GRAPH THEORETIC FRAMEWORK BASED COOPERATIVE CONTROL AND ESTIMATION OF MULTIPLE UAVS FOR TARGET TRACKING

by

MOUSUMI AHMED

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON
May 2012

To my parents Mostaque Ahmed, and Daulat Ara Ahmed, my sister Ferdous Ara Ahmed and my husband Md Mostafijur Rahman who always been a source of my inspiration and who pushed me always to make my dreams come true!

ACKNOWLEDGEMENTS

I would like to thank my supervising professor and mentor Dr. Kamesh Subbarao for his continued guidance and support throughout the years of this dissertation. His continuous encouragement, invaluable insight, and pursuit of excellence in research have always inspired me to think in many different ways to find the best solution. Dr. Subbarao has truly been an inspirational person in my life over the years; His academic lessons on nonlinear control, and Kalman filter come with great appreciation, and have helped to develop my fundamental background. I am very grateful to have had the opportunity to work with him.

I would like to thank my PhD dissertation committee members Dr. David A. Hullender, Dr. Alan Bowling, Dr. Gian Luca Mariottini, and Dr. Panayiotis S. Shiakolas for being part of my dissertation committee and providing invaluable inputs and suggestions to improve the quality of this dissertation. I thank Dr. Hullender for providing me much of basic understanding of dynamic system modeling, and stochastic random processes. Dr. Hullender's class-notes and tools have further helped me with the estimation problem in my dissertation. I appreciate Dr. Mariottini for taking time out to help with my dissertation and have learned much from his computer vision and control course. I am indebted to Dr. Bowling for sharing a different approach to solve the dynamics problem in his advanced and computational dynamics class, and it will definitely continue to assist me in my academic career. I would also like to give a special thanks to Dr. Frank Lewis for teaching me the fundamentals of distributed decision and cooperative control which have significantly helped promote critical thinking in my dissertation.

I would also like to thank the College of Engineering, and the Department of Mechanical and Aerospace Engineering (MAE) for supporting me with the STEM doctoral fellowship and scholarships to carry out my studies at UTA without any difficulties. I must thank the current and past graduate advisors Dr. Seiichi Nomura, Dr. Roger Goolsby, and Dr. Albert Y. Tong for their contributions to my professional planning and completing my dissertation in a timely manner. In particular, I would like to thank Dr. Nancy Michael and Dr. Rob Taylor for giving me the privilege of working with them as a teaching assistant as well as offering their support over the years; Dr. Tiernan for having such a good time with her during summer camp and working for high school students. I would also like to thank all administrative staff members of the MAE department especially Debi Barton, and Lanie Gordon for always being there to provide help, encouragement, and friendship. I am very thankful to Zonta International for awarding me with the Amelia Earhart Fellowship.

I am specially grateful to my current and past lab-mates including Pavan Kumar Nuthi, Ghassan Atmeh, Laura Henderson, Pankaj Sarda, Alok Ashok Rege, Marcin Brodecki, Divya Bhatia, Erik De Vries for always being there to share and discuss ideas, for their friendship and the times we would go out for lunch together; and all my close friends at UTA including Pran, Nipunika, Nusrat, Sadik, Ishtiaq, Kibria, Ferdous and aunty Shampa Lodi for always being there as a family; and Kyle Crumpton for his friendship, encouragement and the daily inquiry on the progress in my dissertation writing, and his help in reviewing my dissertation.

Finally, I am greatly indebted to my parents, my sister, and my extended family members for their sacrifice, love and encouragement throughout the years of my studies. I am especially indebted to Mostafij; without your love and support, none of this would have been possible.

April 5, 2012

ABSTRACT

GRAPH THEORETIC FRAMEWORK BASED COOPERATIVE CONTROL
AND ESTIMATION OF MULTIPLE UAVS
FOR TARGET TRACKING

MOUSUMI AHMED, Ph.D.

The University of Texas at Arlington, 2012

Supervising Professor: Kamesh Subbarao

Designing the control technique for nonlinear dynamic systems is a significant challenge. Approaches to designing a nonlinear controller are studied and an extensive study on backstepping based technique is performed in this research with the purpose of tracking a moving target autonomously. Our main motivation is to explore the controller for cooperative and coordinating unmanned vehicles in a target tracking application.

To start with, a general theoretical framework for target tracking is studied and a controller in three dimensional environment for a single UAV is designed. This research is primarily focused on finding a generalized method which can be applied to track almost any reference trajectory. The backstepping technique is employed to derive the controller for a simplified UAV kinematic model. This controller can compute three autopilot modes i.e. velocity, ground heading (or course angle), and flight path angle for tracking the unmanned vehicle. Numerical implementation is performed in

MATLAB with the assumption of having perfect and full state information of the target to investigate the accuracy of the proposed controller. This controller is then frozen for the multi-vehicle problem.

