An Agent-based Framework for UAV Collaboration

Anup Dargar
Computer Science
North Dakota State University
Fargo, ND 58105
anup.dargar@ndsu.nodak.edu

Ahmed Kamel
Computer Science
North Dakota State University
Fargo, ND 58105
ahmed.kamel@ndsu.nodak.edu

Gordon Christensen
Computer Science
North Dakota State University
Fargo, ND 58105
gordon.christensen@ndsu.nodak.edu

Kendall Nygard
Computer Science
North Dakota State University
Fargo, ND 58105
nygard@cs.ndsu.edu

Abstract

Unmanned air vehicles are becoming an essential tool of modern warfare. Teams of unmanned air vehicles may be called upon to perform tasks that cannot easily be accomplished using an individual air vehicle. In this paper, we present a framework for collaboration among a team of unmanned air vehicles. Under this framework, each individual air vehicle serves as an autonomous agent sensing its environment and responding to changes (perceived threats) in the environment by analyzing the changes, and taking necessary actions. The response can vary from minor corrections in the air vehicle's own path, to complete re-planning of the flight path. As part of the response, additional actions must be taken in terms of notifying other air vehicles within the team of vehicles. The change in the environment must be assessed and a decision must be made to notify all the other members of the team, a subset of the team, or that a notification is not necessary. We present a knowledge-based framework for the analysis of these threats and for planning further actions. We also present a framework for the communication among the different agents. The communication framework attempts to minimize the amount of communications in order to minimize the possibility of detection.

1 Introduction

Scientific and Engineering advances in sensors, propulsion, wireless communication and other areas are rapidly making it possible to develop Unmanned Air Vehicles (UAVs) with sophisticated capabilities. Deployed individually, UAVs offer the potential to search for, detect, and destroy enemy targets in relatively complex environments. They potentially reduce risk to human life, are cost effective, and superior to manned aircraft for certain types of missions. It is desirable for UAVs to have a high level of intelligent autonomy, to carry out mission tasks with little external supervision and control. As individuals, UAVs must be capable of functioning in multiple modes of operation, such as launch, search, waypoint following, precision strike, battle damage assessment (BDA), and cooperative classification of threats. There are many types of missions for which it is potentially advantageous to deploy multiple UAVs working cooperatively in teams. This raises important issues involving tradeoffs between centralization and decentralization of control and decision-making. Cooperative control strategies for resource allocation concern ways to assign mission tasks among multiple UAVs to accomplish force multiplication effects. These strategies vary in the degree of communication required, and the degree of autonomy provided to individual UAVs. It is also highly desirable for cooperating UAVs to quickly and intelligently adapt to new battlefield information provided by sensor suites.

1.1 Path Planning

Path planning for UAVs involves using mathematical procedures for generating sequences of waypoints that can be followed by a vehicle with known flight dynamics to carry out a mission in the presence of enemy threat. The mission might involve search, surveillance, reconnaissance, or delivery of weapons. Solution methods involve computational geometry constructs that superimpose variable density meshes that discretize a continuous geographical area of interest, and employ network and combinatorial optimization search techniques (such as A* search) to generate the paths. The resultant paths avoid enemy threat centers (radar, anti-aircraft artillery) and terrain obstacles such as mountain peaks. The mathematical methods are scalable to quite large problems.

1.2 UAV Autonomy

We view the ability to quickly and autonomously generate high performance paths as being an fundamental capability for individual UAVs, as essential as the ability to sense the environment, and maneuver and regulate the aircraft. However, there are many types of missions for which it is potentially advantageous to deploy multiple UAVs working cooperatively or collaboratively as a team. Cooperative control raises a host of issues, involving such things as signal processing and communications, dynamics of formation control, distributed sensor fusion, and embedded applications. The issues are significantly complicated by the importance of the UAVs being adaptive, accepting new information from sensors on the fly, and incorporating that information into intelligent decision making while underway. Resource allocation strategies vary in the degree of communication required, and the degree of autonomy provided to individual UAVs.

