AIAA Infotech@Aerospace Conference <br>and <br>AIAA Unmanned...Unlimited Conference
6 - 9 April 2009, Seattle, Washington

AIAA 2009-1932

Vision Based Distributed Cooperative Navigation for MAVs in GPS denied areas

Rajnikant Sharma* and Clark N. Taylor†

Brigham Young University, Provo, Utah, 84602, USA

Current Miniature Air Vehicle (MAV) systems rely on the availability of GPS to enable navigation (state estimation) during flight. However, many envisioned MAV usage scenarios require MAVs that can fly in tightly constrained spaces (e.g., urban terrain, indoors, dense jungle, etc.) meaning that GPS may not be available. While inertial sensors can be used to estimate navigation state, their estimates will diverge over time without GPS. In this paper, we propose a method for overcoming this drift in a multiple-MAV scenario. We develop a distributed cooperative navigation system where each MAV is equipped with a bearing-only sensor (an electro-optical camera) and an inertial measurement unit. The camera is used to measure the bearing from other MAVs and/or landmarks which are in its field of view. By communicating local bearing information with other MAVs in their communication range, each MAV can estimate, without drift, its navigation state *assuming* the entire system of MAVs observes two or more landmarks at known locations. The distributed cooperative navigation system explicitly minimizes communication both in terms of message size and number of communication links. We describe sufficient conditions for eliminating drift in the connectivity of the MAVs and how many landmarks are observed. We present simulation results to support the developed theory.

I. Introduction

Advancements in unmanned aerial vehicles, especially Miniature Air Vehicle (MAV) technology have made it possible to keep human pilots out of many dangerous aerial situations. MAVs can be used in place of piloted airplanes for both military and civil missions, including hazardous search and rescue operations, aerial photography, collection of weather data, etc.

Navigation is at the heart of all manned and unmanned air missions; none of the tasks mentioned above are possible without precise navigation. Modern navigation systems for MAVs typically consist of an Inertial Measurement Unit (IMU) and Global Positioning System (GPS). A typical IMU contains three accelerometers and three gyroscopes that provide information about the motion of a moving body. Unfortunately, the process of extracting body position estimates from an IMU leads to significant drift over time particularly with the low-cost, low-weight IMUs that can be placed on-board an MAV. GPS, on the other hand, can be used to bound drift as it provides absolute position information. Therefore, both IMU and GPS are used together for precise navigation, known as hybrid INS/GPS navigation. However, many envisioned MAV usage scenarios require MAVs that can fly in tightly constrained spaces (e.g., urban terrain, indoors, dense jungle, etc.) meaning that GPS may not be available.

There have been many approaches in the past that have attempted to use sensors besides GPS for navigation. Most of these papers have considered the navigation of a single MAV.[1,2] In this paper, however, we discuss a low-cost technique for enabling MAV navigation in scenarios where several MAVs are working cooperatively to fulfill a mission. This cooperative style of mission has been studied extensively for reasons unrelated to navigation. For example, some MAV missions may be quite dangerous, and it is unlikely that a single agent would survive long enough to complete the task. Similarly, some MAV models are inexpensive but prone to failure, in which case it may be more economically feasible to use a large number of cheap MAVs

*Graduate research assistant, Department of electrical and computer engineering, raj.drdo@gmail.com, Student member AIAA.

†Assistant Professor, Department of electrical and computer engineering, taylor@ee.byu.edu, Member AIAA.

rather than risk a single expensive one. Many tasks, such as searching a particular area, can be completed more quickly using multiple MAVs.

In this paper we use multiple MAVs to cooperatively navigate in GPS denied areas. Each MAV is assumed to have a camera on-board (a bearing-only sensor) and an inertial measurement unit (IMU). In addition, we assume that using all the MAVs in the system, two or more fixed landmarks of known location are observed. By communicating local IMU information and MAVs share their bearing from other MAVs and landmarks and IMU information with neighboring MAVs to estimate the position and heading in distributed and decentralized manner.In this work we exploit multisensor data from a group of MAVs for navigation. Multisensor estimation can be achieved in either a centralized or decentralized manner. In centralized estimation, local sensor information from every MAV is fused at a single node to estimate the states of all the MAVs. Although centralized multisensor systems are an improvement on single sensor systems, they have a number of disadvantages including: (1) severe computational loads imposed on the central node, (2) the possibility of catastrophic failure in case of failure of the central node and (3) the communication requirements of transmitting all information to a central node and communicating back all estimated state information. Distributed systems, on the other hand is a data processing system in which all information is processed locally and there is no central processing. In such systems, data fusion occurs locally on each node using only local observations and information communicated from neighboring nodes.

A. Related Work

A significant amount of research has been done in cooperative positioning (estimation of position only, no heading) for robots. In Kurazume et al,[3–5] a set of ground robots was divided into two groups. One group remains stationary and acts as a landmark while the other group moves and performs positioning based on relative position measurement from the stationary group. After a set amount of time, the groups exchange roles. By repeating this process of movement and stoppage, the robot group works towards its destination. Although estimation error in this method is significantly less then stand-alone dead reckoning system this method cannot be applied to fixed-wing MAVs as these MAVs are not capable of stopping in mid-air while other MAVs move.

Spletzer et al[6] uses vision to cooperatively localize and control a group of mobile robots in pre-specified formation. Rekleitis et al in their paper[7] compares several Cooperative localization techniques for different sensors measurements such as range only or bearing only and both range and bearing and the effect of the number of robots. Mourikis and Roumeliotis[8,9] in their work study the accuracy of position estimation for groups of mobile robots performing cooperative localization. They provide analytical expressions for the upper bound on their expected positioning uncertainty for a team comprising of possibly heterogeneous robots. Note, however, that each of these methods assumed a centralized estimation filter.

Nebot et al[10] describes the implementation and development of a decentralized architecture for the programming and control of a team of coordinated robots for cooperative missions. Sanderson[11] extends cooperative positioning work for linearly modelled ground robots by applying a distributed Mean Square Estimator. In this paper robots exchange their relative measurement, turn rates and planed trajectories with other robots in group. They show that cooperation increases the navigation capability of robots and estimation error decreases with increase in number of robots. Roumeliotis and Bekey[12] develops a decentralized Kalman filter approach for localization of groups of mobile robots. They use heterogeneous robots with different sensors and intercommunication capabilities for localization. While the above mentioned approaches[10–12] successfully implement decentralized cooperative localization, they require communication of a global covariance matrix among all nodes in the network, an infeasible communication cost for large networks.

