A Hierarchical Approach to Control a Swarm of Unmanned Aerial Vehicles

Thierry Pamphile  
Department of Mechanical and Materials Engineering  
Florida International University  
Miami, FL 33174  
tpamp001@fiu.edu

Kuo-Chi Lin  
Department of Mechanical, Materials and Aerospace Engineering  
University of Central Florida  
Orlando, FL 32816  
klin@pegasus.cc.ucf.edu

ABSTRACT
In this paper, the problem of controlling a mission of multiple UAVs is studied by considering a swarming hierarchy that relies on a switching mechanism and is based on a combination of the local interactions that the agents have with each other. An analytical model of a point mass UAV in a 2D environment is used to represent the family of mobile agents and to properly combine the low-level behaviors (basic, individual, and group) necessary to yield a family of feasible solutions. The results achieved for such a switching mechanism are verified by computer simulations and suggest that the hierarchical control approach can be used and eventually applied to actual UAVs.

Keywords: Swarming, Switching Mechanism, Hierarchical Control, UAV

1. INTRODUCTION
1.1. Background
The concept of unmanned aerial vehicle was first used in the American Civil War, when the North and the South tried to launch balloons with explosive devices that would fall into the other side’s ammunition depot and explode. The term UAV or unmanned aerial vehicle is used to describe a powered, aerial vehicle that does not carry a human operator, and can fly autonomously or be piloted remotely. In some cases, UAVs can carry a lethal or non-lethal payload depending on the mission it is assigned [1]. This concept was also used by the Japanese for around a month in World War II, when they tried to launch balloons with incendiary and other explosives. During World War II, the United States did use a prototype UAV called Operation Aphrodite. It was an attempt to use manned vehicles in an unmanned mode. However, at that time, the United States did not have the technology to launch or control the aircraft.

1.2. Current Research
Extensive research is being conducted in the UAV area largely due to the desire of reducing or even eliminating the risk that the pilot of a manned vehicle faces when in combat. This is a task that needs to be done while increasing the effectiveness of the UAVs and reducing their cost efficiency. One approach to this solution is to allow the UAVs to function as a swarm which can reduce the complexity of motion control by reducing code size and communication requirements [3]. A swarm in the sense referred to in this report is best described by Clough as a collection of autonomous agents, using local sensing and reactive behaviors, interacting such that a collective, global behavior emerges from their interactions [4]. While not completely understood by scientists, social insects such as ants and bees are capable of carrying out relatively complex tasks without any explicit communication or cooperation [6]. The metaphor can be extended to other systems with similar architecture. A crowd is a swarm of people; traffic can be considered a swarm of cars. The large amount of relatively recent research and information already published on swarm intelligence indicates that this area is one of great interest right now. The interest in these fields will only increase in coming years as the military races toward fewer human soldiers and more autonomous robots by the year 2015 [9].

One downside to this approach is the way the UAVs react during the mission. Since they only depend on their local sensing, their reaction is somewhat unpredictable [2]. Many strategies have been proposed in order to achieve a swarming behavior. One of these strategies is the use of genetic algorithms (GA). GA is a search technique which is based on the biological process of evolution and can optimize by iteratively considering several solutions and selecting the best. Another strategy is using digital pheromones similar to those of insects. For insects, digital pheromones can be defined as chemical scent markers deposited in the environment, evaporated over time and propagated over the neighboring area, while applied to a swarm of UAVs, it requires a network of Unattended Ground Sensors (UGS) to broadcast signals deposited by other UAVs [5].

Our approach towards the swarming of UAVs is to consider what is called behavior-based robotics [6]. The goal is to design a series of behaviors that govern the UAVs response to their local environment.

2. TECHNICAL APPROACH
2.1. Behavior Control
We propose that a swarm of autonomous UAVs (for our particular case MAVs) can accomplish certain missions by properly sequencing low-level behaviors such that the swarm exhibits collective high-level behaviors. The swarm will be a reactive, decentralized system capable of achieving goals that would otherwise be very difficult or impossible for a single, more intelligent and therefore more expensive UAV to accomplish.
This can be done without delegating certain tasks to specific agents. Rather, all agents (unless explicitly denoted, such as a leader UAV) will have identical behavior sets.

