Multi-level Control Architecture and Energy Efficient Docking for Cooperative Unmanned Air Vehicles

Stephen Alexander Young

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Mechanical Engineering

Tomonari Furukawa, Chair
Craig Woolsey
Andrew Kurdila

February 4, 2011
Danville, Virginia

Keywords: Solar Power, UAV, Unmanned, Energy Efficient, Path Planning, High-level Control

Copyright 2011, Stephen Alexander Young
Multi-level Control Architecture and Energy Efficient Docking for Cooperative Unmanned Air Vehicles

Stephen Alexander Young

(ABSTRACT)

In recent years, significant progress has been made in improving the performance of unmanned air vehicles in terms of aerodynamic performance, endurance, autonomy, and the capability of on-board sensor packages. UAVs are now a vital part of both military actions and scientific research efforts. One of the newest classes of UAV is the high altitude long endurance or HALE UAV. This thesis considers the high-level control problem for a unique HALE mission involving cooperative solar powered UAVs. Specifically addressed is energy efficient path planning for vehicles that physically link together in flight to form a larger, more energy efficient HALE vehicle.

Energy efficient docking is developed for the case of multiple vehicles at high altitude with negligible wind. The analysis considers a vehicle governed by a kinematic motion model with bounded turn rate in planar constant altitude flight.Docking is demonstrated using a platform-in-the-loop simulator which was developed to allow virtual networked vehicles to perform decentralized path planning and estimation of all vehicle states. Vehicle behavior is governed by a status which is commanded by a master computer and communication between vehicles is intermittent depending on each vehicle’s assessment of situational awareness. Docking results in a larger vehicle that consumes energy at 21% of the rate of an individual vehicle and increases vehicle range by a factor of three without considering solar recharging.
Contents

1 Introduction 1

2 Background and Review of Literature 6

3 Fundamentals of Multi-Level Estimation/Control 10

  3.1 Formal Problem Statement 10

  3.2 Vehicle Motion Model 11

  3.3 Stochastic Sensor Model 15

  3.4 Gaussian Nonlinear Estimation of Vehicle States 16

    3.4.1 Gaussian and Linear Approximation for Vehicle Motion 18

    3.4.2 Gaussian Approximation for Observation 18

    3.4.3 Centralized vs. Decentralized Estimation 20

  3.5 Summary 21
4 Energy Efficient Path Planning

4.1 Multi-Level Control and Estimation ........................................ 24

4.2 Docking Priority ................................................................. 26

4.3 Energy Based Docking .......................................................... 28

4.3.1 Optimal Path Planning ...................................................... 28

4.3.2 Single Vehicle Optimal Path Planning ............................... 29

4.3.3 Docking Formation ............................................................ 31

4.4 Information Based Decision Making ...................................... 35

4.5 Summary .............................................................................. 37

5 Platform-In-The-Loop Simulator Implementation ......................... 39

5.1 Docking Priority .................................................................. 40

5.2 PILS Implementation ............................................................ 44

5.3 FlightGear Visualization ....................................................... 46

5.4 Summary .............................................................................. 48

6 Platform-In-The-Loop Simulation Results ................................... 51

6.1 Energy Based Docking: Master Simulations ............................. 51
List of Figures

1.1 C-HALE Mission Overview ........................................... 2

3.1 Diagram of a C-HALE fleet ........................................... 11

3.2 Fixed Global Frame and Body Frame Notations ................. 12

3.3 Relative Vehicle Sensor with Field of View ................. 16

3.4 Diagram of the Kalman Filter Recursive Estimation Process . 17

4.1 Multi-Level Control and Estimation Diagram ................. 25

4.2 Possible Dubins Shortest Paths .................................. 30

4.3 Example of Energy-Based Docking Formation Case I ........ 32

4.4 Linked Vehicle Turn Radius ...................................... 33

4.5 A: Sample Power Consumption Curves for a Linked C-HALE Vehicle B: Minimum Power Velocity for Linked C-HALE Vehicle. 34
6.6 PILS-FlightGear Demonstration Information Entropy, Correction Threshold=5

6.7 PILS-FlightGear Demonstration Information Entropy, Correction Threshold=10

6.8 PILS-FlightGear Demonstration Energy State

6.9 PILS-FlightGear Demonstration Energy Improvement
List of Tables

5.1 Fixed Simulation and Vehicle Parameters
5.2 Variable Simulation and Vehicle Parameters
6.1 Fixed Simulation and Vehicle Parameters
6.2 Energy Consumption With Arbitrary vs Automatic Docking Order
Chapter 1

Introduction

The importance of unmanned air vehicles has grown significantly in recent years. UAVs are now a vital part of both military actions and scientific research efforts. Significant progress has been made in improving the performance of such vehicles in terms of aerodynamic performance, endurance, autonomy, and the capability of on-board sensor packages. One of the newest classes of UAV is the high altitude long endurance or HALE UAV. Definitions for HALE vary from organization to organization however: HALE UAVs are generally considered to be capable of the following: flight endurance of more than several days, a ceiling of more than 50,000ft, and area coverage on the order of 100,000km². Achieving these performance characteristics places many unique constraints on vehicles design. The most important of these constraints is the need for low power consumption which restricts the weight, maneuverability, and power consumption of on-board computers and communication equipment. Several current projects for HALE UAVs have focused on solar powered UAVs with high
aspect ratio wing as a means to extend the range of HALE vehicles as much as months or even years. Vehicles such as NASA HELIOS and QinetiQ / Boeing Vulture are capable of very long endurance; however, they require wingspans of several hundred feet. Such large wingspans on very lightweight vehicles presents a significant obstacle to practical and speedy deployment of such a HALE vehicle. As the endurance of solar HALE vehicles grows, so does the wingspan of these vehicles. In some cases wingspans have topped 200 feet which severely limits the number of airfields capable of supporting such vehicles. To address this, the Cooperative HALE (C-HALE) concept was developed.

The C-HALE concept calls for a fleet of small solar powered vehicles with medium aspect ratio to launch independently and physically link together in flight to form a more energy efficient, high aspect ratio vehicle. The C-HALE mission concept presents many unique
control and cooperation problems which are the focus of this thesis. Among the challenges are the need for cooperative control, energy efficient path planning for the inefficient smaller vehicles, and the need for greater autonomy in the vehicles so as to make the vehicle-to-operator ratio feasible.

The technology for solar powered UAVs has been thoroughly explored. In order to operate, such vehicles are restricted in terms of vehicle weight, maneuverability, and power consumption of on-board computers. These restrictions necessitate a high-level control and path planning strategy that considers energy consumption while utilizing limited computational resources. Several energy optimal path planning techniques have been developed in previous work, some specifically addressing solar powered vehicles. Cooperation among vehicles has been developed for decentralized action of cooperative vehicles; however, physical interaction between vehicles is usually treated as something to be avoided rather than a goal. In order to advance the C-HALE concept, the objective of the research described in this thesis has been to:

1. Formulate a multi-level control and estimation architecture

   - Estimation within fleet of vehicle states
   - Centralized and decentralized estimation with communication loss

2. Develop an energy efficient docking technique for cooperative UAVs

   - Pre-docking rendezvous and formation flight
   - Energy based sequential docking
3. Introduce a framework for cognitive behavior

- A way to quantify situational awareness is needed
- Vehicle behavior should change based on situational awareness