In previous work we have developed a cooperative navigation method for MAVs flying in an GPS denied area.[13] IMU data and data and inter MAV range bearing from all MAVs is fused together and position heading is estimated using EKF. By performing nonlinear observability analysis, we have shown that the biases of the IMU become observable, significantly decreasing the amount of drift in navigation state estimates. However, drift is still present over time in this system.

B. Contribution

In this paper we develop a distributed cooperative navigation method for MAVs. We assume each MAV has an on-board camera that collects bearing-only measurements of other MAVs and landmarks. This system has several advantages over prior systems, including:

- Assuming two fixed landmarks are observed by the entire system, the drift does *not* increase over time. Using Lie Derivatives and a graph-based representation of the communication and measurement topology, we derive the necessary and sufficient condition for the observability of the system, demonstrating that drift-free navigation state estimation is feasible in a multi-agent system.
- Using a distributed information filter to estimate navigation state, we are able to minimize the communication requirements of multi-agent navigation. By carefully designing this system, we show that identical performance is achieved as with a centralized filter.

C. Paper Organization

The rest of the paper is organized as follows. Section II described our cooperative navigation system. Section III presents the observability analysis of this system using a graph to represent communication and measurement relationships between agents. Sufficient and necessary conditions for observability are presented in this section. In Section IV, we develop a distributed estimation system that achieves the same results as the centralized filter presented earlier in the paper. Section V presents simulation results verifying our system. Section VI concludes the paper.

II. Bearing Only Cooperative Navigation

The problem which we consider in this paper is a group of N MAVs flying with constant speed in level flight. The MAVs in this group have a camera on-board that can be used to observe the bearing of neighboring MAVs or of fixed landmarks. We also assume that the measurements and communications between MAVs are limited by range. An example of this scenario is shown in Figure 1, where three fixed landmarks (colored circles) are present in the environment with 5 MAVs. In addition to showing a distribution of MAVs and landmarks, Figure 1 also introduces the concept of a graph which connects the MAVs to each other and the landmarks. Each edge in the graph between an MAV and a landmark represents the MAV obtaining measurements of the landmark. (Note that not all MAVs in the environment are observing a landmark.) Each edge in the graph between two MAVs signifies that relative bearing measurements occur between the two MAVs and that they can communicate information between the MAVs. Note that the graph in Figure 1 does not have every MAV communicating with every other MAV, but rather has a sparse set of edges.

Using this basic scenario, we would like each MAV to be capable of estimating its navigation state without drift over time. Let us discuss the basic model used to describe each MAV, followed by the information communicated and used for navigation state estimation. We assume that each MAV has an autopilot on-board that regulates altitude and that all MAVs are flying at approximately the same altitude. To simplify the mathematics in the paper, we also assume that the fixed landmarks are at the same altitude as the MAVs, making the scenario in Figure 1 fully planar.

Because we are assuming the MAVs all lie in a plane, the navigation state for the ith MAV is

$$X_i = [x_i, y_i, \psi_i]^T. \tag{1}$$

The kinematics for each MAV is given by

$$\begin{aligned} \dot{x}_i &= V_i \cos(\psi_i), && (2)\\ \dot{y}_i &= V_i \sin(\psi_i), && (3)\\ \dot{\psi}_i &= \dot{\psi}_{gi} + w_i, && (4) \end{aligned}$$

where $\dot{\psi}_{gi}$ is returned by a gyroscope on-board the MAV

The measurements obtained by each MAV is the bearing from landmarks and other MAVS as measured by a camera. The bearing angle measured between MAV i and object j (whether another MAV or a landmark at a known location) is given by

$$\eta_{ij} = \tan^{-1} \frac{y_j - y_i}{x_j - x_i} - \psi_i + v_{ij}, \tag{5}$$

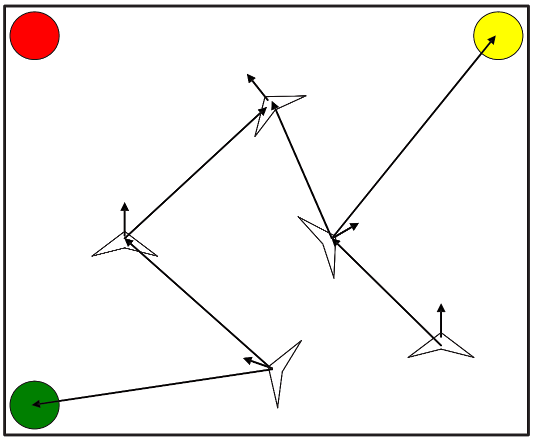

Figure 1. This figure illustrates the general class of problems addressed in this paper. Note that while there is a collection of both MAVs and fixed landmarks, the connections between these nodes are sparse.

where v_{ij} is measurement noise $v_{i,j} \in \mathcal{N}(0, \sigma_v)$. If the i_{th} MAV has n_i MAVs and m_i landmarks in its field of view, then the measurement vector for i_{th} MAV is given by

$$\begin{aligned} Z_i &= [\eta_{i,j}, \eta_{i,k}], & (6)\\ j &= 1, \cdots, n_i, & (7)\\ k &= 1, \cdots, m_i. & (8) \end{aligned}$$

With the measurement and state vector for each MAV defined, we can define the total system state variable as

$$\begin{aligned} X &= [X_1, X_2, \cdots, X_N]^T, & (9)\\ X &\in R^{3N}, & (10) \end{aligned}$$

and the total measurement vector as

$$\begin{aligned} Z &= [Z_1, Z_2, \cdots, Z_N], & (11)\\ Z &\in R^M, & (12)\\ M &= \sum_{i=1}^{N}(n_I + m_i). & (13) \end{aligned}$$

For a centralized version of the system estimator, an Extended Kalman Filter (EKF) is used with the state vector defined in Equation (9) and the measurement vector defined in Equation (11). The full equations for the EKF are:

$$\begin{aligned} \hat{X}(k|k-1) &= \hat{X}(k-1|k-1) + \dot{X}(k-1|k-1)T_s & (14)\\ P(k|k-1) &= F_k P(k-1|k-1) F_k^T + Q_k & (15)\\ K_k &= P(k|k-1) H_k^T (H_k P(k|k-1) H_k^T + R_k)^{-1} & (16)\\ \hat{X}(k|k) &= \hat{X}(k|k-1) + K_k[Z_k - h(\hat{X}(k|k-1))] & (17)\\ P(k|k) &= (I - K_k H_k) P(k|k-1), & (18) \end{aligned}$$

where F_k is the Jacobian of the kinematic equations found in Equations (2-4), $h(\cdot)$ is the measurement equation expressed in (5), and H_k is the Jacobian of that equation.

