A Formulation and Heuristic Approach to Task Allocation and Routing of UAVs under Limited Communication

Article · December 2013

DOI: 10.1142/S2301385014500010

Chelsea Sabo
The University of Sheffield

Kelly Cohen
University of Cincinnati

A Formulation and Heuristic Approach to Task Allocation and Routing of UAVs under Limited Communication

Chelsea Sabo
Department of Computer Science
University of Sheffield, Sheffield, UK S10 2TN
c.sabo@shef.ac.uk

Derek Kingston
Air Force Research Laboratory
Wright-Patterson Air Force Base, Ohio, 45433
Derek.Kingston@wpafb.af.mil

Kelly Cohen
Department of Aerospace Engineering
University of Cincinnati, Cincinnati, Ohio, 45221
kelly.cohen@uc.edu

Abstract

Unmanned Air Vehicle (UAV) teams are anticipated to provide surveillance support through algorithms, software, and automation. It is desirable to have algorithms that compute effective and efficient routes for multiple UAVs across a variety of missions. These algorithms must be realizable, practical, and account for uncertainties. In surveillance missions, UAVs act as mobile wireless communication nodes in a larger, underlying network consisting of targets where information is to be collected and base stations where information is to be delivered. The role of UAVs in these networks has primarily been to maintain or improve connectivity while undervaluing routing efficiency. Moreover, many current routing strategies for UAVs ignore communication constraints even though neglecting communication can lead to suboptimal tour designs. Generating algorithms for autonomous vehicles that work effectively despite these communication restrictions is key for the future of UAV surveillance missions. A solution is offered here based on a variation of the traditional vehicle routing problem and a simple communication model. In this work, the new routing formulation is defined, analyzed, and a heuristic approach is motivated and described. Simulation results show that the heuristic algorithm gives near-optimal results in real-time, allowing it to be used for large problem sizes and extended to dynamic scenarios.

Keywords

UAVs; Task Allocation; Vehicle Routing; Communication Networks; Multi-Agent Decision-Making

Nomenclature

N	=	Number of Requests
M	=	Number of Vehicles
i	=	Request Index
A_i	=	Arrival time of request i
P_i	=	Pickup time of request i
D_i	=	Delivery time of request i

I. Introduction

Surveillance functions are comprised of various means for acquiring and processing information needed by a high-level decision maker [1]. In the not too distant future, UAV teams are anticipated to augment this much needed service more effectively [2]. Situations where UAVs are being increasingly used in supplying surveillance include search and rescue missions, forest fire monitoring, traffic surveillance, agricultural remote sensing, pipe-line monitoring, border patrol, and the military. Because of the numerous applications of UAVs, their presence will continue to grow. Effective and safe algorithms for autonomy are necessary to ensure the continued integration of UAVs into shared air space.

A very important aspect in the design of UAV technologies is the ability to collect and transmit the data collected to a decision making authority such as command and control headquarters. Ideally speaking, a group of collaborating UAVs should be able to communicate "whenever and as much as they need to [2]." UAVs rely on communication to send command and control signals to ensure nominal system operations and to transmit remotely sensed data to guarantee mission efficiency and success. While this should be the standard, it is far from a reality.

One of the advantages to using UAVs for surveillance is due to their increased mobility. Due to this, they primarily use wireless networks to convey information where the quality of the wireless communication is based on the signal of the receiver, noise, and interference [3]. These parameters dictate the rate at which data can be sent and the probability of it being received. The commonly used Shannon-Hartley theorem [4] describes the wireless connection between two nodes in a network. The law states that there is a maximum data rate that be transmitted over a channel without error (C), and it has an inverse relationship with distance (d). The law is governed by the signal-to-noise ratio (SNR) which is influenced by the power between the nodes and the communication link (Eqn. 1). The SNR is a function of link gain (K_0) which is dependent on power, antenna gain and orientations, and radio electronics quality. It is also a function of noise (N_0) and the signal decay exponent (ε) which encompasses the path loss in free space. The effect of the link gain and the noise can be simplified into one parameter (K).

$$SNR = \frac{Power(d)}{N_0} = \frac{K_0}{N_0 d^{\varepsilon}} = \frac{K}{d^{\varepsilon}} \tag{1}$$

It is easy to see from Eqn. 1 that when distance between nodes is short, SNR is high and vice versus. It is this that governs the Shannon-Hartley law (Eqn. 2):

$$C = B \cdot \log_2(1 + SNR) = B \cdot \log_2\left(1 + \frac{K}{d^{\varepsilon}}\right) \tag{2}$$

It is clear from Eqn. 2 that both the channel bandwidth and the distance have a big impact on the effective bandwidth and vary throughout missions as the UAVs move about the network. This inverse relationship between distance and effective bandwidth can be seen from Figure 1 below. Additionally, the link gain and decay exponent influence the effective bandwidth, and although they change and are hard to predict, the fluctuations are small in comparison.

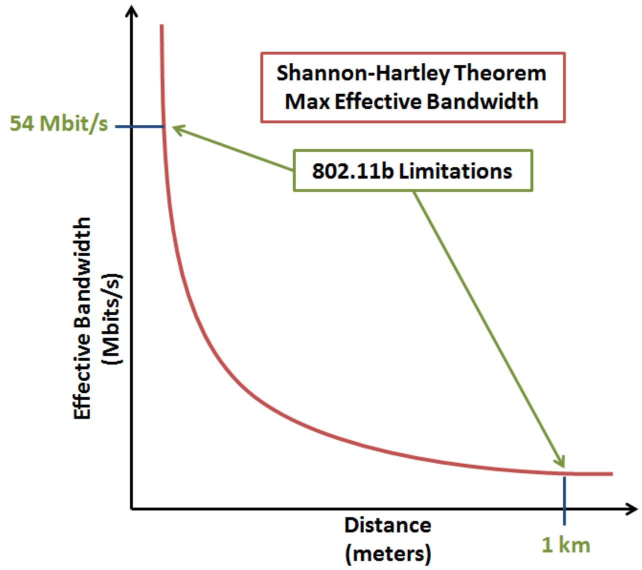

Figure 1: UAV Communication Model and Limitations

In addition to the Shannon-Hartley law which theoretically describes the wireless connection between nodes, there are realistic constraints on this model. Specifically, there are limitations on the maximum effective bandwidth and the maximum distance information can be transmitted over. Typical radios used for UAVs in networks have communication ranges that are compatible with those provided by technologies such as IEEE 802.11b (Wi-Fi) [5]-[8]. It is these radios that impose the limitations on the channel communication where the limits for effective bandwidth and distance are 54 Mbps and 1 km, respectively (Figure 1).

Although low bandwidth 'command and control' signals can generally be delivered reliably, there is a distinct need for a high bandwidth delivery mechanism. Particularly in surveillance missions, UAVs are required to pick up high bandwidth data like video, images, atmospheric samples, and other remote sensing data [9]. The limitations of the communication channels above highlight the low data rate capabilities over long distances. Additionally in certain environments (i.e. urban, mountainous, etc), line-of-sight issues can prevent communication altogether. These issues can make delivery of high bandwidth data slow or nearly impossible even though missions often are time-dependent on this information.

Beyond these limitations which describe the effective bandwidth of a wireless channel, communications are complicated by shadowing, multipath fading, and path loss effects [9]. Shadowing occurs when obstacles are in the line-of-sight and depends on obstacle density. Multipath fading is the result of scatterings and reflections of the signal by objects in the environment. Lastly, path loss is inevitable due to the probability that some information will be dropped depending on the modulation scheme and transmission data rate [3]. These effects can be detrimental to communication and difficult to predict [10]. Additionally, adversaries who attempt to intentionally "jam" communications are a common concern for UAVs. In these situations, missions would benefit from a strategy in which UAVs delivered information within a "trusted" or secure area.

Communication constraints and limits define all UAV activity, and there is a growing interest in the research community to expand current capabilities. By considering communication, the surveillance problem with UAVs can be viewed as a simplified communication network where information needs to be collected from targets and delivered to base stations (Figure 2), and the communication link between the UAV and tower (and between UAVs) can be described by the model above (Figure 1). The objective is then to route data to the destination with minimal delay with sufficient data rate and minimal packet loss.

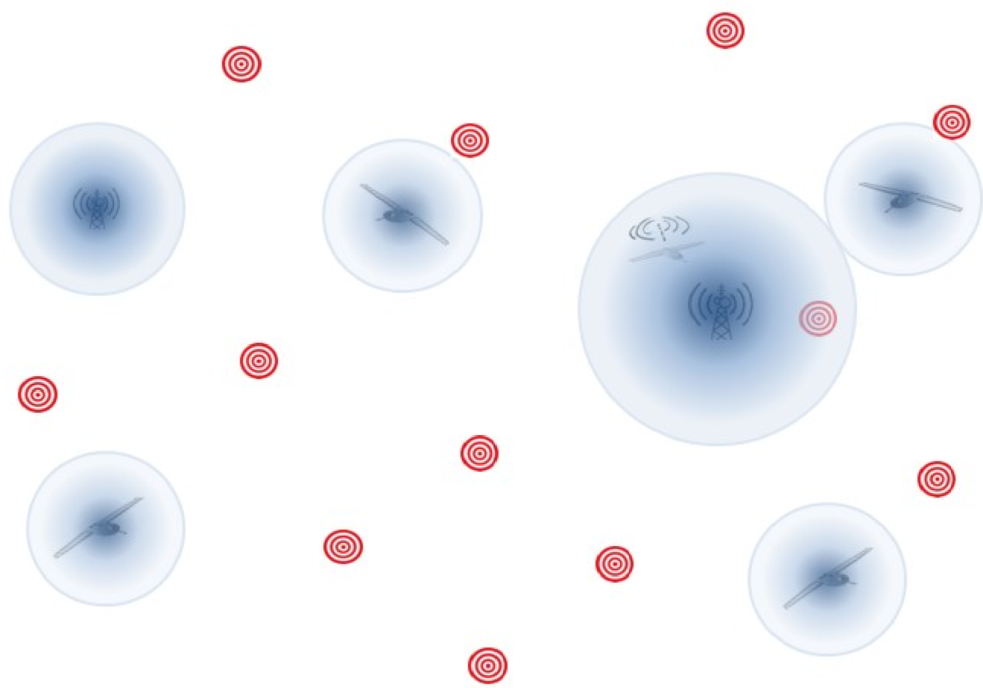

Figure 2: Environment where UAVs need to service targets and communication constraints are considered.

