

Heterogeneous UAV Swarm System for Target Search in Adversarial Environment

Shripad Gade

Department of Aerospace Engineering
Indian Institute of Technology
Mumbai, India 400076
Email: shripad.gade@iitb.ac.in

Ashok Joshi

Department of Aerospace Engineering
Indian Institute of Technology
Mumbai, India 400076
Email: ashokj@aero.iitb.ac.in

Abstract—Unmanned Aerial Systems (UAS) have great potential to aid in search and situation assessment. Here, we present a UAV swarm system performing target search in adversarial environment. It utilizes a Ant-Colony (ACO) and Artificial Potential Function (APF) based decentralised target search strategy. ACO meta-heuristic forms the higher level guidance algorithm and APF provide global and local path planning. Uncertainty maps are used to represent probable target locations. The algorithm is scalable and is shown to be robust to agent loss. Its distributed nature makes it ideal for applications in large scale search operations. Trajectory estimates are factored into prioritization resulting into better target selection and faster search.

NOMENCLATURE

n_{seen}	Number of sightings for complete classification
\mathcal{A}	Set of Agents ($a_i \in \mathcal{A}$)
Ψ_i	Pheromone map of agent a_i
Γ_i	Task list for agent a_i
ϕ_i	Potential Map for agent a_i
$z(i, j, k)_t$	Uncertainty of target at location (i, j, k)
η	Reduction factor for target uncertainty
HZ	Home Zone

I. INTRODUCTION

Automatic Target Recognition (ATR) is autonomous and remote search of targets based on its characteristic patterns. Unmanned Aerial Vehicles (UAV's) have been extensively used for intelligence gathering, surveillance and reconnaissance in recent times. Early ATR systems faced the inhibition due to limited computational capacity. With perpetually increasing computational power of on-board computers, it has now been possible to provide UAV's with significant computational capabilities. UAV's can now perform basic image processing apart from gathering information. Such technological advancements have opened up a host of application fronts for UAS. Several methods of target search including swarm based classification [1]–[3], cooperative classification [5], opportunistic learning [4], coverage methods [6] have been implemented on UAS. In swarm based systems, UAV's capture images and locate potential targets using basic image processing. These possible sites are shared with neighboring UAV agents. UAV's collaboratively visit these sites and confirm target locations.

Centralised algorithms require UAV's to collect sensor data and process it at central servers for decision making. This

model is however riddled with problems of scalability and high communication loads. Data/Agent loss and risk of communications failure can endanger mission success. Centralized systems are also vulnerable to attacks on centralised servers. Swarm methods on the contrary are inherently decentralized making it immune to malevolent attacks. Localized decision making also introduces scalability. It would therefore be prudent to use such decentralized swarms for large scale ATR.

Here we investigate application of Super-Agent led UAV teams for collaborative target search in hostile airspace with popup and cluster threats. Teams are comprised of different types of UAV's, having dissimilar dynamics and sensor characteristics. We present a swarming algorithm for performing target search using uncertainty maps.

A. The Search Problem

Automatic target search in an dynamic, noisy and uncertain environment is to be performed in the presence of hostile zones. Targets are scattered in a bounded domain and are capable of motion in \mathfrak{R}^2 . Information about target location or distribution is not available at the outset. Targets have to be visited multiple times to ensure complete classification ($n_{seen} > 1$). Popup threats become active, at random time instants during search. Cluster treats are group of targets that pop-up simultaneously and are co-located. The UAV agents are capable of moving in \mathfrak{R}^3 . Agents present in the field are equipped with localization equipment like GPS, wireless module for communication, cameras (vision, thermal etc.) as sensors and sufficient onboard computational capacity. Hostile entities are also scattered in the domain and are capable of destroying UAV's that fall within its lethal radius.

II. SUPER-AGENTS: THE SWARM LEADERS

We define Super-Agents (SA's) as agents who outperform normal agents in terms of motion dynamics, sensor characteristics and communication capabilities. The concept of having swarm leaders in search operation arises from the underlying idea that, in information deprived situations; faster the information gathering, faster the target classification. We try to achieve this by providing certain agents (Super-Agents) with better speed, sensors and communication capabilities. Advantage in speed results in faster gathering of information.