In this thesis multi-level control is developed using decentralized extended Kalman filter based estimation of vehicle states. Individual vehicles are governed by a set of behaviors associated with a vehicle status which is commanded by a base station or lead vehicle. An energy efficient strategy for sequentially staging vehicles for docking is developed for the case of multiple vehicles at high altitude with negligible wind. For simplicity the task of sequentially staging vehicles for docking is referred to as docking. This analysis is limited to outer loop control for staging vehicles for a docking procedure that is still being developed. Developing this procedure requires solving a difficult control problem to overcome the aerodynamic interaction of two wingtips in close proximity which is outside the scope. This analysis considers a vehicle governed by a kinematic motion model with bounded turn rate in a planar and constant altitude flight regime. This docking is demonstrated using a platform-in-the-loop simulator on which virtual networked vehicles perform decentralized path planning and estimation of all vehicle states. Vehicles are assumed to have limited computational resources. To address this, an information based decision making scheme has been developed and communication between vehicles is intermittent depending on each vehicle’s assessment of situational awareness. The outline of this thesis is as follows. Chapter 2 presents the background for solar powered vehicles and several aspects of efficient and optimal control. This is followed by a formulation of the fundamentals of decentralized esti-
information and multi-level control. Chapter 4 presents the development of energy efficient path planning for cooperative docking UAVs and information based decision making. Implementation of this work on a real-time Platform-In-the-Loop Simulator is presented in Chapter 5 followed by PILS results in Chapter 6.
Chapter 2

Background and Review of Literature

Applications for unmanned air vehicles have grown greatly in recent years. As such, there is a significant amount of interest in developing new vehicle technologies and control techniques to improve the capabilities of these vehicles. The C-HALE mission includes tasks that overlap many areas of UAV research including solar powered aircraft, cooperative control, efficient search strategy and efficient path planning.

A brief overview of the modern history and development of UAVs as well as future applications and challenges for unmanned systems is given in [24]. In addition to military applications, interest has developed for HALE vehicles as a means of exploration on other planets using solar powered vehicles [11]. The concept of solar powered flight stretches back decades with initial feasibility studies beginning in the mid-1970’s [34, 36]. More detailed discussions of the development of solar powered aircraft including HALE UAVs are found in [5, 10, 20]. General design principles for solar vehicles are discussed in [12]. More
recently, complete aircraft concepts and applications are discussed in numerous sources
[7, 25, 29, 30, 32, 39, 46, 47, 48, 60, 63, 65].

Solar powered flight has generally been achieved using one or more of the following strategies.

- Using extremely high aspect ratios to reduce drag.
- Harvesting energy using solar panels or dynamic soaring in updrafts.
- Using optimal control for efficient path planning or gaining aerodynamic advantages through formation flight.

Discussions of energy efficient flight for aircraft in general in terms of both trajectory planning and dynamic soaring in thermals are found in [2, 3, 4, 43] while less conventional means of increasing efficiency including radio power transmission, alternative propulsion sources, and using formation flight to gain aerodynamic advantages are discussed in [13, 18, 21, 66].

Optimal control is an extremely broad field and techniques have been applied to aircraft and path planning in [6, 14, 17, 45, 50, 62] among many others. In many cases optimization considers either time or fuel state optimality [16, 27, 51, 53] while later works considered multi-objective optimization [19, 61].

Solar powered aircraft have generally been constrained by the performance of energy generation technology. Consequently, special emphasis has been placed on selecting mission types and locations that are well suited to vehicle capabilities [44, 22, 35, 21]. Other works have considered techniques to optimize specific maneuvers and missions [39, 60, 7, 49].

For high-level control and path planning it is often advantageous to consider less complex
vehicle systems in order to perform optimal path planning. This has been addressed as a shortest-curve-to-intercept-problem. Geometric arguments for determining shortest paths were developed in [26] while later, optimal control methods were applied to this problem [9]. More recently [40, 55, 41] developed a means to determine a shortest path by determining the shortest of six possible Dubins paths via root finding. Similar techniques have been applied to high level control and path planning in [57, 64, 8] with both path planning and solar energy collection for a single vehicle being considered in [37].

Intelligence, surveillance, and reconnaissance (ISR) are most common tasks for a C-HALE system. ISR missions using the C-HALE concept will require addressing two major tasks. The first task is to develop a way for vehicles to rendezvous and link together. Once linked, the mission becomes a single vehicle guidance, control, and navigation problem. The task of docking is a unique task that requires not only rendezvous and formation control but also physical interaction between vehicles. Cooperation of UAVs has been addressed in many forms. Of most interest to the C-HALE mission are a number of information theoretic based methods. Initially introduced by Shannon [54], a number of approaches have been developed to coordinate the motion of multiple vehicles based on sensor coverage and information gain [50, 38, 58, 31, 11] within the general framework of probabilistic robotics which is thoroughly formulated in [59] as well as others.

This chapter has presented an overview of previous work in the areas of solar powered flight, optimal control and efficient path planning, and approaches to achieve cooperation within the framework of information theory. The technology for solar powered UAVs has been
thoroughly explored. In order to operate, such vehicles are restricted in terms of vehicle weight, maneuverability, and power consumption of on-board computers. These restrictions necessitate a high-level control and path planning strategy that considers energy consumption while utilizing limited computational resources. Several energy optimal path planning techniques have been developed, some specifically addressing solar powered vehicles. Cooperation among vehicles has been developed for decentralized action of cooperative vehicles; however, physical interaction between vehicles is usually treated as something to be avoided rather than a goal. The problem of docking UAVs is unique and has not yet been explored. The next chapter will present the fundamentals of the C-HALE mission, stochastic vehicle motion and sensor models, and decentralized estimation and control using the extended Kalman filter.
Chapter 3

Fundamentals of Multi-Level Estimation/Control

3.1 Formal Problem Statement

The focus of these formulations is on the docking process for a fleet of C-HALE UAVs. These vehicles will begin the mission at a desired altitude with a random position and energy state. Each vehicle will be assigned a docking priority in order to determine the order for docking and will be denoted by the subscript $i$. Each vehicle or sensor platform, $s_i$, with the state vector $x_{s_i}^{s_i}$, must sequentially intercept and dock with the linked vehicles $s''$. The goal of high level control is to find the control inputs, $u_{1:k}^{s_i:s''}$, that consume the least amount of energy $E^{s'}$ of the fleet $s'$. Energy should be evenly distributed within the fleet as it is assumed that
energy cannot be shared between vehicles. A diagram of the fleet and docking process is given in figure 3.1 and the formal control objective and control constraint is described by

\[
S_i \in S' = \{S_i, S', S''\}
\]

\[
S' = \{S_{i+n}, S_{i+1}, \ldots, S_{n}\}
\]

\[
S'' = \{S_i\}
\]

Figure 3.1: Diagram of a C-HALE fleet.

\[
J\left(u'_{k:k+n-1}\right) = \left\| \sigma_{k+n-1} \left(E'\right) - \left(E'_0 - E'_k\right) \right\| \rightarrow \text{min}
\]

\[
\Phi^{\ast} = \|x''_{k+n} - x''_{k}\| = r_{n, s''}
\]

3.2 Vehicle Motion Model

UAVs move in six degrees of freedom described by the Newtonian equations for the forces along and moments about the principal axes. Using the notation from [45], this motion is given by 3.2. Additionally, close attention must be paid to the relationship between the body
axis and the fixed axis which is described in figure 3.2. For a more thorough description of the translation between the fixed and body frame as well as full derivations of the equations of motion please consult [45].