UAVs in communications networks have been studied extensively from the perspective of communication performance. In the network, UAVs act as mobile nodes to connect stationary nodes. Frew and Brown discuss the role of unmanned aircraft, particularly small UAVs, in meshed airborne communication [5]. It has been shown that using UAVs can restore communication in stressed or fractured networks, improve quality in communication networks, expand the range of communication, and overcome terrain constraints [4]-[23].

One of the main problems for UAV teams in communication networks is to accomplish some objective while maintaining connectivity. Several common objectives include tracking targets, maximizing coverage, rendezvousing in space, or formation/flocking [24]-[29]. The other main problem involving UAVs entails designing trajectories for UAVs to improve communication performance. This requires controlling chains of relays between end nodes [30] or creating relays to support a larger, fixed network [31]. Some work even takes this a step further to exploit UAVs to use as ferries for node mobility in meshed networks. These are referred to as delay-tolerant networking as data is considered to not be transferable as quickly as over a direct link [13]-[23].

While communication constraints for UAVs are often studied in the context of communication networks, it is desirable to understand how these constraints effect task allocation and routing for UAVs. In these problems, maintaining network connectivity is secondary to the task of returning information in a timely manner. Sometimes it just isn't possible or desired to require that the network is connected at all times. Additionally, trajectory planning for UAVs in a network can be difficult and suboptimal since communication can be highly dynamic and hard to predict. If there are uncertainties or failures in the communication links, it could cause mission failure or performance degradation. Currently, solutions to these problems are often simplified in that agent roles don't change or scenarios are too simple or static (does not evolve over time). These simplifications hardly represent the real world and an alternate approach is needed.

As a result of the some of the work in delay-tolerant networks, it has been shown that there is a coupling between resource allocation, optimal tour design, and trajectory optimization [17]. Furthermore, these layers are influenced by the system itself (i.e. environment, communication, and vehicles). Typically resource allocation and tour design is done without the consideration for system models. While some work has been done to show that ignoring systems models can lead to suboptimal tour design [14], there is a lack of literature that addresses how routing strategies should be altered to consider communication. Additionally, some work addresses different routing methodologies for ferries [20], but is done so in the context of communication networks which has little implications to the choice of a global allocation and routing strategy. This work addresses this task allocation and routing problem for UAVs while incorporating communication limitations.

The resource allocation problem associated with sending UAVs to collect information about targets falls within the general framework of Vehicle Routing Problems (VRPs) [32]. Here, the context of the VRP is to deliver goods to customers that have placed orders (or in this case, for UAVs to service target requests) from a central depot. To simplify the problem and put it in the framework of a VRP, the communication range of the UAV is considered to be very small when compared to the base station. This results in a requirement for the UAV to visit nodes corresponding to the targets, or requests, and visit the base station, or central depot, to deliver the data. This "simplification" is close to reality when there is limited power on board (which can be typical of small UAVs), when sensor data is large and cannot be easily transmitted over long distances or due to line-of-sight issues, when it is necessary or desirable for UAVs to deliver information to a more secure connection, or other situations where communication is severely limited.

Vehicle routing comprises a wide range of problems where multiple vehicles are sent to service a given set of customers or requests. Each problem tends to be unique in that it has different customer constraints and requirements and various operating environments. Therefore, solution strategies vary depending on the specific problem. The VRP concerns scheduling a set of vehicles routed to service a set of customers such that a minimum transportation cost is found while meeting operational constraints. In the VRP, all vehicles start and end at a depot (or multiple depots). In many cases, customers need to be serviced within a given time window and vehicles have a limited capacity. Typical objectives are the minimization of the global transportation cost (either distance or time and the corresponding associated driver/vehicle cost); number of vehicles; multiple objectives: travel time, route distance, and vehicle load; or penalties associated with not meeting customer demands. Also, some cases consider dynamic or stochastic environments: those which only have partial information about customers and their demands apriori [32].

Because of its numerous practical applications, there is much literature related to VRPs. However, interest lies in allocating UAVs to service requests and deliver information back to a single depot or multiple depots (similar to the Pickup and Delivery Problem (PDP) since UAVs must be allocated to "pickup" data and then "deliver" it). As it is necessary to get this information back as quickly as possible, a minimum latency objective is desirable. In this situation, it is probable that the UAV will need to make multiple trips similar to the VRPs with multiple trips (VRPMs).

The VRP with multiple trips (VRPM) is a variation of the VRP in which vehicles are allowed to make multiple trips in a single planning period. Typically, the assumption is that a vehicle finishes its route at the end of the planning period when it returns to the depot. However, it has been shown recently that sometimes this assumption does not hold [33]. In this problem, the setup is similar to the VRP except that each of the vehicles' trips is constrained by trip duration. While this problem is similar to the allocation of UAVs with communication constraints (sometimes it is in the vehicle's best interest to make multiple trips from the communication range), there is no restriction than can intuitively be put on the vehicles that maximizing the trip length. In some instances, it might be better to pick up a single request, and in others, it might be better to pick up dozens (which depends on the specific instance). Therefore, this restriction cannot be posed as a constraint on the trip like it is in the VRPM; it is reflected in the objective.

The minimum latency problem is sometimes referred to as the Travelling Repairpersons Problem (TRP). In this problem, the latency of a request in regards to the VRP is the total distance travelled up to the request. The minimum latency VRP is the minimization of the sum of the latency of each request. This problem is classified as cumulative, because the request's cost accumulates until it is serviced [34]. In the TRP, the repairman needs to minimize the time to service all of the requests (which is dependent on how long until he gets to the location). In this effort with UAVs, the latency associated with the delivery of the requests is of interest (rather than the pickup). Therefore, there needs to be a strategy for collecting targets and delivering them under a minimum latency objective.

In the classical VRP, it is assumed that there is only one depot where all the vehicles originate from. The multi-depot VRP requires a fleet of vehicles stationed at multiple depots to service a set of customers. Vehicles need to be assigned not only to customers, but the customers also need to be assigned to depots (from which depot the customer will be served). The objective is to minimize the number of vehicles and the distance travelled. Many solutions to the MDVRP rely on the assumption that each vehicle is assigned to a depot and then cluster requests around a depot [35]. However, these solutions fail to address the much more complex case where a vehicle might not be restricted to a single depot. Beyond this simplifying assumption, this body of work only obtains solutions that are great for minimizing the distance travelled by all the vehicles. There is a need for algorithms that emphasize mission success (i.e. the importance is put on the requests rather than on minimizing route cost) coupled with the complexity of multiple depots.

While Pickup and Delivery Problems (PDPs) are a variation of VRPs, the main distinction between the two is that VRPs are PDPs where either the pickup or the delivery location is located at the depot. Each request in the PDP has a unique delivery location. These problems have had more attention in the last decade, but limited solution strategies have been establish, because the PDP is known to be NP-Hard [36]. In PDPs, each vehicle has a start location, end location, and loading capacity. Each request has a start location (pickup), end location (delivery), and a load size. The vehicles have to be routed such that each request picked up and delivered without violating any constraints [37].

Arguably the hardest, and often most realistic, version of the PDP is the unconstrained dynamic case with large numbers of service requests as is of interest here. Because pickup and delivery problems with time windows (PDPTW) and vehicle capacity [38]-[40] have many practical applications, these have been studied extensively. Generally when discussing UAVs in literature, the restrictions of time windows and vehicle capacity do not exist. It is assumed that each vehicle can take as many pictures or videos of a target (or targets) as needed without exceeding its capacity. Also, the only "time windows" present are due to the need to return information to command and control as quickly as possible, e.g. for tactical reasons or due to cost. Here in the problem formulation with UAVs, it is assumed that these 'time windows' are not tight, and that the overall time in which information is returned is more important (it becomes the objective). Savelsbergh and Sol (1995) provide other examples of realistic problems where there are not restrictive time window constraints [37]. However, these cases are few.

In the formulation with UAVs, the PDP is setup such that there are targets that need to be visited (to collect information from) and a communication range where the information is dropped off. Because this is a range and not a single point depot, each target typically has a unique delivery site that's best based on its location (e.g. the closest point in the range that would create a direct path from the target to the range). The UAVs must be allocated to "pickup" targets and then "deliver" them to their corresponding destination within the communication range. However, each target can also be delivered at another target's delivery location (each target has a "best" delivery location, but is satisfied by being delivered at any point within the range). Because of this nuance of the problem (that any information can be delivered at any number of drop off locations), the few applicable algorithms developed for the unconstrained PDP would not satisfy the given problem. Moreover, the objective functions studied in these cases focus on the minimization of route cost or a similar variation whereas it's important in UAV surveillance problems to deliver information as quickly as possible.

As discussed previously, there is a need for algorithms that are designed to allocate vehicles to operate around realistic communication restrictions. Furthermore, there are other applications in vehicle routing and scheduling that this same behavior is desired. While this falls within the domain of VRPs, there is no literature that addresses this problem specifically or this particular combination of constraints and objectives. Although the problem at hand relates to several variations of the VRP (and is a generalization of these problems), the common formulations and solution strategies to these problems are deficient [32]. It was shown in previous work [41]-[42] that the order of the problem with UAVs is very large (much larger than the traditional VRP) which means the scalability of the solution is of particular interest. This is especially true in dynamic scenarios (most VRPs with UAVs are dynamic in nature) where inputs are changing, and new solutions need to be recalculated quickly. This research was initiated by evaluating approaches to resolve the issue of allocating UAVs while working around these "communication bottle-necks" inherent in today's co-operative UAV networks.