1.3 Missions involving Multiple UAVs

We identify three types of potential advantages of multiple UAVs over a single UAV in accomplishing missions:

Partitioning their workload. During a given block of available mission time, multiple UAVs working independently can potentially carry out more tasks than a single UAV. Some examples include: i) Dividing the workload of a search mission geographically (assigning the UAVs to non-overlapping search areas) or functionally (assigning the UAVs to deploy differing sensors, or the same type of sensors in differing ways), and ii) Dividing the workload among differing mission tasks, such as attacking targets, or carrying out surveillance, or reconnaissance objectives.

Coordinating their workload. Multiple UAVs can achieve synergistic effects or perform missions and tasks that would be impossible with a single UAV. Some examples include: i) Searching areas that are crossed with impenetrable lines of enemy defenses, making it impossible for a single UAV to search the entire area, but easily done by two or more, ii) Acquiring a wide-baseline stereo image, using two cameras precisely synchronized and separated by a prescribed distance, iii) Sharing information acquired by sensors among the members of the team, increasing the value of the choices that individual UAVs make through their autonomous decision-making systems, and iv) Synchronizing or otherwise coordinating the activities of multiple UAVs in time, such as striking a target with multiple weapons at precisely the same time, or using a decoy to deliberately attract attention from the enemy followed by deploying strike vehicles.

Building in redundancy. The likelihood of successfully carrying out a mission is potentially enhanced if certain capabilities are replicated across a team. For example, if a UAV carrying out a task of high importance is disabled, another UAV could take on the task. Or in an architecture with a leader and followers, a line of succession procedure could appoint a new, and capable leader to replace one that becomes disabled.

1.4 Task Allocation among Multiple UAVs

As discussed in [7], we have implemented a mathematical optimization model for assigning task to UAVs that is based on linear network optimization modelling, and has the advantage of very fast computational times and scaling well to larger problem sizes.

It is a time-phased network optimization model designed to produce task assignments for all the UAVs functioning as a team each time it is run. The model is intended to run simultaneously and autonomously at discrete points in time on all of the UAVs, and explicitly assign each to a task. Tasks could include such things as carrying out a target strike, directing a sensor at a detected object to possibly clasify it, searching a particular area, or doing an assesment of battle damage. We begin with a set of UAVs with known amounts of available flight time, and deployed as a team. Tasks fall into known classes, and there are ways to associate values or scores with doing them. The path planner is able to provide probabilities of survival for a given path. Target value, available resources and survival probabilities make up the parameters for the model. Some information about the task is available at the outset, and further information about the tasks and their positions is obtained by the UAVs carrying out searches and classyifying what they find.

Task allocation decisions are determined by solving the network optimization model. The receipt of new task information is viewed as event that triggers the formulation and solving of a fresh network optimization problem that reflects the new current conditions.

1.5 Architectures for Coordinated UAV Behavior

Most proposed architectures for command and control of multiple UAVs adopt a hierarchical architecture. Figure 1 illustrates an architecture that we are developing.

The hierarchy controls the complexity of the coordinated control strategies, yet retains essential modeling constructs that are tractable at each level. Agents at each level have access to the knowledge and databases corresponding to their level, and can also freely access information from the next lower level. An agent at any given level has no need to know about information, knowledge, or data at any level higher than its own.

At the top level of the hierarchy, the Inter-team Cooperative Control Planning Agent is basically responsible for configuring teams of UAVs and providing them with their goals. This agent has visibility of the highest level goals and doctrinal goals for the overall mission. If teams are pre- configured before takeoff, this agent will primarily be responsible for determining if teams should be reconfigured as new information is received and situation awareness improves. The models it invokes are expected to request and receive information from the Intra-team Cooperative Control Planning Agents at the next lower level. Based on this information, this agent may autonomously abandon certain high-level goals in favor of others.

The domain of responsibility for a second-level Intra-team Cooperative Control Planning Agent involves the division of responsibilities among the UAVs working within a configured team. Leadership responsibilities and coordination mechanisms depend on the mission, the models available to support accomplishing the mission, available data, and the capabilities of the UAVs comprising the team.

At the third level, the UAV planning agents function specifically within an individual aircraft. These on-board planners should accept a specific goal that is appropriate for a single UAV, then invoke models for path planning and scheduling to meet the goal.