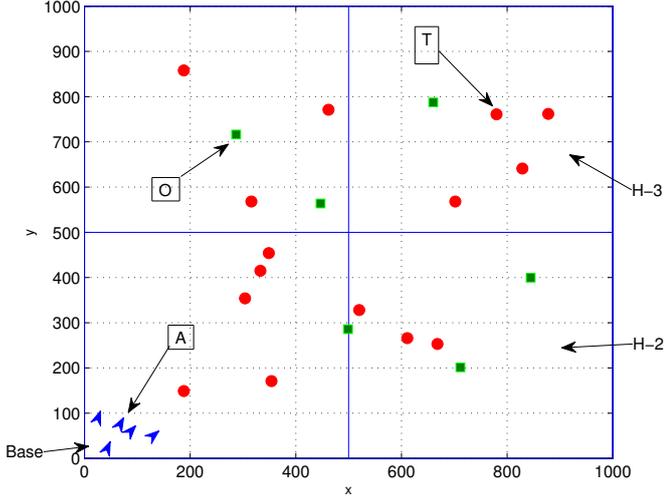


Fig. 1. Simulation Environment.
O - Hostiles, T - Targets, A - Agents, $\{H-2, H-3\}$ - Home Zones

Better sensors with an increased vision range result into quicker target sightings. Better communication with longer ranges will ensure greater information dissemination and hence will better prepare the swarm for search. These leadership qualities can drastically improve search performance. Let us consider set of agents \mathcal{A} deployed in a search domain,

$$a_i = a(\lambda, V, C) \quad \forall a_i \in \mathcal{A} \quad (1)$$

$$\lambda = \frac{\text{Speed SA}}{\text{Speed Agent}} \quad (2)$$

$$V = \frac{\text{SensorRange SA}}{\text{SensorRange Agent}} \quad (3)$$

$$C = \frac{\text{Comm.Range SA}}{\text{Comm.Range Agent}} \quad (4)$$

Agent a_i with standard characteristics is represented by $a(1, 1, 1)$. Super agent with speed advantage of 50%, vision advantage of 50% and communication advantage of 25% is represented as $a(1.5, 1.5, 1.25)$.

III. ALGORITHM

Search Domain is the area of interest that needs to be searched. Search domain is further divided into virtual sections called as Home Zones. Fig. 1 shows the simulation environment. Agents assigned to a home zone are expected to search only inside the allotted home zone except when confirming a gossiped target.

Each agent $a_i \in \mathcal{A}$ maintains the following data sets,

- Target uncertainty map
- Ψ_i is the pheromone landscape of agent a_i and is populated by adding perceived target points obtained from sensors and gossip.
- Γ_i is the task set corresponding to the agent a_i .

- State information of other agents.

Each point $\psi \in \Psi_i$ corresponds to a potential target. Task $\gamma \in \Gamma_i$ corresponds to pheromone point and represents the target status and its priority.

A. Deterministic Foraging

Deterministic Foraging is the de-mining, non-swarming behavior shown by agents when their task set Γ_i is empty. Foraging behavior is expected to exhaustively search areas in the absence of any target information. As soon as targets are sensed, this information is gossiped and swarming behavior sets in. A waypoint based deterministic foraging algorithm is implemented here. Fig. 2 shows waypoint based foraging in simulation.

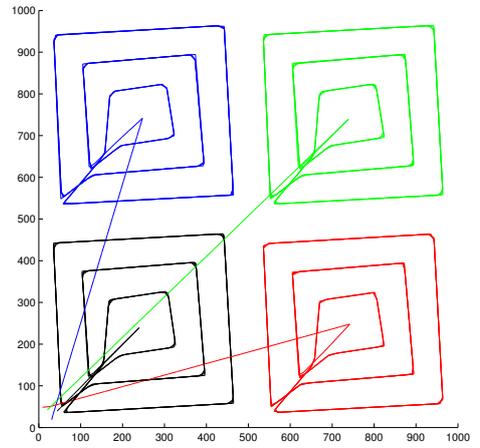


Fig. 2. Deterministic Foraging performed by agents in absence of targets.