Solutions to VRPs tend to be problem specific and require a unique solution approach. Optimal methods like variational calculus, game theory, dynamic programming, and linear programming tend to be good at finding solution that perform well for small, specific cases, but they often don't scale and fail in the case of a little uncertainty. An optimal brute force policy to the UAV problem has only found solutions with up to 9 requests and 6 vehicles [43], and the execution times to these solutions are too large to be practically used in dynamic scenarios (to update a solution every time the problem changes). While search strategies (i.e. tabu search, genetic algorithms, etc) can generally be used for any VRP, these solutions can produce inadequate results when the solution space is very large (as it is here). Furthermore, they can be computationally time consuming to find good results and are non-deterministic. While some approaches have improved recently in optimality and execution time, for example market-based approaches [44], there will always be a trade-off between quick solutions and near-optimal solutions. Again because of the scaling issues with VRP related problems, it has been identified that one of the focus areas for the future of these systems is the need to be robust and scalable [45]. This and the success of heuristic solutions in similar problems [46]-[49] motivated a unique solution based on a heuristic strategy.

Again, the dynamic variation of VRPs (DVRP), where only partial information about customers and their demands are known apriori, is relevant to problems involving UAVs. Because the environment is changing in the dynamic scenario (i.e. new targets arrive), it requires that the solution be re-evaluated every time this happens. Solutions to DVRPs (and especially dynamic PDPs) are generally found by either re-solving any time a new target arrives or using a heuristic for target insertion [50]. This means that the underlying static solution is particularly important since it is either solved repeatedly or used as a base by which to incorporate the changes. It is also important to note that re-solving as the scenario changes requires a solution which can be calculated quickly or in real-time. Short execution times for solutions to realistic scenarios are essential for VRPs with UAVs. Due to the significance of the static case, it is the focus of the work here.

While optimally allocating UAVs is an important growing concern as UAVs are more frequently used, this scheduling issue where the emphasis is on the request arises in many fields. Vehicle routing problems have been studied extensively, but often the objectives are on distance travelled, vehicle cost, customer satisfaction, meeting application specific constraints, or a combination of these. In this research, the emphasis is on the return of the information picked up or received as the goal for UAVs is to return data with a minimum delay. Though similar objectives and problems are studied, to the best of our knowledge, none have addressed this issue even though it has numerous applications. This is often an issue when scheduling delivery vans and trucks; distributing mail and packages; dispatching trucks; etc. Furthermore, additional applications include logistics and robotics.

There is a distinct need for an efficient methodology for collecting information, particularly high-bandwidth data, from targets anywhere in the environment in addition to the need to reliably get this information back in a timely way despite limited communications. This research addresses this objective with a unique VRP formulation (Section II). A polynomial size 2-phase approximate algorithm (cluster first, route second) is presented and analyzed for the new problem (Section III). Simulations are developed for the static case and analyzed based on nearness to optimality and execution time via Monte Carlo runs (Section IV). It is also demonstrated that the proposed solution scales quadratically for the NP hard problem examined. Finally, conclusions and future work are discussed in Section V.

II. Problem Formulation

In this scenario, it is required that multiple UAVs travel beyond their high-bandwidth connection to collect information and return to a communication range to transmit it back to command and control. In these situations, it is necessary to gather this information as quickly as possible; e.g. for tactical reasons or due to cost. The problem then becomes how to allocate these UAVs such that they collect data from a number of targets and how often to return home.

A. Problem Definition

This problem is formulated such that M vehicles are allocated to service N requests. Each of these requests, $i \in N,$ is characterized by:

$N_i^+ \in N^+$	Origin of request i
N^-	Set of all points containing drop-off vertices (all points in the tower range)
$N_i^- \in N^-$	Drop-off location if delivering immediately after servicing request i
$V := N^+ \cup N^-$	Set of all points containing origin and drop-off vertices of requests

while each vehicle, $k \in M,$ is characterized by:

$k^+ \in M^+$	Starting point of vehicle k
$k^- \in M^-$	Ending point of vehicle k
$W := M^+ \cup M^-$	Set of all points containing start and end vertices of vehicles
v^k	Velocity of vehicle k

For all $i, j \in V \cup W$, the associated travel from i to j can be characterized by:

d_{ij}	Travel distance from i to j

A pickup and delivery route R_k for vehicle k is route through V_k, a subset of V, such that:

1. The route starts at k^+
2. Each location in N^+ gets visited exactly once
3. Some N_i^+ needs to be visited before N_i^- for all $N_i^+ \cup N_i^- \in V_k$
4. The route ends at k^-.

Additionally, a plan is a set of routes $R := \{R_k | \ k \in M\}$ such that: R_k is a route for vehicle k for each $k \in M$ and $\{V_k | \ k \in M\}$ is a partition of V. Given a route R, the cost associated with the route is given by an objective function $J_T(R)$. Then, the pickup and delivery problem becomes:

$$min\{J_T(R) | \ R \ is \ a \ pickup \ and \ delivery \ plan\}$$

B. Variables

Three different variables are associated with this problem. The first is equal to 1 if request i is assigned to vehicle k and 0 otherwise:

$$z_i{}^k \quad where \quad i \in N, k \in M.$$

The second is equal to 1 if vehicle k travels from i to j and 0 otherwise:

$$x_{ij}{}^k \quad where \quad (i,j) \in (V \times V) \cup \{(k^+, j) | j \in V\} \cup \{(j, k^-) | j \in V\}, k \in M.$$

The third variable is equal to 1 if at request i vehicle k visits the communication range and 0 otherwise:

$$u_i{}^k \quad where \quad i \in N, k \in M.$$

While the setup is similar to the VRP and PDP, an additional variable is assigned for this problem that allows each UAV to decide whether to visit the communication range at a request before visiting the next request. The go/no go decision to visit the communication area is represented by u and can be either 0 or 1. Therefore in addition to the variable x, the UAV would also need to make the decision u of whether to go directly to the next target ($u = 0$) or whether to go through the communication range to transmit data ($u = 1$). This is such that data can be transmitted back to the home base periodically throughout the mission.

The problem then becomes (where R is a feasible pickup & delivery plan and J_T is the objective function as subsequently defined):

$$min\{J_T(z, x, u)\}$$

C. Constraints

The minimization of $J_T(x)$ is subject to the following constraints:

$$\sum_{k\in M} z_i^k = 1 \qquad \text{for all } i \in N \tag{3}$$

$$\sum_{j\in V\cup W} x_{ij}^k = \sum_{j\in V\cup W} x_{ji}^k = z_i^k \qquad \text{for all } i \in N, k \in M \tag{4}$$

$$\sum_{j\in V\cup\{k^-\}} x_{k^+j}^k = 1 \qquad \text{for all } k \in M \tag{5}$$

$$\sum_{i\in V} u_i^k \geq 1 \qquad \text{for all } k \in M \tag{6}$$

$$x_{ij}{}^k \in \{0,1\} \qquad \text{for all } i, j \in V \cup M^+, k \in M \tag{7}$$

$$z_i{}^k \in \{0,1\} \qquad \text{for all } i \in N, k \in M \tag{8}$$

$$u_i^k \in \{0,1\} \qquad \text{for all } i \in N, k \in M \tag{9}$$

Equation (3) ensures that each request is only assigned to one vehicle. Equation (4) ensures that each vehicle only travel to a request location if that request is assigned to that vehicle. Equation (5) ensures that each vehicle starts at the correct locations. Equation (6) ensures that each vehicle visits the communication range at least once. Equations (7) - (9) require that the decision variables be in binary form.

D. Assumptions

To allow for a good comparison across solution methodologies, all UAVs and targets are considered homogeneous. That is, all UAVs have similar speeds, sensors, and communication abilities, and all targets only need to be serviced (visited), and there is no need for loitering, approaching at specific directions, or sweeping the area for targets. Additionally, collision avoidance is considered a non-issue as the UAVs could work at different altitudes. No flight dynamics are considered, and therefore, the UAVs simply need to fly straight paths to and from the targets (or depot). This allows for the target locations to be the corresponding state of the UAVs and the cost to go from state to state, the distance between locations. However in this formulation, the cost to go from state to state doesn't necessarily need to be the Euclidean distance (for example, it could be Dubins paths).

For a given scenario, the number of targets, the number of UAVs available, and their respective locations can vary. Therefore, to allow for any circumstance, these parameters are not restricted (i.e. it is not assumed that the number of targets is greater than the number of UAVs).

In this formulation, there is one communication tower with a given range that allows for transmission of data, and communication within the tower range is homogeneous and instantaneous. Therefore, there is no need for the UAV to linger within the area to transmit data (i.e. service times are zero), and data transmission is equivalent at all points in the range. While each UAV realistically has a communication range (or circle), it is relatively much smaller and therefore, considered to be a point that coincides with the UAV location here. Furthermore, the capacity of the vehicles is considered unlimited for now since the focus is on collecting information, and the capacity on UAVs is generally much larger than the space needed for this data.

Due to the added decision of whether to visit the communication tower between tasks, a sub-optimization problem presents itself at each stage. That is, which point in N^{-} the UAV needs to visit when it returns to the communication range. Because the communication range is a circle and not a single point, the point within the range (or on the circle) needs to be found that minimizes the objective function (see next section). The sub-optimization problem becomes finding the best path between two request vertices via the communication range to minimize time. A heuristic on which point to visit along the communication range was implemented as it was found that the difference between obvious choices (the direct path to the range versus taking the minimum distance route via the range) was minimal and the additional computational cost to find the optimal point is very large [43]. Therefore, the point along the communication range that is used as the delivery point after visiting a request (N_i^{-}) will always be the point that minimizes the distance between the target and the range.