Let’s consider a point-mass UAV in a 2D environment. For our model, a few assumptions have to be taken into account. First, we will only use bang-bang control, where the only governing attributes are thrust ($T$) and bank angle ($\sigma$). Both are limited to three values: \( \{T_{\text{min}}, T_0, T_{\text{max}}\} = \{0.5T_0, T_0, 1.5T_0\} \) where $T_0 = 0.25\text{N}$ and $\sigma = [-\pi/12, 0, \pi/12]$. Acceleration due to gravity $g$ is $9.81\text{m/s}^2$, initial speed $v$ is set to $5\text{m/s}$, air drag $D$ is $0.01\text{kg m/s}^4\text{v}^2$, and mass $m$ is $0.201\text{kg}$.

![Figure 1. A Point-mass UAV in a 2D Environment (Speed $v$, Heading Angle $\psi$, Mass $m$).](image)

We can separate the high level behaviors in three major levels: the basic behavior level, the individual level behavior and the group level behavior.

### 2.2. Low-Level Maneuvers

Consider an object centered at $\left(x_{\text{obj}}, y_{\text{obj}}\right)$. Let $v$ represent the velocity of the UAV and $r$ the position of the center of the object with respect to the position of the UAV $\left(x, y\right)$. Approaching or avoiding the obstacle is determined by calculating the cross product and dot product of the UAV’s velocity and its relative location to the center of the object and adjusting the UAV’s bank angle and thrust accordingly.

#### 2.2.1. Approach

To approach the object, the UAV must turn in its direction and speed up to catch up if it is in front of it. Thus, if the object is to the right, bank right; else bank left. If the object is in front of the UAV, increase thrust; else reduce thrust:

$$
\begin{align*}
\begin{vmatrix}
T \\
\sigma
\end{vmatrix} =
\begin{cases}
\frac{T_{\text{max}} \cdot v \cdot r > 0}{T_{\text{min}} \cdot v \cdot r \leq 0} \\
\frac{\pi/12, v \cdot r | > 0}{-\pi/12, v \cdot r | \leq 0}
\end{cases}
\end{align*}
$$

#### 2.2.2. Avoid

To avoid the object, the UAV must turn away from it and slow down if it is in front of it. Thus, if the object is to the right, bank left; else bank right. If the object is in front of the UAV, reduce thrust; else increase thrust:

$$
\begin{align*}
\begin{vmatrix}
T \\
\sigma
\end{vmatrix} =
\begin{cases}
\frac{T_{\text{min}} \cdot v \cdot r > 0}{T_{\text{max}} \cdot v \cdot r \leq 0} \\
\frac{-\pi/12, v \cdot r | > 0}{\pi/12, v \cdot r | \leq 0}
\end{cases}
\end{align*}
$$

### 2.3. Basic Behaviors

The following basic behaviors are modeled as part of the low-level behaviors of a UAV in a swarm: Obstacle Avoidance, Area Spreading, Target Tracking, Cluster Forming, Leader Following, Boundary Avoidance, Collision Avoidance and Path Following.

#### 2.3.1. Obstacle Avoidance

If the UAV’s distance relative to the obstacle is less than avoidance distance, it avoids the obstacle by using equation (3).

#### 2.3.2. Area Spreading

If a UAV’s distance relative to the centroid is smaller than dispersion distance, it avoids the centroid by using equation (3).

#### 2.3.3. Target Tracking

If a UAV’s distance relative to the target is greater than the allowed distance, it approaches the target by using equation (2).