\[ F_x - mgS_\theta = m (\ddot{x} + q \dot{z} - r \dot{y}) \]
\[ F_y + mgC_\theta S_\Phi = m (\ddot{y} + r \dot{x} - p \dot{z}) \]
\[ F_z + mgC_\theta S_\Phi = m (\ddot{z} + p \dot{y} - q \dot{x}) \]
\[ M_\phi = I_x \ddot{\phi} - I_{xz} \dot{r} + qr (I_z - I_y) - I_{xz} pq \]
\[ M_\theta = I_y \ddot{\theta} + rq (I_x - I_z) + I_{xz} (p^2 - r^2) \]
\[ M_\psi = -I_{xz} \ddot{\psi} + I_z \dot{r} + pq (I_y - I_z) + I_{xz} qr \] (3.2)

From here there are several assumptions that can be made to simplify the problem to a more manageable set of equations of motion. First, it is known that the docking process for
C-HALE UAVs is intended to occur when the fleet of vehicles are within a specified docking area at a fixed altitude. The altitude at which this will occur is well above 50,000 ft. At this altitude, the wind velocity is much less than the vehicle speed. Therefore, disturbances due to wind are considered to be a low level control problem and are not considered for high level control. It can also be assumed that an autopilot system produces a steady flight path while following commanded velocity and maneuvering rates. In other words only a trajectory planning problem using a kinematic motion model is considered. Absolute velocity in the fixed frame in terms of the Euler angles and body frame velocity components are given by

\[
\begin{bmatrix}
\dot{x}_f \\
\dot{y}_f \\
\dot{z}_f
\end{bmatrix} = \begin{bmatrix}
C_\theta C_\psi & S_\theta S_\psi C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\theta S_\psi \\
C_\theta S_\psi & S_\theta S_\psi S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\theta C_\psi \\
-S_\theta & S_\theta C_\theta & C_\phi C_\theta
\end{bmatrix} \begin{bmatrix}
\dot{x}_b \\
\dot{y}_b \\
\dot{z}_b
\end{bmatrix}
\]

(3.3)

Since docking is also a constant altitude process, assume \( \dot{z}_f = \dot{z}_b = 0 \), and by selecting that the wind axis is coincident with the \( x_bz_b \) and side slip is zero, the motion model for the C-HALE vehicles becomes the three degree of freedom kinematic model:

\[
\begin{bmatrix}
\dot{x}_f \\
\dot{y}_f \\
\dot{z}_f
\end{bmatrix} = \begin{bmatrix}
C_\theta C_\psi & S_\theta S_\psi C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\theta S_\psi \\
C_\theta S_\psi & S_\theta S_\psi S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\theta C_\psi \\
-S_\theta & S_\theta C_\theta & C_\phi C_\theta
\end{bmatrix} \begin{bmatrix}
\dot{x}_b \\
\dot{y}_b \\
\dot{z}_b
\end{bmatrix} \Rightarrow \begin{bmatrix}
\dot{x}_f \\
\dot{y}_f \\
\dot{z}_f
\end{bmatrix} = \begin{bmatrix}
\cos \psi \\
\sin \psi
\end{bmatrix} \begin{bmatrix}
\dot{x}_b \\
\dot{y}_b
\end{bmatrix}
\]

(3.4)
\[ \dot{s}_i = s_i v \cos s_i \psi \]
\[ \dot{y}_i = s_i v \sin s_i \psi \]
\[ \dot{\psi}_i = u_2 \] (3.5)

where \( S_i \) denotes sensor platform (or vehicle) \( i \), \( s_i x = [x_k \ y_k \ \psi_k]^T \) is the state vector which includes Cartesian position and heading and \( s_i u = [s_i V \ s_i \dot{\psi}]^T \) is the control input vector including velocity and maneuvering rate.

Vehicle motion is constrained by physical limitations on maneuvering rate based on the maximum allowable wingtip velocity differential, which is a function of the number of linked vehicles, \( n \), given by:

\[ s'' \dot{\psi} \leq s'' \dot{\psi}_{\text{max}}(n) \] (3.6)

and limitations on velocity, which are independent of the number of linked vehicles, given by

\[ s_i V_{\text{min}} \leq s_i V \leq s_i V_{\text{max}} \] (3.7)

Representing this as a discrete time process that contains zero mean multivariate Gaussian noises, \( w_{s_i} \) yields:

\[ x_{s_i}^k = f^{s_i} \left( x_{s_i}^{k-1}, u_{s_i}^{k-1}, w_{s_i}^{k-1} \right), \quad w_{s_i}^k \sim N \left( \bar{w}_{s_i}^{k-1}, \sum_{w_{s_i}^{k-1}} \right) \] (3.8)
\[
x^s_i = \begin{bmatrix} x_k \\ y_k \\ \psi_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \psi_{k-1} \end{bmatrix} + \begin{bmatrix} \cos(\psi_{k-1}) & 0 \\ \sin(\psi_{k-1}) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_{a_{k-1}} \\ \psi_{k-1} \end{bmatrix} \Delta t + w^s_i \tag{3.9}
\]

### 3.3 Stochastic Sensor Model

Each vehicle is equipped with sensors that can independently measure the global position of the vehicle and the relative position between vehicles. Both types of sensor are assumed to provide an observation with zero mean multivariate Gaussian noises. The general form for vehicle \(i\) sensing vehicle \(i\) (i.e. Global self sensor) is given by

\[
s^i_i z^s_i = s^i_i h^s_i \left(x^s_i, s^i_i v^s_i \right)
\tag{3.10}
\]

Likewise, the relative position sensor provides an observation of the location of a neighboring vehicle \(j\) also with zero mean multivariate Gaussian noises. Additionally, the relative position sensor has a finite field of view or observable region, \(s^X\).

\[
s^i_j z^s_j = \begin{cases} 
s^i_j h^s_j \left(x^s_i, x^s_j, s^i_j v^s_j \right) & x^s_j \in s^X \\ 0 & x^s_j \notin s^X \end{cases}
\tag{3.11}
\]

The noises of the self and relative position sensors are given respectively by

\[
s^i v^s_i \sim N \left(s^i v^s_i, \sum_{\substack{s^i v^s_i}}} \right), \quad s^i v^s_j \sim N \left(s^i v^s_j, \sum_{\substack{s^i v^s_j}}} \right)
\tag{3.12}
\]
In order to estimate the states of vehicles within the fleet, which are governed by a discrete time stochastic model, a recursive estimator must be formulated. Since the vehicle’s properties and dynamics will be well known we can choose a simple and computationally efficient estimator. One of the most popular such estimators is the Kalman filter, the process for which described in [3.4]. A more thorough discussion of this formulation can be found in [59].

In order to utilize the Kalman filter there are certain properties that must be assumed about the process and the states being estimated.

1. As with all Bayesian filters, the Kalman filter requires that the Markov assumption holds which is to say that future state is independent of past if the current state $x_k^i$ is known.
2. State transition must be a linear process with Gaussian noise. Although the state transition described by equation 3.9 is obviously nonlinear, this will be addressed by introducing the extended Kalman filter later.