Using this setup of the Kalman Filter, it is possible to estimate the navigation state of every MAV in a centralized filter. In the following section, we perform a non-linear observability analysis of this system which demonstrates that the navigation state of every MAV is locally weakly observable assuming that two or more landmarks of known location are observed and the graph of MAVs is connected. Because the system is observable, drift will not grow with time but will instead be a function of the noise parameters of the measurement and kinematic steps in the Kalman filter.

III. Observability Analysis

A system is observable if its states can be distinguished in the state space and can be obtained from a finite sequence of measurements. Observability analysis is a well-established method for determining whether the information available from measurements is sufficient to estimate the states.[14] If the entire state is not observable, the observability analysis can also be used to determine which states or linear combination of states are unobservable. Therefore, we undertake an observability analysis of the cooperative navigation system described in Section II to measure the effectiveness of the system and determine if any of the system states will drift over time. If the system is fully observable, it means that the state noise will be limited by the input measurement noises and will not increase with time.

In the two sections that follow, we briefly review how observability is computed for non-linear systems and introduce some terminology used to describe the connections in our cooperative navigation system.

A. Computing Observability for a non-linear system

Because our system is non-linear (e.g., bearing-only measurements) we cannot apply traditional linear techniques for evaluating observability. Instead, we can prove local observability in a non-linear system as introduced in.[15] The observability of nonlinear systems can be determine using Lie derivatives. Let

$$l(x) = \begin{bmatrix} L_f^0(h) \\ \cdots \\ L_f^{n-1}(h) \end{bmatrix} \tag{19}$$

be a column vector of Lie derivatives where

$$\begin{aligned} L_f^0(h(x)) &= h(x) \\ L_f^1(h(x)) &= \left\langle \frac{\partial}{\partial x}[L_f^0(h(x))], f(x) \right\rangle \\ &\cdots \\ L_f^{n-1}(h(x)) &= \left\langle \frac{\partial}{\partial x}[L_f^{n-2}(h(x))], f(x) \right\rangle . \end{aligned}$$

Note that the $< a, b >$ operator denotes the inner product of the a and b vectors, and that we are assuming a single measurement is produced by $h(x)$. If more than one measurement is produced, each measurement will add its own set of rows to a concatenated observability matrix.

The observability matrix $\mathcal{O}$ using Lie derivatives is computed as

$$\mathcal{O} = \frac{\partial l(x)}{\partial x}. \tag{20}$$

A system is observable if the observability matrix is full rank. If $\mathcal{O}$ is not full rank, the row space of the matrix gives the observable modes, while the unobservable modes are in the row null space of $\mathcal{O}$.

B. Graph-based representation of our system

In our scenario, we represent all MAVs as nodes in a graph and edges as a bearing measurement between two nodes. We call this graph a *relative position measurement graph (RPMG)*. The RPMG is an ordered pair (X, E) consisting of a set X of vertices as (the MAVs in the system) and E, the edges, representing bearing measurements between two nodes. If there n MAVs in the group then

$$X = \{X_1, X_2, \cdots, X_n\} \tag{21}$$

$$E = \{\eta_{ij} : i, j \in \{1, \cdots, n\}\} \tag{22}$$

If η_{ij} is an edge then we say X_i and X_j are adjacent. A graph is called complete if every pair of vertices are adjacent. A path of length r from X_i to X_j in a graph is a sequence of $r+1$ distinct vertices starting with X_i and ending with X_j. If there is path between any two vertices of a graph then the graph is connected. We assume that our system maintains a connected, but not complete, graph between the MAVs.

C. Sufficient Conditions for Observability of Cooperative Navigation

In this section we derive sufficient conditions for observability using graph theory and Lie-Derivatives. We prove two important theorems regarding the observability using several Lemmas.

Theorem 1: If the RPMG representing the MAVs is *connected*, the rank of the nonlinear observability matrix $\mathcal{O}$ is $3(n-1)$, where n is the number of MAVs in the graph.

Proof: To prove this theorem, we first describe and prove the following four lemmas. Once these four lemmas have been proved, we can prove the theorem described above.

Lemma 1.1: For a connected RPMG with two MAVs, $rank(\mathcal{O}) = 3$.

Proof: Using Lie Derivatives of the measurement function given in Equation (5) for the relative bearing between two MAVs and the prediction step described in Equations (2-4), we obtain a non-linear observability matrix of

$$\mathcal{O}_{2MAVs} = \begin{bmatrix} H_1 & -1 & -H_1 & 0 \\ H_2 & H_1F_1 & -H_2 & -H_1F_2 \\ H_3 & (H_2+H_1)F_1 & -H_3 & -(H_2+H_1)F_2 \end{bmatrix}, \tag{23}$$

where

$$\begin{aligned} H_1 &= \begin{bmatrix} -a & b \end{bmatrix} \\ a &= \frac{y_1-y_2}{R^2} \\ b &= \frac{x_1-x_2}{R^2} \\ R^2 &= (x_1-x_2)^2+(y_1-y_2)^2 \\ e &= \begin{bmatrix} -1 & 0 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} H_2 &= J_1^T(f_1(x_1)-f_2(x_2)) \\ J_1 &= \frac{\partial H}{\partial X} = \begin{bmatrix} -2ab & -(a^2-b^2) \\ (a^2-b^2) & -2ab \end{bmatrix} \\ F_1(\psi_1) &= \begin{bmatrix} -V_1\sin\psi_1 \\ V_1\cos\psi_1 \end{bmatrix} \\ F_2(\psi_2) &= \begin{bmatrix} -V_2\sin\psi_2 \\ V_2\cos\psi_2 \end{bmatrix} \\ f_1(x_1) &= \begin{bmatrix} V_1\cos\psi_1 \\ V_1\sin\psi_1 \end{bmatrix} \\ f_2(x_2) &= \begin{bmatrix} V_2\cos\psi_2 \\ V_2\sin\psi_2 \end{bmatrix} \end{aligned}$$

