MULTI-AGENT FORMATION CONTROL FOR UAV SWARMING

Omar Ilaya*, Cees Bil*
***School of Aerospace, Mechanical, Manufacturing Engineering**
RMIT University

Keywords: *UAV Swarm Control, Control Design*

Abstract

This paper presents a novel swarm-based defence system for the interception of a moving targe such as an aircraft or missile. The paper discusses issues regarding the synthesis of an agent-based control system for the swarm-behaviour of cooperative UAV systems. The problem, representative of various other multi-agent systems, is considered as an optimum formation reconfiguration problem involving the decentralization of the control task. A general method towards the development of multi-agent control systems is then presented with reference to the notional defence system.

1. Introduction

The pilotless nature and increasing mission capabilities of unmanned aerial vehicles (UAVs) is generating a strong interest in airborne operations over remote and hostile territories. In recent years, UAVs have been used by the military to perform intelligence, surveillance, and reconnaissance (ISR) operations, close-air support (CAS), suppression of enemy air defences (SEAD), and precision strike operations. The growing interest towards smaller and cheaper UAVs presents an emerging capability to operate deeper within hostile and politically denied airspaces. The relatively stealthy and expendability of these UAVs make them an optimum candidate for extremely dangerous operations. The trend towards UAVs suggests an unprecedented level of automation in future combat scenarios.

Current UAV technologies lack the sufficient autonomy to efficiently and effectively deploy multiple UAVs concurrently. Sophisticated UAV platforms today used in military operations depend on a remote crew of 1-3 users. In future combat situations, where multiple UAVs cooperate to achieve a common objective, the required cognitive effort of the operators' increases exponentially as the number of unmanned systems increases.

In this paper, the concepts of swarming are introduced as a method to reverse the operator-to-system ratio observed in multi-agent systems. A notional swarm-based defence system using multiple small-scale UAVs is presented to demonstrate the capability of swarming UAVs. Issues regarding the development and control of swarming systems are discussed followed by the general framework for the synthesis of swarming UAVs.

2. Problem Domain

This paper considers the problem of cooperative control of multiple UAVs for the role of target interception. The swarm consists of small UAVs with low-observability characteristics. Low-observability allows the close-proximity operations of UAVs to the enemy systems and supports the assumption of a non-evading target. Furthermore, it is assumed a constant swarm population is observed throughout the operation; i.e. the swarm does not suffer from attrition. Figure 1 shows the functional diagram for the notional swarm-based defence system.