B. Pheromone Map

Pheromone map is a target knowledge map. It stores information about sensed targets, obstacles and hostile entities. Each of the target points, obstacles and/or hostiles is represented as a pheromone point. Pheromone point $\psi \in \Psi_i$ is represented as,

$$\psi = \{loc, pher, phertime, vel, bear, weight\} \quad (5)$$

where, $\psi.loc \in \mathcal{R}^2$ is the location of the potential target. $\psi.pher \in \mathcal{R}$ is the pheromone associated with the point and depends upon the severity and confidence perceived by the agent. $\psi.phertime$ is the time of first pheromone deposition and is used in comparing relative life of information. $\psi.vel$ and $\psi.bear$ are estimated target velocity magnitude and bearing as seen by agent. $\psi.weight$ is a Boolean value assigned during prioritization.

Agent a_i gathers information about potential targets either through sighting or through gossip. If it is a first sighting, agent adds pheromone point to that location. Otherwise, it updates the pheromone landscape with information already present and new information. Pheromone values are programmed to decay

with time. It removes false identifications and noise from Ψ . An exponential decay model is used. Pheromone model and update equations from [1] have been used

C. Uncertainty Map

Uncertain and noisy environments usually require multiple sightings for the target to be completely classified. Target Uncertainty Map quantifies how much is known about existence of targets in a given area. It has uncertainties of target being present at perceived target location. Every sighting reduces the target uncertainty by a factor η (see Eq. 6).

$$z(i, j, k) = \eta z(i, j, k) \quad (6)$$

This factor depends upon altitude and linearly increases with altitude (see Eq. 7). A target gets completely classified when its uncertainty value becomes smaller than the threshold ($z(i, j, k) < \text{classification threshold}$).

$$\eta = \eta_0 + \frac{(\eta_1 - \eta_0)}{(h_1 - h_0)}(h - h_0) \quad (7)$$

D. Gossip

Gossip is a mechanism employed by agents to communicate with each other. Uncontrolled flooding type of gossip is employed here. An agent keeps broadcasting possible target information flooding the environment with data. Whenever agents are in vicinity, they receive data and update their pheromone and uncertainty maps.

E. Prioritization Strategy

Swarming involves, use of simple coordination rules, between agents leading to emergence of intelligent behavior. Ant Colony metaheuristic is based on food-foraging behavior of ants. We use the concept of pheromones, in prioritizing targets. ACO metaheuristic therefore provides high-level guidance to the swarm agents.

The priority of each target is dynamically calculated for each possible task $\gamma \in \Gamma_i$. These priorities are evaluated based upon the current task, the other possible tasks, their distance estimates and the probability that other agents might prioritize them. As soon as a target gets prioritized its weight is switched to 1, all others are switched to 0. This ensures that just one minima exists (one active target) for any agent a_i . We consider three different prioritization strategies - greedy search, look-ahead strategy [1] and shared rewarding.

1) *Greedy Strategy*: Greedy strategy is based on the philosophy of doing what is best for you irrespective of anything else. Agents move towards the target with maximum pheromone intensity. They do not take into account presence of other agents or the chance of them having prioritized the same target.

2) *Look Ahead Strategy*: In Look Ahead Strategy [1], agents consider the possibility other agents having prioritized same target. An agent a_i will prioritize a target if and only if, the number of agents who could classify the target before a_i are lesser than the minimum number of sights required for confirming a target.

3) *Shared Rewarding Strategy*: Shared Rewarding implies that if multiple agents prioritize the same target, they will have to share the reward obtained by confirming a target. The reward of visiting a target (α_i) is inversely proportional to the time taken (T_i) to reach the target. Total reward of confirming a target is the pheromone value ($\gamma.pher$) of the target. It ensures severe targets with high confidence get higher priority. Let us consider that n agents have information about a target γ . The reward for each agent can be given by Eq. 8. Agents prioritize a target that is associated with maximum reward.

