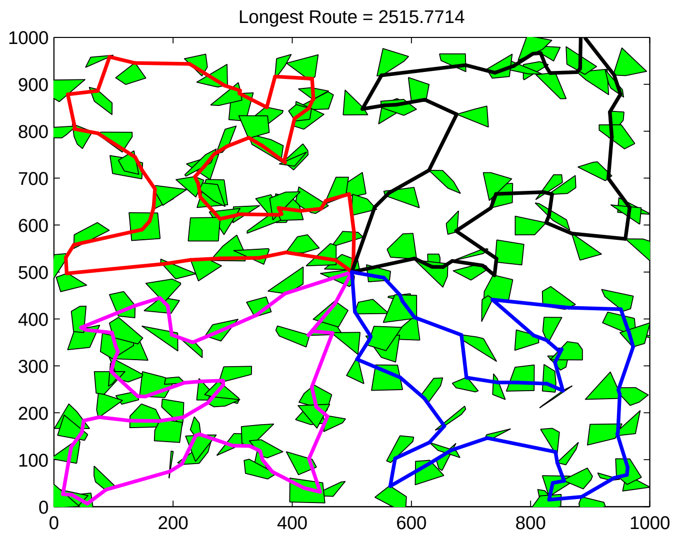
The solutions to the PVMTSP, using the GFCM, for small and large polygons are shown in Fig. 9. The presence of a lot of straight lines passing through the polygons in the solution shown in Fig. 9(b) is indicative of the effectiveness of the algorithm in finding the optimal connection points on each polygon.

Table 1. Comparison of results obtained for PVMTSP

	Small polygons		Large polygons	
	Largest distance	Computational time (in seconds)	Largest distance	Computational time (in seconds)
Fuzzy clustering method (FCM)	2919	6.0	2498	8.2
Genetic fuzzy clustering method (GFCM)	2754	80.8	2376	96.3
Genetic fuzzy clustering using approximate cost function	2743	8.02	2456	8.0



(a) Polygon radius = 10 units



(b) Polygon radius = 30 units

Fig. 9. PVMTSP solution with (a) small and (b) large polygons

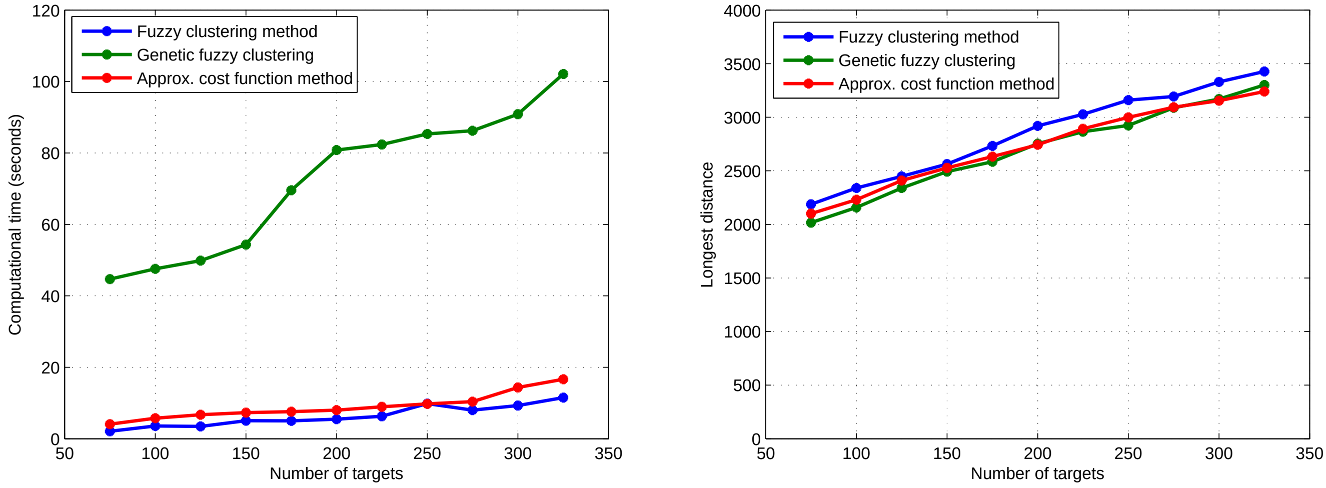
Fig. 10 shows the variations in computational time and the longest distance as the number of targets are changed in the case of small polygons. From Fig. 10(b), it can be seen that both genetic fuzzy approaches give better distances as compared to the fuzzy clustering method. The computational time increases steadily as the number of targets are increased. The computational time is significantly higher compared to the other two methods. The approximate cost function method has a similar computational time profile as the fuzzy clustering method while giving better distances. Thus, overall the approximate cost function method performs better in the case of small polygons.