, and

$$H_3 = \begin{bmatrix} f_1^T(x_1) & -f_2^T(x_2) \end{bmatrix} \begin{bmatrix} -2a & 2a \\ -2b & 2b \end{bmatrix} (a-b)^2$$

To determine which linear combinations of states are observable/unobservable in this matrix, we converted the matrix into reduced row echelon form (RREF), yielding

$$RREF(\mathcal{O}_{2MAVs}) = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & y_1 - y_2 \\ 0 & 1 & 0 & 0 & -1 & x_2 - x_1 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix} \tag{24}$$

Note that this same observability matrix is achieved whether the measurement is uni-directional (MAV i only observes MAV j, but MAV j does not observe MAV i) or bi-directional (MAVs i and j both obtain relative bearing measurements of each other). As is obvious from the RREF matrix, the observability matrix has rank 3, proving Lemma 1.1.

Lemma 1.2: For a RPMG which is a two-level tree with n nodes (e.g. Figure 3), the rank of the nonlinear observability matrix is $3(n-1)$.

Proof: If the RPMG is a two-level tree with n nodes, then one node is at the first level, and the remaining $n-1$ nodes are at the second level. As a tree, the node on level one must have a direct connection to every node on level 2, and those are the only connections in the tree. Therefore, there are $n-1$ edges in the graph.

To determine the observability matrix for this graph, we can modify the observability matrix from Equation (23) to include multiple edges. First, we modify the ordering of the observability matrix so that the effects on node 1 are stored in the last 3 columns of $\mathcal{O}$. The edge from node 1 (the root of the tree) to node j ($j \in (2 \dots n)$) will introduce rows into the observability matrix of the form

$$\mathcal{O}_j = \left[\begin{array}{c|cc|c|cc} & H_1 & -1 & & -H_1 & 0 \\ \mathbf{0}_{3\times3(j-1)} & H_2 & H_1 F_1 & \mathbf{0}_{3\times3(n-j-1)} & -H_2 & -H_1 F_2 \\ & H_3 & (H_2 + H_1) F_1 & & -H_3 & -(H_2 + H_1) F_2 \end{array} \right]. \tag{25}$$

Note that this equation is identical to Equation (24) except that columns of zeros have been added to the matrix for the other nodes besides j and 1 in the RPMG. For each node j in the graph, three new rows will be introduced into $\mathcal{O}$. These rows are linearly independent of the other rows due to the placement of the non-zero components in different columns of the observability matrix. Therefore, the total rank of this matrix will be $3(n-1)$, proving Lemma 1.2.

Lemma 1.3: For a connected RPMG with three vertices, the rank of the observability matrix is 6 and is unchanged regardless of the specific edges used to make the graph connected.

Proof: Using the measurement and prediction functions described in Section II, we computed the observability matrix for the four connected RPMGs shown in Figure 2. We name the observability matrix for each subfigure $\mathcal{O}_a$, $\mathcal{O}_b$, $\mathcal{O}_c$, and $\mathcal{O}_d$ respectively. To analyze these matrices, they were converted into RREF. The RREF for the four different graphs are of the form:

$$RREF(\mathcal{O}) \quad = \quad \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & a \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & b \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & c \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & d \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix},$$

where a, b, c, and d are numbers or expressions depending on the specific measurements taken. While these numbers may be different, because the general form of the matrices are the same and there are six independent rows, we have

$$rank(\mathcal{O}_a) = rank(\mathcal{O}_b) = rank(\mathcal{O}_c) = rank(\mathcal{O}_d) = 6, \tag{26}$$

proving Lemma 1.3.

Lemma 1.4: A connected RPMG with n nodes has an equivalent observability matrix with a two-level tree RPMG with n nodes.

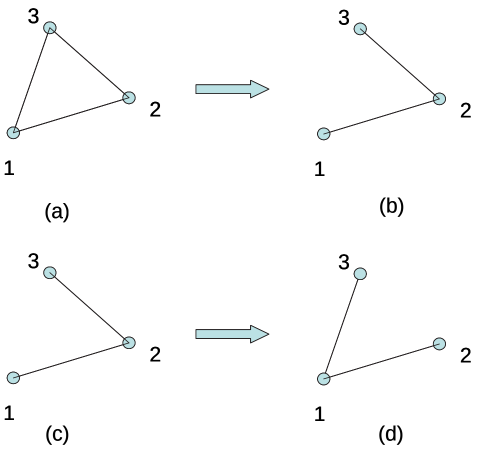

Figure 2. The observability conditions between these four possible configurations of a connected, 3-vertex RPMG are identical. In particular, we can perform redundant edge removal (the transition from a to b) or a side exchange (the transition from c to d) and not affect the observability of the system.

Proof: Using Lemma 1.3, we can modify an arbitrary RPMG and create a different RPMG that has an observability matrix of the same rank. To prove Lemma 1.4, we propose an algorithm that can convert an arbitrary connected graph to a two-level tree *without* modifying the observability rank properties of the system. We propose using two steps, illustrated in Figure 2: redundant edge removal (Figure 2(a) to (b)) and side exchanges (Figure 2(c) to (d)). Using Lemma 1.3, we know that performing either of these steps does not affect the rank of the observability matrix.

Our proposed algorithm is as follows:

1. Randomly select a node to be the root node of the two-level tree.
2. Find every node with a minimum distance (in number of edges traversed) of two from the root node.
3. For every node with minimum distance 2, perform a side exchange to make the minimum distance from the root node to that node equal to 1.
4. Repeat steps 2 and 3 until step 3 returns the empty set.
5. For every two nodes i and j that are adjacent to the root node, check for a edge which connects those two nodes. If an edge is present, remove it.

An illustration of this algorithm is shown in Figure 3.

This algorithm works due to the fact that the original RPMG is connected. If the graph is connected, every node can be placed in one of three distinct sets of nodes: (1) the root node, (2) the set of nodes with minimum distance 1 from the root node, and (3) the set of nodes with a *finite* distance greater than 1 from the root node. Each time a side exchange step occurs, it has two effects. First, it moves a single node from the greater than 1 distance set into the distance 1 set. Second, it decreases the minimum distance of any nodes which were connected to the root node through the node that was removed from the set, decreasing the minimum distance to these nodes by 1. Because the distance to all nodes at the beginning was finite, this algorithm will eventually move all nodes to distance 1 away from the root node, making it a two-level tree. Using Lemma 1.3, we know that each step this algorithm takes does not affect the rank of the observability matrix of the system. Therefore, this algorithm proves Lemma 1.4 by example.