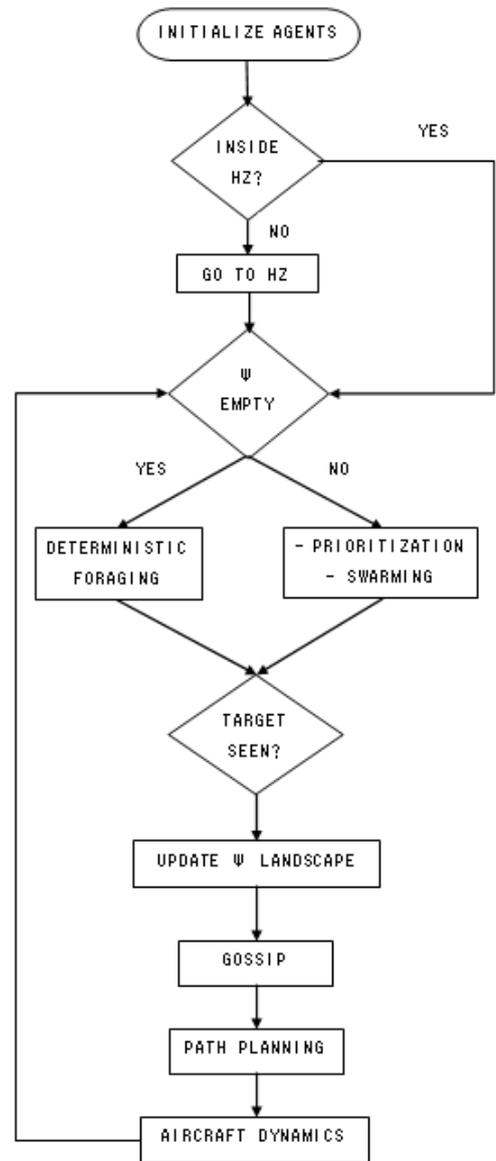


Fig. 3. Algorithm Flowchart

$$\alpha_i = \frac{\gamma \cdot pher}{T_i \sum_{j=1}^n (\frac{1}{T_j})} \quad (8)$$

F. Trajectory Estimation

Look ahead and shared rewarding strategies both depend a lot upon the time taken by agents to reach pheromone points. Due to agent dynamics, the actual path taken will be different from straight line distance. We present two methods to estimate possible UAV trajectories.

1) *Circular Path Estimate*: Circular Path Estimate or the BigCircle estimate (see Fig. 4) is a trajectory such that the agent and target lie on a circle and agent's velocity at that instant is tangent to the circle.

2) *Fastest Turn Estimate*: Fastest turn estimate (see Fig. 5) employs first a turn with minimum possible radius to align UAV's velocity vector to target direction and then a straight line trajectory towards the target.

G. Artificial Potential Function

Artificial Potential Field (APF) has proved to be an essential tool in path planning and trajectory generation in cluttered environments [7], [8]. Artificial Potential Functions provide global and local path planning solution. Targets are provided with negative potential and obstacles, hostile entities and other agents are provided with positive potential. For a prioritized target, APF provides a trajectory that takes it closer to targets, avoids hostile airspace and avoids collisions with neighbouring agents. Harmonic functions are used here. Harmonic functions being solutions of Laplace equation do not generate local minimum in cluttered environments [7].

$$\phi_{Target} = - \sum_{i=1}^{n_{Target}} k_T w_i^T \frac{\gamma \cdot pher}{2\pi} \log(r_i) \quad (9)$$

Eq. 9 represent potentials, with k_T being weights assigned to importance of targets.

IV. SIMULATION AND RESULTS

Simulations for a scaled search problem were performed in MATLAB. Search is conducted in a square domain with a side of 30 km (= 1000 u) and is divided into 4 home regions of equal area as shown in Fig. 1. UAV agents have a speed of 45m/s (= 1.5 u/s). Vision sensors used have a range of 3 km (= 100 u). Communication modules present in UAV's have a range of 6 km (= 200 u). Each simulation is run for 5000 s (= 5000 ticks). A tick is an execution step. During each step, agent processes sensor information, updates pheromone landscape, gossips and prioritizes targets. 20 different target situations are generated and the results shown are average of the 20 simulations.

Number of targets and hostiles are variable and randomly placed within the search domain. Popup targets and cluster target profiles are shown in Fig. 6. Each cluster has 8 targets with low separation. Two target distribution (sparse (8 targets +

Parameter	Value
Severity, s	0,1,2
Confidence, c	0,1,2
Weight to Severity, w_1	0.5
Weight to Confidence, w_2	0.5
Pheromone Decay Rate, β	0.01
Weight to old information, κ	0.5
Classification Threshold	0.1
Uncertainty Reduction factor, η	[0.5 0.8]