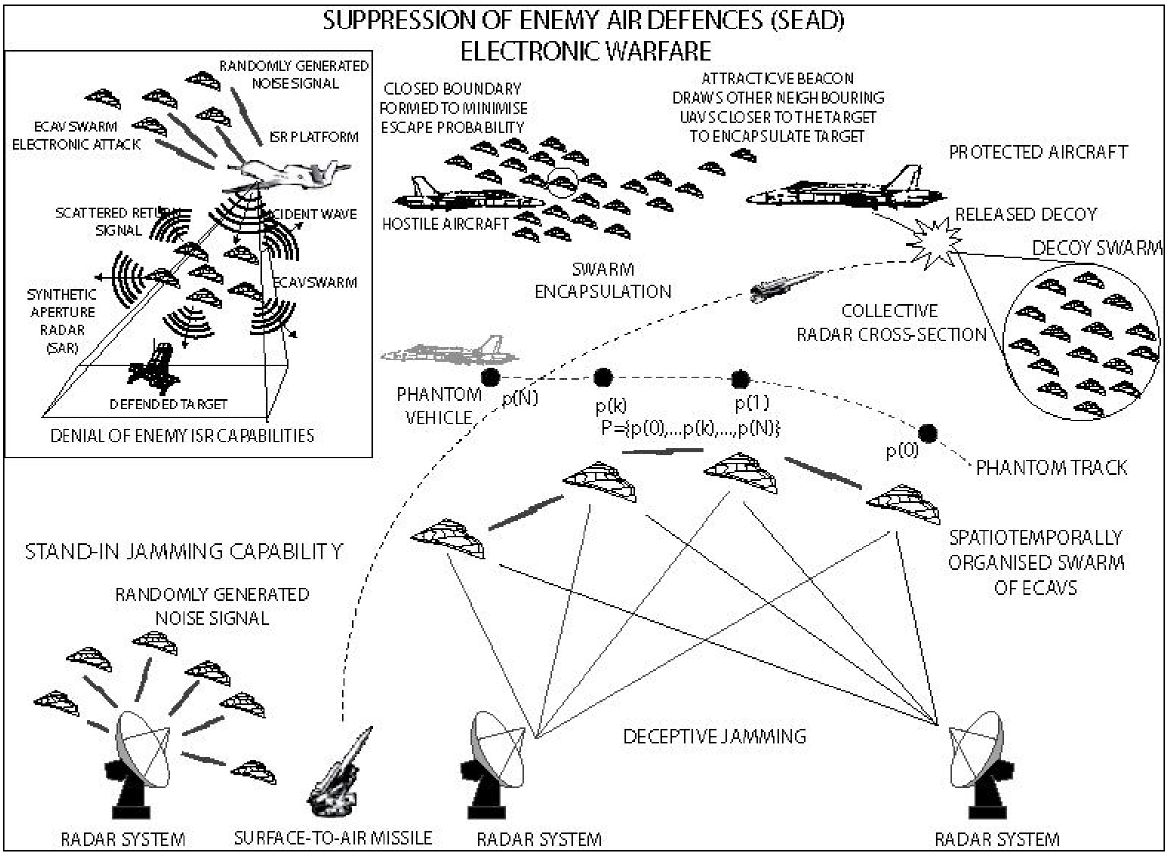


Fig. 1. Functional diagram of notional swarm-based defence system.

The intercept problem involves the identification of a suitable formation that minimises the escape probability of the target (figure 2). In this paper, an optimum intercept formation considers the encapsulation of a target and involves a closed regular polygon with the centre coincident to the target position. To simplify the controller design, it is further assumed that the shape of the formation is provided by an external controller. This corresponds to the command and control node in a defence network.

The controller design problem is to synthesise a decentralised control algorithm for the swarm agents that emulates the self-organising behaviour of natural swarming systems. Specifically, given an initial swarm configuration of UAVs, find an optimal set of trajectories that map the initial configuration of each UAV to an optimal final configuration corresponding to the set of vertices describing the shape of the formation and respects the modularity of the system. The problem is representative of a decentralised optimisation problem involving formation control.

Although the problem defined is specific to the application of a defence system, the decentralised formation control problem is representative of many multi-agent systems. For UAVs and cellular robotics, the desired formation of the agents can be altered for specific applications. In distributed sensory networks involving swarms UAVs, the optimal formation for convergence could be a strongly connected lattice structure. In fuel efficient strategies, the formation for a swarm of UAVs would be a tight cluster that minimises the induced drag of neighbouring aircraft.

The concept of a formation control problem is not limited to multi-agent systems existing on a Euclidean manifold. The concept of formation is a generalisation of a specific configuration for the agents in the swarm, and could be used to describe information flow paradigms in ad hoc communication networks or the signal processing in phased-arrays.

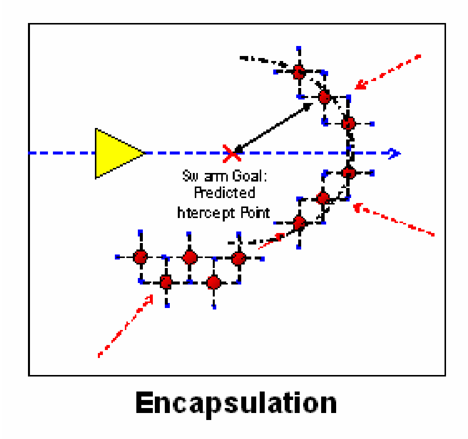


Fig. 2. Encapsulation concept.

Although this investigation has been limited to a swarm of UAVs provided with the desired formation shape, future work aims to eliminate the involvement of the command and control node by allowing the UAVs to evolve an optimum formation independently. In this regard, the swarm becomes fully autonomous and utilise high-level objectives to synthesise an appropriate configuration.

In the following sections, design issues and considerations for the synthesis of a decentralised control strategy for formation control are presented.

3. Design Considerations

In the following section, the design issues considered are presented by topic to highlight various requirements that should be addressed in the development of swarming systems.

3.1 Organisations

An organisation provides the framework that describes the agent interactions. It describes the roles of each agent, their expected behaviour, and levels of authority. In this regard, an organisation can be considered as the pattern of relationships defining the flow of information and control amongst the various subsystems.

[8] describes four main categories typical of most multi-agent systems. These include i) hierarchy, ii) community of experts, iv) market-based, and v) scientific community organisation models.

In the hierarchical organisation, decision-making and control is vertically decomposed along the dimensions of decreasing control authority. The agents of the system represent subordinates that receive high-level commands from a designated central problem solver [8]. Typically, hierarchical structures are observed in centralised control schemes and are of little interest in fully distributed swarming systems.

The community of experts' model considers each agent of the swarm as a specialist in one aspect of the problem domain. The agents interact through rules of order and behaviour and can adjust their solutions to reach a mutual agreement [8].

