Cooperative control of UAV swarm via information measures

Haoyang Cheng, John Page and John Olsen
The School of Mechanical and Manufacturing Engineering, University of New South Wales, Sydney, Australia

Abstract

Purpose – This study aims to investigate the rule-based decentralised control framework for a swarm of UAVs carrying out a cooperative ground target engagement mission scenario.

Design/methodology/approach – This study is to investigate the rule-based decentralised control framework for missions which require high-level cooperation between team members. The design of the authors' control strategy is based on agent-level interactions. Different to a centralized task assignment algorithm, the cooperation of the agents is entirely implicit. The behaviour of the UAVs is governed by rule sets which ultimately lead to cooperation at a system level. The information theoretic measures are adopted to estimate the value of possible future actions. The prediction model is further considered to enhance the team performance in the scenario where there are tight coupled task constraints.

Findings – The simulation study evaluates the performance of the decentralised controller and compares it with a centralised controller quantitatively. The results show that the proposed approach leads to a highly cooperative performance of the group without the need for a centralised control authority. The performance of the decentralised control depends on the complexity of the coupled task constraints. It can be improved by using a prediction model to provide information such as the intentions of the neighbours that is not available locally.

Originality/value – The achievable performance of the decentralised control was considered to be low due to the absence of communication and little global coordinating information. This study demonstrated that the decentralised control can achieve highly cooperative performance. The achievable performance is related to the complexity of the coupled constraints and the accuracy of the prediction model.

Keywords UAV, Decentralized control, Simulation, Multi-agent system, Flight, Flight control, Decentralized control

Paper type Research paper

Nomenclature

$\boldsymbol{F}$	transition matrix
$\boldsymbol{z}$	observation vector
$\boldsymbol{w},\boldsymbol{Q}$	process noise and its covariance, respectively
$\mathbf{v}$	measurement noise
$\mathbf{i}$, $\mathbf{I}$	information state contribution and its associated information matrix, respectively
$\mathbf{c}, \mathbf{R}$	observation noise and covariance matrix, respectively
$\hat{\mathbf{y}}, \mathbf{Y}$	information state and information matrix, respectively
$\mathbf{x},\mathbf{P}$	target state and state covariance, respectively
$H(X)$, $h(X)$	entropy of discrete random variables and continuous random variable X, respectively
$I(x, z)$	mutual information obtained about a random variable x with respect to a random variable z
α	bearing angle of the target measured by the vehicle
ξ, η	distances of the target in x and y directions
$\dot{\xi}$, $\dot{\eta}$	velocities of the target in x and y directions

International Journal of Intelligent Unmanned Systems
Vol. 1 No. 3, 2013
pp. 256-275

2049-6427
DOI 10.1108/IJIUS-01-2013-0001

1. Introduction

With recent technological advances in autonomous control and communication, multi-robotic systems are receiving a great deal of attention due to their increased ability to carry out complex tasks in a superior manner when compared to single-robotic systems. Multiple autonomous agents working in a group exceed the sum of the performance of the individuals. With rapid technology development in unmanned aerial systems (UAS), onboard communication and computation capacity, intensive research has been carried out on the cooperative control of groups of UAVs. The results of this research on such systems can be found in search and rescue, intelligence, surveillance and reconnaissance (Bourgault *et al.*, 2003; Furukawa *et al.*, 2006; Nigam and Kroo, 2008; Ceccarelli *et al.*, 2007). The cooperative control of UAVs is a complex problem that is dominated by uncertainty, limited information and task constraints. Both centralised and decentralised, decision and control algorithms have been developed to address this complexity. Centralised decision and control algorithms optimise timing and task constraints but require intensive computation and robust communication. Decentralised decision and control algorithms trade optimality and predictability with robustness and adaptation to environmental changes. A self-organised (SO) system, or swarm, is typically a decentralised control system made up of autonomous agents that are distributed in the environment and follow stimulus response behaviours (Garnier *et al.*, 2007). Examples from social insects, such as foraging and the division of labours show that SO systems can generate useful emergent behaviours at the system level.

Significant research effort has been invested in recent years into the design and simulation of intelligent swarm systems (Bonabeau *et al.*, 1999). Intelligent swarm systems can generally be defined as decentralised systems, comprised of relatively simple agents which are equipped with the limited communicational, computational and sensing abilities required to accomplish a given task (Altshuler *et al.*, 2006). Applying these concepts to a UAV swarm was first explored by Frelinger *et al.* (1998). They examined whether modern communication, sensors and technologies in robotic architecture would permit the development of decentralised control to command a swarm of low cost munitions. Gaudiano *et al.* (2003) extended the work of Frelinger. In their paper, they adopted random, repulsion, pheromone and global decentralised control strategies. In the research of Price (2006), ten self-organisation rules were implemented whose weight factors were collected into a single fitness function. This function was further refined using a genetic algorithm (GA) within the simulation. A similar technique was also adopted by De Vries and Subbarao (2011) who used a potential function to generate steering commands to control a swarm of quadrotors. Another widely adopted mechanism is digital pheromone maps that imitate the foraging behaviour of ants. Digital pheromones are modelled on the pheromone fields of the individual vehicles. By synchronising their maps the UAVs coordinate to avoid redundant searches (Erignac, 2007). Similar to digital pheromone maps, Scerri *et al.* (2007) used a binary Bayesian grid filter to represent the probability that there is a radio frequency emitter within the global map and applied a Rapidly expanding Random Tree path planner to determine UAV paths for locating targets.

The previous research that applies swarm methodology was limited to relative simple scenarios which do not have complex task constraints. In this study, the problem of assigning multiple UAVs to time-dependant cooperative tasks is investigated. A time-dependent cooperative task is one that requires multiple agents to perform separate subtasks simultaneously, or within the available time windows of cooperative agents (i.e. two or more agents are assigned subtasks within the same mission, and the agents have limited time windows to complete their tasks). Such problems have mainly been investigated by using centralised task assignment algorithms. Kingston and Schumacher (2005) in their paper solved this problem with a Mixed Integer Linear Programme (MILP) that addressed task timing constraints and agent dynamic constraints to generate a flyable path. A GA is used to efficiently search the space of possible solutions and provide the cooperative assignment (Shima and Schumachery, 2005). The following assumptions and restrictions were added to their studies: first, targets had constant heading. This assumption can be justified for targets travelling along known roads. Second, targets were stationary since the sensor footprint is much larger than the distance travelled by the target during the simulated scenario. Since MILPs are inherently non-deterministic polynomial time NP-complete, this algorithm scales exponentially with the number of UAVs. Moreover, in the presence of moving targets, the whole solution must be recalculated as the states of the targets change, making this approach unfeasible as a real-time solution. Third, the centralised assignment algorithm can be computed by either a centralised agent (e.g. command centre or team leader) or redundantly by each of the UAVs. With either approach, up-to-date information on each UAV's state consisting of position, heading and velocity, has to be shared across the whole network. Any fault in vehicle state measurements and vehicle to vehicle communications might cause uncoordinated assignments or inconsistent team decision making.

The main contribution of this research is the investigation of the decentralised control framework that enables efficient operations for a large-scale UAS performing a cooperative task when a centralised decision maker cannot or does not exist. The proposed methodology is designed based on agent level interactions through the combination of direct interaction and indirect interaction. Direct interactions are the obvious interactions where agents, in given circumstances, send out requests and other agents who received the requests later respond. Indirect interactions are made via stigmergy. Stigmergy is the mechanism within which individual behaviour modifies the environment, which in turn modifies the behaviour of other individuals (Bonabeau *et al.*, 1999). In this study, stigmergy is used to coordinate the behaviour of the UAVs during the tracking stage via mutual information. Information theory has been widely used in the area of sensor management. Mutambara and Durrant-Whyte (1994) proposed a decentralised data fusion algorithm using information theoretic measures. They later considered sensor management and control essentially as maximising the information gain of the network as a whole (Durrant-Whyte and Grocholsky, 2003). A combination of particle filtering for non-parametric density estimation, information theoretic measures for comparing possible action sequences and artificial physics for providing approximate cooperation between sensor nodes were presented to addresses the problem of sensor management for a large network of agile sensors (Kreucher *et al.*, 2007). Ryan *et al.* (2007) develops a receding horizon control formulation with multiple step lookahead for a team of sensors to cooperatively search for and track a target, which has the potential to produce far better solutions by accounting for sensor motion constraints and delayed payoff. The Fisher information matrix was also considered to quantify the information provided by the measurements, which lead to trajectories designed for UAV tracking ground targets with vision based sensors (Ponda *et al.*, 2009). Information theoretic measures have also been applied in exploration missions to generate informative paths for unmanned platforms (Singh *et al.*, 2009; Yang *et al.*, 2013). For this work, we do not address the path planning problem for tracking UAVs. The tracking UAV only predicts the mutual information gain for the next control step.