TABLE I. SIMULATION PARAMETERS

	Potential Function	Weights
T	$-\sum_{i=1}^{n_T} k_T w_i^T \frac{\gamma \cdot pher}{2\pi} \log(r_i)$	$k_T = 100 \quad \forall r_i \in \mathfrak{R}$
H	$\sum_{j=1}^{n_H} k_O w_i^O \frac{\gamma \cdot pher}{2\pi} \log(r_j)$	$k_O = 10^{09} \quad 0 < r_j < 1.5R_L$ $k_O = 10^{08} \quad 1.5R_L < r_j < 2R_L$ $k_O = 0 \quad r_j > 2R_L$
W	$\sum_{l=1}^{n_W} k_W \frac{1}{2\pi} \log(r_l)$	$k_W = 10^{09} \quad r_l < 5u$ $k_W = 10^{03} \quad 5u < r_l < 20u$ $k_W = 0 \quad r_l > 20u$

TABLE II. ARTIFICIAL POTENTIAL FUNCTIONS
T-TARGETS, H-HOSTILES, W-WALLS

12 popup targets), dense (16 targets + 34 popup targets)) cases are considered with 12 obstacles in each case. Each swarm has 12 agents. Leader guided swarm has 4 $a(1.5, 1.5, 1.5)$ leaders. Parameters used in simulation are tabulated in Table I.

Domain wall is also provided with a steep artificial potential barrier to avoid agents from straying out of the search domain. The potential functions are used to evaluate the gradient direction. The speed of the UAV is kept fixed and hence only the direction from the potential map is needed. The potential functions used in simulation for obstacle, target and wall are given in Table II.

The presence of leaders is beneficial since it increases the rate of information generation and transmission. Since gossip is employed; targets once seen, they can be classified collaboratively. Hence, performance of the swarm in a way can be brought down to t_{eff} (effective time), which can be defined as the time taken for first sighting.

$$t_{eff} = \frac{d_{eff}}{velocity} = \frac{distance - SA \cdot VisRange}{SA \cdot Vel} \quad (10)$$

Presence of leaders directly affects t_{eff} which depends upon d_{eff} (effective distance, defined as minimum distance to first sighting) and $velocity$ (see 10). Super agents with vision advantage can see a target from a much farther distance reducing the value of d_{eff} . Speed advantage further reduces t_{eff} and increases search performance.

Fig. 7 shows the search performance comparison of leader guided swarm and homogeneous swarm in dense and sparse

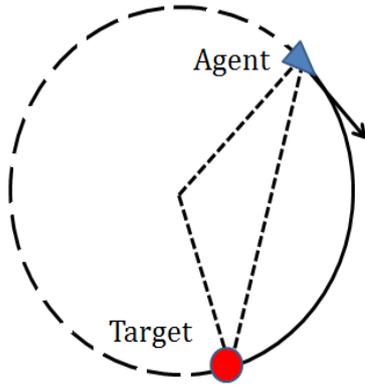


Fig. 4. Big Circle trajectory is shown. Radius of this trajectory is dependent upon agent location and velocity and target location.

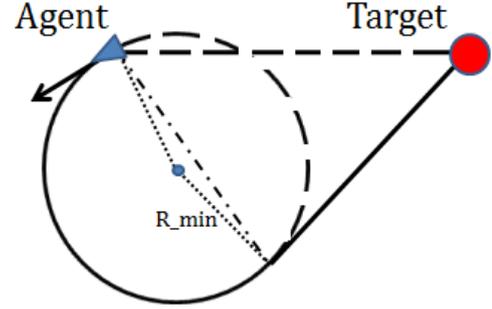


Fig. 5. Fastest Turn trajectory is shown. R_{min} is minimum turn radius and depends on the dynamics of the UAV.

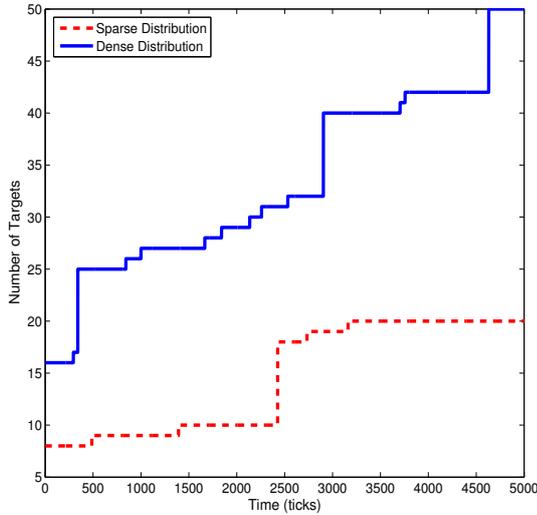


Fig. 6. Evolution of number of targets. Steep changes in target number are due to cluster popup and single step change due to popup targets.