While in this work we are assuming a single depot, zero service times, and all requests to be equivalent (no priorities on any requests), they do impact the problem formulation and solution. This is beyond the scope of this work and an area of study for the future.

E. Objective Function

The cost function is formulated in the context of the minimization problem such that the total time that it takes for all the targets to be visited and then delivered is minimized (minimum delivery latency). The general form of the individual target cost holds for both the static and dynamic case and is essentially the sum of the time since arrival (where A_i is the arrival time of request *i* assumed to be 0 in this work) until the target is picked up and the time from pickup until delivery (where P_i and D_i are a result of the decision variables):

$$J_T = \sum_{i=1}^{N}(P_i - A_i) + (D_i - P_i) = \sum_{i=1}^{N}(D_i - A_i) \tag{10}$$

With multiple UAVs, the total service time cost is the sum of the cost of each individual UAV, J_k:

$$J_T = \sum_{k=1}^{M} J_k \tag{11}$$

Here, the cost for each UAV, J_k, becomes of a function of the targets assigned to that UAV. The targets assigned to each UAV k is a set ordered by visit denoted by V_k where $V_k \subset V$ and with $n_k = |V_k|$. The total cost for each UAV is a sum of the cost for each target (g_i) in V_k divided by the UAV's velocity:

$$J_k = \sum_{i=1}^{n_k} \frac{g_i}{v^k} \tag{12}$$

The total target cost is a function of the costs to go from one state to the next. The decision variables x and z are embedded in these costs and are denoted by the subsequent equations where C_I is the cost to go from a UAV's initial position to a request, C_F is the cost to go from a request to a delivery location, C_D is the cost to go from one request to the next directly with delivery, and C_T is the cost to go from one request to the next via the range. While it is possible to assign a vehicle to end at a particular vertex (k^{-}), it is assumed in this formulation that the vehicle is "done" during its last visit to the range. Incorporating this additional cost to go from the range to the end point is trivial and excluded here.

$$C_I = d_{ij} \;\; where\ i \in M^+ \;\; j \in N^+ \tag{13}$$

$$C_F = d_{ij} \;\; where\ i \in N^+ \;\; j \in N^- \tag{14}$$

$$C_D = d_{ij} \;\; where\ i \in N^+ \;\; j \in N^+ \tag{15}$$

$$C_T = d_{ij} + d_{jk} \;\; where\ i, k \in N^+ \;\; j \in N^- \tag{16}$$

Because the total time for a target to be delivered to the communication range depends on when it is picked up and then delivered, the cost for each target is the sum of the time until the UAV gets to the target and the time to get from the target to the communication range. The time until pickup (when the UAV services the target) is based on the decisions made since the target appeared until it is picked up. Because of the added decision at each state of whether to visit the communication range or not (u_i), the final time could vary. That is, if $u_i = 1$, the final time cost would be the cost to go from the request directly to the tower, and if $u_i = 0$, the final time would accumulate until the next time the decision was to visit the range ($u_i = 1$). Therefore, the cost of each target has the following form (Eqn. (17)):

$$g_i = \left\{ \begin{matrix} u_i \cdot \left[C_I + C_F + \sum_{l=1}^{i-1} [u_l \cdot C_T + (1 - u_l) \cdot C_D] \right] \\ +(1 - u_i) \cdot \min_{l=i+1:n_k} g_l \end{matrix} \right\} \tag{17}$$

As seen in Eqn. (17), not only does the cost of each target rely on past decisions, but it also relies on future decisions. If the decision is to go to the communication range at that state, the cost is equal to the total time past until the UAV has reached that state plus the cost to reach the tower. Otherwise, the cost is equal to the next state at which the decision is to go to the communication range ($u_k = 1$). Due to this dependency on future decisions, this methodology has to be solved backwards (from state N to 1) if solved optimally. Also for a given solution candidate, the coefficients need to be computed recursively which means the objective function cannot be solved efficiently.

III. Heuristic Methodology

The heuristic algorithm is developed here with the motivation to minimize the time that each target has to wait before delivery. It was shown that even efficient optimal methods are too computationally complex to scale to realistic problem sizes or to allow for quick solving in dynamic environments thus motivating a solution based on heuristics [42]. While the dynamic version of the problem is of interest, only the underlying static problem and solutions are studied in this work. Again, this is due to the fact that dynamic solutions are typically found by either re-solving the static problem repeatedly or using heuristics to adapt the static solution for the changing environment.

These heuristics developed here are based on how targets are grouped between visits to the communication range (clustered) and how these clusters are assigned to UAVs (routed), and are similar to other 2-phase heuristic solutions to Vehicle Routing Problems (VRPs) where the solution is decomposed into two natural phases: 1) clustering of target vertices into feasible routes and 2) actual construction of routes.

Clustering is a common technique used in formulating solutions to VRPs. This is because the problem solutions clearly show the division of requests into groups: the objective of any clustering problem. As a result, cluster-first, route-second algorithms are common place in VRP literature [32]. The main challenge is clustering is to group similar objects while distinguishing between objects that are clearly different. It is obvious that the choice of the parameter/s that measure whether objects are similar or different impacts the resulting clusters significantly. In typical VRPs, Euclidean distance is a sufficient measure, and this lends itself to several, common clustering techniques. Additionally, the number of clusters plays a huge role in the results. Many techniques require the user to input the number of clusters up front while few are able to build clusters without this.

Some common spatial clustering techniques include partitioning, hierarchical, and locality-based methods. Partitional clustering divides objects in *k* clusters by minimizing how different objects are from their clusters. Hierarchical algorithms build clusters based on a measure of similarity (typically Euclidean distance) in a tree structure until some cut-off measure is reached [51]. This can also be done in a top-down approach where all clusters are grouped initially and clusters are divided until a cut-off is reached. Locality-based algorithms group objects based on local relationships like density and can handle arbitrary cluster shapes but suffer from the curse of dimensionality [52] – [54].

This problem differs from conventional VRP solutions in that the number of clusters is not necessarily equal to the number of vehicles. Therefore, clusters tend to be much smaller and not intuitive. It was shown in previous work that the size of the clusters can vary depending on the specific scenario and that using Euclidean distance only was an insufficient measure of similarity [41]. That is, the size of clusters depends on the distribution and number of targets and vehicles. Therefore, a clustering technique that doesn't rely on the user inputting the number of clusters upfront and can use arbitrary parameters to measure similarity is needed. A unique approach to clustering is developed here based on HAC and fuzzy logic.

A. Fuzzy HAC Clustering

The key to an effective solution using a cluster-first, route-second heuristic for this problem is correctly defining targets that belong to similar clusters. Here, a HAC algorithm is adopted to build clusters of targets, because the solution is deterministic and do not require the user to define the number of clusters at the start. Once the similarity measure between each set of requests is determined, a similarity cutoff measure is calculated at which to stop clustering requests.

Because the similarity of requests and the similarity cutoff in this case can depend on a variety of things: number of vehicles and requests, density of requests, proximity to the communication range, etc, a Fuzzy Inference System (FIS) is used to create both a measure of similarity between requests and to determine the cutoff. Fuzzy logic [55], based on multi-valued logic, provides a unique method for encoding knowledge about continuous variables by manipulating inputs to outputs with if-then rules by using heuristic knowledge and human experience. The two parameters used to find the similarity between two requests (or two clusters of requests) is the ratio of their distances from the range and the angle between the two (Figure 3). These parameters were chosen due to the fact that they will apply regardless of the environment's size. While these parameters were chosen based on expertise of the problem, there are a variety of techniques developed for accomplishing feature selection autonomously [56] – [58].

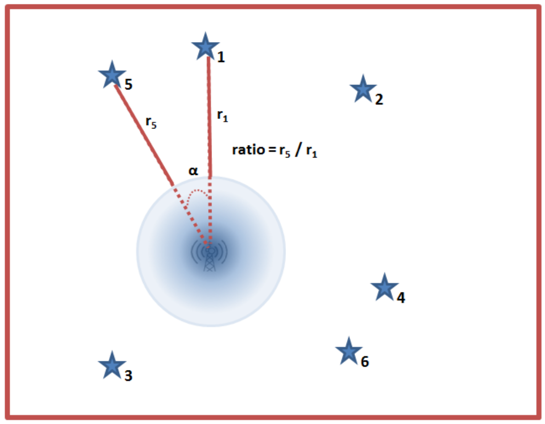

Figure 3: HAC Similarity Parameters

The biggest influence on the clustering besides these parameters was found to be the ratio of UAVs to targets. Much larger errors in the performance of the heuristic when compared to the optimal solutions were found to be when the ratio of UAVs to targets was either very high or very low [41]. However, it was determined that the ratio of UAVs to targets only affects the similarity cutoff (which is specific to the scenario), but does not actually affect the similarity between two requests. Therefore, the clustering is broken into two, cascading FISs: one to determine the similarity between all requests and another to determine the similarity cutoff measure. The inputs and outputs into each FIS are shown in Figure 4.