In the market-based organisation, each agent effectively competes for resources in the system through a process of bidding. This approach is highly favourable in resource constrained multi-agent systems and can be used for task allocation paradigms. It also has applicability in permutation problems such as formation reconfiguration [8].

The scientific community model uses a swarm of processing nodes that produce a viable solution each. Through the mutual sharing of solutions amongst the individual problem solvers, each problem solver can test and refine neighbouring solutions until a global consensus is arrived amongst the swarm. This is the basis of particle-swarm optimisation techniques that search for global optimums [8].

In dynamic multi-agent systems, such as swarms of UAVs, agents are free to move between the organisations. Organisations must therefore be represented as dynamic nets to accommodate the motion of mobile agents.

Swarming systems naturally imply a flat decentralised control organisation. The organisation and local interactive effects are modelled through the concept of neighbourhoods. A neighbourhood defines the regions of locality. Using a dynamic neighbourhood model the organisation structure remains flat and adaptable to the changing dynamics of the environment. A consideration in the boundary definition of the neighbourhood is the radius of influence. The radius of influence models the information and control flow between neighbouring agents. It represents a bounded region of local interactivity. The shape and size of this region is defined by the sensory, communication, and actuation capabilities of the vehicle. In this regard, it can be directed between specific agents in the swarm, or undirected to any agents within a specific locality.

3.2 Agent-Level Cognitive Design

Typically, agents can be categorised based on their cognitive abilities. Reactive systems are those where no internal representation of the external environment is maintained. Agents of this type respond to the immediate environment and make no regard for past instances or future consequences of their immediate actions. Responses are activated through stimulus-based behaviour. Generally, the control algorithms governing their behaviour can be computationally simple and highly responsive.

Through local interactions with other agents complex collective behaviour emerges. This is analogous to the principles of swarming systems [8]. Despite their relatively simple design, robustness and adaptability, their realisation into practical applications is limited by the very same principles that make them favourable: emergent behaviour through local interactions.

Purely reactive systems only consider local information and cannot predict the effect of their local behaviour on the global behaviour of the swarm. This leads to unpredictable and often sub-optimal results. Methods to assess the performance and stability of the swarm are reliant on stochastic and empirical formulations. This lack of understanding of the agents' relationships with the environment, neighbouring agents, and the swarm collective, make it difficult to engineer purely reactive swarm systems precisely.

Unlike reactive agents, deliberative agents use and maintain an internal representation of the external environment. Based on this internal representation, they are able to plan and decide on the appropriate behaviour. They can also predict the effect of their decision on future outcomes and the impact on the global behaviour. Swarms of deliberative agents could potentially converge towards the desired global optimal solutions [8].

The heightened cognitive abilities of deliberative agents require sophisticated intelligent control paradigms that can be computationally extensive. The demand produced by these algorithms on power consumption and processing time limit their use on small-scale mobile agents operating in dynamic and uncertain environments.

Hybrid systems combine elements from both types of system into a unified architecture that exploits the decision-making capabilities of deliberative agents and the responsiveness of reactive agents. Hybrid architectures are represented by various layers of abstraction. Most hybrid systems can be represented by a three-tiered hierarchy (figure 3). The hybrid architecture is characterised by increasing precision down the hierarchy and increasing intelligence up the hierarchy.

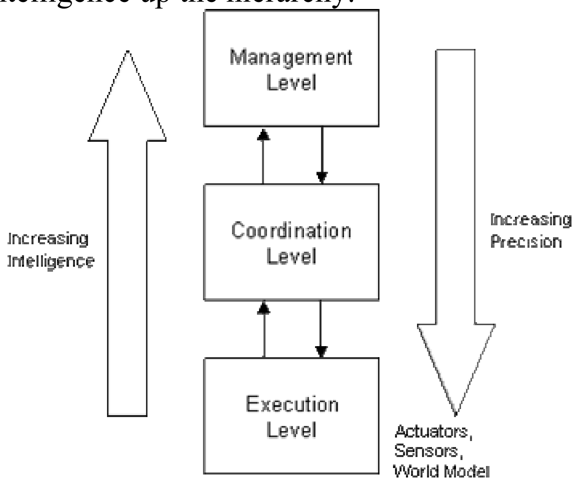


Fig. 3. Hybrid control architecture.