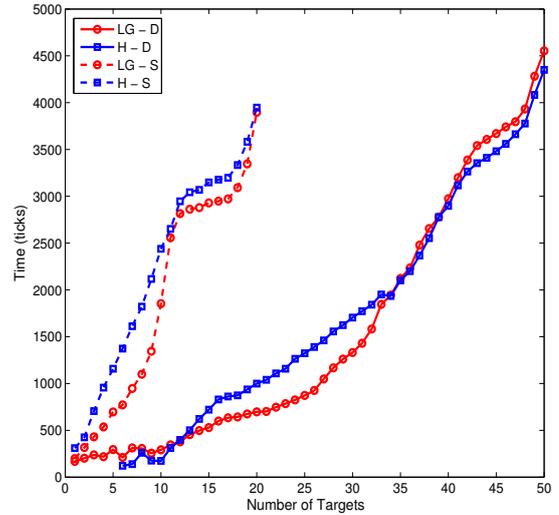


Fig. 7. Leader guided (LG) and Homogeneous swarm (H) performance comparison for dense (D) and sparse (S) target density

target density scenario. System performs better in the presence of leaders and requires less time for search. Benefit of leaders however is lower in densely distributed target zones. In dense target distributions, first sighting happens rather quickly for large number of targets, on its own.

Fig. 8 shows comparison of prioritization strategies in terms of time to search. In dense target distributions look-ahead strategy performs better than shared rewarding and greedy strategy. Shared Rewarding is only slightly better than greedy strategy since it takes into account prioritization by other agents. In sparse target distribution, time taken is similar for most time, however as target cluster pops up, greedy strategy performs better. Since, cluster has 8 targets placed closely, high pheromone deposition in that area, makes greedy strategy better.

Fig. 9 shows effect of trajectory estimation in search performance in different target distributions. Although both estimation methods provide with similar search performance,

fastest turn estimate provides us with marginally better search performance. In dense distribution scenario, Fastest Turn provides performance very similar to Big Circle estimate. However, the performance of Fastest Turn is better for sparse distribution scenario.

Cases	UNCT	%
Guided - Dense	160/1000	16 %
Guided - Sparse	13/400	3.25 %
Unguided - Dense	225/1000	22.5 %
Unguided - Sparse	24/400	6 %
Guided - Dense - Look Ahead	160/1000	16 %
Guided - Dense - Shared Rewarding	100/1000	10 %
Guided - Dense - Greedy	96/1000	9.6 %
Guided - Sparse - Look Ahead	13/400	3.25 %
Guided - Sparse - Shared Rewarding	8/400	2 %
Guided - Sparse - Greedy	19/400	4.75 %

TABLE III. UNCONFIRMED TARGETS
UNCT: UNCONFIRMED TARGETS

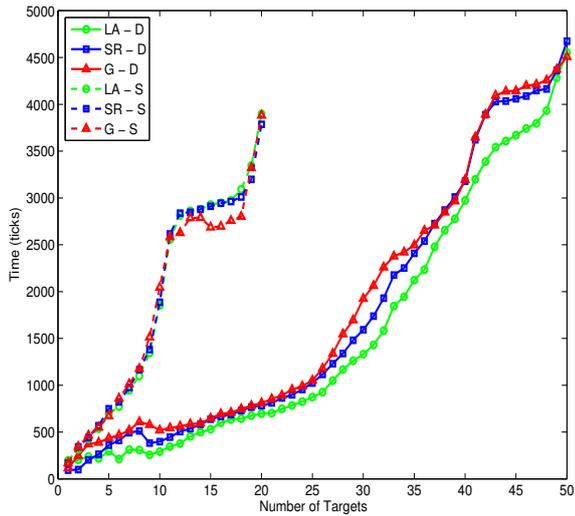


Fig. 8. Comparison of prioritization strategies, Look Ahead (LA), Shared Rewarding (SR) and Greedy (G) for dense (D) and sparse (S) target density