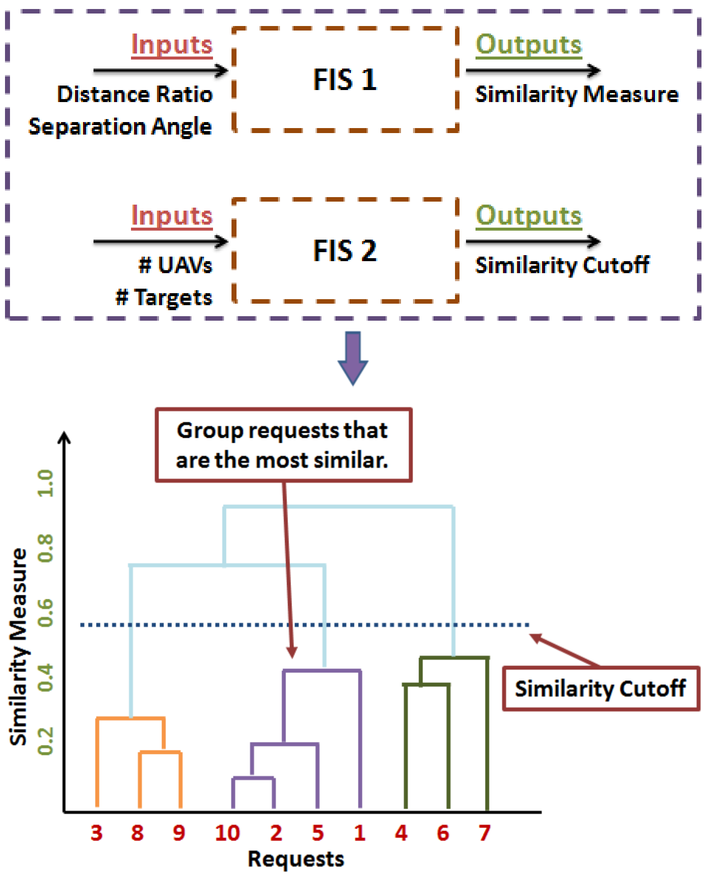

Figure 4: Heuristic Clustering Algorithm

The first FIS used here inputs information about the ratio of the distances of the two requests from the range and the angle separating the two requests to get a crisp output for similarity. The ratio of distances and angle separation are both described by either very small, small, medium, high, or very high. The output measure of similarity is spread equally between 0 and 1 and is described as very small, small, medium, high, or very high. The second FIS uses the ratio of the number of vehicles to the number of requests as an input to determine the similarity cutoff. The ratio of UAVs to requests is broken into 15 Membership Functions (MFs) named 1 to 15 where 1 is the smallest ratio and 15 is the highest. This range is broken into so many MFs to adequately distinguish between similarity cutoffs for different scenarios, and the MFs are concentrated over the ratio values. Comparable to the UAV to request ratio input, the similarity cutoff measure, is spread equally between 0 and 1 and is described by 15 MFs. The MFs for the inputs and outputs are shown below (Figure 5 - Figure 9). These MFs were all subsequently tuned using a Genetic Algorithm (GA) to gain better performance [59]. However since it is often the case that the results from tuning MFs using a GA are nonsensical, the MFs used to initiate the GA are shown here (with the final MFs being slight departures from these).

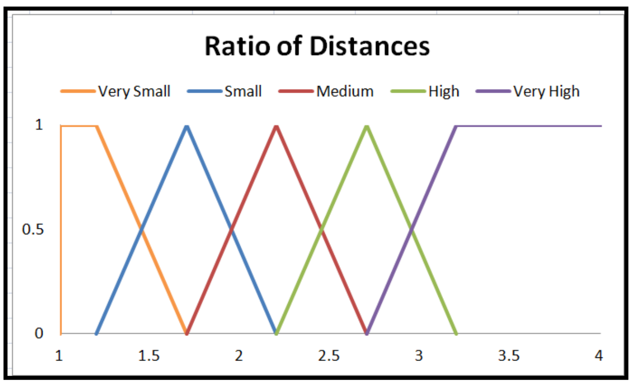

Figure 5: FIS 1, Input 1: Ratio of Distances to the Communication Range of the Two Requests

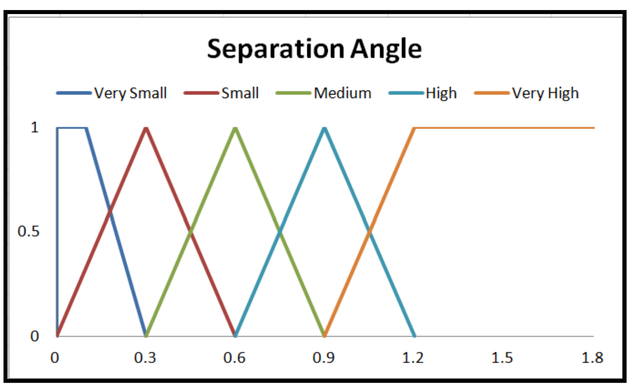

Figure 6: FIS 1, Input 2: Angle of Separation between two Requests

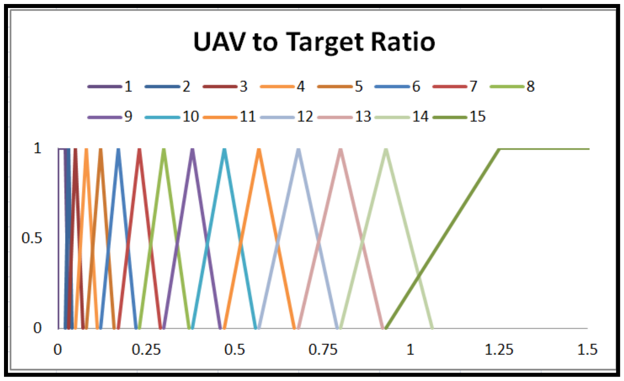

Figure 7: FIS 2, Input 1: Ratio of Number of UAVs to Number of Requests

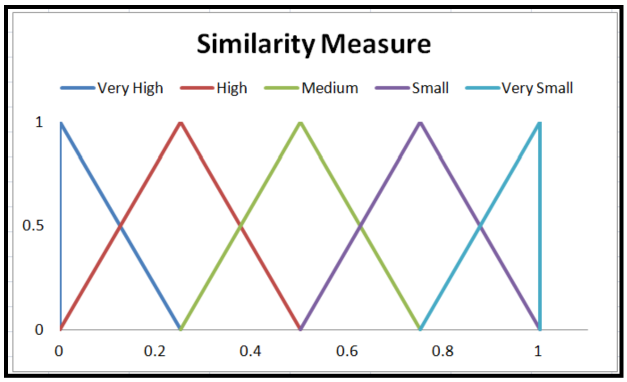

Figure 8: FIS 1, Output: Similarity Measure

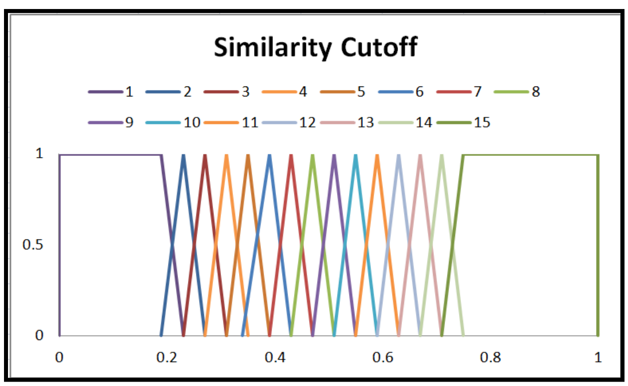

Figure 9: FIS 2, Output: Similarity Cutoff

Rules relating the inputs and outputs for the FIS are set up in the form of if-then statements and are based on heuristics and human experience. The rules for the fuzzy inference system can be summed up in some simple decision making logic. There are a total of 25 rules for FIS 1 and 15 rules for FIS 2 this setup. The rules for FIS 1 (Table 1) follow the logic that if two requests are farther apart (either in distance ratio or angle of separation), then they are less similar and vice versa. The rules for FIS 2 (Table 2) follow the logic that if there is a small ratio of UAVs to targets the similarity cutoff should be larger (clusters will be bigger). Additionally, the opposite is true: if the ratio is large, the cutoff will be smaller. That is, if there are a lot of UAVs compared to the number of targets, the clusters tend to be much smaller. The rules implemented with the FISs can be summarized below.

Table 1: FIS 1: Rules for Determining the Output Similarity Measure

		Ratio of Distances from Communication Range				
		Very Small	**Small**	**Medium**	**High**	**Very High**
Angle of Separation	**Very Small**	Very High	Very High	High	Medium	Low
	Small	Very High	High	Medium	Low	Low
	Medium	High	Medium	Medium	Low	Very Low
	High	Medium	Low	Low	Very Low	Very Low
	Very High	Low	Low	Very Low	Very Low	Very Low

Table 2: FIS 2: Rules for Determining the Similarity Cutoff Measure

Ratio of UAVs to Targets															
Input:	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Output:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

B. Vehicle Routing

In the static problem, the assignment of clusters to targets is done as a finite horizon scenario. Instead of assigning vehicles to clusters simultaneously, each assignment is done individually and consecutively until all targets have been assigned.

At each "stage," each UAV gives a bid on the cluster that they would best service next. This bid is done by calculating the information within that cluster, c, divided by the distance to the cluster from the UAV for each cluster. Then, each UAV's bid is given by the maximum value over all the clusters (Eqn. 18). This represents which cluster each UAV would visit "next." A stage is any point at which a UAV is available for use. That is, these bids are done at the initial stage and then each time a UAV returns to the communication range. Once each UAV gives their bid, they are assigned to their cluster they bid for. To resolve cases where 2 UAVs are assigned to the same cluster, the cluster is given to the first UAV that would arrive there. The bidding is done at the each stage to allow for circumstances in which a UAV could drop-off one cluster and still pickup the next before a different UAV could get it. Each UAV is subsequently assigned to clusters as it arrives back to the communication range. This is done until all clusters are visited.

$$B_k = \max_{c=1\ldots N_c}\left(\frac{I_c}{dist(X_k, X_c)}\right) \quad for\ all\ k = 1 \ldots M \tag{18}$$

Once the targets are assigned to each UAV based on the previous algorithm, each UAV must decide in what order to visit the targets within the cluster. Because clusters are defined as those targets with which a UAV would visit all together and then visit the communication range at the end, it takes the decision variable u_k out of the problem. More accurately, we know that the decision variable u_k will be 0 for all targets in the cluster except the last. Therefore, to determine the shortest path for a UAV within a cluster a quick, approximate TSP solver was used [60]. A GA was used for these purposes due to its ease of implementation and reliability for the small problem sizes studied here. However, other good heuristics have been developed to solve TSPs quickly and efficiently. One of the most commonly used solvers was developed by Lin and Kernighan in 1973, called the Lin-Kernighan algorithm [46], and is suggested for use as a more reliable solver for larger problem sizes.