The upper-most level, commonly referred to as the management layer, is responsible for decision-making and planning. It is characterised by the highest level of intelligence and uses the high-level objectives defined by the user to allocate tasks to the various subsystems within the vehicle. In a multi-agent framework, the management layer could also be responsible for coordinating the social behaviour of the agent. The lowest-level is the reactive layer and makes and executes decisions based on raw sensor data. In a swarming UAV application, the reactive agent would be responsible for immediate actions such as inter-agent collision avoidance. The reactive layer represents the agent's interface to the immediate environment and contains the necessary sensors and actuators. This is sometimes referred to as the execution layer. The middle-layer, or coordination layer, provides an interface between the high-level decision-maker with the execution layer. Its primary role serves to facilitate information processing between the raw-data of the execution layer with the symbolic abstractions of the management layer.

3.3 Behaviour Modelling

Behaviours are used to define the roles and responsibilities of the system and provide an understanding of the control design goals. Behaviour modelling can be considered from the system level, host level and agent level. Agent-level behaviours define the mappings between the input and output of the agent and describe how they interact with the environment. Agent-level behaviours for UAV systems could include collision-avoidance and move-to-target.

Host-level behaviours consider the collective interactions of the agents, and are synonymous to emergent behaviour. Examples could include formation convergence and formation reconfiguration.

System-level behaviours consider the interoperability of the swarm (and its host-level behaviour) with other subsystems. In a swarm-based defence network, the swarm would be supported by a mission control element that specifies mission objectives. This specification of mission objectives ensures that the swarm-behaviour is purposeful for the defence network. In the defence scenario, system-level behaviours could include swarm encapsulation.

Development of a swarm-based application requires the definition of the various behaviours exhibited at each level of abstraction. The definition of behaviours provides an understanding of the role of the swarm and the agents, and compliments the organisation design.

With respect to the problem domain, the system-level behaviour would be to encapsulate a moving target. The host-level design would encompass the necessary behaviours demonstrated by each mission element. The battle management, command, control and communication, and intelligence node (BM3CI) assesses the battle space and provides an optimal intercept point and shape formation for the swarm. The swarm, initialised in some region in the battle space, would then need to converge to that formation. The agent-level behaviours pertain to the individual vehicles and could include collision avoidance, move to target, maintain a fixed inter-vehicle distance, and converge on desired formation shape. Based on these simplistic agent-behaviours and local interactions, the collective swarm would demonstrate the capability to converge to the desired formation that intercepts and encapsulates a target of interest.

Realisation of these behaviours requires the synthesis of appropriate control techniques and models.

3.4 Information Flow

[1] distinguishes two types of information available to each agent; perceptive and communicated. Perceptive information pertains to information obtained via sensory channels and could include the information obtained from proximity sensors or imaging devices. Communicated information regards any information transferred between two or more systems. This could include the inter-agent communication between neighbouring vehicles via wireless protocols, or the long-range reception of mission objectives from external mission control elements. The type of information and its range of availability are dependent on the selected hardware and the behaviours defined at system-level, host-level, and agent-level.

Selection and design of sensory and communication suites involve the identification of the type of information and the purpose. Proximity sensors such as laser range finders can be used to detect and measure distances to nearby obstacles, whilst GPS-based navigation hardware provides the local vehicle position with respect to some global reference frame.

The information flow within the swarm of agents can be regarded as the information accessible by each agent. The information flow influences the behaviour of each agent and the consequent emergent behaviours of the collective swarm. The shape and size is modelled using the concept of neighbourhoods described earlier and is tightly coupled to the sensory and communication design of the vehicles.

The information flow affects the stability and performance of the swarm [2]. The information flow of an agent is modelled as the in-degree of a connected graph. The magnitude of the in-degree matrix corresponds to the information regard of an agent. For agents with high in-degree values, the available information could correspond to a global knowledge of the swarm state. Low in-degree values correspond to localised areas of knowledge with sparse information flow between spatially disjoint neighbours. High densities of information flow (e.g. fully connected swarm systems with global information accessible by every agent); swarm systems tend to exhibit unstable behaviour with the propagation of string instabilities. Flock coherence is difficult to achieve and maintain as the field of regard of each agent considers every agent in the swarm. This is a particular problem of large swarm populations.

In swarms consisting of agents of limited fields of regard and sparse information flow, the emergent behaviour can fail to converge to the desired result. Solutions of this swarm type are often sub-optimal. Thus, design of sensory and communication protocols are an important consideration for the stability and performance of UAV swarms. However, the tight coupling between agent-level behaviour and host-level behaviour is poorly understood. This uncertainty complicates the synthesis of communication and sensory capabilities of swarms for optimal behaviour. In the next section, several of the issues of emergent behaviour on formation reconfiguration are addressed.