3. Observations of the states must be linear with Gaussian noise.

4. Prior belief $\bar{x}_{0|0}$ must be normally distributed.
3.4.1 Gaussian and Linear Approximation for Vehicle Motion

Previously, a nonlinear non-Gaussian motion model was formulated and is given by:

\[ x_{k}^{s_{i}} = f^{s_{i}}(x_{k-1}^{s_{i}}, u_{k-1}^{s_{i}}, w_{k-1}^{s_{i}}) \] (3.13)

If the effect of un-modeled dynamics is assumed small, then the state transition of \( f^{s_{i}} \) is independent of state and noise. Thus the motion model conforms to the Kalman filter formulation:

\[ x_{k}^{s_{i}} = f^{s_{i}}(x_{k-1}^{s_{i}}, u_{k-1}^{s_{i}}) + w_{k-1}^{s_{i}} \] (3.14)

As mentioned previously, the Kalman filter requires linear state transition. To address this, the extended Kalman filter linearizes the state transition function, \( f^{s_{i}} \), about the previous state, \( x_{k-1}^{s_{i}} \).

\[ A_{k-1} = \frac{\partial f^{s_{i}}(x_{k-1}^{s_{i}}, u_{k-1}^{s_{i}})}{\partial x_{k-1}^{s_{i}}} \bigg|_{x_{k-1}^{s_{i}}=x_{k-1}^{s_{i}|k-1}} \] (3.15)

\[ x_{k}^{s_{i}} \cong A_{k-1} x_{k-1} + B_{k-1} u_{k-1} + w_{k-1} \] (3.16)

3.4.2 Gaussian Approximation for Observation

Similarly, the observation process must be approximated as being linear with Gaussian noise that is independent of state. This assumption generally holds for GPS for the global sensor and for the camera for a relative sensor.
The non-Gaussian observation model:

\[ s_i z^s_{ik} = s_i h^s_i (x^s_{ik}, s_i v^s_{ik}) \]  
\[ s_i z^s_{ik} = \begin{cases} 
  s_j h^{s_j} (x^s_{ik}, x^s_{jk}, s_i v^s_{ik}) & x^s_{kj} \in s_i X \\
  0 & x^s_{kj} \notin s_i X 
\end{cases} \]  

becomes:

\[ z^s_{ik} = h^s_i (x^s_{ik}) + v^s_{ik} \]  

since there are nonlinearities in the observation:

\[ z^s_{ik} = C^s_i (x^s_{ik}) + v^s_{ik} \]  

where the global sensor is given by:

\[ s_i x^s_i = I x^s_{ik} + s_i v^s_{ik} \]  

and the relative sensor is given by:

\[ s_i z^s_j = s_i C^s_j [x^s_{kj} - x^s_{ik}] + s_i v^s_{kj} \]  
\[ s_i C^s_j = \begin{cases} 
  I & x^s_{kj} \in s_i X \\
  0 & x^s_{kj} \notin s_i X 
\end{cases} \]
3.4.3 Centralized vs. Decentralized Estimation

With the motion and observation models developed, the final consideration for the Kalman filter is where the estimation process will occur. Since the C-HALE vehicles are severely restricted in computation and communication capacities, some high level control tasks will necessarily be performed by a single vehicle or a base station computer. Thus an estimation model should allow vehicle observations to be shared within the fleet and with base station when possible and incorporated into the on-board estimation seamlessly. Centralized estimation at the base station is a simple matter of gathering the observation matrices from each vehicle as shown in (3.24) and (3.25)

$$z_k = C_k x_k + v_k \quad (3.24)$$

$$\begin{bmatrix}
  s_1 z_k \\
  \vdots \\
  s_n z_k 
\end{bmatrix} = \begin{bmatrix}
  s_1 C_k \\
  \vdots \\
  s_n C_k 
\end{bmatrix} \begin{bmatrix}
  x_k \\
  \vdots \\
  x_k 
\end{bmatrix} + \begin{bmatrix}
  s_1 v_k \\
  \vdots \\
  s_n v_k 
\end{bmatrix} \quad (3.25)$$

Similarly, the on-board estimation is formulated with an augmented observation matrix consisting of matrices shared among the fleet when possible or necessary.

$$s_i z_k = s_i C_k^{s_i=1:n_x} x_k + s_i v_k \quad (3.26)$$
\[
\begin{bmatrix}
    s_1 z_{k}^{s_1} \\
    s_1 z_{k}^{s_2} \\
    \vdots \\
    s_1 z_{k}^{s_n}
\end{bmatrix}
= 
\begin{bmatrix}
    s_1 C^{s_1} & 0 & \ldots & -s_i C^{s_i} & \ldots & 0 \\
    0 & s_2 C^{s_2} & \ldots & -s_i C^{s_i} & \ldots & 0 \\
    \vdots & \vdots & \ddots & \vdots & \ldots & \vdots \\
    0 & 0 & \ldots & s_i C^{s_i} & \ldots & 0
\end{bmatrix}
\begin{bmatrix}
    x_{k}^{s_1} \\
    x_{k}^{s_2} \\
    \vdots \\
    x_{k}^{s_n}
\end{bmatrix}
+ 
\begin{bmatrix}
    s_1 \nu_{k}^{s_1} \\
    s_1 \nu_{k}^{s_2} \\
    \vdots \\
    s_1 \nu_{k}^{s_n}
\end{bmatrix}
\tag{3.27}
\]

In each case, when communication is not available the observation matrices shared from vehicles are replaced with a zero matrix and by the nature of the Kalman filter formulation, the states of other vehicles are updated via dead reckoning. This will cause uncertainty of the states to grow with time until communication is reestablished.

3.5 Summary

In this chapter, a formal problem statement for the high-level control task for docking C-HALE UAVs was presented. A kinematic motion model for the UAV was formulated and approximated as a linear Gaussian probabilistic process for the Kalman filter. A stochastic observation model was developed and methods for centralized and decentralized state estimation were presented. The next step towards solving the problem statement is to develop a process that will allow each vehicle to intercept and dock with the larger vehicle in an energy efficient manner. The next chapter will provide an overview of the proposed multi-level
control architecture, formulate docking priority as a means of selecting vehicle order during docking, present the proposed process of energy efficient docking, and introduce information based decision making.
Chapter 4

Energy Efficient Path Planning

With the general framework for estimation and control developed, intercept and docking of multiple UAVs can now be addressed. The constraints of the C-HALE mission dictate that intercept and docking incorporate a number of factors when determining an optimal path for intercept of multiple vehicles. Specifically, an optimal trajectory should minimize the energy consumption of the vehicles within the fleet as well as result in an identical energy state of each vehicle after docking.

The scope of this investigation is limited to high level control, decision making and path planning; collectively referred to as autonomy. The end goal of energy based docking is to successively bring vehicles within an arbitrary distance of one another on a parallel course. From this point, vehicles will follow a different trajectory optimized to reduce aerodynamic interaction between wingtips, which is outside the scope of this thesis. Formulations for energy based docking will address path planning from the point a vehicle reaches altitude
until the full vehicle is assembled.

4.1 Multi-Level Control and Estimation

Before formulating the high-level control and path planning for C-HALE vehicles it is important that these tasks be provided context within the larger control and estimation architecture. A diagram of this architecture is given in Figure 4.1. The multi-level label refers to the fact that each process in this figure occurs at a different frequency. High frequency tasks are at the bottom with motion being a continuous time process. Likewise, the lowest frequency tasks are at the top with a discrete decision maker that commands new behaviors only when significant changes to the fleet occur such as a new vehicle completing docking or a new external threat being detected. In the middle level are tasks such as path planning and multi-step prediction that occur only when the belief of a state becomes high and the vehicle needs to re-plan its actions.

A summary of the multi-level control and estimation process is as follows. Beginning with at the summing junction in the Multi-Level Control box, low level control represents an autopilot system. According to the assumptions made in the formulation of the kinematic vehicle model, the autopilot first takes a desired trajectory from the high level controller; then, governed by the 6-DOF system dynamics, commands control inputs which produce motion that is near the commanded trajectory. These control forces produce motion that
is noisy but is approximated by a 3-DOF kinematic model. The multi level estimation begins with sensors observing the new vehicle state with noise. The extended Kalman filter algorithm compares the predicted states with the observed states and estimates the current position. This process can accept inputs from multiple sensors and cooperative UAVs. The remaining boxes in this diagram have yet to be formulated.