Distributed or decentralized cooperative control is discussed in the context of multi-agent systems. A consensus based cooperative control is studied; such consensus based control problem can be viewed from the algebraic graph theory concepts. The communication structure between the UAVs is represented by the dynamic graph where UAVs are represented by the nodes and the communication links are represented by the edges. The previously designed controller is augmented to account for the group to obtain consensus based on their communication. A theoretical development of the controller for the cooperative group of UAVs is presented and the simulation results for different communication topologies are shown. This research also investigates the cases where the communication topology switches to a different topology over particular time instants. Lyapunov analysis is performed to show stability in all cases.

Another important aspect of this dissertation research is to implement the controller for the case, where perfect or full state information is not available. This necessitates the design of an estimator to estimate the system state. A nonlinear estimator, Extended Kalman Filter (EKF) is first developed for target tracking with a single UAV. The uncertainties involved with the measurement model and dynamics model are considered as zero mean Gaussian noises with some known covariances. The measurements of the full state of the target are not available and only the range, elevation, and azimuth angle are available from an onboard seeker sensor. A separate EKF is designed to estimate the UAV's own state where the state measurement is

available through on-board sensors. The controller computes the three control commands based on the estimated states of target and its own states. Estimation based control laws is also implemented for colored noise measurement uncertainties, and the controller performance is shown with the simulation results.

The estimation based control approach is then extended for the cooperative target tracking case. The target information is available to the network and a separate estimator is used to estimate target states. All of the UAVs in the network apply the same control law and the only difference is that each UAV updates the commands according to their connection. The simulation is performed for both cases of fixed and time varying communication topology. Monte Carlo simulation is also performed with different sample noises to investigate the performance of the estimator. The proposed technique is shown to be simple and robust to noisy environments.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Unmanned Vehicles are considered to be a viable alternative option to replace the repetitive or life threatening tasks which would otherwise be performed by a human-piloted vehicle. The existing form of unmanned vehicles include Unmanned Aerial Vehicles (UAVs), Unmanned Underwater Vehicles (UUVs), and Unmanned Ground Vehicles (UGVs). There are many potential applications of using autonomous unmanned vehicles in both military and commercial sectors such as surveillance, reconnaissance, rescue and fire protection, underwater exploration, space exploration, atmospheric data acquisition, homeland security such as coast and border patrol. Modern UAVs are using in many applications such as the U.S. Department of Energy uses Altus UAVs in climate research to studying cloud interaction with the earths solar and thermal radiation to heat or cool the earth [1], Aerosonde UAVs and the UAVs of the same family are used for meteorological, environmental observations, and surveillance operation, and Global Hawk, Predator, and Dark Star are used for military missions in many applications. The role of UAVs has remarkably increased over the last decade due to its technological advances of on-board level of autonomy. The level of autonomy decides the potential use of UAVs in practical scenarios and there is critical need for additional research and development to maximize its capabilities. Achieving fully autonomous UAVs is still a big challenge and it involves both theoretical and practical challenges.

In this study, a theoretical framework and realistic implementation of $3D$ controller is performed for a cooperating group of Unmanned Aerial Vehicles (UAVs)

using an advanced nonlinear control technique (backstepping). Designing control algorithms in $3D$ environment accounts for many limitations and complexities for real systems. This research also accounts for the system and measurements being corrupted by uncertainties or the cases with limited state information knowledge about the target is known which necessitates to develop the estimation algorithms and achieve the robust performance of controller even under noisy environments. An extensive and thorough study is performed in this chapter which includes the current state, and existing technologies for UAVs, and the methods studied in the literature related to path following and guidance problem, cooperative guidance and control approach for multiple UAVs, measurement uncertainties involved with UAVs, and the available estimation techniques studied for state estimation of UAVs to appropriately handle these uncertainties.

1.1 General Description of Guidance Laws for UAVs for Path Following and Target Tracking

Over the recent years many approaches have been studied by researchers for UAV guidance and control systems. Solving such problems for nonlinear systems is much more challenging as compared to linear systems, and still a less explored area for UAVs. Lyapunov vector field approach is one of the well known guidance techniques for UAVs studied by several researchers [2–9] in recent years. In [2, 3], vector fields are constructed based on the Lyapunov stability theory and with the use of these vector fields circular loiter attractors are generated for path tracking. In [4], the control laws are developed for a group of unmanned aircrafts based on the Lyapunov guidance vector field approach in coordinated standoff tracking application. In [6], the vector field approach is used for path following of Miniature Air Vehicles (MAVs) in 2D, where the velocity and altitude of fixed-wing MAV is kept constant. The

control laws are developed which generate course commands for straight and circular path following of MAVs. A similar approach is then employed in [7] for curved path following of MAVs. This work demonstrates the practical limitation on choosing higher ground speed which significantly increases the turning radius and makes it difficult to follow any arbitrary curved path. An appropriate choice on the feedback gains is necessary for this particular scenarios.