3.5 Formation Reconfiguration

Formation reconfiguration regards the transition from an initial state to a final state. With regard to UAV swarms of a defence network, the initial configuration of the swarm would correspond to the initial distribution of UAVs in the environment. The final configuration of the UAV swarm would then be an optimal distribution of UAVs along the desired formation shape provided by the mission control element. A significant problem of interest in this research is the synthesis of emergent behaviour among a swarm of decentralised agents from an initial configuration to an optimal permutation of the final desired configuration. The research is primarily interested in synthesising the necessary behaviour for each agent to plan and execute local actions that optimally satisfy the global objective with local information that preserves the modularity characteristics of swarming systems.

Fig. 4. Problem domain of formation reconfiguration.

The problem is presented as a permutation problem solved by a decentralised network of nodes distributed among the UAV swarm. Given an initial distribution of UAVs in the environment and a final unlabelled or un-enumerated desired configuration for the swarm, the aim of for each agent is to identify a potential target position in the final configuration that minimises a cost objective of the mission requirements. This problem can be solved by a decentralised swarm algorithm that uses local information and local cost objective functions to converge to a globally optimised solution. A significant drawback of this approach is the decomposition of the global objective among the decentralised nodes. Decomposition of the global objective requires a thorough understanding of the relationship between the agent-level behaviour and the host-level behaviour. The emergence of behaviour is unpredictable and does not necessarily satisfy the global objective of the system. Moreover, UAVs are dynamic agents and the transition between configurations should be considered as an evolution in time. In this regard, if initial and final configurations can be represented using graphs, then the transition between configurations is a dynamic graph. At any period during the transition, the instantaneous configuration is modelled as a graph. [5] refers to this transition between graphs as a graph process.

The modelling of graph processes between two configurations maps the emergence of behaviour and provides a preliminary understanding of the relationship between agent-level and host-level behaviour. This understanding facilitates the synthesis of agent-based controllers and reduces the inherent uncertainty and unpredictability of swarming systems. The reduction in uncertainty and the consequent predictability of the system realises potential swarming systems into practical applications beyond the simulation environment of the laboratory.

3.6 Cooperative Synthesis

The modelling of graph processes provides a preliminary to the synthesis of cooperative and self-organising behaviour of the swarm. Cooperative planning of the swarm is required to synthesise decentralised and optimum behaviour that converges to a permutation of the final configuration. A critical enabler to cooperative planning is the mutual sharing of trajectory information between neighbouring UAVs. UAVs in the swarm utilise the planned trajectories of neighbouring agents to refine their own trajectories for a locally optimal results. These trajectories are obtained using graph process modelling techniques and updated by recursive search algorithms. The continual sharing and updating of trajectory information among neighbouring agents converges to an optimal set of trajectories that define the graph process from an initial configuration to an optimal permutation of the final desired configuration.

4. Swarm Engineering Approach

The following presents a general formulation of the problem addressing the formation control of a swarm of UAVs for the notional defence system described in section 2. In the following formulation, behaviours are simplified to preserve generality and applicability to a variety of swarming systems.

Consider a swarm of n agents randomly distributed in the environment constrained by the following dynamics:

$$\begin{cases} \dot{q}_i = p_i \\ \dot{p}_i = u_i \end{cases} \tag{1}$$

The configuration of agents in the environment is modelled using a graph theoretic approach. A graph G is a pair consisting of a set of vertices, or nodes, denoted by $V = \{1,2,\ldots,N\}$, and a set of edges $E \subseteq V \times V$, where $e = (v,w) \in E$ and $v,w \in V$. The set of vertices in space correspond to the set of UAVs distributed in the environment. Communication links between adjacent neighbours are denoted using edge notation. Here, research is limited to undirected graph representations of mutual sensory and communication links. The following property of undirected graphs is also observed $\forall (v_i, v_j) \in E \Rightarrow (v_j, v_i) \in E$. Using this representation, the in-degree and out-degree of every agent is equal.

To preserve the concept of localities, the set of neighbours for agent v_i is defined by:

$$N_i = \{j \in V : a_{ij} \neq 0\} = \{j \in V : (i,j) \in E\} \tag{2}$$

where a_{ij} defines the adjacency matrix of the graph given by:

$$a_{ij}(q) = \rho\left(\left\|q_j - q_i\right\| / r\right) \tag{3}$$

[7] defines a smooth influence map (C^k) for $\rho(z)$ using the following:

$$\rho(z) = \begin{cases} 1, & z \in [0, \delta] \\ \frac{1}{2^k}\left[1 + \cos\left(\pi \frac{(z-\delta)}{(1-\delta)}\right)\right]^k, & z \in (\delta, 1] \\ 0, & otherwise \end{cases} \quad (4)$$

It is observed that $a_{ii} = 0$ and $(i,i) \notin E$. The neighbourhood representation makes no distinction between the sensory and communication capabilities of the UAV and does not delineate any difference between the perceived and communicated information. This formulation of the neighbourhood principles also observes the scalability of a dynamic organisation in that it accommodates the passage of agents between adjacent neighbourhoods.