Our main focus in this study is to evaluate the efficiency of task allocation of the overall group resources over the multiple targets.

The remainder of this paper is structured as follows. Following the problem statement in Section 2, Section 3 describes the rule-based control framework. Simulation results and analysis are presented in Section 4 and a summary of the study is given in Section 5.

2. Problem statement

The problem of interest is a dynamic variant of the known cooperative moving target engagement mission, described and analysed by Kingston and Schumacher (2005). In this scenario, a group of UAVs is operating over a defined area tracking and attacking several ground based moving targets. Instead of assuming the targets are stationary relative to the UAVs as in previous research, the target motion is represented using a state-space model with motion uncertainty. Each UAV is assumed to be equipped with radar-based Ground Moving Target Indicator (GMTI) to measure the locations and range rate of the ground targets relative to the UAVs and a sensor to detect the presence of other UAVs within a predefined distance. This mission requires that the target be continuously tracked by two vehicles during the attack phase while an additional UAV launches a guided weapon. The information from the tracking vehicles is fused to form a precise target location for targeting the weapon. Since the cooperative search algorithm for distributed, independent swarm UAS has been addressed in previous research (Pack and Mullins, 2003; Erignac, 2007), we remove the global search stage from the scenario assuming the initial positions of the targets are available to the UAVs.

The GMTI sensor footprint in this study is assumed to be a circular sector-shape with a minimum and maximum radius. The GMTI sensors are Doppler based which means that a moving ground target can only be detected and tracked if the range rate relative to the vehicle is above some minimum detection velocity. Thus the tracking UAVs have to stay within some offset angle from the heading of the moving target, to maintain this relative velocity. Figure 1 shows the heading of the target and the associated target's region of detectability (light grey area). Within the target detection region, the onboard GMTI radars can detect and track the target's motion. The sensor footprint (dark blue area) is shown pointing from the side of the UAV. As the UAV flies along the arc of a circle centred on the target, the sensor footprint can lock on the target

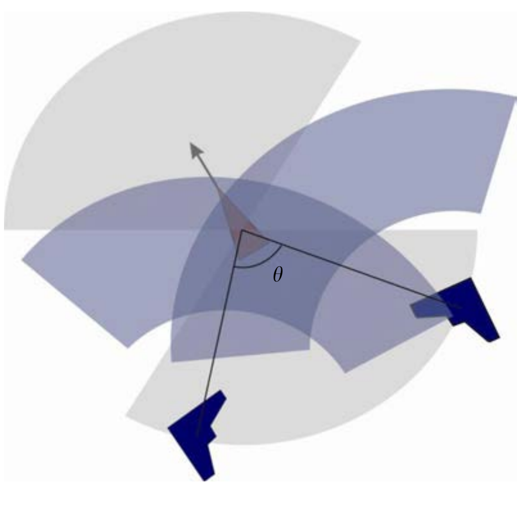

Figure 1.
Schematic diagram for cooperative tracking

constantly. To further complicate the problem, the difference in bearing angles θ of the tracking UAVs to the target, are required to be greater than a certain value to reduce the target location estimation error. The target kinematic model and GMTI measurement model are presented in the following.

The primary objective of target tracking is to estimate the states of the target. A target is usually treated as a point object in tracking problems. The kinematic model of the target describes the evolution of the target states. The target kinematic model considered in this study is a commonly used model called the discrete white noise acceleration model (Rong Li and Jilkov, 2003). The corresponding state-space model is given by:

$$\mathbf{x}(k+1) = \mathbf{F}(k)\mathbf{x}(k) + \mathbf{G}(k)\mathbf{w}(k), \tag{1}$$

where the state vector $\mathbf{x} = [\,\xi, \dot{\xi}, \eta, \dot{\eta}]$. ξ and η are the distances of the target in x and y directions from the origin point. The corresponding velocities are thus $\dot{\xi}$ and $\dot{\eta}$.$\mathbf{w}(k)$ is the constant velocity of the target during the time period k, $\mathbf{w}(k)\Delta T$ is the increase in the velocity of the target during this period, while $\mathbf{w}(k)\Delta T^2/2$ is the average acceleration of the target during this period. $\mathbf{w}(k)$ is modelled as a white Gaussian noise sequence denoted by $\mathbf{w}(k) \sim N(0, \mathbf{Q}(k))$, where:

$$\mathbf{Q}(k) = \begin{bmatrix} \sigma_\xi^2 & 0 \\ 0 & \sigma_\eta^2 \end{bmatrix} \tag{2}$$

The standard deviation of process noise σ_ξ and σ_η is set to be 0.5 metres per second. The transition matrix is:

$$\mathbf{F}(k) = \begin{bmatrix} 1 & \Delta T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta T \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

and the vector gain multiplying the process noise is given by:

$$\mathbf{G}(k) = \begin{bmatrix} \Delta T^2/2 & 0 \\ \Delta T & 0 \\ 0 & \Delta T^2/2 \\ 0 & \Delta T \end{bmatrix} \tag{4}$$

To simplify the problem, we assume that the dynamic model of the target is known to the UAVs. In many practical situations, the dynamic model of the target being tracked is not available to the tracker or it may possibly be time-varying, e.g. manoeuvring targets. The techniques used to deal with such problems can be found in Bar-Shalom *et al.* (2001).

In this study, the UAVs are assumed to be equipped with GMTI sensors (Bar-Shalom, 2000). The measurement equation is given by:

$$\mathbf{z}(k) = \mathrm{h}(\mathbf{x}(k), k) + \mathbf{v}(k) \tag{5}$$

The measurement vector $\mathbf{z}(k)$ comprises positions in the x and y directions and range rate $\dot{r}$ where:

$$\dot{r}(k) = \dot{\xi}(k)\cos\alpha(\mathbf{x}(k), \mathbf{x}_v(k)) + \dot{\eta}(k)\sin\alpha(\mathbf{x}(k), \mathbf{x}_v(k)) \tag{6}$$

where $\alpha(\mathbf{x}(k), \mathbf{x}_v(k))$ is the bearing angle of the target measured by the vehicle at time k.