Using Lemma 1.4, we can convert an arbitrary connected graph to a graph with equivalent observability rank that is a two-level tree. Using Lemma 1.2, we know this equivalent graph has an observability matrix with rank $3(n-1)$, proving Theorem 1.

Theorem 2: A connected RPMG can achieve full observability ($rank(\mathcal{O}) = 3n$) if two or more landmarks of known location are observed anywhere within the system.

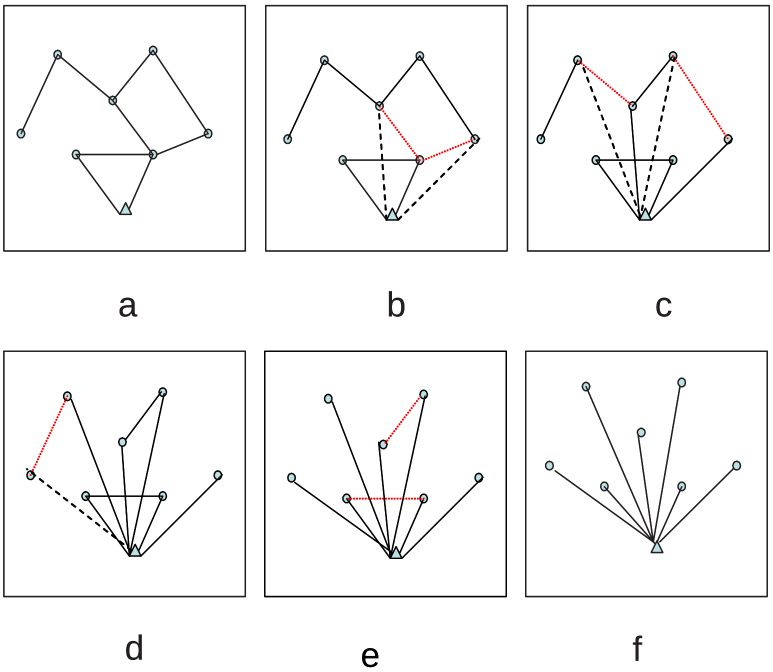

Figure 3. An example of converting an arbitrary, connected RPMG with a two-level tree as described in Lemma 1.4. The subfigures correspond with: (a) the original RPMG, (b) Side exchanges to bring in two nodes distance 2 from the root node, (c) 2 more side exchanges, (d) the final side exchange, resulting in a graph where every "non-root" node is distance 1 away from the root, (e) the removal of any redundant edges and (f) the final, two-level tree.

Proof: To prove this theorem we first analyze the effects of adding a single landmark of known location to the system. This is followed by a brief analysis of adding more landmarks to the system.

Lemma 2.1: The first landmark added to the system increases the rank of the observability matrix by 2.

Proof: Let MAV X_i in RPMG connected to a known landmark X_{lk}. Let η_{ik} be the bearing measurement between them. The zero order Lie Derivative is given as

$$L_f^0(h) = \eta_{ik} = \tan^{-1}\frac{y_k - y_i}{x_k - x_i} - \psi_i \tag{27}$$

$$\frac{\partial L_f^0(h)}{\partial X_i} = \begin{bmatrix} -a & b & -1 \end{bmatrix} \tag{28}$$

where $a = \frac{y_i - y_k}{R^2}$, $b = \frac{x_i - x_k}{R^2}$, and $R^2 = (x_i - x_k)^2 + (y_i - y_k)^2$. The first order Lie Derivative is:

$$L_f^1(h) = \frac{\partial \begin{bmatrix} -a & b & -1 \end{bmatrix}}{\partial X_i} f_i(X_i) \tag{29}$$

$$\frac{\partial L_f^1(h)}{\partial X_i} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \end{bmatrix} \tag{30}$$

where $H_{11} = f_{xi}^T \begin{bmatrix} 2ab \\ a^2 - b^2 \end{bmatrix}$, $H_{12} = f_{xi}^T \begin{bmatrix} a^2 - b^2 \\ 2ab \end{bmatrix}$, $H_{13} = \begin{bmatrix} -a & b \end{bmatrix} \begin{bmatrix} -V1 \sin\psi_1 \\ V1 \cos\psi_1 \end{bmatrix}$, and $f_{xi}^T = [V_i \cos\psi_i \quad V_i \sin\psi_i]$. The second order Lie Derivative is a scale factor of the first order Lie Derivative, adding no new information to the Observability matrix. Therefore, the rows

$$\left[\begin{array}{c|ccc|c} \mathbf{0}_{1\times(j-1)} & -a & b & -1 & \mathbf{0}_{1\times(n-j)} \\ \mathbf{0}_{1\times(j-1)} & H_{11} & H_{12} & H_{13} & \mathbf{0}_{1\times(n-j)} \end{array}\right] \tag{31}$$

will be added to the observability matrix of the entire system. Because these rows are located in only a single MAV's column, and all rows due to MAV to MAV measurements have vectors spanning two MAVs' columns (see Equation (25)), this measurement adds 2 to the rank of the observability matrix, proving Lemma 2.1.

Theorem 2 Proof: Using the proof of Lemma 2.1, we know that each landmark has the potential to add 2 to the rank of the observability matrix. However, once a single landmark has been added, only 1 row with some linearly independent components need to be added to $\mathcal{O}$ to achieve full rank. As each landmark has two rows of potentially independent information, adding a second landmark is generally sufficient.

Note that there are some degenerate conditions when adding a second landmark does not lead to a full rank observability matrix. For example, if a single MAV and two landmarks are co-linear, and that MAV is the only one observing those landmarks, then the system is still not observable. When different MAVs are observing the two landmarks, a degeneracy can exist, but it is an involved function of the landmark locations, the MAV locations and their relative measurements. Understanding the physical configurations that are degenerate will be the focus of future work. However, in the multiple simulations we have run, the degenerate condition appears to be rare when different MAVs are observing the two landmarks. In addition, if more than two landmarks are present, the probability of the system being fully observable is also increased.