#### 2.3.4. Cluster Forming

A function is implemented that calculates the centroid of all UAVs. If a UAV’s distance relative to the centroid is greater than aggregation distance, it approaches the centroid. First, find angle $\alpha$ between the $x$-axis and the displacement from the UAV to the centroid $r_c = (x_c - x)i + (y_c - y)j$:

$$
\alpha = \tan^{-1}\left(\frac{y_c - y}{x_c - x}\right).
$$

Let $\beta$ be the mean of the orientation of all UAVs within aggregation distance including this UAV:

$$
\beta = \frac{\sum \psi_i}{n}.
$$
Knowing that $\gamma$ represents the normalized sum of the difference between $\alpha$ and UAV orientation $\psi$ and the difference between $\beta$ and $\psi$:

$$\gamma = \frac{(\alpha - \psi) + (\beta - \psi)}{2\pi},$$

(6)

Then, the required controls are given by:

$$\begin{bmatrix}
T \\
\sigma
\end{bmatrix} =
\begin{cases}
T_{\text{max}}, \mathbf{r} \cdot \mathbf{v} > 0 \\
T_{\text{min}}, \mathbf{r} \cdot \mathbf{v} \leq 0 \\
-\pi/12, \gamma > 0 \\
\pi/12, \gamma \leq 0
\end{cases},$$

(7)

### 2.3.5. Leader Following

Assuming that all the UAVs know the position of the leader, if their distance relative to the leader is greater than following distance, they approach it by using equation (2).

### 2.3.6. Boundary Avoidance

If the UAV is less than or equal to boundary avoidance distance $d_{\text{boundary}}$ away from any simulation boundary, then $\sigma = \pi/12$ and $T = T_{\text{max}}$ [7].

$$\begin{bmatrix}
T \\
\sigma
\end{bmatrix} =
\begin{cases}
T_{\text{max}}, t \mod t_{\text{boundaryInterval}} = 0 \\
T_0, t \mod t_{\text{boundaryInterval}} \neq 0 \\
-\pi/12, t \mod t_{\text{boundaryInterval}} = 0 \\
0, t \mod t_{\text{boundaryInterval}} \neq 0
\end{cases},$$

(5)

### 2.3.7. Collision Avoidance

A function is implemented that calculates the position of the closest agent. If the agents’ positions relative to each other are smaller than collision avoidance distance, they avoid each other by using equation (3).

### 2.3.8. Path Following

If a UAV’s distance relative to the path is greater than the allowed distance, it approaches the path by using equation (2).

### 2.4. Individual-Level Behaviors

The individual-level behaviors are the basic building blocks of this approach. They are obtained by combining the basic behaviors. Once we have enough of them, designing a mission only comes down to choosing the appropriate behavior [1]. Similar to the basic behaviors, the individual-level behaviors are condition-based or behavior-based behaviors, meaning that they come into effect depending on the environment. The individual-level behaviors rely on the following elements: rules and variables, actions and controls, constraints, parameters, priority weights.

#### Rules and Variables

The rules and variables depend on the environment and their local sensing. The variables can be positions and velocities of other UAVs, obstacles, target, etc [1]. Furthermore, the rules are a variety of if….else statements.

#### Actions and controls

The actions and controls result directly from the rules and variables. Types of actions include approach target, avoid obstacle, speed up, slow down, etc [1].

#### Constraints

Constraints for this approach are due to the capability of the system [1]. For example, if a UAV doesn’t detect the presence of all the UAVs, during a cluster forming or dispersion action, its reaction will be different from when it detects all of the UAVs [8].

#### Parameters

The parameters can be defined as the constants [1]. They are very important because they are the leading cause of a behavior emergence. For example, collision avoidance distance, obstacle avoidance distance are the parameters or constants predefined that enable these behaviors when the conditions are not satisfied.

#### Priority

Priorities help the UAVs decide which behavior is more relevant or more important [1]. For instance, if the group behavior is to track a target, than target track will have the greatest priority. The individual-level behaviors inherit the basic behaviors that form it.

### 2.4.1. Swarming-Clustering-Wondering

This individual-level behavior is a combination of collision avoidance, cluster forming, obstacle avoidance and boundary avoidance as illustrated in Figure 4 (a):

- Obstacle avoidance has the highest priority
- Collision avoidance is called if obstacle avoidance is not executed
- Cluster forming is called if neither obstacle avoidance nor collision avoidance are executed
- Boundary avoidance is called if none of the previous basic behaviors are executed.