The remaining tasks for high-level control can be broken down into several distinct tasks. In order to achieve energy efficient path planning:

1. Docking order must be determined
• Must consider location and energy state of each vehicle

• Order must be the same among vehicles, i.e. the base station must force the order on the fleet

2. Plan an intercept path for the linked $s''$ and linking vehicle $s_i$

• Intercept point must be mutually determined

• Path should reduce energy consumption and distribution

In addition, a method to quantify belief or situational awareness must be developed in order to enable rule-based behavior and control laws to be adapted to the current situation.

### 4.2 Docking Priority

In order to implement energy efficient docking, the issue of docking order must be addressed. Due to the nature of the task being attempted, two unmanned vehicles at 60,000ft touching wingtips together and physically linking, it is not practical to attempt more than one vehicle docking at a time. Therefore, while the first pair is docking, other vehicles will approach the linked vehicle. Eventually multiple vehicles will be following the linked UAV, and it is only natural that each vehicle should be placed in order according to some measure of need in terms of energy state.

One advantage of a C-HALE vehicle is that it is possible that individual vehicles could be
deployed from various small airfields rather than a single large airfield. Due to a number of factors such as weather, distance between airfields, and restricted areas that must be navigated, it is highly likely that the individual vehicles will arrive at the area for docking with very disparate energy levels. Although it is generally preferable that the vehicle with the lowest amount of energy remaining dock first so as to conserve the most energy for the vehicle most in need, it is possible that, in some situations, the vehicle with the lowest energy state is farthest from the fleet and that waiting for this vehicle to arrive before docking may consume the energy of other vehicles in the fleet unnecessarily. In such a situation it may be best to proceed with docking of some vehicles while waiting for other lower energy vehicles to arrive.

The proposed method for assigning docking priority considers both an individual vehicle’s energy state as well as distance from the fleet. The reference point for the center of the fleet should also incorporate the energy state of the vehicles. The ”center of energy” is defined by equation (4.1). Here $E_0$ is the initial energy state of each vehicle and $\bar{E}_i$ is the estimated current energy state of the individual vehicles within the fleet. The docking priority can now be defined by equation (4.3) where $\Omega_d$ is a weighting parameter that can be used to change the relative importance of vehicle energy state versus distance, $\bar{R}^i$, from the center of energy, $CE$, based on the specific scenario and/or vehicle being used.

\[
CE = \frac{\sum_{i=1}^{n_s} \bar{x}_i \left( E_0 - E_k^i \right)}{\sum_{i=1}^{n_s} \left( E_0 - E_k^i \right)}
\]

\[
\bar{R}^i = \sqrt{(\bar{x}_i - CE_x)^2 + (\bar{y}_i - CE_y)^2}
\]
\[ Q_d = \Omega_d \frac{1}{R} + \frac{1}{E} \]  

(4.3)

### 4.3 Energy Based Docking

Recall from Chapter 3 that the formal problem statement for docking is described by:

\[
\begin{align*}
J(u_s'_{k:k+n_k-1}) &= \left\| \sum_{\kappa=0}^{n_k-1} \left( \sigma_{k+n_k-1} \left( E_s' \right) \right) + \omega_1 \left( E_s' - E_s'_{k+n_k-1} \right) \right\| \rightarrow \min \\
\Phi^{s_i} &= \left\| x_{k+n_k-1}^{s_i''} - x_{s_i}^{s_i} \right\| = r_{s_i,s_i''}
\end{align*}
\]  

(4.4)

#### 4.3.1 Optimal Path Planning

The objectives for energy-optimal docking given by the multi-objective inequality (4.4) can be rewritten as a single cost function.

\[
J(u_s'_{k:k+n_k-1}) = \sum_{\kappa=0}^{n_k-1} \left( \omega_1 \sigma_{k+n_k-1} \left( E_s' \right) + \omega_2 \left( E_s' - E_s'_{k+n_k} \right) \right) \]  

(4.5)

This cost function consists of terms describing the total energy consumed as well as the standard deviation of the energy states of all vehicles within the fleet and is subject to equality constraint, \( \Phi \), which describes the final location of a linking vehicle being at some desired distance, \( r_{s_i,s_i''} \), from the linked vehicle.

\[
\Phi^{s_i} = \left\| x_{k+n_k-1}^{s_i''} - x_{s_i}^{s_i} \right\| = r_{s_i,s_i''}
\]  

(4.6)

Solving for an optimal path based on this cost function and constraint above can be accomplished via the control parameterization method which uses functions such as Matlab’s
fmincon function. This would yield a series of successive optimal paths for the linked and linking vehicles until docking is complete. The nature of the C-HALE mission concept, however, limits the practicality of this method. Mission-imposed constraints on weight and power consumption severely limits computing power available on-board each vehicle. As formulated here, determining an optimal path requires centrally planning a path for all vehicles simultaneously. While such a process might be possible for a small number of vehicles, the C-HALE mission could use as many as eight to ten vehicles which presents a significant scalability problem.

In order to address the problem statement, sub-optimal energy-based docking must be developed that, while not optimal, is capable of reducing total energy consumption and minimizing energy distribution objectives present in equations 3.1. Therefore, we must formulate a method for determining a single vehicle optimal path. Thus the cost function is rewritten as:

\[ J \left( u_{k,k+n-1}^i \right) = \sum_{k=0}^{n-1} \| (E_{0}^s - E_{k,k+n-1}^s) \| \]  
\[ \Phi^s_i = \| x_{k+n-1}^{s''} - x_{k+n-1}^{s_i} \| = r_{s_i,s''} \]  

(4.7)  

(4.8)

4.3.2 Single Vehicle Optimal Path Planning

If the speed of the chasing vehicle is steady, single vehicle path planning reduces to the well-studied shortest curve to intercept problem. Dubins [26] developed a geometric argument for determining shortest paths while later Boissonnat et al [9] achieved this result using optimal control methods. More recently, several Matlab algorithms [15, 28, 40, 41] have
been developed to achieve efficient path planning by determining the shortest of six possible Dubins paths via simple root finding. Each of these paths is given in figure 4.2.