The sensor measurement matrix is given by:

$$h(\mathbf{x}(k), k) = \begin{bmatrix} & 1 \quad 0 & & 0 \quad 0 \\ & 0 \quad 0 & & 1 \quad 0 \\ 0 & \cos\alpha(\mathbf{x}(k), \mathbf{x}_v(k)) & 0 & \sin\alpha(\mathbf{x}(k), \mathbf{x}_v(k)) \end{bmatrix} \tag{7}$$

$\mathbf{v}(k)$ in Equation (5) is the Gaussian measurement noise vector denoted by $\mathbf{v}(k) \sim N(0, \mathbf{R}(k))$, where:

$$\mathbf{R}(k) = \begin{bmatrix} \sigma_\xi^2(k) & \sigma_{\xi\eta}^2(k) & 0 \\ \sigma_{\eta\xi}^2(k) & \sigma_\eta^2(k) & 0 \\ 0 & 0 & \sigma_{\dot{r}}^2 \end{bmatrix} \tag{8}$$

The error statics for sensor measurements are given in terms of the range standard deviation σ_r, the range rate standard deviation $\sigma_{\dot{r}}$ and the bearing angle standard deviation σ_α, which are known. In this exercise, they are set to correspond to 5 metres, 1 metre per second and 0.001 radians. With these and the position variances, the covariance can be calculated as:

$$\begin{aligned} \sigma_\xi^2(k) =& r(\mathbf{x}(k), \mathbf{x}_v(k))^2\sigma_\alpha^2 \sin\alpha(\mathbf{x}(k), \mathbf{x}_v(k))^2 \\ &+ \sigma_r^2 \cos\alpha(\mathbf{x}(k), \mathbf{x}_v(k))^2 \end{aligned} \tag{9}$$

$$\begin{aligned} \sigma_\eta^2(k) =& r(\mathbf{x}(k), \mathbf{x}_v(k))^2\sigma_\alpha^2 \cos\alpha(\mathbf{x}(k), \mathbf{x}_v(k))^2 \\ &+ \sigma_r^2 \sin\alpha(\mathbf{x}(k), \mathbf{x}_v(k))^2 \end{aligned} \tag{10}$$

$$\begin{aligned} \sigma_{\eta\xi}^2(k) =& (\sigma_r^2 - r(\mathbf{x}(k), \mathbf{x}_v(k))^2\sigma_\alpha^2) \sin\alpha(\mathbf{x}(k), \mathbf{x}_v(k)) \\ &\times \cos\alpha(\mathbf{x}(k), \mathbf{x}_v(k)) \end{aligned} \tag{11}$$

where range of the target $r(x(k), \mathbf{x}_v(k))$ measured from each vehicle is evaluated at $\mathbf{x}(k) = \hat{\mathbf{x}}(k|k-1)$.

3. Decentralised cooperative control

In this section, a rule-based control framework, which effectively generates cooperative behaviours of multiple UAVs is presented. A swarm system can be defined as a decentralised system made of autonomous agents that are distributed in the environment and follow simple probabilistic stimulus-response behaviours (Garnier *et al.*, 2007). The individuals within the swarm system are not able to assess a global situation and control the tasks to be carried out by the other agents, i.e. there is no supervisor in a self-organising swarm system. For example, each time an agent performs an action, the local environment is modified by this action. The new environmental configuration will then influence the future actions of other agents. This process leads to the emergent properties of the colony at the system level. In this rule-based approach, each vehicle can switch between four behavioural states; these being cruise, standby, attack and track. In each state, the behaviour of the UAVs is governed by local rules. The switch between behavioural states is triggered by changes of local environment, e.g. target states and request received from neighbours.

3.1 Information theoretic cooperative tracking

In order to track the targets, the UAVs in track mode have to satisfy the sensor geometry constraints and maintain separated line-of-sight angles to the target. The estimation error in the position of the moving target can be reduced by multiple sensors with separated line-of-sight angles to the target, with preferably orthogonal views (Schumacher, 2005). This is treated in previous research by restricting the difference in bearing angles of the UAVs to the target to be >45 degrees. Different to the previous research, the angle separation requirement is not enforced by constraints. Since the performance of the target estimation is dependent upon the positions from which measurements are taken relative to the target and the information content of a set of measurements can be quantified and recursively computed by an information filter, our methodology is to let the tracking UAVs fly trajectories that locally maximise the information gain for the next measurement. By doing this, the angle separation will be formed automatically.

The standard version of the Kalman filter uses the estimation of states $\mathbf{x}(i)$ and calculates the gain with a recursive computation of the state covariance $\mathbf{P}(i|j)$ (Crassidis, 2004). The information filter is derived from the Kalman filter in terms of the information states vector $\hat{\mathbf{y}}(i|j)$ and the information matrix $\mathbf{Y}(i|j)$. The information state vector and information matrix are defined as:

$$\hat{\mathbf{y}}(i|j) = \mathbf{P}^{-1}(i|j)\hat{\mathbf{x}}(i|j) \tag{12}$$

$$\mathbf{Y}(i|j) = \mathbf{P}^{-1}(i|j) \tag{13}$$

Because the target motion state and the sensor measurement used in our study has a non-linear relationship (as given in Equation (5)), we utilise an extended information filter (Mutambara, 1999). The prediction step of the filter is given by:

$$\hat{\mathbf{y}}(k|k-1) = (1 - \mathbf{\Omega}(k)\mathbf{G}^T(k))\mathbf{F}^{-T}(k)\hat{\mathbf{y}}(k-1|k-1) \tag{14}$$

and

$$\mathbf{Y}(k|k-1) = \mathbf{M}(k) - \mathbf{\Omega}(k)\mathbf{\Sigma}(k)\mathbf{\Omega}^T(k) \tag{15}$$

where

$$\mathbf{M}(k) = \mathbf{F}^{-T}(k)\mathbf{Y}(k-1|k-1)\mathbf{F}^{-1}(k) \tag{16}$$

$$\mathbf{\Omega}(k) = \mathbf{M}(k)\mathbf{G}(k)\mathbf{\Sigma}^{-1}(k) \tag{17}$$

and

$$\mathbf{\Sigma}(k) = \mathbf{G}^T(k)\mathbf{M}(k)\mathbf{G}(k) + \mathbf{Q}^{-1}(k) \tag{18}$$

while the estimation step can be written as:

$$\hat{\mathbf{y}}(k|k) = \hat{\mathbf{y}}(k|k-1) + \mathbf{i}(k), \tag{19}$$

$$\mathbf{Y}(k|k) = \mathbf{Y}(k|k-1) + \mathbf{I}(k) \tag{20}$$

The information state contribution $\mathbf{i}(k)$ and its associated information matrix $\mathbf{I}(k)$ are given, respectively, as:

$$\begin{aligned} \mathbf{i}(k) =& \nabla\mathbf{h}_x(k)^T(k)\mathbf{R}^{-1}(k)(\mathbf{z}(k) - \mathbf{h}(\hat{\mathbf{x}}(k|k-1), k) \\ &+ \nabla\mathbf{h}_x(k)\hat{\mathbf{x}}(k|k-1)) \end{aligned} \tag{21}$$

and

$$\mathbf{I}(k) = \nabla\mathbf{h}_x(k)^T(k)\mathbf{R}^{-1}(k)\nabla\mathbf{h}_x(k) \tag{22}$$

where the Jacobian $\nabla\mathbf{f}_x(k)$ is evaluated at $\mathbf{x}(k-1) = \hat{\mathbf{x}}(k-1|k-1)$and $\nabla\mathbf{h}_x(k)$ is evaluated at $\mathbf{x}(k) = \hat{\mathbf{x}}(k|k-1)$. Compared with a Kalman filter, the estimation step of the information filter simplifies while the complexity in the prediction step increases. The initial condition for the information filter is set as $\hat{\mathbf{y}}(0|0) = 0$regardless of $\hat{\mathbf{x}}(0|0)$ (Bar-Shalom *et al.*, 2001).

Information filters play an important role in decentralised data fusion problems that involve multiple information sources. As for a decentralised system, data fusion and vehicle control occur locally at each individual agent with local observation and the information communicated from neighbouring agents. In the conventional approaches to state estimation, it is difficult to capture the statistical relationships that exist between estimates produced by combinations of observations (Durrant-Whyte and Grocholsky, 2003). The information filter formulation has an additive structure that can be exploited by distributed decision-making algorithms.