IV. Results & Analysis

In this section, the simulations performed and their results are described. At this time, only one tower range is considered, and it is located centrally (though this does not affect the results). Furthermore, the number of UAVs and targets are inputted by the user, but the locations are given at random. All simulations described and presented here were implemented in MATLAB programming language and on a Windows machine with an Intel Core i5 2.4 GHz processor and 4.0 GB of RAM.

Sample solutions for various cases are shown below in Figure 10. The different color lines represent the tours of the different vehicles *(UAV 1 = red, UAV 2 = blue, etc)*. As can be expected, requests close together are grouped in clusters and picked up in a single tour. Furthermore it can be seen that often the vehicle takes more than 1 tour.

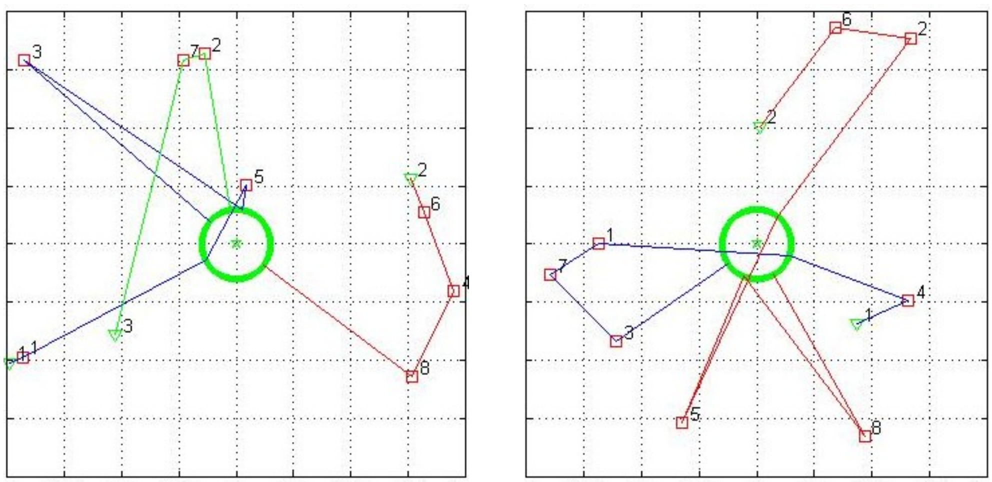

Figure 10: Heuristic Solution for Various Cases, M = 3, N = 8 (left) and M = 2, N = 8 (right)

A. Nearness to Optimality

A value traditionally used as a metric of performance for heuristics to combinatorial optimization problems is the nearness to the optimal policy based on the objective function. Here the average case performances are given through a Monte Carlo analysis of 3000 runs for each case. In the figure below (Figure 11) the average percent greater time waited for the heuristic algorithm over optimal for 2 to 8 targets and 1 to 5 UAVs is shown. As seen, the algorithm gives near-optimal results. Also, the results show the performance of algorithm when the ratio of UAVs to targets is both large and small. It can be seen that the cost for the heuristic solution has the same trend as the optimal solution. The average cost for a scenario generally reduces as UAVs are added and increases with the number of targets. Also, the cost of the heuristic solutions is on the same scale as and within about 95% of the optimal solution cost for the cases tested.

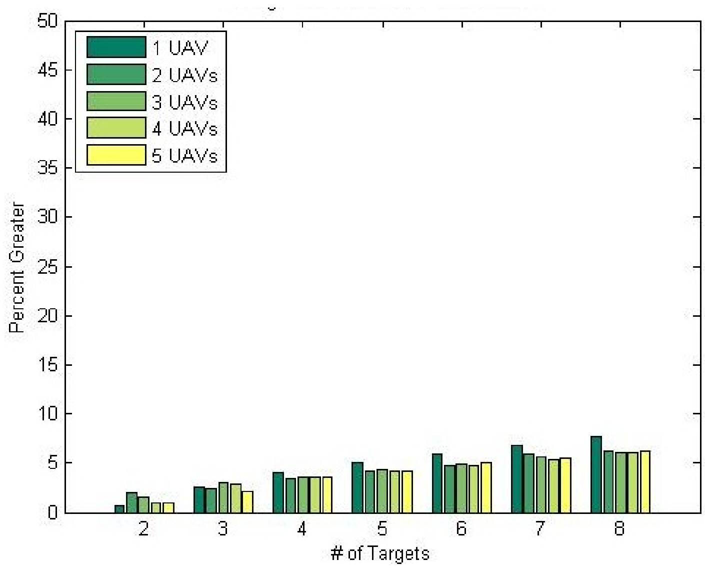

Figure 11: Average Percent Error on Cost Analysis for Small Problem Sizes

Because of the complexity of this problem, it makes it very difficult to provide optimal solutions for medium and large problem sizes. In turn, it is tough to analyze how the performance of the heuristic is scaling as problem sizes increase. To get a rough idea, the average cost per target was plotted (Figure 12). In addition to consistent performance for small problems, it can be seen in Figure 12 that the algorithm tends to be performing pretty consistently for medium and large problems. That is, the cost per target is relatively consistent as the number of targets increases. While it is increasing, the cost appears to be increasing either linearly or at a decreasing rate. This implies that as the problem gets more complicated, the algorithm is maintaining its performance. These results are expected, because the membership functions in the clustering algorithm were tuned to scenarios ranging from 1 to 10 vehicles and 1 to 100 requests. Overall in general, the heuristic-based solution provides excellent results in significantly reduced execution time from the optimal solution.

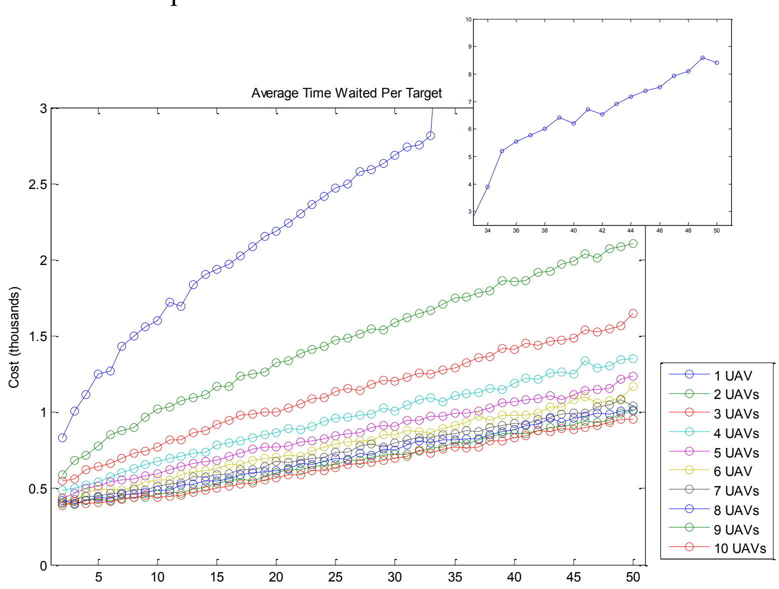

Figure 12: Heuristic Average Cost Per Target

B. Scalability

An important part of the analysis to allocation problems is to examine how the solution strategy to the static problem scales. Therefore, we are concerned with the order of the algorithm (and how long it takes to run) and how the algorithm is performing (nearness to optimality) as problem sizes get larger (i.e. more targets and vehicles).

Problem Order

The Vehicle Routing Problem is a combinatorial optimization problem which is known to be computationally complex [61] for any size problem that is not trivial. This is one of the most studied combinatorial optimization problems since its introduction in 1959 due to its numerous practical applications and known difficulty. For example, the most effective exact algorithm know to date can solve problems with up to 50 customers reliably. Furthermore, implementations of effective algorithms into the distribution of goods in transportation departments have shown considerable improvement (from 5-20%) in costs [61].

The running time to compute the optimal policy of VRPs is dependent on the size of the problem and is an important consideration in the selection of an algorithm. As the solution space increases, even with the best algorithms the time to find the optimal solution increases accordingly. Furthermore, many dynamic solutions are found by re-optimizing static cases every time the scenario is updated. This would be unrealistic to do in a dynamic environment if the running time is long. Therefore, algorithms are classified based on their efficiency (order). A solution is considered efficient if a polynomial-time solution is found.

Optimal – Brute Force

The methodology used in this research to calculate the optimal solution is done by direct enumeration of all possibilities and a check to see which one minimizes the objective function [Eqn. (10)]. Therefore, the optimal strategy needs to run through all possible solutions to find the best. A brute force strategy is used due to its directness and its guarantee to find the best combination.

For a single UAV, the number of possibilities increases with the number of targets. With the added communication constraint, the number of possibilities becomes $N! \cdot 2^N$ with order being $\mathcal{O}(N! \cdot 2^N \cdot N \cdot M)$ when taking into consideration the cost function complexity. For multiple UAVs, each UAV can be assigned from 0 to N targets. For each number of targets assigned, the specific target assignment can vary. For a random scenario, the number of targets assigned, which targets, and the order they're visited in can vary based on the location of the UAVs and the targets. The number of ways *N* targets can be assigned to *M* UAVs is given by $\#_1 = \sum_{k=1}^{N+M-1} \frac{[N+M-(k+1)]^{[M-2]}}{M-2}$ where $[v]^{[w]} = v(v-1)\ldots(v-w+1)$. Then for a given number of targets for each UAV, the number of possibilities by which to assign specific targets is given by $\#_2 = \prod_{j=1}^{M} \binom{m_j}{n_j}$ where $m_j = N - \sum_i^j n_{i-1}$ and $N = \sum_{i=1}^{M} n_j$. With $\#_1$, $\#_2$, and the number of possibilities for the single UAV problem, the order of the multiple UAV problem becomes $\mathcal{O}(\#_1 \cdot \#_2 \cdot 2^N \cdot N \cdot M)$. When taking the limit of this as N and M become very large, the $\#_1$ and $\#_2$ terms dominate, and the problem becomes intractable. Using the brute force method and searching through all possibilities, the execution time to find the optimal solution will increase exponentially as the environment gets more complex. Therefore, scalability is the main motivation for finding a solution that is near optimal but tractable.