Finally, the UAV regulating agents are charged with providing command sequences for the aircraft itself, to do such things as following trajectories, turning sensors on an off, executing maneuvers, changing speed, and releasing weapons.

2 UAV communication

In order to cooperatively plan and execute a mission, UAVs must share some information involved in executing both the task allocation model and the path planner. The path planner must have access to any new threat or target info or UAV position that would effect the path of the UAV. The probabilities produced by the path planner, and weapons and fuel availability must be accessible to the task allocation model. This is potentially a large quantity of communication in a changing environment.

Commonly, one of two methods would be used to control the communication: polling or subscription/publish. Using subscription, a UAV agent would subscribe to the information another UAV agent has by sending a message requesting any changes to the information base. That UAV would then forward the requested information as it came in. This may lead to inefficiency if the UAV must subscribe to more information than is necessary. Polling requires the UAV agent to request each piece of information on a regular basis. If there is no information the request is wasted.

In a hostile environment, cummunication must be kept to a minimum. Excessive communication may make it easy to track and destroy the UAV. Both polling and subscription present inefficiencies by introducing unneccesary communication.

As an alternative, we are developing ways to evaluate and generate new information with various heuristics which model the peer UAVs' need for information. If a UAV determines that it can provide information that would affect another UAV, it volunteers the information as soon as possible. Delays are possible if the information must be generated. Subscriptions and requests for information can be reduced or eliminated.

As described above, a task allocation algorithm is used to assign each UAV a task. This algorithm is run on every UAV. Given the same initial values, every UAV will determine the same assignments. However, two situations must be accounted for: data unique to a UAV, and a support algorithm that must be distributed because of time.

Data unique to a UAV may include a threat which was previously unknown. In this case, by the criteria stated above, the UAV will heuristically detemine which UAVs may be affected by the new information and transmit the information only to them. The affected UAVs would then re-evaluate their task allocations and modify them if appropriate.

In order to parameterize the task allocation algorithm, the path planner is used. The path planner is

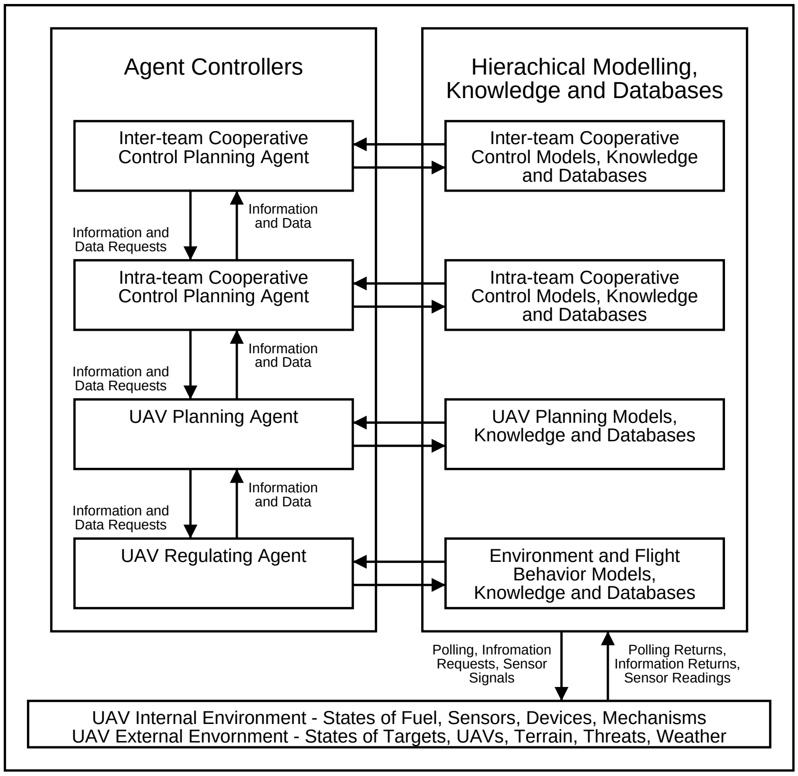

Figure 1: Hierarchical Cooperative Control Hierarchy for UAVs. Requests for information propagate downward, response propagate upward.

able to determine mission probabilities given a UAV-target pair. This planner can take a significant amount of time to execute. If there are m UAVs and n targets then for the task allocation algorithm to run $m * n$ path planners must be run. Running $m * n$ planners on each UAV could be excessively slow. Allowing each UAV to run only the planners for its own paths exploits the natural parallelism of the system, reducing the complexity to n and saving critical time.