Constructing these paths is performed by drawing two circles tangent to the current pose and two that are tangent to the desired final pose. Then, a root finding technique is used to determine the location a straight path that is tangent to each permutation of left and right circles from the current pose and desired pose yielding the LSL, RSR, LSR, and LSL paths. In the case where the distance between current and desired pose is less than 4R, a
similar process used to determine the three tangent circles connecting the two poses to yield the LRL and RLR path. Thus the single vehicle optimal path planning is simply a matter of selecting and following the shortest of the 6 Dubins paths to the predicted future state of the target vehicle which is given by finite horizon multi-step prediction:

\[
\bar{x}_{s'}^{k+nk|k} = f_{s'} \left( x_{s'}^{k+nk-1|k}, u_{s'}^{k+nk-1} \right) 
\]

(4.9)

\[
\Sigma_{x_{s'}^{k+nk|k}} = A_{s'}^{k+nk-1} \Sigma_{x_{s'}^{k+nk-1|k}} A_{s'}^{k+nk-1\text{T}} + \Sigma_{w_{s'}^{k+nk|k}}
\]

(4.10)

\[
A_{k+nk-1} = \frac{\partial f_{s'} \left( x_{s'}^{k+nk-1|k}, u_{s'}^{k+nk-1} \right)}{\partial \bar{x}_{s'}^{k+nk-1|k}}
\]

(4.11)

### 4.3.3 Docking Formation

The rate of energy consumption for vehicles can be approximated using the drag equation with the zero-lift drag coefficient and Oswald efficiency number being estimates of a vehicle still under development. The required power, \( P \), for the linked vehicle, \( S'' \), consisting of \( n \) linked vehicles is given by equation (4.12). From this expression, the energy-optimal velocity for linked and unlinked vehicles can be determined and is given by (4.13)

\[
P_{s''} (V, n) = [F_{D_0} + F_{D_i}] V
\]

(4.12)

\[
P_{s''} (V, n) = \frac{1}{2} \rho V^2 (nS) [C_{D_0} + C_{D_i}] V
\]

\[
C_{D_i} = \frac{C_L^2}{\pi e (nAR_i^2)}, \quad C_L = \frac{(nW)}{\frac{1}{2} \rho V^2 (nS)}
\]

(4.13)

\[
V_{s''_{p_{\min}}} (n) = \sqrt{\frac{4}{3\pi n^2 \rho^2 A R_i^2 C_d A}} \frac{W_i^2}{}\]

(4.14)
Equation 4.14 shows that the energy-optimal velocity of a vehicle is inversely related to the square root of the number of vehicles linked. This results in the benefit that the most energy-efficient velocity decreases as the number of vehicles linked increases. A simple optimal energy-based docking process can be achieved by commanding the linked vehicle to fly at its most efficient velocity while the linking vehicle intercepts it, also while traveling at minimum power velocity which is faster that the linked vehicle.

As the number of linked vehicles increases, the maneuvering rate of the linked vehicle will decrease. As a result, the linked vehicle should fly straight and level while the unlinked vehicle pursues and maneuvers into position for docking.

![Image of energy-based docking formation](image)

**Figure 4.3: Example of Energy-Based Docking Formation Case I.**

In the case where multiple vehicles are in position and ready to dock at the same time, as in figure 4.3, this method would require more complex maneuvering of a vehicle in formation in
order to maintain relative distance with the linked vehicle while waiting to dock. In essence, unlinked vehicles, which are faster, would be required to weave back and forth to cover a greater distance while remaining in roughly the same position relative to the linked vehicle.

In order to avoid the use of more complex maneuvers by the following vehicles it is desirable to have all vehicles travel at the same velocity. This means that some set of vehicles, either the linked set or the unlinked set, will be required to travel at a speed that is not its most efficient. It is necessary to determine which set should fly in a non-optimal energy state.

Figure 4.5 shows a set of sample energy curves for a linked vehicle consisting of up to eight cellular vehicles and is based on equation 4.14. These curves are based on approximate vehicle parameters that fall within the scope of the C-HALE mission of interest. Once the
first two vehicles are linked, the energy consumption per vehicle at the minimum power velocity for a single vehicle drops by nearly forty percent which is the basis for the C-HALE concept, therefore the linked vehicle should be the one flying at a non-optimal energy state.

In effect, this gives highest priority to the energy consumption of the unlinked vehicle. This also has the benefit of slightly reducing energy distribution among vehicles in the fleet as well as allowing the linked vehicle to fly faster than its stall speed. The proposed near-optimal energy-based docking process is described by figure 4.6. This places priority on preserving energy in the less efficient unlinked vehicle while reducing any unnecessary maneuvering.
4.4 Information Based Decision Making

When developing high level control laws, it may become necessary to adjust these laws based on the quality of information available. In the case where a vehicle has a very accurate estimate of cooperative vehicle states, external threat states and environmental conditions it is said to have high situational awareness. In this case, a vehicle can act more confidently and aggressively during path planning. Conversely, when situational awareness is low, vehicles should act more cautiously and conservatively. In practical terms, a "confident" vehicle will assign a lower avoidance radius to an obstacle, communicate less frequently and the information from this vehicle's sensors will be weighted more heavily by a base station or operator.

The state estimation discussed above provides more rich information than just the states of
vehicles within and external to the fleet. The covariance matrix of the estimated states, $\Sigma_{x_{k|k}}$, presents an opportunity to quantify situational awareness. With vehicle states estimated, we need to know how good the estimate is. We can do this using information entropy. For systems with Gaussian noise represented by the covariance matrix, $\Sigma_{x_{k|k}}$, information entropy is given by [54, 59]:

$$H(\chi) = \ln \sqrt{(2\pi e)^n} \left| \sum_{k|k} x_k \right|$$  \hspace{1cm} (4.15)

We can use this scalar value, $H$, as a threshold to command different behaviors through rule based high level control more easily than the simply using the full covariance matrix. Also, by selecting a different subset, $\chi$, we can introduce different behaviors that respond to very specific aspects of situational awareness. For example, if a vehicle $V_{i=1}$ is estimating the states of $V_{i=1,n}$ and is trying to link with vehicle $V_3$ it need not consider the information entropy of the estimation of vehicles $V_{i\neq1,3}$, the subset $\chi$ is chosen to include only the estimated states

---

**Figure 4.7: Proposed Energy-Based Docking.**
of the vehicles of interest.

The primary implementation of information entropy in the C-HALE mission simulations was to use information entropy to reduce the amount of communication between vehicles. The limitations of computation power and weight available for communication equipment for the C-HALE mission necessitates that unnecessary communication among vehicles be reduced as much as possible. Since communication between vehicles occurs primarily before docking and is used to correct the estimate of vehicle locations, an entropy based rule is introduced by choosing $\chi$ to be the position and heading of each vehicle individually:

$$\chi^j = \begin{bmatrix} \bar{x}^j_l & \bar{y}^j_l & \bar{\psi}^j_l \end{bmatrix}^T.$$  

Then the covariance matrix is condensed to contain only the covariances of the states of interest. Thus the information entropy of the estimated position and heading of each vehicle in the fleet can be used to initiate communication if the value rises above some predetermined threshold.

### 4.5 Summary

This chapter has presented an overview of the proposed multi-level control architecture. We have proposed energy based docking shown in 4.7 that uses the docking priority given by equation 4.3 to command the status for each vehicle, $s_i : ns$. In the linked vehicle status, the path is straight and level flight ad the velocity of minimum power for a single vehicle given by 4.14 where $n = 1$. In the chasing status, the vehicle follows a Dubins shortest path trajectory to the predicted future location of the linked vehicle given by 4.11. Finally
the remaining vehicles fly straight and level at \( v_{p_{\text{min}}}^{s_i} \). During docking vehicle behaviors can be developed to respond to an assessment of situational awareness through quantifying the belief of estimated states.

The following chapter will present the real-time implementation of the work proposed here. This implementation required the development of the Platform-In-the-Loop simulator which allows networked computers to interact as virtual vehicles while docking is simulated and visualized using a flight simulator software package.
Chapter 5

Platform-In-The-Loop Simulator

Implementation

This chapter describes the implementation of the formulations developed previously. First, a simulation was constructed on a single computer to evaluate docking and determine the docking priority tuning parameter, $\Omega_d$. This simulation considered all vehicle states as deterministic and Kalman filtering was not included. Next, a Platform-In-the-Loop-Simulator, PILS, was developed to demonstrate docking and mimic estimation uncertainty and communication between vehicles. Finally, the PILS was augmented with additional computers to allow real-time visualization of the C-HALE mission.
5.1 Docking Priority

The first evaluation of the proposed energy efficient docking was done with a series of pattern simulations with a single computer. The goal of this simulation was first to demonstrate docking could successfully complete sequential docking. After this was demonstrated, a series of parametric studies were performed in order to determine the correct weighting factor, $\Omega_d$, for determining docking order. An example of these simulations is shown in Figure 5.1.