Fig. 11 shows the variations in computational time and the longest distance as the number of targets are changed in the case of large polygons. From Fig. 11(b), it can be seen that all three algorithms perform similarly in the case of large polygons. The genetic fuzzy approaches are slightly better for a lesser number of targets. But at higher target numbers, the fuzzy clustering method is better, although the difference is not that significant. The computational time increases steadily as the number of targets are increased. The computational time is significantly higher compared to the other two methods. Also for the case of large polygons, the approximate cost function method has a similar computational time profile as the fuzzy clustering method.

5. Conclusions & Future Work

This research has shown the applicability of genetic fuzzy systems to path planning. While fuzzy logic by itself works well for this set of problems, tuning the parameters involved to satisfy a specific requirement might need a lot of trial and error to be done by the researchers. Incorporating GA to tune these parameters solves this problem.

The GFCM shows an improvement in optimal distance as compared to the fuzzy clustering method [3]. The main disadvantage of the GFCM is the higher computational time. This was because of the need to evaluate the distances for each UAV during every iteration of the clustering code. This was solved by using an approximate cost function instead of evaluating the distance. The performance of the algorithms were shown for small and large polygons. This reduces the computational time by a factor of 10 thus making it comparable with that of the fuzzy clustering method. The results could be further improved by using a better cost function than Eq. 20, especially in the case of large polygons. The low computational time makes this algorithm useful for applications requiring real-time updates. For example, if one of the UAVs is shot down or if a new set of targets are added to the existing mission, the GFCM with approximate

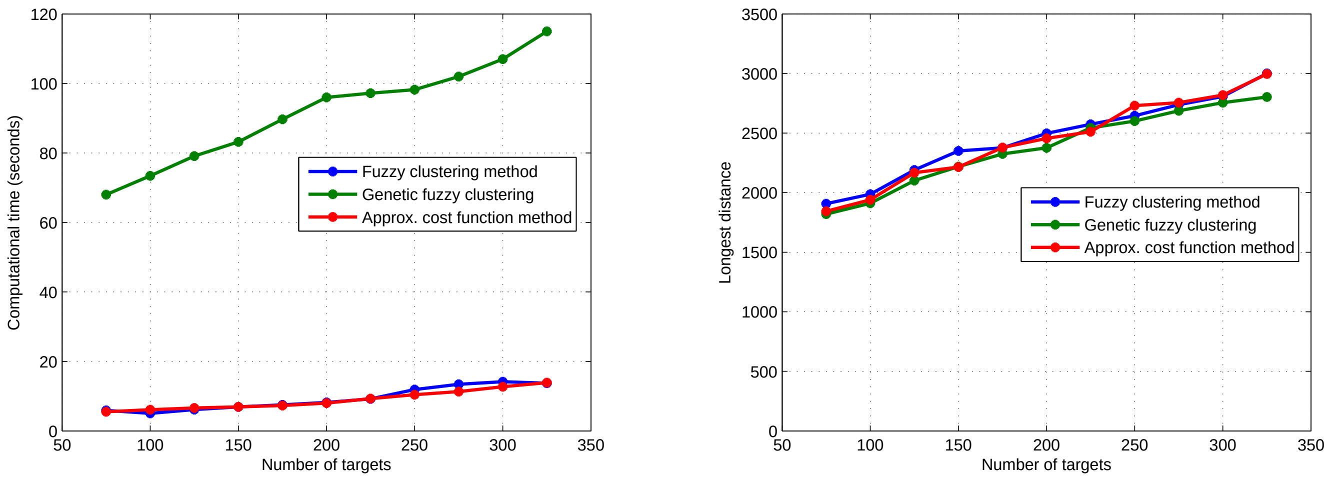


(a) Computational time

(b) Longest distance

Added sub-figure label descriptions

Fig. 10. Computational time and longest distance in relation to number of targets for small polygons



(a) Computational time

(b) Longest distance

Added sub-figure label descriptions

Fig. 11. Computational time and longest distance in relation to number of targets for large polygons

cost function could be used to rearrange the UAVs to cover the target areas. Although the GFCM used symmetric membership functions for this research, it can very well be extended to asymmetric membership functions. This will increase the number of parameters that need to be tuned using GA.