### 2.4.2. Swarming-Clustering-Dispersion

This individual-level behavior is a combination of area spreading, cluster forming and collision avoidance and is shown in Figure 4 (b):

- Collision avoidance has the highest priority
- Cluster forming is called if collision avoidance is not executed
- Area spreading is called if neither collision avoidance nor cluster forming are executed.

### 2.4.3. Swarming-Clustering-Homing

This individual-level behavior is a combination of collision avoidance, cluster forming and target track as displayed in Figure 4 (c):

- Target track has the highest priority
Collision avoidance is called if target track is not executed
Cluster forming is called if neither target track nor collision avoidance are executed

2.4.4. Swarming-Clustering-Following
This individual-level behavior is a combination of collision avoidance, cluster forming and leader following.
- Leader following has the highest priority
- Collision avoidance is called if leader following is not executed
- Cluster forming is called if neither leader following nor collision avoidance are executed

This individual-level behavior is a combination of collision avoidance, cluster forming and path following.
- Path following has the highest priority
- Collision avoidance is called if path following is not executed
- Cluster forming is called if neither path following nor collision avoidance are executed

In addition to these behaviors, obstacle avoidance behavior was added to the behavior Swarming-Clustering-Homing.

2.4.6. Swarming-Clustering-Homing
This individual-level behavior is a combination of collision avoidance, obstacle avoidance, cluster forming and target track as shown in Figure 4 (d):
- Obstacle avoidance has the highest priority
- Target track is called if obstacle avoidance is not executed
- Collision avoidance is the next step in the hierarchy
- Cluster forming is executed if none of the other behaviors are executed

2.4.7. Swarming-Clustering-Following_2
This individual-level behavior is a combination of leader following, collision avoidance, obstacle avoidance and cluster forming:
- Obstacle avoidance has the highest priority
- Leader following is called if obstacle avoidance is not executed
- Collision avoidance is the next step in the hierarchy
- Cluster forming is called if neither leader following nor collision avoidance are executed

2.4.8. Swarming-Clustering-Trekking_2
This individual-level behavior is a combination of collision avoidance, cluster forming and path following.
- Obstacle avoidance has the highest priority
- Path following is called if obstacle avoidance is not executed
- Collision avoidance is the next step in the hierarchy
- Cluster forming is called if neither path following nor collision avoidance are executed

2.5. Group-Level Behaviors
Group-level behaviors or swarms emerge from a collection of individual-level behaviors. In other words, in order to achieve the swarming behaviors, a number of homogeneous UAVs with identical individual-level behaviors are used. This low-level collaboration yields a higher-level reaction.

Figure 2. Original Hierarchy

3. SIMULATIONS
3.1. Analysis
The ideas and behaviors were modeled using MATLAB. MATLAB, short for "matrix laboratory", is a data-manipulation software package that allows data to be analyzed and visualized using existing functions and user-designed programs. Matlab uses two modeling strategies. The first is to write the code as with other compiling languages like visual c++, visual.net, etc. the second is to use Simulink. Simulink is a block library tool for modeling, simulating and analyzing dynamic systems.

For this project, the codes were written out entirely. The first step was to design generic calculation functions such as centroid which calculates the center of mass, draw and arrow which draws the UAVs and find closest which calculates the position of the closest UAV. A function is a routine or a set sequence of steps that is part of a larger computer program. After defining these functions, the basic behaviors were also modeled as functions, this way when combining the basic behaviors to form the individual-level behaviors, it was only necessary to call the functions. Besides the basic behaviors, an existence algorithm was added to the code. The primary use of this existence algorithm is to check if the UAVs exist within the parameters allocated and if not, to remove them from the simulation. This allows identifying any collision with other UAVs or obstacles.

As mentioned above, the UAVs are modeled as arrows; the obstacles are modeled as circles. Doing this reduces hassles associated with implementing obstacles and obstacle avoidance behaviors. For example, it is much easier to program obstacle avoidance with a circle because each radial line is of identical length. No matter where a UAV approaches the obstacle, the distance from the UAV to the obstacle can be found using the distance between center points of the UAV and obstacle [8].