Heuristic

The sizes of the optimal solutions are the main motivation for a solution based on heuristics. Because this problem has such a large solution space, the order of a heuristic solution is also very important. As stated previously, any algorithm that is polynomial in time is considered efficient. Previous work [43] has shown that the order of the clustering technique is at most $\mathcal{O}(N^2)$, and that the order of the routing is $\mathcal{O}(M)$. That is, the routing is linear with the number of vehicles. Therefore, the order of the heuristic algorithm presented here is at most $\mathcal{O}(M \cdot N^2)$, which is polynomial in time and significantly reduced from the original order of the problem. This is validated further in the next section on Execution Time.

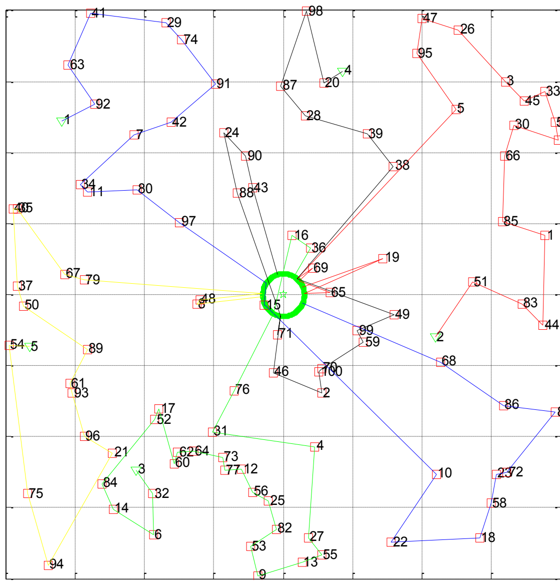

Figure 13: Large Scale Validation of Heuristic Algorithm – N = 100 and M = 5

Because the order of the problem is so small, it allows for much larger cases to be solved in a "reasonable" length of time. An example of a relatively large scale validation of the algorithm is shown below in Figure 13 where *N = 100* and *M = 5* that was obtained within only 11 seconds of execution time (in MATLAB). Larger problems may be of interest for UAVs, and the short execution time for this problem confirms its scalability.

Execution Times

Because the clustering technique is at most order $\mathcal{O}(N^2)$, the execution time for the heuristic algorithm (Figure 14) increases accordingly. To gain further perspective of the growth of the execution time, both a linear and quadratic function was fitted to the graph. It was found that the R^2 value for each was 0.9906 and .9974, respectively. Since an R^2 value closer to 1 represents a better fit, the growth is slightly more quadratic than linear. However, a linear fit is almost as good as the quadratic implying that the growth of the execution time is much less than originally anticipated. Additionally, this is obviously much less than the order of the brute force problem and is the cause of the execution time being significantly reduced. Furthermore, it can be seen that the increase in execution time has little relationship to the number of UAVs.

Most importantly, the execution times are on the scale of milliseconds rather than seconds or minutes (or even years and lifetimes for large problems) like the brute force solution. With an appropriate sampling time for dynamic scenarios, the solutions to dynamic problems can easily be recalculated as the environment changes. Therefore, the execution times drive the sampling time for these scenarios, and since the execution times for moderately sized problems are all less than 1 second, a sampling time as low as 1 second can be used for the dynamic problem (which is appropriately small for problems of this nature).

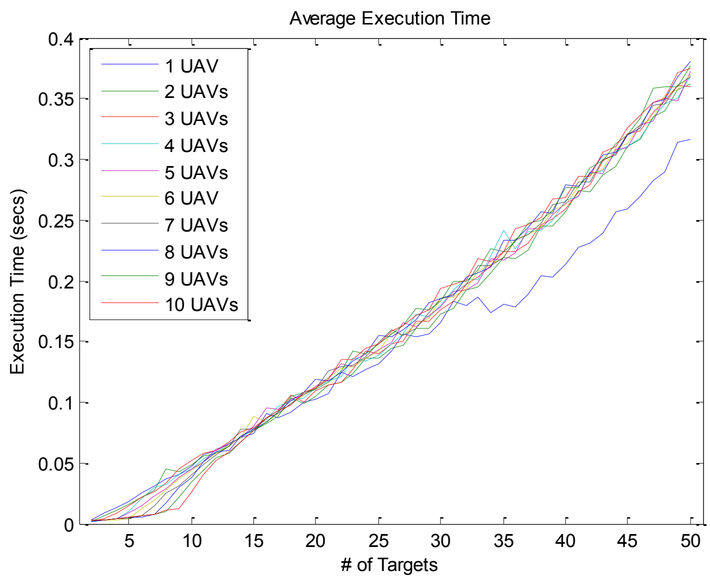

Figure 14: Execution Time for a Heuristic Solution

V. Conclusions & Future Work

In reality, UAV communication is limited; whereas task allocation and routing problems with UAVs ignore this restriction. Typically the focus of mission planning is on minimizing mission costs while communication restrictions are often ignored. However, communication range can be restricted by line-of-sight, power, or a need for uninterrupted transmission. In this work, a solution strategy based on heuristics was proposed for a variation of the VRP that minimized the total time that all the targets have to wait to be picked up and delivered to a communication range. This problem was proposed as an alternative and effective routing solution when communication is very restricted or it is desirable to wait until there is a reliable and safe connection.

It is well known that optimal solutions for realistic scenarios of VRPs can be computationally complex, and trying to maintain the objective of minimizing the latency in a dynamic environment makes this even more difficult. Here, we have preserved this objective by using a heuristic solution that approximates the behavior of the optimal solution. It has been shown based on the static scenarios studied in this work that a heuristic solution on average provides results that are within about 95% of the optimal case in significantly reduced execution time. For more practical scenarios where there is uncertainty, a robust algorithm is required, which can be obtained by a good heuristic solution. Therefore, a new clustering algorithm based on Hierarchical Agglomerative Clustering and Fuzzy Logic was implemented.

An addition benefit to this approach is that the algorithm developed for the static case can be extended to work in a dynamic environment. It was shown in previous work [62] that a solution to the static case for this problem will also provide a solution for the dynamic case. This implies that with a fast solver, the optimal solution to dynamic problem can be found by re-optimizing whenever the environment changes (i.e. new targets appear). Therefore, the solution to the static problem found previously can be applied to the dynamic problem by re-solving as the environment updates. This holds as long as the solution can be obtained before the environment re-updates which will be the case using an appropriate sampling time. Validation and analysis of the performance in dynamic scenarios is currently under investigation.

Because this new approach to routing UAVs is motivated by a very strong assumption about communications, this work begs the question of how good this assumption is and when it applies. Future work will examine under which communication conditions this routing strategy is the most appropriate. As discussed earlier, a simple model was assumed where communication is restricted to a very small range. While there will be circumstances in which communication is so limited that this approach is desirable, future work will attempt to find the exact boundaries for distance, bandwidth, UAV velocity, and data size where this transition between routing approaches lay.

Acknowledgments

Authors acknowledge that this work was partially funded by the Air Force Summer Faculty Fellowship Program (SFFP) and the Dayton Area Graduate Studies Institute (DAGSI).

References

1. R. A. Best, "Intelligence, Surveillance, and Reconnaissance (ISR) Programs: Issues for Congress," CRS Report for Congress, CRS Web, RL32508, February 22, 2005, Library of Congress.
2. Shima, T., and Rasmussen, S., Editors, "UAV Cooperative Decision and Control – Challenges and Practical Approaches," Advances in Design and Control, SIAM, 2009.
3. A. Goldsmith, "Wireless Communications," Cambridge University Press, New York, NY, 2005.
4. B. Taub and D. L. Schilling, "Principles of Communication Systems," New York: McGraw-Hill, 1986.
5. E. Frew and T. Brown, "Airborne Communication Networks for Small Unmanned Aircraft Systems", Proceedings of the IEEE, Vol. 96, No. 12, Dec., 2008.
6. S. Lu, H. Ting-lei, "A Summary of Key Technologies of Ad hoc Networks with UAV Node," IEEE, pp. 944-949, 2010.
7. T. X. Brown, B. Argrow, C. Dixon, and S. Doshi, "Ad Hoc UAV Ground Network (AUGNet)," AIAA 3rd Unmanned Unlimited Technical Conference, Workshop and Exhibit, Chicago, Illinois, Sept., 2004.
8. E. P. Freitas, T. Heimfarth, I. F. Netto, C. E. Lino, C. E. Pereira, A. M. Ferreira, F. R. Wagner, and T. Larsson, "UAV Relay Network to Support WSN Connectivity," International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, IEEE, 2010.
9. E. Modiano, "16:36 Communication Systems and Network," Lecture Notes, Massachusetts Institute of Technology, Cambridge, MA, 2012.
10. H. C. Christmann and E. N. Johnson, "Design and Implementation of a Self-Configuring Ad-Hoc Network for Unmanned Aerial Systems," AIAA Infotech@Aerospace, May, 2007.
11. P. Zhan, K. Yu, and A L. Swindlehurst, "Wireless Relay Communications with Unmanned Aerial Vehicles: Performance and Optimization," IEEE Transactions on Aerospace and Electronic Systems, Vol. 47, No. 3, July, 2011.
12. M. F. J., D. Hampal, and S. DiPierro, "Unmanned Aerial Vehicle (UAV) Communications Relay," IEEE, 1996.
13. Dixon, D. Henkel, E. W. Frew, and T. X. Brown, "Phase Transitions for Controlled Mobility in Wireless Ad hoc Networks," AIAA Guidance, Navigation, and Control Conference and Exhibit, Colorado, Aug., 2006.
14. T. Carfang, E. Frew, and T. Brown, "Improved Delay-Tolerant Communication by Considering Radio Propagation in Planning Data Ferry Navigation," AIAA Guidance, Navigation, and Control Conference, Toronto, Canada, Aug. 2-5, 2010.
15. Henkel and T. Brown, "On Controlled Node Mobility in Delay-Tolerant Networks of Unmanned Aerial Vehicles," International Symposium on Advance Radio Technologies (ISART), Boulder, CO, 2006.
16. Frew and T. Brown, "Networking Issues for Small Unmanned Aircraft Systems", Journal of Intelligent and Robotic Systems, Vol. 54, No. 1-3, 2008.