The results from this study show improvements for the utilization of genetic fuzzy techniques for solving these types of problems. Quick run-times are maintained by the GFCM with the approximate cost function while performance is increased. While the run-times are slightly higher for the GFCM with approximate cost function, both it and the FCM increase linearly in computational cost with the number of targets. Additionally, the difference between the two times is so subtle that speed optimization of the coding could provide enough benefit to overcome the FCM. While maintaining an efficient and linearly increasing computational cost, the GFCM with approximate cost function is able to improve upon the already extreme levels of performance of the FCM. This marks a significant increase in the capabilities of fuzzy systems to solve these types of routing problems.

References

[1] A. W. Vick, Genetic fuzzy controller for a gas turbine fuel system, PhD thesis, University of Cincinnati (2010).

[2] A. Walker, K. Cohen and P. Putman, Fuzzy logic attitude control of a magnetically actuated cubesat, *Proceedings of the AIAA Infotech@Aerospace Conference, no. AIAA-2013-5059. Boston, MA*, (August 2013).

[3] N. Ernest, K. Cohen and C. Schumacher, UAV swarm routing through genetic fuzzy learning methods, *Proceedings of the AIAA Infotech@Aerospace Conference, no. AIAA-2013-4730. Boston, MA*, (August 2013).

[4] N. Ernest, K. Cohen and C. Schumacher, Collaborative tasking of uavs using a genetic fuzzy approach, *Proceedings of the 51st Aerospace Sciences Meeting, no. AIAA-2013-1032. Grapevine, TX*, (January 2013).

[5] K. J. Obermeyer, Visibility problems for sensor networks and unmanned air vehicles, PhD thesis, University of California Santa Barbara (2010).

[6] S. Lin and B. W. Kernighan, An effective heuristic algorithm for the traveling-salesman problem, *Operations research* **21**(2) (1973) 498–516.

[7] M. Albayrak and N. Allahverdi, Development of a new mutation operator to solve the traveling salesman problem by aid of genetic algorithms, *Expert Systems with Applications* **38**(3) (2011) 1313–1320.

[8] S.-M. Chen and C.-Y. Chien, Parallelized genetic ant colony systems for solving the traveling salesman problem, *Expert Systems with Applications* **38**(4) (2011) 3873–3883.

[9] Y. Nagata and D. Soler, A new genetic algorithm for the asymmetric traveling salesman problem, *Expert Systems with Applications* **39**(10) (2012) 8947–8953.

[10] Y. Nagata, Edge assembly crossover: A high-power genetic algorithm for the traveling salesman problem, *Proc. 7th ICGA*, (1997), pp. 450–457.

[11] A. Sathyan, N. Boone and K. Cohen, Comparison of approximate approaches to solving the travelling salesman problem & its application to UAV swarming, *International Journal of Unmanned Systems Engineering (IJUSEng)* **3 No. 1** (2015) 1–16.

[12] B. L. Golden, G. Laporte and É. D. Taillard, An adaptive memory heuristic for a class of vehicle routing problems with minmax objective, *Computers & Operations Research* **24**(5) (1997) 445–452.

[13] N. Christofides, A. Mingozzi and P. Toth, Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations, *Mathematical programming* **20**(1) (1981) 255–282.

[14] E. Kivelevitch, K. Cohen and M. Kumar, Market-based solution to the allocation of tasks to agents, *Procedia Computer Science* **6** (2011) 28–33.

[15] E. Kivelevitch, K. Cohen and M. Kumar, On the scalability of the market-based solution to the multiple traveling salesmen problem, *Proceedings of the 2012 AIAA Infotech@ Aerospace Conference, no. 2012-2543. Garden Grove, CA*, (June 2012).

[16] S. M. Mitchell, N. D. Ernest and K. Cohen, Comparison of fuzzy optimization and genetic fuzzy methods in solving a modified traveling salesman problem, *Proceedings of the 2013 AIAA Infotech@ Aerospace Conference, no. 2013-4664. Boston, MA.*, (August 2013).