Task reallocation is triggered by new information. This in turn triggers path planners on the affected UAVs. Each UAV is able to determine which other UAVs will be rerunning task allocation and sends its updated path probabilities only to those UAVs. After all path planners and CTPs complete the UAV will progress to its assigned target.

We want to minimize communication in the hostile environment. With this in mind, we divide the information into three types: that which needs to be shared with all other UAVs, that which does not need to be sent to any and information which is only shared with only some of the UAVs. New information of the first two types are quickly handled by the respective response. That of the last type can be evaluated on the fly by a heuristic which determines the appropriate/affected UAVs.

3 Knowledge Based Agents

Knowledge-based systems are increasingly playing an important role in the advancement of all fields of science, and technology as well as business and economy. Consequently, knowledge based architectures have undergone tremendous development in recent years resulting in several successful paradigms that are currently in widespread use. The most promising of these paradigms fall under the general category of “Task Specific Architectures”. These methods have been successfully applied in numerous domains, such as industrial design, agricultural planning and financial planning. A common underlying scheme behind these applications is that they rely on heuristic knowledge of the domain rather than precise numeric models. This characterization implies that knowledge based systems are inherently less precise than other techniques. However, under circumstances where a pre-

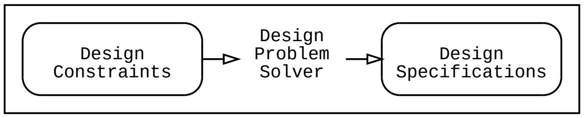

Figure 2: Information Processing Task of Routine Design

cise solution method is not available, and in situations where a precise model exists but is computationally too complex to implement, it becomes crucial to rely on knowledge-based alternatives.

3.1 Background on Key Knowledge Based Systems Methodologies

In our overall problem solving architecture, we integrate a number of Generic Tasks (GT's). We will present here as an example, two GT problem-solving types; hierarchical classification ([3],[8]) and routine design ([2],[5]).

Hierarchical classification is intuitively a knowledge organization and control technique for selecting among a number of hierarchically organized options. Hierarchical classification relies on developing a hierarchical taxonomic view of the problem on hand and classifying arising situations into a category within this taxonomy. As a selection methodology, it is readily understood, and will not be further described here. For more detail on hierarchical classification see ([3],[8]). Routine design is not as intuitive and will be described in more detail here.

The Routine Design GT forms a high level planning template for the generation of design assemblies as well as for the generation of plans of actions. We have previously used Routine Design as the backbone of an engineering design system [4] as well as for an agricultural planning system [6].

The basic intuition underlying Routine Design is that a successful Artificial Intelligence technique should follow the same method of reasoning as humans do to be able to use predictions to form hypotheses about how a human designer (planner) would behave in situations that have not been observed or analyzed and act accordingly.

A consequence that follows from this intuition is the use of hierarchical structures of design specialists to perform design, each responsible for a particular part of the overall plan. Hierarchies are used not because the design (or planning) activity is intrinsically hierarchical, but because hierarchical decomposition is a typical means utilized to manage complexity.

The input is a set of planning constraints, and the output should be a full set of specifications for the required plan. The information-processing task can be summarized as follows and as shown diagrammatically in Figure 2:

- Working on a problem that has been done many times before, each time with different but similar requirements, until the problem solving knowledge has been compiled into a form that allows efficient solution of the problem, and
- Design proceeds with each sub-problem by selecting from previously known sets of well-understood alternatives. In a sense, plans generated using a routine design framework are novel assemblies of plan segments of well understood nature.