IV. Distributed Decentralized Cooperative Navigation

In the prior section, we proved sufficient conditions for drift-free navigation state estimation in a cooperative navigation system. Unfortunately, the system as presented assumes that a centralized estimator is used to determine the estimates of all states. In this section we present a distributed and decentralized version of the estimator using the Information Filter.[16] We begin by converting our Kalman Filter estimator to the Information Form. We express this in a decentralized form where every MAV computes the complete state of the system. A distributed form of the cooperative navigation system is then created so that only the local state and those of neighboring MAVs have to be considered at a specific MAV. By distributing the computation in this fashion, the communication requirements of the system are minimized, making the system scalable as the number of nodes increase.

A. Conversion to Decentralized Information Filter

The Information Filter is essentially a Kalman Filter expressed in terms of measures of information about the parameters (state) of interest rather than direct state estimates and their associated covariances. The two key information-analytic variables are the information matrix and the information state vector, where the term information is used according to the definition of Fisher information. The information matrix is equal to the inverse of the covariance matrix $P(k)$ as follows

$$Y(k) = P^{-1}(k). \tag{32}$$

Likewise, the information state vector y_k can be easily derived as the product of the information matrix and the state estimate as follows:

$$\begin{aligned} y(k) &= Y(k)\hat{x}(k) && (33)\\ &= P^{-1}(k)\hat{x}(k) && (34) \end{aligned}$$

At this point, the Information Filter can be obtained by rewriting the Kalman Filter in terms of the information state vector and the information matrix

$$\begin{aligned} \hat{y}(k|k-1) &= Y(k|k-1)f(\hat{x}(k-1|k-1), u(k)) && (35)\\ Y(k|k-1) &= \left[F_k Y^{-1}(k-1|k-1)F_k^T + Q_k\right]^{-1} && (36)\\ \hat{y}(k|k) &= \hat{y}(k|k-1) + \sum_{l=1}^{N}\sum_{j=1}^{M} i_{l,j}(k) && (37) \end{aligned}$$

$$Y(k|k) \quad = \quad Y(k|k-1) + \sum_{l=1}^{N}\sum_{j=1}^{M} I_{l,j}(k), \tag{38}$$

where the N is the number of MAVs in the system and M is the number of known landmarks or other MAVs observed by MAV l. The associated information matrix and state information contribution from local observations are, respectively, given by

$$I_{l,j}(k) \quad = \quad H^T R_{l,j}^{-1}(k) H, \tag{39}$$

$$i_{l,j}(k) \quad = \quad H^T R_{l,j}^{-1}(k)[\upsilon_{l,j}(k) + Hx(k|k-1)]. \tag{40}$$

The vector $\upsilon_{l,j}(k)$ represents the local innovations

$$\upsilon_{l,j}(k) = z_{l,j}(k) - h_{l,j}(\hat{x}(k|k-1)). \tag{41}$$

In decentralized cooperative navigation, each MAV is capable of estimating the global navigation state without need of a central processor.This is enabled by the fact that the information from each MAV is additive (see Equation (37)). Therefore, the information from each node is broadcast to every other node, allowing each node to compute a global state estimate that is equivalent to the results of a centralized filter by simply summing all received information.

While this method for enabling decentralized navigation theoretically sound, it requires significant communication of information between nodes in the network. As the number of MAVs (N) increases, the amount of information that must be communicated will also increase. Therefore, this system is not scalable. Therefore, in next section we detail a distributed system where the communication requirements between a set of nodes remains constant even as N increases.

B. Distributed and Decentralized Cooperative Navigation (DDCN)

To create a distributed filter that yields local estimates of state equivalent to that obtained by a centralized filter, we will use the concept of transformation matrices outlined in [16]. First, we assume that each MAV moves independent of other MAVs such that,

$$\dot{X}_i = f_i(X_i, u_i) \tag{42}$$

and the linearized discrete model is given as,

$$X_i(k+1) = X_i(k) + F_i(k)(X_i(k))T_s \tag{43}$$

As the MAVs move independently the complete system can be written in diagonal form as

$$X(k+1) = \begin{bmatrix} X_1(k+1) \\ \vdots \\ X_N(k+1) \end{bmatrix} = \begin{bmatrix} X_1(k) \\ \vdots \\ X_N(k) \end{bmatrix} + T_s \begin{bmatrix} F_1(k) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & F_N(k) \end{bmatrix} \begin{bmatrix} X_1(k) \\ \vdots \\ X_N(k) \end{bmatrix} \tag{44}$$

Even though the local dynamic models of MAV are independent the measurement model depends on states from other MAVs as the camera measures relative bearing. Therefore, we combine the local state of the MAV i with the navigation state of its M_i neighbors, obtaining the total local state X_{Li}, defined as

$$X_{Li} = [X_i, X_j]_{j=1,\cdots,M_i}. \tag{45}$$

To create a DDCN filter, it is necessary to create a transformation matrix T_i which maps the global state vector to the total local state X_{Li}, yielding

$$X_{Li}(k) = T_i X(k). \tag{46}$$

Note that T_i is of size $3(M_i + 1) \times 3N$, where N is the total number of MAVs in the entire system. The entries of T_i are all zeros except for local 3×3 identity matrices that map the global state variables into the ordering required for the local state vector.

Given, T_i, we can rewrite the local dynamic model as

$$F_i(k) = T_i F(k) * T_i^{\dagger}, \tag{47}$$

where $T_i^{\dagger}$ represents the Moore-Penrose psuedo-inverse of T_i. Because our T_i matrices perform only a movement of data (no linear combinations of states are used), $T_i^{\dagger} = T_i^T$. The measurement model for each MAV can also be rewritten as

$$H_i = HT_i^T. \tag{48}$$

With the local information vector defined, with its prediction and update steps, the only part lacking to create a DDCN system with identical results to a centralized filter is to determine what effect measurements in other MAVs have on the current MAV. First, the information from the other MAV is transformed into the global parameter space using $T_j^{\dagger}$, where j is the index for the other MAV. The information from the global space is then transformed into the current MAV that is performing its estimation using T_i. By defining $T_{ji} = T_i T_j^{\dagger}$, we have the following equations for transferring information between MAVs:

$$\begin{aligned} i_i(z_j(k)) &= T_{ji} i_j(z_j(k)) & (49)\\ I_i(z_j(k)) &= T_{ji} I_j(z_j(k)) T_{ji}^{\dagger}, & (50) \end{aligned}$$

where

$$\begin{aligned} i_j(z_j(k)) &= H_j^T(k) R_j^{\dagger} z_j(k) & (51)\\ I_j(k) &= H_j^T(k) R_j^{\dagger} H_j(k). & (52) \end{aligned}$$

The importance of developing the DDCN system in this fashion is that as the system grows, T_{ji} will often be zero, removing the requirement for communication between nodes j and i. To understand when T_{ji} is zero, we need to remember that T_i is a function of the MAVs that are observed by MAV i. If the set of MAVs observed by MAVs i and j are completely disjoint, then T_{ji} is zero meaning no information needs to flow between nodes j and i. Also note that the procedure we have followed to distribute the Kalman Filter is general, guaranteeing that the state estimates in each MAV are equivalent to what a centralized filter would return. In this scenario, this means that a distributed estimator will return navigation state estimates that do not drift over time.