For N sensor information sources, the posterior information state and information matrix are obtained from:

$$\hat{\mathbf{y}}(k|k) = \hat{\mathbf{y}}(k|k-1) + \sum_{i=1}^{N} \mathbf{i}_i(k) \tag{23}$$

$$\mathbf{Y}(k|k) = \mathbf{Y}(k|k-1) + \sum_{i=1}^{N} \mathbf{I}_i(k) \tag{24}$$

where $\mathbf{i}_i(k)$ is the information matrix and $\mathbf{I}_i(k)$ represents the information state contributions of the sensors $i = 1, \ldots, N$. This is assuming that the sensors are fully connected with no communication failures or delays. The posterior target state estimation is obtained from:

$$\hat{\mathbf{x}}(k|k) = \mathbf{Y}^{-1}(k|k)\hat{\mathbf{y}}(k|k) \tag{25}$$

The entropic information of the estimated target j states is given by:

$$i(k) = \frac{1}{2}\log[(2\pi e)^{-n}|\mathbf{Y}^{i,j}(k|k)|] \tag{26}$$

The log is to the base 2 and entropy is expressed in bits (Cover and Thomas, 2006). The mutual information gain $I^{i,j}_{Track}(k)$ for vehicle i tracking target j is calculated from:

$$I^{i,j}_{Track}(k) = \frac{1}{2}\log\left[\frac{|\mathbf{Y}^{i,j}(k|k-1)+\mathbf{I}^{i,j}(k)|}{|\mathrm{Y}^{i,j}(k|k-1)|}\right] \tag{27}$$

From Equation (22), it can be found that the information matrix $\mathbf{I}^{i,j}(k)$ results directly from the geometric position of a UAV relative to the target. Thus, the mutual information will give an a priori measure of the information to be gained from potential sensing actions.

To test and verify this methodology, we setup a simulation with two UAVs tracking one target from different line-of-sight angle separations. The simulation results of the track estimation error against the separation angle using the sensor model presented in the previous subsection are as follows. Figure 2a shows that the minimum estimation error was achieved with a separation angle between the two UAVs of approximately 90 degree line-of-sight. Figure 2b shows the entropic information plotted against separation angle. The figure suggests that the information measure on the target is directly related to the estimation error. It follows that the more information we have on the target, the more accurate the estimation of both its position and velocity, provided that the sensors are appropriately displaced. This result verified that navigating towards the location from where the observation is more informative (i.e. mutual information gain calculated from Equation (27) can lead to more accurate estimation about the target's states and also maintain the desired angle separation requirement.

3.2 Four basic behaviours

A. Cruise. The UAV is in cruise mode if it is not engaged with any target. This behaviour has three guidance rules that navigate the UAV towards the nearest target, depending on the request it received:

- If there is no request received, as shown in Figure 3a, the UAVs will fly toward the nearest target and switch to standby mode once it reaches the standoff distance.
- If the UAV receives a request for tracking support from the standby vehicle, as shown in Figure 3b, it will fly towards the tracking region of the target and switch to track mode once it arrives. If the UAVs are already in the tracking region when it receives the request, they will switch to track mode immediately.

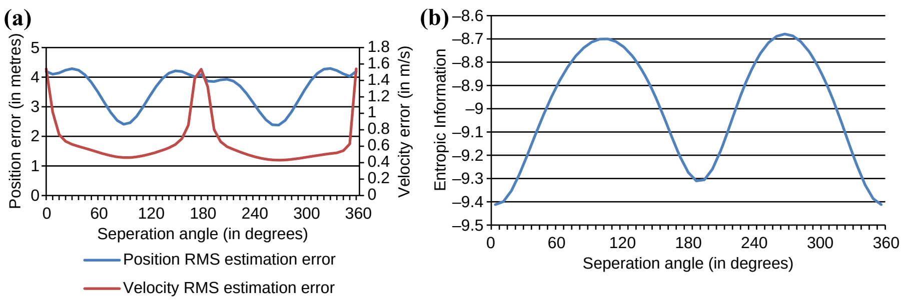

Figure 2.
(a) Target position root mean square (RMS) estimation error and target velocity RMS estimation error; (b) target entropic information

- If the UAVs receive request for tracking support, at the same time it also receives the local track data from another tracking vehicle. As shown in Figure 3c, it will start running the track data through the data fusion process and fly toward the location from where the observation can be most informative. Once it arrives, it switches to track mode.

B. Standby. In the standby mode, the UAVs circle around the target with a predefined standoff distance. The UAVs receive local track data from all the tracking UAVs and use decentralised the data fusion algorithm to process the track (Figure 4). Once the target track is defined accurately enough for the guided weapon, the standby UAVs switch to attack state. This accuracy is measured by the entropic information on the target.

C. Attack. The attack mode is simply modelled as the UAVs heading towards the target and launching the weapon from a predefined distance from the target (Figure 4). The UAVs tracking the target must track the target for the duration of the weapon flight. Once the target is destroyed, all the UAVs engaged to this target switch back to cruise state.

D. Track. When a UAV is in the track mode, its motion is governed by two rules:

- The first rule the tracking UAVs follow is to place the target within the sensor footprint while remaining within the detectable line-of-sight angle to the target heading.
- The second rule is to increase the entropic information on the target. The control and estimation process is conducted in discrete time. The tracking UAVs fly at constant velocity with their heading as the decision variable. They will request local track information, i.e. information states $\hat{y}(k|k)$and the information matrix

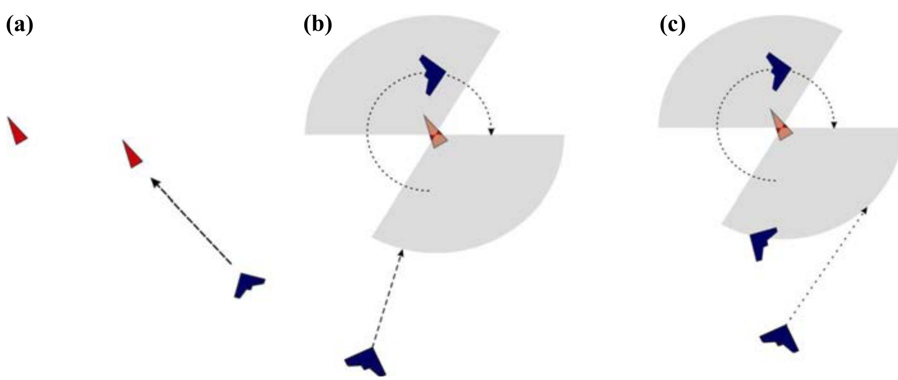

Figure 3.
(a) Guidance law for cruise mode; (b) guidance law for cruise mode; (c) guidance law for cruise mode

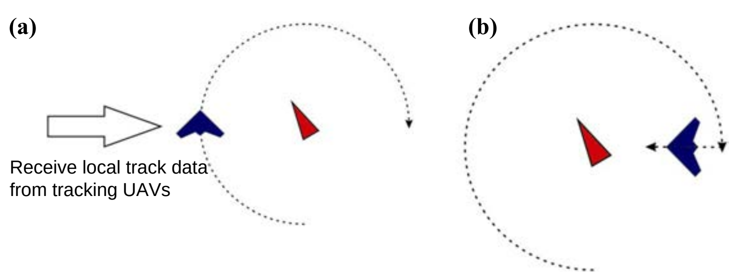

Figure 4.
(a) Standby mode; (b) attack mode

$Y(k|k)$ from another tracker and compute the desired heading command that maximise the mutual information gain for the next observation step. This process will work recursively until the track is accurate enough to trigger the attack.

In terms of communication, the track information is only shared among the two tracking UAVs and the attacking UAV. Figure 5 shows a schematic block diagram of the sensor and communication network for engaging a target. As shown in the diagram, two tracking UAVs make observations on the targets and maintain local tracks. The local tracks are shared between two tracking UAVs and used for calculating mutual information. The information states and information matrices are sent to the attacking UAVs to fuse a global track. Algorithm 1 summarises the behaviours that are followed by each UAV and the conditions under which they switch modes (Figure 6).