In this screen shot two vehicles have completed docking while the other two chase the linked vehicle. Notice that the two vehicles that have completed docking are consuming energy much more slowly than the two vehicles that have not docked. The vehicle trailing the farthest back is following its own planned shortest path to intercept the linked vehicle. Finally, the energy plot in the top right shows that the two docked vehicles were not the two with the lowest energy. This is due to the fact that these two vehicles were closest together at the beginning of the simulation and the docking priority formula gave these two vehicles the highest priority.

A series of parametric simulations were conducted in order to determine the weighting factor, $\Omega_d$, for determining docking order formula. The parameters for these simulations are given in Tables 5.1 and 5.2. The parameters of interest were mean starting distance and docking weight factor. Since the location and energy state of vehicles when docking will have a significant effect on the amount of energy consumed during docking, a total of 250 simulations were performed for each combination of parameters with random start locations and energy states. A total of twenty mean starting distances and weight factors were simu-
lated for a total of 105,000 full docking simulations.

The results of these parametric studies are shown in Figures 5.2 and 5.3. From the energy distribution results, Figure 5.3, it is clear that the initial vehicle locations have no effect on final energy distribution. In Figure 5.2, it was expected that energy consumption would rise with starting distance; however, the relationship between the docking weight parameter and distance was not. This suggests that the docking priority weight should change with starting distance.

Figures 5.4 and 5.5 show the two extremes of starting distance in terms of energy consumption and distribution. In Figure 5.5, the vehicles started with an average spacing of
Table 5.1: Fixed Simulation and Vehicle Parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link Time</td>
<td>60 s</td>
</tr>
<tr>
<td>Wingspan</td>
<td>75 ft</td>
</tr>
<tr>
<td>Weight, W</td>
<td>1000 lb</td>
</tr>
<tr>
<td>Area, S</td>
<td>1000 sqft</td>
</tr>
<tr>
<td>Aspect Ratio, AR</td>
<td>5</td>
</tr>
<tr>
<td>Oswald Efficiency Factor, e</td>
<td>0.8</td>
</tr>
<tr>
<td>Parasite Drag Coefficient, $C_{D_0}$</td>
<td>0.009</td>
</tr>
</tbody>
</table>

Table 5.2: Variable Simulation and Vehicle Parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>88 ft/s</td>
<td>210 ft/s</td>
</tr>
<tr>
<td>Docking Weight, $\Omega_d$</td>
<td>0.001</td>
<td>100</td>
</tr>
<tr>
<td>Maneuvering Rate, $\dot{\psi}$</td>
<td>0.15 rad/s</td>
<td>0.37 rad/s</td>
</tr>
<tr>
<td>Power for Docking</td>
<td>22 kWh</td>
<td>25 kWh</td>
</tr>
<tr>
<td>Mean Start Distance</td>
<td>100 m</td>
<td>3000m</td>
</tr>
</tbody>
</table>
3km compared to 250m in 5.4. These two Figures suggest that the docking priority weight parameter should range from 8x10E-3 when vehicles are close to 8x10E-2 when vehicles are farther apart. For all further simulations, the weighting parameter was chosen to be the intercept of the two curves for the specific starting distance being simulated.
5.2 PILS Implementation

With the docking priority weighting complete, the proposed energy efficient docking can now be implemented on the Platform-In-the-Loop-Simulator, or PILS. The goal of the PILS is to simulate as nearly as possible the effects of estimation uncertainty and communication between a fleet of cooperative virtual UAVs.

Hardware and information flow diagrams are given in Figures 5.7 and 5.6, respectively. When simulating docking, a group of four computers represent virtual vehicles. They perform estimation of the states of vehicles within the fleet using false and noisy sensor readings, $s_i z_k$, provided by the master computer. After estimation, the vehicles will plan a trajectory and
Figure 5.4: Energy Consumption and Distribution At 250m Start Distance

command control inputs, \( u_k \), based on the status given by the base station. These control inputs are received by the master computer which, at higher frequency than the virtual vehicle, adds noise and determines the true position to which noise is added to form the sensor reading for time step \( k+1 \).

The computers used to represent the virtual vehicles are ASUS eee Box net top PCs which are relatively inexpensive and low powered computers. These were selected for the Nvidia Ion GPU which can be used for parallel grid based recursive Bayesian estimation; however, these are not utilized in the energy based docking for C-HALE vehicles. In general, the computers selected for the virtual vehicles are intended to mimic the available computing
power on board a lightweight solar powered UAV. Each computer is connected using Cat5e cable and a gigabit Ethernet switch. Communication between vehicles was performed using the UDP protocols in the Instrument Control Toolbox in Matlab.

5.3 FlightGear Visualization

Vizualization of the docking simulations was performed using FlightGear, an open source flight simulator. In order to accomplish this, the PILS was supplemented with four additional computers shown as FG Clients in Figures 5.8 and 5.9. These computers took the true state
of the vehicle from the master computer and displayed the vehicle on the monitor. This was accomplished using the FlightGear interface Simulink blockset. This blockset displays the vehicle and outputs vehicle location to a multiplayer server which allows each monitor to show the location of all vehicles within the fleet.
5.4 Summary

This chapter has presented the development of real-time implementation of energy efficient docking. This implementation required the development of the Platform-In-the-Loop simulator which allows networked computers to interact as virtual vehicles while docking is simulated and visualized using FlightGear flight simulator software package. The following chapter will present and discuss the results of the PILS simulations.
Figure 5.8: Information Flow for PILS-FlightGear Implementation.
Figure 5.9: PILS-FlightGear Implementation Hardware Configuration.
Chapter 6

Platform-In-The-Loop Simulation

Results

This chapter presents results from the simulations of energy based docking. First, a series of simulations evaluated docking in terms of the energy performance of the docking priority as a means of selecting docking order. Next, several screen shots of the PILS results demonstrate successful docking of virtual networked vehicles. Finally, a series of figures demonstrate the use of information entropy as a means of initiating communication between vehicles.

6.1 Energy Based Docking: Master Simulations

In order to evaluate the process for determining docking order, a series of simulations were performed to compare the order chosen using docking priority to all possible docking orders.
Table 6.1: Fixed Simulation and Vehicle Parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link Time</td>
<td>60 s</td>
</tr>
<tr>
<td>Maximum Start Radius</td>
<td>1500 m</td>
</tr>
<tr>
<td>Docking Weight Parameter $\Omega_d$</td>
<td>8E-3</td>
</tr>
<tr>
<td>Energy V1</td>
<td>24.57 kWh (80%)</td>
</tr>
<tr>
<td>Energy V2</td>
<td>24.19 kWh (66%)</td>
</tr>
<tr>
<td>Energy V3</td>
<td>23.44 kWh (37%)</td>
</tr>
<tr>
<td>Energy V4</td>
<td>23.82 kWh (52%)</td>
</tr>
</tbody>
</table>

For a fleet of four vehicles there are twenty-four possible docking orders. For a series of 100 random initial vehicle locations, docking was simulated using each of the twenty-four possible orders. The full list of parameters for these simulations is given in table 6.1. These simulations did not consider sensor noise.