17. Kuiper, "Mobility and Routing in a Delay-tolerant Network of Unmanned Aerial Vehicles", Linkoping Studies in Science and Technology, Department of Computer Science, Thesis No. 1356, Apr., 2008.
18. W. Zhao and M. Ammar. "Message ferrying: Proactive routing in highly-partitioned wireless ad hoc networks," *9th IEEE International Workshop on Future Trends of Distributed Computing Systems*, May, 2003.
19. W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," Proceedings of ACM Mobihoc, 2004.
20. W. Zhao, M. Ammar, and E. Zegura, "Controlling the Mobility of Multiple Data Transport Ferries in a Delay-Tolerant Network," IEEE INFOCOM, 2005.
21. P. M. Olsson, J. Kvarnstrom, P. Doherty, O. Burdakov, and K. Holmberg, "Generating UAV Communication Networks for Monitoring and Surveillance," 11th International Conference on Control, Automation, Robotics, and Vision, Singapore, Dec, 2010.
22. P. M. Olsson, J. Kvarnstrom, P. Doherty, O. Burdakov, and K. Holmberg, "Positioning Unmanned Aerial Vehicles as Communication Relays for Surveillance Tasks," Linkoping Studies in Science and Technology, Department of Computer Science.
23. P. M. Olsson, "Positioning Algorithms for Surveillance Using Unmanned Aerial Vehicles," Linkoping Studies in Science and Technology, Department of Computer Science, Thesis No. 1476, Apr., 2011.
24. M. M. Zavlanos, M.B. Egerstedt, and G. J. Pappas, "Graph Theoretic Connectivity Control of Mobile Robot Networks," *IEEE Journal*, Vol. 99, No. 9, pp. 1525-1540, 2011.
25. N. Michael, M. M. Zavlanos, V. Kumar, and G. J. Pappas, "Maintaining Connectivity in Mobile Robot Networks," *Springer Tracts on Advanced Robotics*, Vol. 54, pp. 117-126, 2009.
26. A. Ghaffarkhah and Y. Mostofi, "Communications-Aware Navigation Functions for Cooperative Target Tracking," *American Control Conference (ASS)*, June, 2009.
27. T. Razafindralambo and D. Simplot-Ryl, "Connectivity Preservation and Coverage Schemes for Wireless Sensor Networks," *IEEE Transactions on Automatic Control*, Vol. 56, No. 10, pp. 2418-2428, 2011.
28. S. S. Ponda, L. B. Johnson, H. L. Choi, and J. P. How, "Ensuring Network Connectivity for Decentralized Planning in Dynamic Environments," *Proceedings of the AIAA Infotech@Aerospace Conference*, St. Louis, MO, March, 2011.
29. S. S. Ponda, L. B. Johnson, A. N. Kopeikin, H. L. Choi, and J. P. How, "Distributed Planning Strategies to Ensure Network Connectivity for Dynamic Heterogeneous Teams," *IEEE Journal on Selected Areas in Communications*, 2012.
30. Y. Yan and Y. Mostofi, "Robotic Router Formation in Realistic Communication Environments – a Bit Error Rate Approach," *IEEE Military Communications Conference*, 2010.
31. E. Craparo and J. P. How, "Throughput Optimization in Mobile Backbone Networks," *IEEE Transactions on Mobile Computing*, March, 2010.
32. Laporte, "The Vehicle Routing Problem: An Overview of Exact and Approximate Algorithms," 1991.
33. Sen., K. Bulbul, "A Survey on Multi Trip Vehicle Routing Problem," International Logistics and Supply Chain Congress, Istanbul, Turkey, Nov., 2008.
34. Kara, B.Y. Kara, M. K. Yetis, "Cumulative Vehicle Routing Problems," First International Conference on Combinatorial Optimization and Applications, COCOCA, China, 2007.
35. Gillett and J. Johnson, "Multi-Terminal Vehicle-Dispatching Algorithm," *Omega*, Vol. 4, pp. 711–718, 1976.
36. R.M. Garey, D.S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness," *Bell Laboratories*, Murray Hill, N.J., 1979.
37. M.W.P Savelsbergh, M. Sol, "The General Pickup and Delivery Problem," *Transportation Science*, Vol. 29, pp. 17-29, 1995.
38. W. Nanry, J. W. Barnes, "Solving the Pickup and Delivery Problem with Time Windows using Reactive Tabu Search," *Transportation Research Part B*, Vol. 34, pp. 107-121, Jul, 2000.
39. S. Ropke, D. Pisinger, "An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows," Transportation Science, Vol. 40, No. 4, pp. 455-472, Nov, 2006.
40. H. Li, A. Lim, "A Metaheuristic for the Pickup and Delivery Problem with Time Windows," *IEEE 13th International Conference on Tools of Artificial Intelligence,* pp. 160-167, Nov, 2001.

41. C. Sabo and K. Cohen, "SMART Heuristic for Pickup and Delivery Problem with Cooperative UAVs," Air Force Research Laboratory, AIAA Infotech@Aerospace, St. Louis, Missouri, 29-31 March 2011, AIAA-2011-1464.
42. C. Sabo, M. Kumar, K. Cohen, and D. Kingston, "VRP with Minimum Delivery Latency using Linear Programming," 2012 AIAA Infotech@Aerospace, Garden Grove, California, 19-21 June 2012, AIAA-2012-2561.
43. C. Sabo, D. Kingston, and K. Cohen, "Minimum Service Time for UAV Cooperative Control Subject to Communication Constraints," Air Force Research Laboratory, 2010 AIAA Infotech@Aerospace, Atlanta, Georgia, 20-22 April 2010, AIAA-2010-3344.
44. E. Kivelevitch, K. Cohen, and M. Kumar, "The Scalability of the Market-Based Solution to the Multiple Traveling Salesmen Problem," 2012 AIAA Infotech@Aerospace, Garden Grove, California, 19-21 June 2012.
45. D. Estrin et al., "Embedded, Everywhere: A Research Agenda for Networked Systems of Embedded Computers," *Computer Science and Telecommunications Board*, National Academy Press, 2009.
46. S. Lin and B. W. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Operations Research*, Vol. 21, p.p. 498–516, 1973.
47. S. Arora, "Nearly linear time approximation scheme for Euclidean TSP and other geometric problems," *Proc. 38th IEEE Annual Symposium on Foundations of Computer Science*, p.p. 554–563, FL, October 1997.
48. N. Christofides, "Bounds for the travelling-salesman problem," *Operations Research*, Vol. 20, p.p. 1044–1056, 1972.
49. N. Ernest and K. Cohen, "Fuzzy Clustering based Genetic Algorithm for the Multi-Depot Polygon Visiting Dubins Multiple Travelling Salesman Problem," 2012 AIAA Infotech@Aerospace, Garden Grove, California, 19-21 June 2012.
50. G. Ghiani, F. Guerriero, G. Laporte, and R. Musmanno, "Real-Time Vehicle Routing: Solution Concepts, Algorithms, and Parallel Computing Strategies," *European Journal of Operational Research*, Elsevier, Vol. 151, 2003, pp.1-11.
51. K.C. Gowda and G. Krishna, "Agglomerative Clustering using the concept of Mutual Nearest Neighborhood," *Pattern Recognition*, Vol. 10, Issue 2, p.p. 105-112, 1978.
52. J. Han, M. Kamber, and A. Tung, "Spatial Clustering Methods in Data Mining: A Survey," *Geographic Data Mining and Knowledge Discovery,* Taylor and Francis, 2001.
53. E. Kolatch, "Clustering Algorithms for Spatial Databases: A Survey," University of Maryland, Department of Computer Science, 2001.
54. E. Chandra and V. P. Anuradha, "A Survey on Clustering Algorithms for Data in Spatial Database Management Systems," *International Journal of Computer Applications,* Vol. 24, No. 9, June 2011.
55. Zadeh, L. A., "Fuzzy Sets", *Information and Control*, Vol. 8, 1965, pp. 338-353.
56. M. Dash and H. Liu, "Feature Selection for Classification," *Intelligent Data Analysis*, Vol. 1, No. 1-4, pp. 131-156, 1997.
57. H. Liu and L Yu, "Toward Integrating Feature Selection Algorithms for Classification and Clustering," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 4, pp. 491-502, 2005.
58. M. Hall, "Correlation-Based Feature Selection for Machine Learning," Dissertation, University of Waikato, 1999.
59. M. D. Vose, "The Simple Genetic Algorithm: Foundations and Theory," Massachusetts Institute of Technology Press, 1999.
60. Kirk, Traveling-Salesman Problem – Genetic Algorithm, 2007.
61. P. Toth, D. Vigo, "The Vehicle Routing Problem," Monographs of Discrete Mathematics and Applications, Siam, Philadelphia, PA, 2002.
62. C. Sabo and K. Cohen, "Dynamic Allocation of Unmanned Aerial Vehicles with Communication Constraints," 2012 AIAA Infotech@Aerospace, Garden Grove, California, 19-21 June 2012, AIAA-2012-2455.