4. Simulation and analysis

4.1 Lyapunov potential field vs information measures

In this cooperative moving target engagement scenario, the ability to generate accurate tracks on the targets is critical to mission performance. The guidance rules have to enable the tracking UAVs to fly trajectories that increase the amount of information provided by the measurements and improve target states estimation, resulting in proper target tracking and an accurate target location estimate, without violating the dynamics constraints and sensor restrictions. In this study, we compare the proposed cooperative tracking method with decentralised method based on potential field. The potential field method has been investigated to generate a guidance vector field that provides the desired velocity of the unmanned systems (Rimon and Koditschek, 1992; Dale, 2003; Nelson *et al.*, 2006). The specific method we adopted in this comparison is from Frew *et al.* (2008) work. Their method is based on a Lyapunov guidance vector field that produces stable convergence to a loiter circle of predefined radius. Cooperative

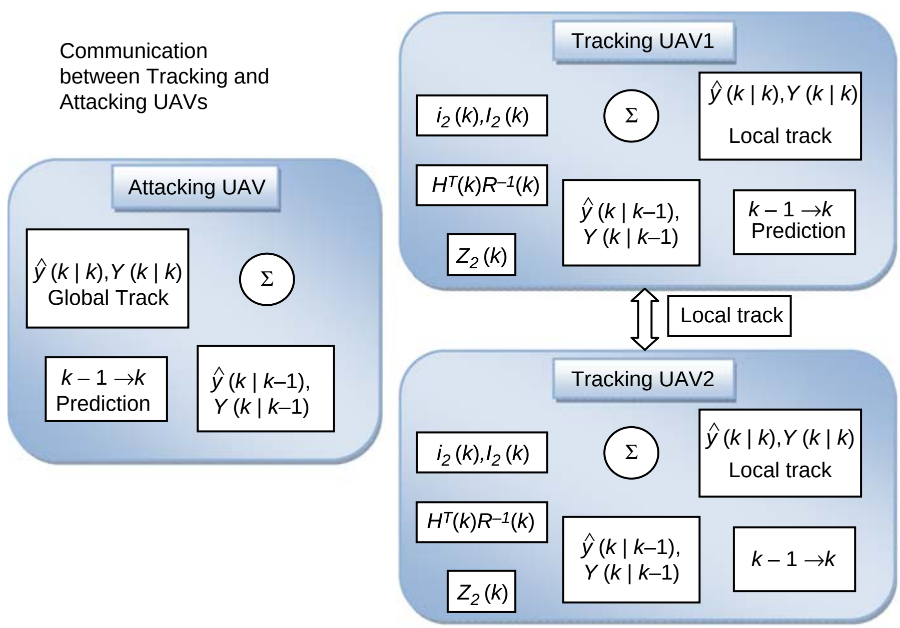

Figure 5.
Schematic block diagram of data fusion

```
Algorithm 1
1:  switch behaviour
2:  case Cruise (default behaviour at the beginning of the mission)
3:        if  (no request received)
4:              Fly toward the nearest target; Change to standby once it reaches the standoff distance.
5:        elseif (tracking requestreceived)
6:              Fly toward tracking region of the target; Change to track once it arrives.
7:        elseif (tracking request and receive ŷ (k | k),Y(k | k) from other tracking UAV received)
8:              Compute the desired heading command that maximise the mutual information gain for
                the next observation step; Change to track once it arrives.
9:        end if
10: case Standby
11:       Circle around the target; Send request for tracking support; Process the target track;
          Change to attack when the level of accuracy is achieved.
12: case Attack
13:       Release weapon; Change back to cruise.
14: case Track
15:       if (target tracked by single UAV)
16:             Track target by maintain sensor coverage.
17:       elseif (target tracked by cooperative UAVs)
18:             Perform cooperative tracking (step 8); Change back to cruise once the target is
                destroyed.
19:       end if
20: end while
```

Figure 6.
Pseudo code for rule-based control architecture

tracking by multiple unmanned aircraft is achieved through additional phasing control law which adjusts the speed of the vehicles to maintain a desired relative phase on the loiter circle. The phase separation is set to be 90 degrees to minimise the achievable error variance (Gu *et al.*, 2006).

Figure 7 shows the flight trajectories generated by these two methods while tracking a ground moving target from stand-off distance. It can be seen from Figure 7a, by following the guidance vector field, that the UAVs are attracted to the loiter circle about the target vehicle, and maintain the desired phase angle separation. Such coordination is achieved by constantly sharing the position between the two UAVs. In contrast, the UAVs using the proposed method head toward the tracking region in the early stage without any knowledge of other UAVs. As the UAV 2 is initially located nearer to the target, it starts tracking the target earlier than UAV 1. Figure 7b shows that UAV 1 receives track data from UAV 2 and plans its next move to maximise information which causes UAV 1 to loiter counter clockwise about the target.

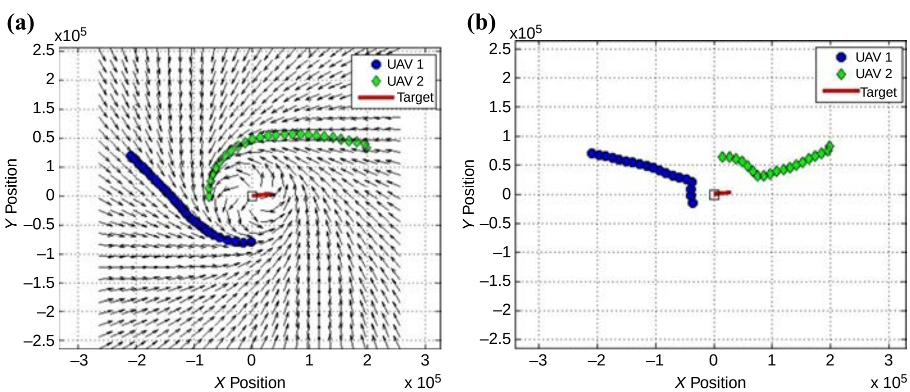

Figure 7.
(a) Cooperative tracking via Lyapunov potential field; (b) cooperative tracking via information measures

In order to investigate the performance of these two techniques, the Monte Carlo method uses a total sample size of 100. The two UAVs are randomly generated between 40 kilometres and 55 kilometres from the target. We record the time when the entropic information on the target reaches −8.8 which indicates the accuracy of the estimation. The simulation results show that the UAVs spend an average of 748 seconds to reach the threshold using potential field method while they spend an average of 348 seconds using information measures. This indicates the effectiveness of the information theoretic framework applied to cooperative tracking. Even without coupled objective functions and dynamics, i.e. no planning information is shared between UAVs; the UAVs can still demonstrate the capability to perform cooperative tracking in a time critical scenario. In contrast, the Lyapunov potential field method is more suitable for persistent tracking as it can produce an asymptotic stability to a desired loiter pattern.

4.2 Centralised control vs swarm control

In this section, the simulation results of the complete cooperative moving target engagement scenario are presented. First an example scenario is used to explain the proposed rule-based control framework. This scenario includes seven UAVs and three ground targets. The mission area is a square whose area is 100 square kilometres. The UAVs fly at a constant speed, governed by flight dynamics, of 110 metres per second. Their behaviours are governed by the rule sets presented in Section 3.2. The targets are initially randomly distributed over the mission area and their motions are defined by Equation (1). Figure 8a-c display stages of the simulated scenario at three points in time.

At 188 seconds, UAVs 1 (red) and 4 (yellow) were designated to assist in tracking target 1 in cooperation with UAV 7 (orange). The coloured area displays the sensor footprint of the tracking UAVs. The UAVs' sensor footprints have the same colour as the UAVs themselves. It can be noted that UAV 1 changed its heading once UAV 4 started tracking as a result of the tracking rule, i.e., heading towards the location where the measurement could yield more information. This action caused UAVs 1 and 4 to maintain an appropriate line-of-sight separation angle while tracking the target. On the right side, UAV 6 (cyan) were tracking target 3 while UAV 5 (magenta) was in standby mode waiting for tracking support from the second vehicle. As can be seen from the flight path, UAV 2 (green) was heading towards target 2 initially, and then changed its course to help tracking target 3 after it received a tracking request from standby UAV 5.

After 213 seconds, UAV 1 started tracking target 3 after target 1 was destroyed and the track information became accurate enough for UAV 5 to attack target 3. UAV 2 switched back to target 2 because tracking support was no longer needed for target 3.