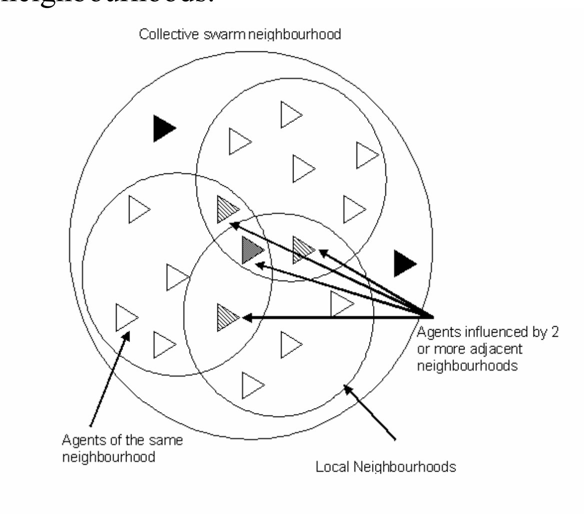
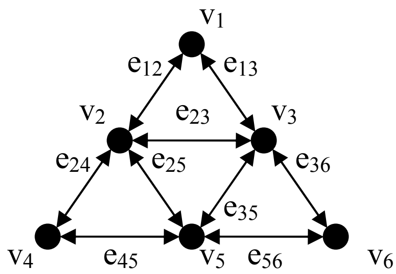


Fig. 5. Regions of localities.

The objective of the swarm, as outlined in section 2, is to converge to the desired formation. For the purpose of encapsulating a target, the optimum formation considered is a closed ring. The mission control element provides a representation of the formation to the swarm that is interpretable by the agent's controller. Preserving the graph-theoretic approach to the formation control problem, the desired formation is represented as a graph. Figure 6 shows an example of a formation and its corresponding adjacency matrix



$$A = \begin{bmatrix} 0 & e_{12} & e_{13} & 0 & 0 & 0 \\ e_{12} & 0 & e_{23} & e_{24} & e_{25} & 0 \\ e_{13} & e_{23} & 0 & 0 & e_{35} & e_{36} \\ 0 & e_{24} & 0 & 0 & e_{45} & 0 \\ 0 & e_{25} & e_{35} & e_{45} & 0 & e_{56} \\ 0 & 0 & e_{36} & 0 & 0 & e_{56} \end{bmatrix}, \quad i \neq j$$

Fig. 6. Example of graph representation of a pyramid formation.

The optimisation problem can be stated as: given an initial configuration G_0 and a final configuration G_f, determine the set of trajectories and control input that minimise:

$$\int_{t0}^{tf} L(q(t), u(t), t)dt \quad (5)$$

The above formulation is representative of a Bolza optimisation problem [6, 3]. The Mayer cost function often associated with the Bolza problem has been omitted since the problem involves a fixed final state. The optimisation problem can be customised to accommodate various mission objectives such as minimum time for synchronous convergence, or minimum net energy expenditure [6].

At this stage, the solution involves a decentralised search algorithm. Currently, research is being conducted to investigate a method that uses affine connection control systems to model the transition process between neighbouring agents. A property of affine connection control systems is the optimisation problem that naturally exists in the formulation [1]. Affine connection systems have been used in the past as a formulation for various mechanical systems. Solutions of the underlying optimisation problem of the affine formulation can then provide the optimal control inputs and associated trajectories to transform from an initial configuration to a final configuration [4].

Although the optimisation problem can be solved by various search algorithms, the design problem now presented is the identification and decentralisation of a suitable search algorithm for application into small-scale UAVs.

6. Preliminary Results

In the following section, preliminary results demonstrating the formation reconfiguration problem in a team of planar UAVs are shown. A particle swarm optimiser was used to solve the optimisation problem associated with the formation reconfiguration problem described in section 2. The optimisation problem was formulated using the energy deviation function given in equation 6.