V. Results

In this section we present simulation results to validate the theory developed in the previous sections. For our simulation, we have placed five MAVs in the environment together with two landmarks of known location. The MAVs are flying in circles and are connected with with the topology shown in Figure 4. The communication topology and RPMG topology are the same and do not change over time. The initial heading of the MAVs is chosen randomly such that $|\psi_i| < \frac{\pi}{2}$. Other important simulation parameters used are:

- $V_i = 10m/sec$, $i = 1, \cdots, 5$
- $\sigma_w = 0.1rad/sec$
- $\sigma_v = 0.1rad$

To construct the DDCN system, we first construct the reduced order local model consisting of local relevant states using Figure 4. For example, the reduced order state and local model of node 1 is given as

$$\begin{aligned} X_{L1} &= [X_1 \ X_2]^T & (53)\\ T_1 &= \begin{bmatrix} I_3 & 0 & 0 & 0 & 0 \\ 0 & I_3 & 0 & 0 & 0 \end{bmatrix} & (54)\\ F_{L1} &= \begin{bmatrix} F_1 & 0 \\ 0 & F_2 \end{bmatrix} & (55) \end{aligned}$$

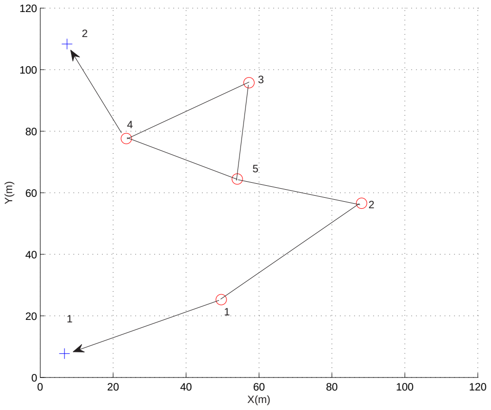

Figure 4. This figure represents the general simulation environment used for our results. The environment contains five MAVs and two land marks, connected as shown in this Figure. The red circles represents the MAVs, the plus signs represents landmarks, and the thin black lines show the measurements between nodes and the communication links between MAVs.

where I_3 is 3×3 identity matrix and F_1 and F_2 are jacobians of MAV 1 and MAV 2 respectively. Similarly we derive the local state and model for the other four nodes.

Our results consist of two main Figures. Both graphs are error plots in X, Y, and ψ for the 5th MAV. In Figure 5, we plot the results of running the DDCN system without any landmarks present. In these plots are three lines. First, we show the error if the IMU from a MAV 5 is used in isolation of the other IMUS (the black line). As is shown, the error from this method is extremely high. However, by simply measuring the location of the MAVs, the error is dramatically decreased (the solid blue lines between the error plots in Figure 5). The second important point to notice from this plot, however, is that the error bounds are increasing over time. This demonstrates the necessity of adding fixed landmarks to eliminate drift over time.

In Figure 6, we have simulated the same scenario but have now included measurements of the two landmarks shown in Figure 4. Note that the error bars in the X and Y location estimate is no longer increasing with time. This is due to the drift-free nature of DDCN when 2 or more landmarks are present in the environment. These results serve as empirical validation of the observability analysis in Section III.

VI. Conclusion

In this paper we have formulated a distributed and decentralized cooperative navigation algorithm for MAVs flying in GPS denied area. The approach uses inter MAV bearing and bearing from landmarks with local IMU data to estimate position and heading of all MAVs using a distributed and decentralized information filter. Internodal transformation matrices used in DDCN estimation explicitly minimizes communication both in terms of message size and number of communication links and the state estimates are equivalent to the fully connected communication topology. Using Lie Derivatives and graph theory we have shown that two different landmarks are sufficient for the full observability of the system if the nodes form a connected graph. We have presented simulation results showing that the estimation error when using DDCN estimation and 2 fixed landmarks does not drift over time.

Acknowledgement

This work funded by AFOSR Young Investigator Award number FA9550-07-1-0167

References

[1] Zufferey, J.-C. and Floreano, D., "Toward 30-gram Autonomous Indoor Aircraft: Vision-based Obstacle Avoidance and Altitude Control," *Proc. IEEE International Conference on Robotics and Automation ICRA 2005*, 2005, pp. 2594–2599.

[2] Kurdila, A., Nechyba, M., Prazenica, R., Dahmen, W., Binev, P., DeVore, R., and Sharpley, R., "Vision-based control of