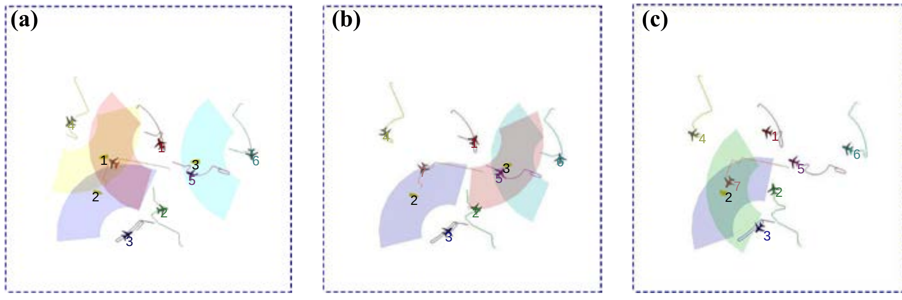

Figure 8.
(a) Simulated scenario at time 188 seconds; (b) simulated scenario at time 213 seconds; (c) simulated scenario at time 279 seconds

After 279 seconds, UAVs 2 and 3 were tracking target 2 cooperatively while UAV 7 was attacking. The information filter results for target 1 are presented in Figure 9. It shows the position estimation error and standard deviation bounds for *x*- and *y*-axis. As can be seen from the lower plot, the *y*-axis estimation results deteriorated as the line-of-sight of UAV 1 to target 1 approached the horizontal. This is due to the fact that cross-range velocity is difficult to detect using a Doppler-based radar. This is unavoidable when using one vehicle to track the target because of the circular motion of the vehicle around the target. Both axis estimation results improved quickly after 160 measurements, that is when UAV 4 joined UAV 1 in tracking the same target. This is especially the case for the *y*-axis estimation. This result shows that the proposed method can coordinate the trajectories of the tracking UAVs which increases the accuracy of the estimation.

The Monte Carlo method with a sample size of 100 is used to evaluate the proposed control framework compared with the centralised control strategy proposed in (Kingston and Schumacher, 2005) in two scenarios. Each scenario includes seven cases with the number of UAVs progressively increasing from 6 to 12. In scenario 1, three targets and the whole team of UAVs were randomly distributed over the mission area at the beginning of the simulation. In scenario 2, the UAV swarm enters the mission area from the left side in a tight formation. The number of targets is increased to four. At the end of each simulation, the time for eliminating all targets was recorded. The computational intensity is one of the main issues with the centralised algorithm. It is found out that the problems of 12 vehicles (include both UAVs and targets) are solvable in <30 seconds by using the CPLEX solver on a personal computer. As the size of the problems increased to 20 vehicles, the time spent on finding solutions increased to 90 seconds. We estimate that it will take

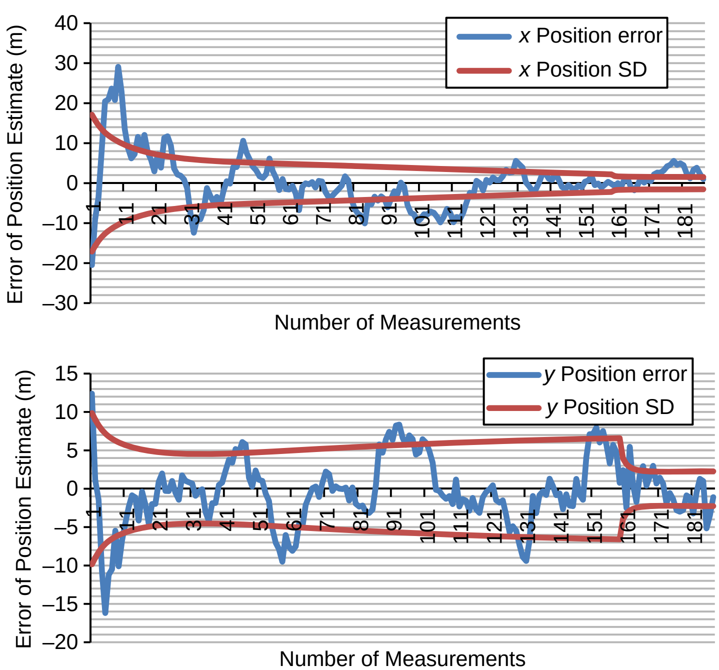

Figure 9.
Position estimation error and standard deviation (SD) of target 1

20 minutes to optimise the problems of 30 vehicles. Thus a maximum 12 UAVs and four targets are considered in our study for the comparison of centralised and decentralised algorithms.

The average of the final task completion time for each simulated scenario are summarised in Figure 10. The error bars show the 95 per cent confidence interval for the mean value based on the sample data. It can be seen from Figure 10a that the curve representing the rule-based UAV swarm descends faster than the centralised controlled swarm, i.e. the advantage of the centralised controller becomes less significant as the number of UAVs increase. For a small size group of agents, the swarm is less efficient as centralised controller, especially for mission scenarios with strong timing and coupling constraints. Such results are expected as the centralised controller optimising the common objective shared by the whole team and be able to resolve the conflicts if the tasks are coupled, while the decentralised controller could only operate under local and environment-driven rules due to the absence of global coordination information. As the number of UAVs increase, the UAVs only need to engage the targets that are near to them. This means that the tasks that the UAVs are to perform become less coupled than in the situation where one UAV has to perform multiple tasks on multiple targets.

The simulation results of scenario 2 are given in Figure 10b. As can be seen, the gap between these two curves does not decrease as the number of UAVs increase. This is mainly because in scenario 2 the whole team of UAVs starts at one side of the mission area in close formation. Such a setup causes the UAVs to lose the advantages of spreading over the mission area. Therefore, the task assignment problem becomes more tightly coupled than in the scenario 1. This makes it impossible for the decentralised controller to achieve the performance of the centralised controller if the problem can be solved in real-time. Such a drawback of the swarm system leads to the investigation of an integrating predictive model applied to individual agents to enhance the mission performance.

4.3 Predictive swarm

In this swarm system, individual agents neither control nor are aware of tasks to be carried out by other agents. Even though this design enables agents to perform separate subtasks cooperatively, the achievable performance is low. This is mainly because in the early state of the mission, the UAVs fly towards the nearest target without knowing that there is a possibility that other UAVs may arrive earlier and destroy the target before they arrive. This causes an excessive number of the UAVs to

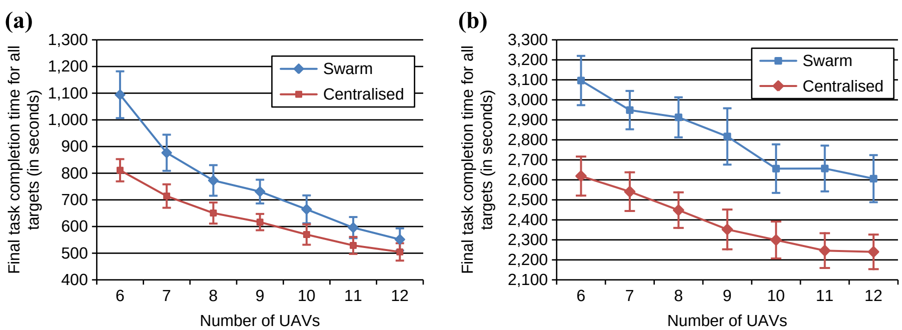

Figure 10.
(a) Monte Carlo results of scenario 1; (b) Monte Carlo results of scenario 2

waste time on a target which can be better engaged by other members in the team. To address this issue, a prediction model which allows the UAVs to predict the intentions of their neighbours is added to the decentralised controller. This prediction model uses the mixed integer linear programming formulation (Kingston and Schumacher, 2005) and works as a preliminary assignment algorithm. When a UAV is in cruise mode, it senses its neighbours' states, such as position, heading and velocity, and uses them as the input to compute the assignment. The prediction model will run at the beginning of the mission to allocate an initial assignment. After that, it will be triggered by three events, namely, the UAVs change back to cruise mode from any of the other three modes, they detect new neighbours or the neighbours' movement deviates from the previous prediction. By setting up these rules, the load on computation is expected to be reduced and to make the UAVs better able to adapt to uncertainty. Even though this prediction model uses the same formulation as the centralised approach in previous research, they operate differently. First, this prediction model is only concerned with one UAV and its near neighbours, while the centralised approach is concerned with the whole group. This makes the predictive decentralised controller scale as well as decentralised controller. Second, the assignment resulting from the prediction model is just a temporary assignment. Since the UAVs are in the cruise mode after receiving assignments from the prediction model, they will still respond to any request to turn into trackers. Although the cooperation of the swarm still relies on the interactions that take place among the group, the prediction model will give the agents an increased level of situation awareness. Thus the agents would accept an assignment that does not maximise their individual utility if they can maximise the overall team utility. For instance, the agents will choose the targets which may not be the nearest but by engaging such targets would potentially reduce the overall time of finishing the mission. The following is a simulation model used to demonstrate the increased performance by using predictive swarm comparing with simple swarm presented early.