$$E(q)=\sum_{i=1}^{n}\sum_{j\in N_i}\phi\left(\left\|q_j-q_i\right\|-d_{ij}\right)+\sum_{i=1}^{n}\sum_{j\in N_i}\phi\left(n_{ij}\cdot\left(p_j-p_i\right)\right) \tag{6}$$

where $\phi(\cdot)$ is some pairwise potential given by equation 7 and shown in figure 7:

$$\phi(z)=\sqrt{z^2+1}-1 \tag{7}$$

and n_{ij} is the unit vector given by

$$n_{ij}=\left(q_j-q_i\right)/\left\|q_j-q_i\right\|$$

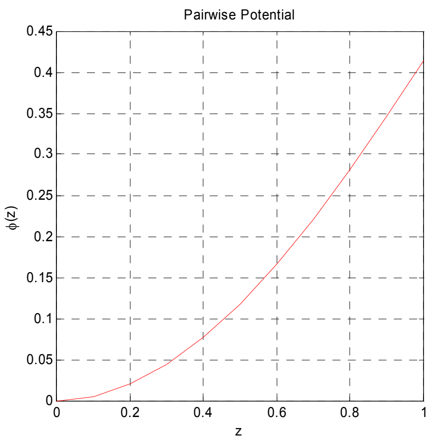


Fig. 7. Pairwise potential used for energy deviation function.

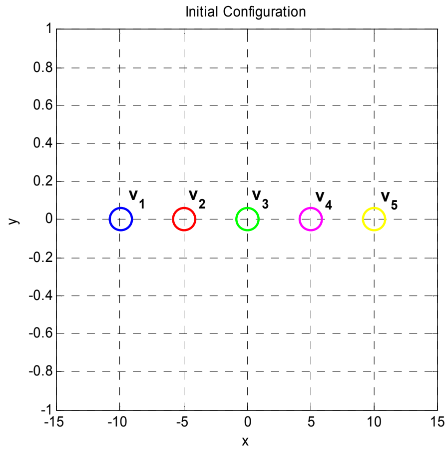


Fig. 8a. Initial configuration of UAVs.

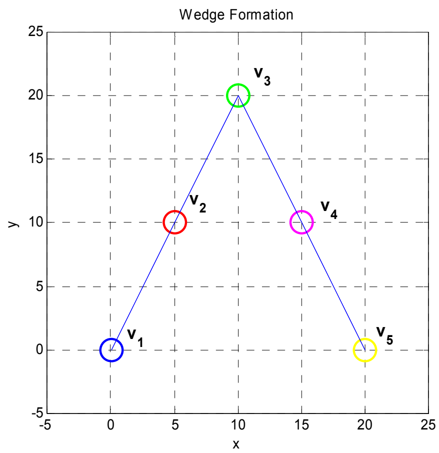


Fig. 8b. Desired wedge configuration.

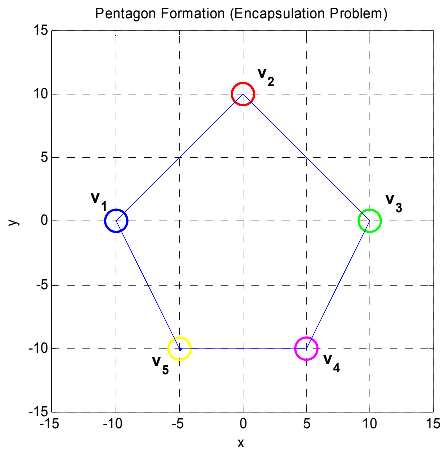


Fig. 8c. Desired pentagon configuration.

The energy deviation describes the correspondence of a graph configuration to another desired configuration. The first term of the energy deviation uses the edge conditions that define the connectivity of the formation graph and describes the convergence of the swarm to the desired graph formation. A 1-1 correspondence between two configurations is observed when the energy deviation function equals zeros (i.e. $E(q)=0$). At this point, the initial swarm distribution is said to have converged to a permutation of the desired configuration. The second term of the energy deviation function describes the kinetic energy of the system. The inclusion of this term aims to minimise the energy required to transform the initial graph formation to the final graph formation.

In the simulations that follow, a swarm of 5 fully actuated UAVs with dynamics and control described by equation 9 and equation 10 respectively were initialised in a straight-line (figure 9a).

$$\begin{aligned} \dot{x}_i &= v\cos\theta_i \\ \dot{y}_i &= v\sin\theta_i \\ \theta_i &= u_i \end{aligned} \tag{9}$$

where:

$$\begin{aligned} &U = \{u \in R^m : u_{\min} \le u_i(k) \le u_{\max} \\ &u_{\min} < 0 < u_{\max}, i = 1,\ldots,5\}, \quad \forall t \in [0,T] \end{aligned} \tag{10}$$

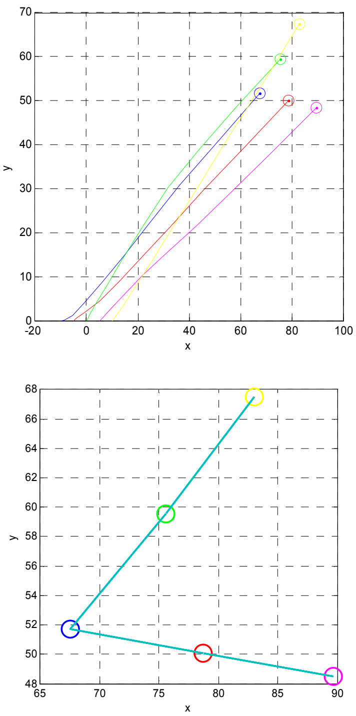
The two desired graph configurations for each simulation are also shown in figure 9b-9c. The enumerated graphs shown for each formation represent one possible permutation of the desired configuration and do not correspond to the optimal permutation for the graph transformation.