To simulate the group level behavior, a for loop was used. In most imperative computer programming languages, a for loop is a control structure which allows code to be executed iteratively. For
loops, are typically used when the number of iterations is known before entering the loop. A total time of 100 seconds was used for each behavior simulated.

3.2. Results

The following behaviors were successfully simulated: Swarming-Clustering-Homing, Swarming-Clustering-Wondering, Swarming-Clustering-Dispersion, Swarming-Clustering-Trekking, Swarming-Clustering-Following and Swarming-Clustering-Homing 2. A second and third Swarming-Clustering-Wondering was also simulated. In the first one, the UAVs were placed in a lined up position. This is a very basic simulation where collision avoidance only comes into effect after the UAVs pass the obstacle where they would resume to a lined up position similar to the first one but not exactly the same. In order to verify if it would work in any environment, the UAVs were scattered all around in the second one. In this we could observe the effect of collision avoidance early on. This idea was also used for the other simulations. Although the response of the UAVs to their environment is somewhat unpredictable, they follow the steps allocated and eventually get the job done. For example, for a cluster forming behavior, if a UAV is far in front of the group, it doesn’t usually turn back around himself or make a sort of U turn to head back to the pack. Some UAVs may turn around their own mass and head back to the group while others may turn around the whole group and end up at the end of the pack trailing the others.

Another important observation is the priority of the behaviors. For the behavior Swarming-Clustering-Homing 2, for the initial run, reaching the target was set as the highest priority. This resulted in the UAVs completely ignoring the obstacle and going straight through it. The solution to this problem: changing the priority of the basic behaviors. By giving obstacle avoidance behavior the highest priority, the UAVs functioned as desired, avoiding the obstacle and reaching the target. A similar phenomenon was observed in the leader following behavior. In the behavior, the UAVs avoided the obstacle while following the leader but instead of resuming the mission after avoiding the obstacle, they kept repeating the same behavior, going in a loop around the obstacle. This problem emphasizes the importance of setting the priorities correctly. Although the objective of the swarm was to reach the target, it wasn’t its highest priority. This also puts in question the accuracy of the other swarming behaviors. Although they worked this time, that doesn’t mean that they will always work in every environment, especially with a more complex task at hand.

Besides the priorities, other considerations have to be revised. For instance, a third run of Swarming-Clustering-Homing was executed. In this run, the UAVs speed was increased by 0.5 m/s from 5m/s to 5.5m/s. This small change in the speed produced results completely different from the previous ones. In the first ones, the UAVs avoided the obstacle as soon as they got within the obstacle avoidance range. For the third one, the leading UAVs turn around and sort of head back to the group. This in turns affect the performance because the UAVs have to go into collision avoidance mode, which for some lead to moving completely away from the obstacle but then heading back due to the aggregation behavior. Thus, at the end, the swarming behavior is still executed but with a greater time span.

Figure 3. Final Hierarchy

![Figure 3. Final Hierarchy](image)

![Figure 4. (a) Swarming-Clustering-Wondering](image)

![Figure 4. (b) Swarming-Clustering-Dispersion](image)

![Figure 4. (c) Swarming-Clustering-Homing](image)
4. CONCLUSIONS
The behavior-based architecture considered is simple and relatively easy to implement. Although all of the desired behaviors were not simulated, the ones that were simulated produced satisfactory results with the constants and parameters originally defined. Nevertheless, as stated above, this may not be the case for every situation. The simplest parameter change or new feature addition can change the simulation drastically. As a result, this approach as to be further researched in order to yield more accurate statistics that reflect general and realistic responses. Some of the future prospects include monitoring the priority settings to insure that the sequence is flawless, designing more individual behaviors, implementing boundary and thrust conditions. Also, velocity tracking of other UAVs as opposed to only one UAV during collision avoidance may prove to enhance the behaviors modeled.

5. DISCLAIMER
This material is based upon work supported by the National Science Foundation under Grant No. 0353918 and the University of Central Florida through the Research Experience for Undergraduates in Intelligent and Autonomous Robotic Systems, May 23 through July 29, 2005. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

6. REFERENCES