As shown in Figure 11a and b, the UAVs with the simple swarm controller were all flying towards target 1 because the rule set tells them to engage the nearest target if there is no request received. In contrast, the UAVs with predictive swarm controller split into two group of three. One group headed towards target 1 and another group chose target 2.

As can be seen from Figure 11c, until target 1 was destroyed, the UAVs began to move towards target 2. On the other hand, as shown in Figure 11d, three UAVs chose the target 2 at the beginning of the mission was already halfway to target 2.

After target 1 was destroyed, the UAVs with the simple swarm controller all chose target 2. As for the UAVs with the predictive swarm controller, three UAVs that previously engaged target 1 decided to go to target 3 after they attacked target 1. As can be seen from Figure 11e and f, it took one and half minutes longer for the UAVs with the simple swarm controller to reach target 2.

Finally, target 3 was destroyed. The UAVs with the predictive swarm controller finished the mission almost five minutes earlier. The simulation study of scenario 2 is repeated including the predictive decentralised controller this time. As can be seen in Figure 12, the predictive decentralised controller provided better performance than the simple responsive swarm in all cases.

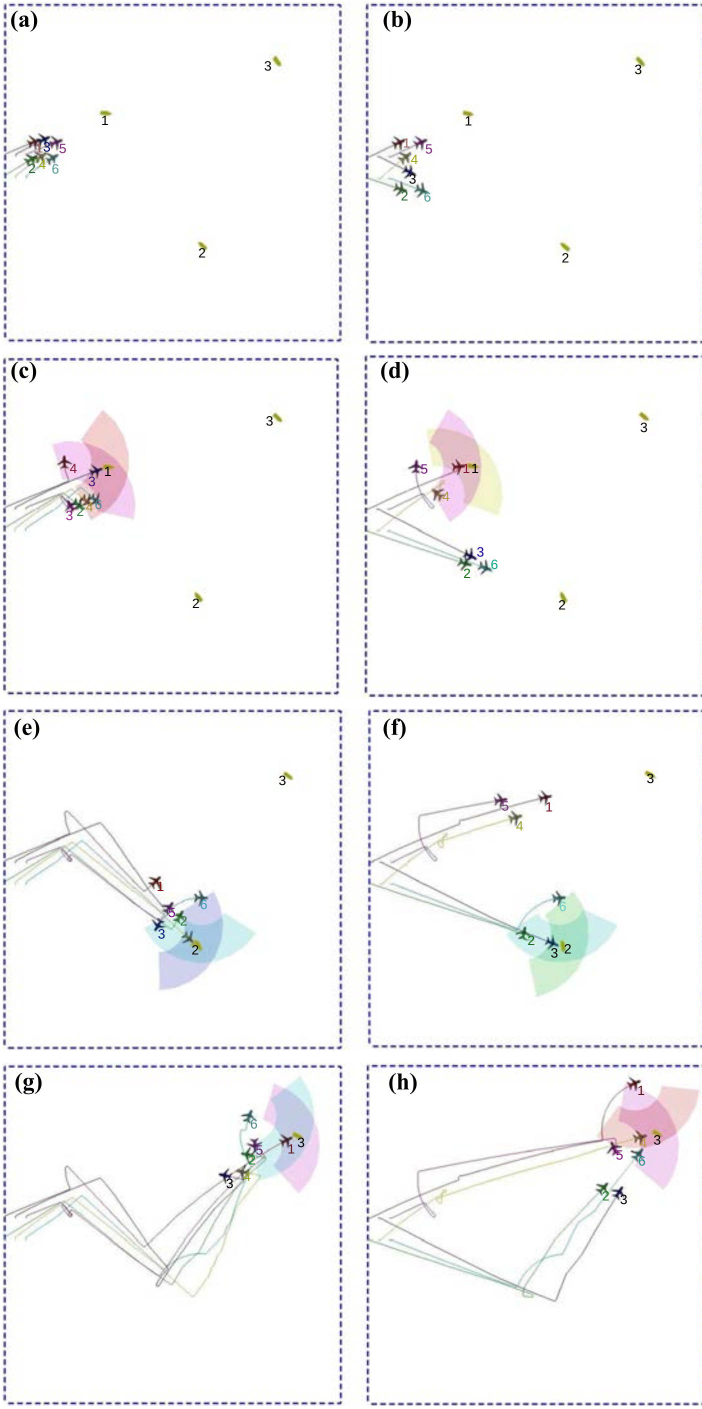

Figure 11.
(a) Simple responsive swarm at 90 seconds; (b) predictive swarm at 90 seconds; (c) simple responsive swarm at 250 seconds; (d) predictive swarm at 250 seconds; (e) simple responsive swarm at 590 seconds; (f) predictive swarm at 500 seconds; (g) simple responsive swarm at 1,115 seconds; (h) predictive swarm at 850 seconds

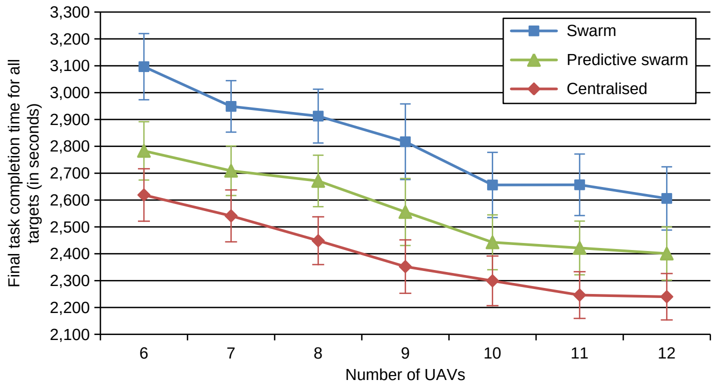

Figure 12.
Repeated scenario 2 with predictive decentralized controller included

5. Conclusion

This paper investigated the use of decentralised controllers for cooperative moving target engagement missions. Different to a centralised task assignment algorithm, the cooperation of the agents is entirely implicit. The behaviour of the UAVs was governed by rule sets which ultimately lead to a highly cooperative performance of the group without the need for a centralised control authority. The information theoretic measures are adopted to estimate the value of possible future actions and make the UAVs fly trajectories that locally maximise the information content for the next measurement. Two types of decentralised controllers were proposed, namely the responsive swarm controller and the predictive swarm controller.

The simulation study evaluates the performance of the decentralised controller against the centralised controller quantitatively.

The results show that the performance of the decentralised controller depends on the complexity of the coupled task constraints. If the increased number of UAVs could cause the tasks to be less coupled, the decentralised controller would become competitive with the centralised controller in large size problems. If the complexity of the coupled task constraints does not reduce as the size of the swarm increases, the predictive model can be integrated into the decentralised controller to let the agents estimate the intentions of their neighbours and choose activities which enhance the overall team utility accordingly. The improvement in the performance by using predictive swarm controller was shown in the simulation results. However, the benefits of redundancy and robustness offered by a swarm system, has not been demonstrated in current design. In the future work, the adaptive ability of these two types of controllers due to uncertainties such as communication constraints, manoeuvring targets and threats will be investigated.