A particle swarm optimiser (PSO) was used to solve the optimisation problem using the energy deviation function. Each particle of the PSO represented a candidate solution consisting of the time history control input for each UAV. The PSO was allowed to run for a horizon time of 10 seconds.

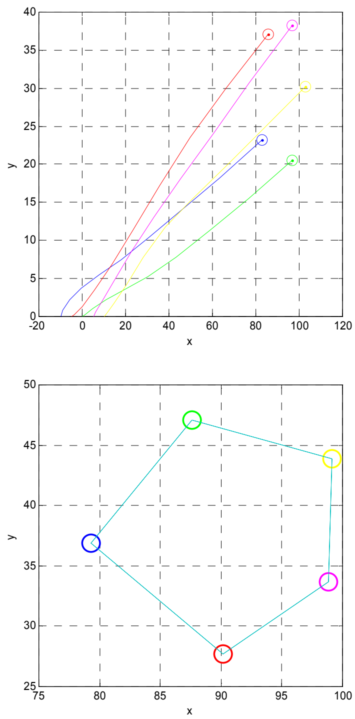
Figure 10 shows the optimal trajectories for each UAV derived by the PSO algorithm using the energy deviation function for the wedge and pentagon formations.

7. Remarks

The final permutation of each formation also demonstrated the optimal permutation of each graph configuration. This was validated using a recursive search algorithm for all permutations of the final configuration. In larger population sizes, such a process would be computationally expensive. The success of the PSO to find the optimal permutations of each configuration and the control input histories for each vehicle simultaneously, demonstrates promising results for implementation into larger populations. Future work aims to use the energy deviation function described in the previous section in an affine connection framework and use the evolutionary search techniques of PSO to solve for the resulting affine connection equations. Solving the system of equations could potentially provide analytical expressions for the optimal transformation process and yield time-parameterised trajectories that describe the motion of each vehicle through the environment. To synthesise the self-organised behaviour of natural swarming systems, the search technique would then be distributed locally to each vehicle. Current research is investigating methods to distribute search algorithms among agents with limited perspective.



a) Wedge Formation



b) Pentagon Formation

Fig. 10. Optimal trajectories and optimal permutations for the a) wedge formation, and b) pentagon formation.

8. Conclusion and Future Work

It was discussed how the concepts of swarming can be applied to a team of UAVs for a swarm-based defence system. In most applications, the problem presents itself as a formation or pattern control problem requiring decentralisation and coordination. The emergent nature of cooperating agents renders the operation of swarming systems unpredictable. Future work in the development of swarming UAVs requires a thorough understanding of the individual agent-level behaviour and the emergent system-level behaviour to encourage support in applications beyond the laboratory. A significant aspect of the control design problem is the synthesis of autonomy and intelligence to characterise the necessary decision-making, and decentralised task-allocation of swarming systems.

9. References

[1]. Fax J. *Optimal and Cooperative control of vehicle formations*, California Institute of Technology, 2002.

[2]. Fax J, Murray R. Information flow and cooperative control of vehicle formations, *IEEE Transactions on Automatic Control*, Vol. 49, No. 9, 2004.

[3]. Hao Y. *A practical framework for formation planning and control of multiple unmanned ground vehicles*, University of Delaware, 2004.

[4]. Lewis A. Affine Connection Control Systems, *Proc. of IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*, Princeton, NJ, 2001.

[5]. Muhammad A. *Graphs, simplicial complexes and beyond: Topological tolls for multi-agent coordination*, Georgia Institute of Technology, 2005.

[6]. Olfati-Saber R, Dunbar W, and Murray R. Cooperative control of multi-vehicle systems using cost graphs and optimization, *Proc. American Control Conference*, Pasadena, California, 2003.

[7]. Olfati-Saber R. *Flocking for multi-agent dynamic: systems algorithms and theory*, California Institute of Technology, 2004.

[8]. Sycara K. Multiagent systems, *AI Magazine*, The American Association for Artificial Intelligence, pp. 79-92, 1998