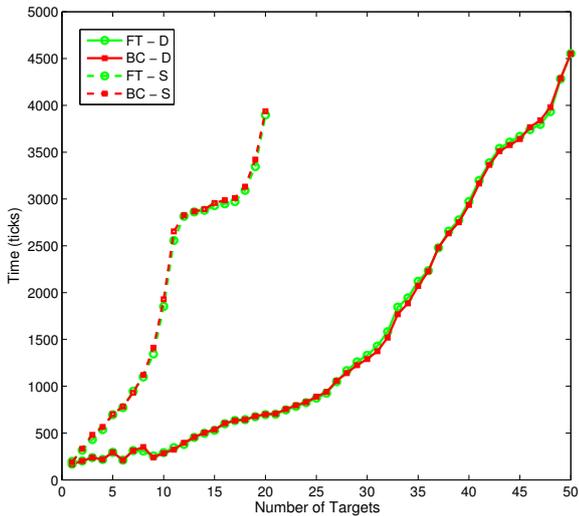


Fig. 9. Comparison of trajectory estimates, Fastest Turn (FT) and Big Circle (BC) for dense (D) and sparse (S) target density

An analysis of number of targets that are left unclassified was also performed. The results are tabulated in Table III. Leader Guided swarms had a much lower number of unconfirmed targets. Although, Shared Rewarding takes higher time than Look Ahead, it has a very low percentage of unconfirmed targets. Numbers obtained here are higher (16 %, 22.5 %), due to popup and popup cluster threats. Since, popups are activated at later time instants, they take more time to classify.

V. CONCLUSIONS

We presents a solution for ATR in adversarial environments with popup and cluster threats. Our work also tries to show how Super-Agents can act as Leaders to the swarm and

increase search performance. Look Ahead strategy has better search performance in terms of time to search. But the number of unconfirmed targets is low for Shared Rewarding strategy, for both sparse and dense target distributions. Shared Rewarding a greed based synergistic strategy acts as a bridge between look ahead and greedy strategies, incorporating benefits of both. Shared Rewarding is hence a better prioritization strategy. Trajectory estimation although does not drastically influence search performance, does make it better for sparse density environments.

Swarming algorithm provided above can have tremendous applications in search and situation assessment. It is superior to other ATR algorithms due to the allowed heterogeneity in the agent pool. It allows us to expand agent definition and include other types of agents like All Terrain Vehicles (ATV's), Autonomous Underwater Vehicles (AUV's) and Human Agents as well.

REFERENCES

- [1] P. Dasgupta, *A Multi agent Swarming System for Distributed Automatic Target Recognition Using Unmanned Aerial Vehicles*, IEEE Transactions on Systems, Man and Cybernetics, Vol. 38, No.3, May 2008.
- [2] Akat, S. B., Gazi, V., and Marques, L. *Asynchronous particle swarm optimization-based search with a multi-robot system: simulation and implementation on a real robotic system*. Turkish Journal of Electrical Engineering and Computer Sciences, 18(5), 749-764, 2010.
- [3] Krishnanand, K. N., and Ghose, D.. *A glowworm swarm optimization based multi-robot system for signal source localization*. In Design and Control of Intelligent Robotic Systems (pp. 49-68). Springer Berlin Heidelberg, Chicago, 2009.
- [4] Y. Yang, A. Minai and M. Polycarpou, *Decentralised Cooperative Search in UAV's using Opportunistic Learning*, Proceedings of AIAA GNC Conference and Exhibit. 2002.
- [5] P. Chandler, M. Pachter and S. Rasmussen, *UAV Cooperative Control*, In Proceedings of ACC, pp. 1853-1856, June 2001
- [6] H. Choset, *Coverage for Robotics: A survey of recent results*, Annals of Mathematics and Artificial Intelligence, 2001.
- [7] J. Kim and P. Khosla, *Real Time Obstacle Avoidance Using Harmonic Potential Functions*, IEEE Transactions on Robotics and Automation. Vol. 8 No., June 1992.
- [8] M. Hao, T. Yantao and Z. Linan, *A Hybrid Ant Colony Optimization Algorithm or Path Planning of Robot in Dynamic Environment*, International Journal of Information Technology, Vol. 12, No. 3, 2006.