References

Altshuler, Y., Yanovsky, V., Wagner, I. and Bruckstein, A. (2006), "Swarm intelligence – searchers, cleaners and hunters", in Nedjah, N. and Mourelle, L.D.M. (Eds), *Swarm Intelligent Systems*, Springer, Berlin, Heidelberg, pp. 93-132.

Bar-Shalom, Y. (2000), *Multitarget/Multisensor Tracking: Applications and Advances*, Artech Print on Demand, Norwood, MA.

Bar-Shalom, Y., Li, X.R. and Kirubarajan, T. (2001), *Estimation with Applications to Tracking and Navigation*, Wiley-Interscience, New York, NY.

Bonabeau, E., Dorigo, M. and Theraulaz, G (1999), *Swarm Intelligence – From Natural to Artificial Systems*, Oxford University Press, Boca Raton, FL.

Bourgault, F., Furukawa, T. and Durrant-Whyte, H.F. (2003), "Coordinated decentralized search for a lost target in a Bayesian world", *Intelligent Robots and Systems, IROS, 2003. Proceedings, 2003 IEEE/RSJ International Conference, Vol. 1, 27-31 October*, pp. 48-53.

Ceccarelli, N., Enright, J.J., Frazzoli, E., Rasmussen, S.J. and Schumacher, C.J. (2007), "Micro UAV path planning for reconnaissance in wind", *American Control Conference, ACC, 9-13 July*, pp. 5310-5315.

Cover, T.M. and Thomas, J.A. (2006), *Elements of Information Theory*, John Wiley & Sons Inc, Hoboken, NJ, available at: http://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=158159&site = ehost-live.

Crassidis, J. (2004), "Sequential state estimation", *Optimal Estimation of Dynamic Systems*, Chapman and Hall/CRC, London.

Dale, L. (2003), "Lyapunov vector fields for UAV flock coordination", *2nd AIAA "Unmanned Unlimited" Conference and Workshop & Exhibit, American Institute of Aeronautics and Astronautics, San Diego, CA, 16 September.*

De Vries, E. and Subbarao, K. (2011), "Cooperative control of swarms of unmanned aerial vehicles", 49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, American Institute of Aeronautics and Astronautics, Orlando, FL, 4-7 January.

Durrant-Whyte, H. and Grocholsky, B. (2003), "Management and control in decentralised networks", *Information Fusion. Proceedings of the Sixth International Conference of, 2003,* pp. 560-566.

Erignac, C.A. (2007), "An exhaustive swarming search strategy based on distributed pheromone maps", *AIAA Infotech@Aerospace 2007 Conference and Exhibit, American Institute of Aeronautics and Astronautics, Rohnert Park, CA, 7-10 May.*

Frelinger, D.R., Joel, K. and William, S. (1998), *Proliferated Autonomous Weapons: An Example of Cooperative Behavior*, RAND Corporation, Santa Monica, CA.

Frew, E.W., Lawrence, D.A. and Morris, S. (2008), "Coordinated standoff tracking of moving targets using Lyapunov guidance vector fields", *Journal of Guidance, Control, and Dynamics*, Vol. 31 No. 2, pp. 209-306.

Furukawa, T., Bourgault, F., Lavis, B. and Durrant-Whyte, H.F. (2006), "Recursive Bayesian search-and-tracking using coordinated uavs for lost targets", *Robotics and Automation, ICRA. Proceedings, 2006 IEEE International Conference on, 15-19 May*, pp. 2521-2526.

Garnier, S., Gautrais, J. and Theraulaz, G. (2007), "The biological principles of swarm intelligence", *Swarm Intelligence*, Vol. 1 No. 1, pp. 3-31.

Gaudiano, P., Shargel, B., Bonabeau, E. and Clough, B.T. (2003), "Control of UAV swarms: what the bugs can teach us", *2nd AIAA "Unmanned Unlimited" Systems, Technologies, and Operations, American Institute of Aeronautics and Astronautics, San Diego, CA, 15-18 September.*

Gu, G., Chandler, P.R., Schumacher, C.J., Sparks, A. and Pachter, M. (2006), "Optimal cooperative sensing using a team of UAVs", *IEEE Transactions on Aerospace and Electronic Systems,*, Vol. 42 No. 4, pp. 1446-1458.

Kingston, D.B. and Schumacher, C.J. (2005), "Time-dependent cooperative assignment". *Proceedings of the 2005 American Control Conference, Vol. 6, 8-10 June*, pp. 4084-4089.

Kreucher, C.M., Hero, A.O., Kastella, K.D. and Morelande, M.R. (2007), "An information-based approach to sensor management in large dynamic networks", *Proceedings of the IEEE, Vol. 95*, pp. 978-999.

Mutambara, A.G.O. (1999), "Information based estimation for both linear and nonlinear systems", *Proceedings of the 1999 American Control Conference, Vol. 2, 2-4 June*, pp. 1329-1333.

Mutambara, A.G.O. and Durrant-Whyte, H. (1994), "Distributed decentralized robot control", *1994 American Control Conference, Vol. 2, 29 June-1 July*, pp. 2266-2267.

Nelson, D.R., Barber, D.B., Mclain, T.W. and Beard, R.W. (2006), "Vector field path following for small unmanned air vehicles", *2006 American Control Conference, 14-16 June*, 7pp.

Nigam, N. and Kroo, I. (2008), "Persistent surveillance using multiple unmanned air vehicles", *2008 Aerospace Conference, IEEE, 1-8 March*, pp. 1-14.

Pack, D.J. and Mullins, B.E. (2003), "Toward finding an universal search algorithm for swarm robots", *Intelligent Robots and Systems (IROS 2003). Proceedings, 2003 IEEE/RSJ International Conference, Vol. 2, 27-31 October*, pp. 1945-1950.

Ponda, S.S., Kolacinski, R.M. and Frazzoli, E. (2009), "Trajectory optimization for target localization using small unmanned aerial vehicles", *AIAA Guidance, Navigation, and Control Conference, 10-13 August, Chicago, IL.*

Price, I.C. (2006), *Evolving Self-Organized Behavior for Homogeneous and Heterogeneous UAV or UCAV Swarms*, Air Force Institute of Technology, Wright Ptrsn AFB, OH.

Rimon, E. and Koditschek, D.E. (1992), "Exact robot navigation using artificial potential functions", *IEEE Transactions on Robotics and Automation*, Vol. 8 No. 5, pp. 501-518.

Rong Li, X. and Jilkov, V.P. (2003), "Survey of maneuvering target tracking. Part I. Dynamic models", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 39 No. 4, pp. 1333-1364.

Ryan, A.D., Durrant-Whyte, H. and Hedrick, J.K. (2007), "Information-theoretic sensor motion control for distributed estimation", *ASME 2007 International Mechanical Engineering Congress and Exposition, Seattle, WA, 11-15 September.*

Scerri, P., Glinton, R., Owens, S., Scerri, D. and Sycara, K. (2007), "Geolocation of RF emitters by many UAVs", *AIAA Infotech@Aerospace, 7-10 May, Rohnert Park, CA*.

Schumacher, C. (2005), "Ground moving target engagement by cooperative UAVs". *Proceedings of the 2005, American Control Conference, Vol. 7, 8-10 June*, pp. 4502-4505.

Shima, T. and Schumachery, C. (2005), "Assigning cooperating UAVs to simultaneous tasks on consecutive targets using genetic algorithms", *AIAA Guidance, Navigation, and Control Conference and Exhibit*, *San Francisco, CA, 15-18 August.*

Singh, A., Krause, A., Guestrin, C. and Kaiser, W.J. (2009), "Efficient informative sensing using multiple robots", *J. Artif. Int. Res*, Vol. 34 No. 1, pp. 707-755.

Yang, K., Keat Gan, S. and Sukkarieh, S. (2013), "A Gaussian process-based RRT planner for the exploration of an unknown and cluttered environment with a UAV", *Advanced Robotics*, Vol. 27 No. 6, pp. 431-443.

Corresponding author
Haoyang Cheng can be contacted at: haoyang.cheng@student.unsw.edu.au