[17] N. Ernest and K. Cohen, Fuzzy clustering based genetic algorithm for the multi-depot polygon visiting dubins multiple traveling salesman problem, *Proceedings of the 2012 AIAA Infotech@ Aerospace, no. AIAA-2012-2562, AIAA. Garden Grove, CA* (June 2012).

[18] C. Sabo, D. Kingston and K. Cohen, A formulation and heuristic approach to task allocation and routing of uavs under limited communication, *Unmanned Systems* **2**(01) (2014) 1–17.

[19] R. Hosseini, S. D. Qanadli, S. Barman, M. Mazinani, T. Ellis and J. Dehmeshki, An automatic approach for learning and tuning gaussian interval type-2 fuzzy membership functions applied to lung cad classification system, *Fuzzy Systems, IEEE Transactions on* **20**(2) (2012) 224–234.

[20] O. Cordón and O. F. Herrera, A general study on genetic fuzzy systems, *Genetic Algorithms in Engineering and Computer Science, (Periaux, J., Winte, G., Eds.)*, (John Wiley and Sons, 1995), pp. 33–57.

[21] H. Surmann, A. Kanstein and K. Goser, Self-organizing and genetic algorithms for an automatic design of fuzzy control and decision systems, *In Proc. EUFIT'93*, Citeseer (1993).

[22] M. A. Lee, On genetic representation of high dimensional fuzzy systems, *Uncertainty Modeling and Analysis, 1995, and Annual Conference of the North American Fuzzy Information Processing Society. Proceedings of ISUMA-NAFIPS'95., Third International Symposium on*, IEEE (1995), pp. 752–757.

[23] A. Sathyan, N. Ernest and K. Cohen, Genetic fuzzy approach for control and task planning applications, *AIAA Infotech @ Aerospace Conference, no. AIAA-2015-0887. Kissimmee, FL*, (January 2015).

[24] J. Beardwood, J. H. Halton and J. M. Hammersley, The shortest path through many points, *Mathematical Proceedings of the Cambridge Philosophical Society*, **55**(04), Cambridge Univ Press (1959), pp. 299–327.

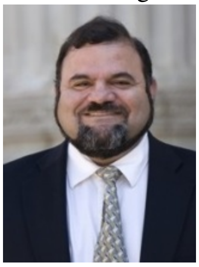
[25] M. A. Figliozzi, Planning approximations to the average length of vehicle routing problems with varying customer demands and routing constraints, *Transportation Research Record: Journal of the Transportation Research Board* **2089**(1) (2008) 1–8.

[26] N. Boone, A. Sathyan and K. Cohen, Enhanced approaches to solving the multiple traveling salesman problem, *Proceedings of the AIAA Infotech@Aerospace Conference, no. AIAA-2015-0889. Kissimmee, FL*, (January 2015).

Modified References 2, 3, 4, 11, 15, 16, 17, 20, 23, 26


Anoop Sathyan is a PhD student of Aerospace Engineering at the University of Cincinnati. He is currently involved with the SIERRA Project (Surveillance for Intelligent Emergency Response Robotic Aircraft). He did his Bachelor's from College of Engineering, Trivandrum and his Master's from Indian Institute of Science, Bangalore. His current research involves using image processing techniques to identify and locate forest fires. His research interests include intelligent systems, genetic fuzzy, deep learning, convolutional neural networks, path planning algorithms, UAVs, and image processing.


Nicholas Ernest holds a PhD in Aerospace Engineering from the University of Cincinnati, where he received both his B.S. and M.S. in the same major as well. He is the CEO of Psibernetix Inc., a startup tech company specializing in artificial intelligence for the Department of Defense topics as well as bioinformatics. Ernest is also a member of the AIAA Intelligent Systems Technical Committee. His research interests include intelligent systems, computational science, cooperative control, and genetic fuzzy systems.


Kelly Cohen,Professor of Aerospace Engineering, has been at the University of Cincinnati (UC) since 2007. He obtained his PhD in aerospace engineering at the Technion-Israel Institute of Technology in 1999. At UC, he teaches the following undergraduate courses: introduction to systems engineering, fundamental control theory, and modeling and simulation of dynamic systems; as well as the following graduate courses: fuzzy logic control, systems engineering analysis, and optimal control. He is the Principal Investigator of the SIERRA project. He is also involved with UC Health on using fuzzy logic for neuroscience applications. His research interests include intelligent systems, fuzzy logic cooperative multi-agent control, unmanned aerial vehicles, feedback flow control, low order modeling, and system identification.