Figures 6.1 and 6.2 show the results of each permutation of docking order with the automatically selected docking order determined by equation 4.3 highlighted in yellow. The black line shows the mean energy consumption of the vehicle for all orders and simulations.

From figure 6.1 it is clear that the automatic order selection causes vehicle 3, which from table 6.1 began with the least energy, to consume less energy during docking while vehicle 1, which began with the most energy, consumes more energy. Figure 6.2 shows the final energy standard deviation for all simulations with the automatic docking order results highlighted.
in yellow. As desired, the automatic docking order selection resulted in a reduced standard deviation with few exceptions that correspond to instances where the vehicle with the most energy, vehicle 1, was close to the center of the fleet and was commanded to dock earlier than vehicles with less energy.
Figure 6.2: Effects of Arbitrary vs Automatic (Yellow) Docking Order on Final Energy Standard Deviation.

6.2 Energy Based Docking of Networked Virtual Platforms

Results of the PILS implementation discussed in Chapter 5 are shown in figures 6.3 and 6.4. Figure 6.3 shows the PILS during a docking simulation. The virtual vehicles each display their local map with the estimated location of all vehicles within the fleet as well as estimated energy state and the information entropy of the estimated states. The PILS simulation is capable of successfully docking each vehicle with real-time visualization of the vehicles in
Table 6.2: Energy Consumption With Arbitrary vs Automatic Docking Order.

<table>
<thead>
<tr>
<th></th>
<th>Arbitrary Order</th>
<th>Auto Order</th>
<th>% Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy Consumed V1</td>
<td>0.9190 kWh</td>
<td>1.1187 kWh</td>
<td>+21.72%</td>
</tr>
<tr>
<td>Energy Consumed V2</td>
<td>0.9190 kWh</td>
<td>0.9668 kWh</td>
<td>+5.20%</td>
</tr>
<tr>
<td>Energy Consumed V3</td>
<td>0.9198 kWh</td>
<td>0.7837 kWh</td>
<td>-14.87%</td>
</tr>
<tr>
<td>Energy Consumed V4</td>
<td>0.9190 kWh</td>
<td>0.8270 kWh</td>
<td>-10.01%</td>
</tr>
<tr>
<td>Final Energy STD</td>
<td>0.504 kWh²</td>
<td>0.343 kWh²</td>
<td>-31.94%</td>
</tr>
</tbody>
</table>

FlightGear as shown in figure 6.4.

Notice that in the PILS screen shot shown in figure 6.4 the vehicle at the far end of the fleet appears to be ahead of the rest of the formation. This is a result of time lag in the multiplayer server. As yet, no solution has been found to reduce this ”jitter” in the simulations with flightgear visualization.

6.3 Effect of Information Entropy on Communication

As described in Chapter 4, information entropy is a useful scalar value that quantifies the uncertainty of an estimated state. The calculation of information entropy was given in equation 4.15. Figures 6.5, 6.6, and 6.7 show the time history of information entropy during a FlightGear demonstration of the energy based docking of C-HALE vehicles. In these simulations the threshold for communication varied from zero to ten meaning that each
vehicle did not communicate until the information entropy of the state of a specific vehicle passed the threshold. The correction threshold is the value that information entropy must reach before the vehicle initiates communication with the master computer to receive the position of the other vehicle and correct the estimated state. The sudden drops in information entropy occur when a vehicle enters the field of view of the sensor platform. Correction can then occur at every time step. Notice that the information entropy of vehicles 1 and 2 estimated by vehicle 3 drop at the same time. This is due to the fact that in this simulation vehicles 1 and 2 enter the field of view of vehicle 3 nearly simultaneously. Similarly, since
vehicles 1 and 4 are on opposite sides of the formation they never enter each others field of view resulting in a constant value for information entropy.

As the correction threshold rises, the correction frequency falls. The correction interval produces a sawtooth pattern in the information entropy. The main tradeoff here is between speed of computation and accuracy of the state estimation. The simulations so far are too general in terms of sensor, vehicle, and communication parameters to determine an optimal correction threshold; however, we have demonstrated the expected behavior of the relationship. Another issue that has not been overcome is timing between virtual vehicles. Currently the PILS cannot not determine the time difference between toe clock on each vehicle which means that it has been impossible to compare the estimated vehicle state
with the true state on the master computer. As a result, the communication threshold that produces an acceptable level of state error has not yet been determined.

6.4 Effect of Docking Priority on Energy State - PILS Demonstration

Figures 6.8 and 6.9 show the results of another PILS-FlightGear demonstration. In this case, a time history of energy consumption was recorded. Figure 6.8 shows both the time history of energy state and power being consumed while Figure 6.9 shows the difference in the time
These results reflect improvement in energy state by using the proposed method for determining docking order similar to the results shown in 6.1. The results of the PILS-FlightGear demonstrations demonstrate that the proposed architecture for multi-level control is suitable for real-time implementation on a real aircraft.
Figure 6.7: PILS-FlightGear Demonstration Information Entropy, Correction Threshold=10.
Figure 6.8: PILS-FlightGear Demonstration Energy State.

Figure 6.9: PILS-FlightGear Demonstration Energy Improvement.
Chapter 7

Conclusions and Future Work

This thesis has addressed various tasks within the C-HALE mission. A multi-level control architecture was formulated with decentralized estimation of vehicle states. An energy based docking process was developed that achieves energy efficient docking using shortest-path trajectory planning rather than more computationally expensive optimal control. As part of docking, a method was created to determine the order for docking which results in more evenly distributed energy within the fleet. Information based decision making was used to reduce communication. The use of information entropy has allowed rule based control laws to be adjusted according to a scalar assessment of situational awareness. A Platform-In-the-Loop simulator was used to perform a full simulation of cooperative docking of virtual networked cooperative UAVs in real-time.

Cooperative docking of HALE UAVs is a unique task. As a result, it was not possible to compare the performance of the proposed energy based docking with other methods. A series
of numerical studies did reveal some useful insights. Docking of vehicles results in a larger vehicle that consumes energy at a rate of 21% per vehicle compared to an unlinked vehicle and increases vehicle range by a factor of three without considering solar recharging. Each kiloWatt-hour of energy produced through solar cells will extend the range of the linked vehicle by 13.64 km while an unlinked vehicle will only achieve 4.35 km of additional range.

When determining the proper weight parameter for establishing docking order, it was found that the proper weight parameter is highly dependent on the starting distance of the vehicles when considering final energy state while the final energy distribution of the fleet was unaffected by start distance. Using docking priority to determine the order of docking has also been shown to significantly reduce the final energy distribution within the fleet.

Information entropy has been shown to be an effective means of altering rule-based control laws based on situational awareness; however, due to issues with asynchronous simulation, the proper communication threshold was not determined.

This work has addressed several tasks of the C-HALE mission, but there are many issues that have not been resolved. The logical extension of this work lies in determining how to change the control architecture after the vehicle has linked. Once linked, the vehicle needs only to estimate a single vehicle’s state, and continuing the decentralized estimation would waste energy. In practice, the vehicle would use the inactive flight computers to perform additional tasks of an ISR mission which would likely require search and tracking of a target. This estimation task would need to transition from an extended Kalman filter approach to a nonlinear non-Gaussian technique. Another area of future work would be to improve the
docking process to use coordinated optimal control which in this work was not considered due to the computational expense and lack of scalability in planning optimal control inputs of large fleets of vehicles. Finally, information based decision making has been implemented for a single behavior which was to reduce communication. Future work should investigate other possible uses of modifying high-level control behaviors based on situational awareness.
Bibliography