3.2 Use of Generic Task Agents for Control of UAVs

As part of the overall architecture, Generic Task based knowledge based systems will be integrated with more precise numeric models. Under circumstances where sufficient planning time is available, precise numeric models will be the preferred choice. However, under circumstances where time is of essence, the complexity of the numeric models will be avoided by relying on knowledge based results. Within this framework, a hierarchical classification system is integrated as an independent agent within each of the UAVs for the classification of emergent situations. For example, as a result of a UAV being notified of the existence of a new threat, it would quickly classify the situation as one where it needs to re-plan its route, abandon its current mission, ignore the threat, ...etc. Another knowledge-based component that will be integrated within each UAV is a routine planning agent to plan the sequence of actions (plan the new path, notify the command center, notify other UAVs, ...etc.) to be followed by the UAV after classifying the current situation.

4 Example

One potential scenario involves three UAVs (a, b and c) enroute to attack three targets.

1. During flight, b discovers a threat. b determines that the threat does not affect a but does affect c. After sending a message to c; b and c each replan paths to their respective targets. Both c and b are able to find a path of similiar quality

to their previous paths and adjust themselves accordingly.

2. Alternatively, b may not have been able to find a path of similiar quality. In this case, b would determine that the group needs to reevaluate tasks and sends this information to both a and c. All three evaluate paths to all targets and share their information with the other two UAVs. If a viable solution is found, each UAV will assign itself to the appropriate target, otherwise, the mission will be aborted by all three.

In the first case, only the information about the new threat must be transmitted to c only. Replanning for this new threat is handled by the UAVs independently.

The second case presents a case in which more UAVs will need to cooperate in order to solve the problem. A total of nine messages are sent. First, the message from the first case (1). Then b shares its problem with maintaining the current plan with each other UAV (2). Finally, each UAV shares with the other UAVs its capabilities for each target $(3 * 2)$.

Without the heuristic determination of who is affected by new information, both cases send eight messages. b would send out the new threat information to both a and c (2). Upon recalculation, each UAV shares with the other UAVs its capabilities for each target $(3 * 2)$. So, if case 2 occurs less than 86% of the time then the communication with heuristics will involve less communication.

5 Future work

The linear network optimization model has already been implemented, but additional work must be done to break the independence of sequential tasks in this model to make it realistic for the application. This can be possibly done iteratively, or by assigning subsequences rather than single tasks. Combinatorial optimization approaches, such as the generalized assignment problem, are well-established as application models in routing and scheduling. This approach is of value when simultaneously striking a target with multiple UAVs, to increase the probability of destroying the target. Work has to be done on this approach and implemented. Neuro-dynamic programming is a relatively new class of dynamic programming methods for control in sequential decision problems under stochastic condition. They have been effective in problems with large state spaces in which accurate models are difficult to formulate. Bertsekas [1] has pioneered the approach. This approach is currently being investigated and will be implemented soon. The hierarchy agent architecture discussed is a good starting point for investigation. More helper agents could be added at each level to handle the complex task of that level and to make the UAV very veratile in handling alien condition. The communication issue between agents on the same UAV is not complex and can be handled easily, but new and latest techniques need to be explored for communication between two seperate UAVs. We have tried to reduce the communication between them in this paper but safer transporting medium is still to be explored.

References

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena, 1996.

[2] D. Brown and B. Chandrasekaran, *Design Problem Solving: Knowledge Structures and Control Strategies*, Pitman, 1989.

[3] T. Bylander and S. Mittal, "CSRL: A Language for Classificatory Problem Solving and Uncertainty Handling," *AI Magazine*, Vol. 7, pp. 66-77, 1986.

[4] A. Kamel, *Generating Multiple Design Alternatives of Composite Materials using a Generic-Task Approach*, Michigan State University, 1994

[5] A. Kamel and J. McDowell and others, "Multiple Design: An Extension of Routine Design for Generating Multiple Design Alternatives," *Artificial Intelligence in Design '94*, 1994.

[6] A. Kamel and K. Schroeder and others, "Integrated Wheat Crop Management Based on Generic Task Knowledge Based Systems and CERES Numerical Simulation" *AI Applications*, Vol. 9, pp. 17-28, 1995.

[7] Kendall Nygard and Phillip R. Chandler and Meir Pachter, "Dynamic Network Flow Optimization Models for Air Vehicle Resource Allocation," *American Control Conference*, 2001.

[8] J. B. Sticklen and B. Chandrasekaran and others, "Modularity of Domain Knowledge," *Expert Systems: Research and Applications*, Vol. 1, 1987.