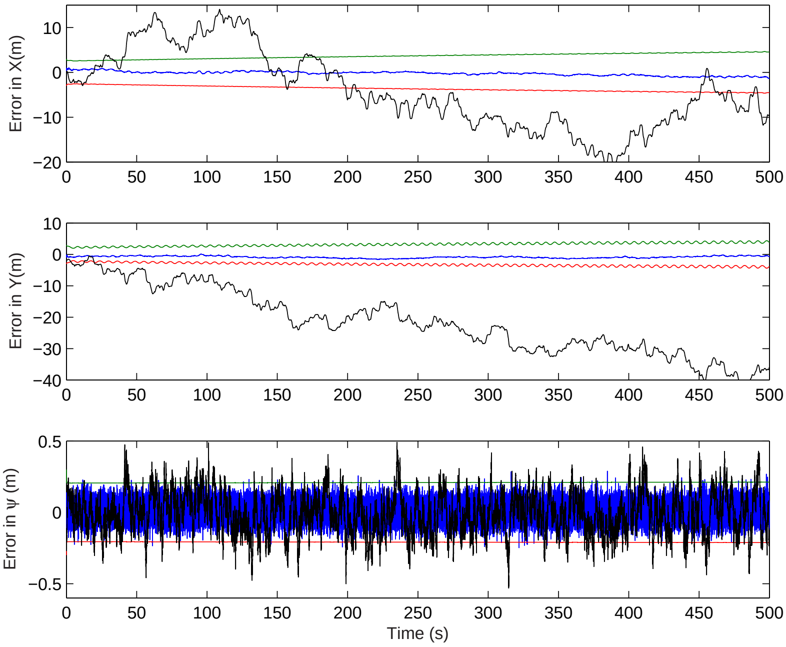

Figure 5. Error plots without any landmark measurement. (1) Black curve represents the error plot for IMU measurement only (no bearing measurement), (2) Blue curve represents the error plot for states estimated by DCNS(IMU and bearing measurement), (3) Green curve represents the $+2\sigma$ bound, and, (4)Red curve represents the -2σ bound.

micro-air-vehicles: progress and problems in estimation," *Proc. CDC Decision and Control 43rd IEEE Conference on*, Vol. 2, 14–17 Dec. 2004, pp. 1635–1642.

[3] Kurazume, R., Nagata, S., and Hirose, S., "Cooperative positioning with multiple robots," *Proc. IEEE International Conference on Robotics and Automation*, 8–13 May 1994, pp. 1250–1257.

[4] Kurazume, R., Hirose, S., Nagata, S., and Sashida, N., "Study on cooperative positioning system (basic principle and measurement experiment)," *Proc. IEEE International Conference on Robotics and Automation*, Vol. 2, 22–28 April 1996, pp. 1421–1426.

[5] Kurazume, R. and Hirose, S., "Study on cooperative positioning system: optimum moving strategies for CPS-III," *Proc. IEEE International Conference on Robotics and Automation*, Vol. 4, 16–20 May 1998, pp. 2896–2903.

[6] Spletzer, J., Das, A., Fierro, R., Taylor, C., Kumar, V., and Ostrowski, J., "Cooperative localization and control for multi-robot manipulation," *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 2, 2001, pp. 631–636 vol.2.

[7] Rekleitis, I. M., Dudek, G., and Milios, E. E., "Multi-robot cooperative localization: a study of trade-offs between efficiency and accuracy," *Proc. IEEE/RSJ International Conference on Intelligent Robots and System*, Vol. 3, 30 Sept.–5 Oct. 2002, pp. 2690–2695.

[8] Mourikis, A. and Roumeliotis, S., "Analysis of positioning uncertainty in reconfigurable networks of heterogeneous mobile robots," *Proc. IEEE International Conference on Robotics and Automation ICRA '04*, Vol. 1, 2004, pp. 572–579 Vol.1.

[9] Mourikis, A. and Roumeliotis, S., "Performance analysis of multirobot Cooperative localization," Vol. 22, No. 4, 2006, pp. 666–681.

[10] Nebot, P., Gomez, D., and Cervera, E., "Agents for cooperative heterogeneous mobile robotics: a case study," *Proc. IEEE International Conference on Systems, Man and Cybernetics*, Vol. 1, 5–8 Oct. 2003, pp. 557–562.

[11] Sanderson, A. C., "A Distributed Algorithm for Cooperative Navigation among Multiple Mobile Robots," *Advanced Robotics*, Vol. 12, 1998, pp. 335–349.

[12] Roumeliotis, S. I. and Bekey, G. A., "Collective localization: a distributed Kalman filter approach to localization of groups of mobile robots," *Proc. IEEE International Conference on Robotics and Automation ICRA '00*, Vol. 3, 24–28 April 2000, pp. 2958–2965.

[13] Sharma, R. and Taylor, C., "Cooperative navigation of MAVs In GPS denied areas," *Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems MFI 2008*, 20–22 Aug. 2008, pp. 481–486.

[14] Thomas Kailath, A. H. S. and Hassibi, B., *Linear Estimation*, Prentice Hal, 2001.

[15] Hermann, R. and Krener, A., "Nonlinear controllability and observability," Vol. 22, No. 5, Oct 1977, pp. 728–740.

[16] Mutambra, A. G., *Decentralized Estimation and Control for Multisensor Systems*, CRC Press LLC, 1998.

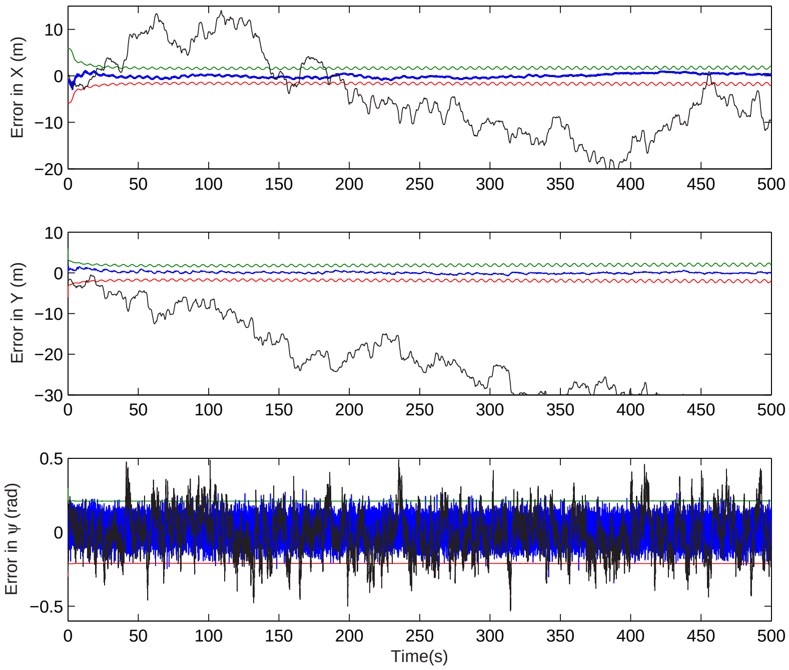

Figure 6. Error plots with two known landmark measurement. (1) Black curve represents the error plot for IMU measurement only (no bearing measurement), (2) Blue curve represents the error plot for states estimated by DCNS(IMU and bearing measurement), (3) Green curve represents the +2σ bound, and, (4)Red curve represents the -2σ bound.