The principle of Voronoi diagram is used to develop a path planner in [5] which generates way-point paths through the obstacles and a trajectory smoother is used to smooth dynamically through the corners of the path. A combination of tangent vector field guidance, and Lyapunov vector field guidance is proposed in [10] to track a target by cooperative UAVs where the problem is solved as an optimal control problem.

Proportional Navigation (PN) based guidance law has been applied for large classes of engagement geometries. Different variants of PN based guidance laws can be found the literature such as Ideal Proportional Navigation (IPN), Pure Proportional Navigation (PPN), True Proportional Navigation (TPN), and Optimal Guidance Law -an of which perform well for non-maneuvering targets. In TPN, the commanded acceleration is applied perpendicular to Line of Sight (LOS). The closed-form solution using TPN law was first proposed in 11 for the the case with constant velocity target in a plane. It showed that a missile intercepts a target if the initial conditions lie within a determined circle. In PPN, the commanded acceleration is applied perpendicular to the velocity vector of the missile. A unified approach is proposed in [12] where all these different guidance laws are special kinds of the proposed guidance laws.

A proportional based guidance, and dynamic inversion based control laws are proposed in [13] for highly maneuvering trajectory-tracking UAV. In [14], a novel algorithm is proposed for 3D path generation and tracking of UAVs. The path gen-

eration is developed based on Dubins ideas and the tracking algorithm is developed based on the line of sight algorithm. Vehicle kinematics is represented based on the Serret-Frenet formulation expressed in terms of path parameters and designed the guidance law for UAVs to follow the desired path by simultaneously correcting the course angle and course track angle to zero. A nonlinear H_∞ missile guidance law is developed in [15] based on the dissipative theory, the guidance laws works better as compared to APN when the target acceleration is unknown or poorly estimated.

Feedback linearization is also used to design the guidance scheme for nonlinear systems [16–18]. A guidance scheme is developed via feedback linearization technique for pursuer evader engagement scenario in [18]. The derivative of line of sight rate along both pitch and yaw direction are forced to zero by feeding back the line of sight and rate of line of sight which can be viewed as a regulator problem. A novel composite based guidance law is proposed in [17] where nonlinear inversion based guidance law for flight angles during initial phase of engagement, and a feedback linearization based guidance law using LOS rates for the pursuer to navigate along LOS till achieve interception during terminal phase of flight. For both cases, it requires less pursuer acceleration than PN based guidance law.

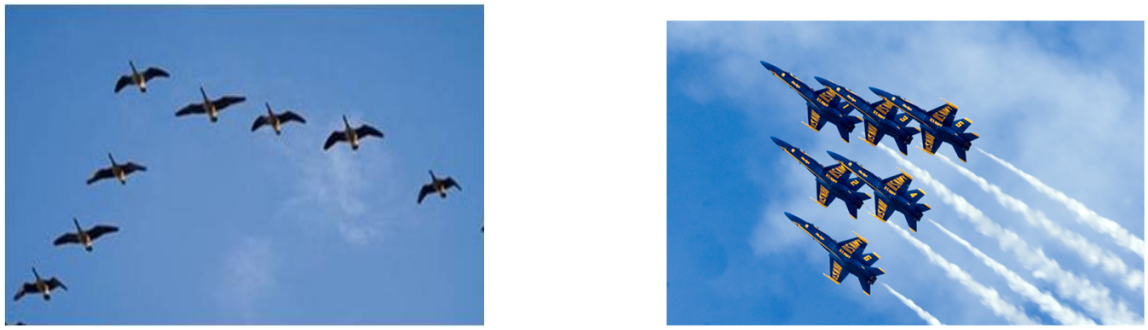
The backstepping technique pioneered by Kokotovic [19] is another nonlinear control technique recently studied by researchers [20–22]. Backstepping technique designs the controller by synthesizing the appropriate Lyapunov function. It starts with the desired virtual control inputs and then backs out the controller recursively to obtain the actual control input. As compared with the feedback linearization technique, it offers more flexibility and the form of the designed closed loop system does not necessarily need to be linear. The control law to compute the roll rate command for a small UAV in $2-D$ is designed based on the backstepping approach in [21]. The backstepping approach is also studied in [20] for nonlinear trajectory tracking

with fixed wing UAVs in $2-D$ case. In [22], the guidance law synthesizes the high-gain backstepping and variable structure method for missile-target $2-D$ engagement problem. In [23], control laws for Autonomous Underwater Vehicle (AUV) are developed based on backstepping technique. [23] also uses similar approach to design the path following controller for underactuated surface vessel. The ship kinematics are expressed in terms of the path parameter based on Serret-Frenet formulation. Backstepping techniques applied to nonlinear $3-D$ problem are relatively less studied so far. We apply the backstepping technique to design the $3-D$ nonlinear guidance controller for UAVs.

1.2 Cooperative Path following and Target Tracking

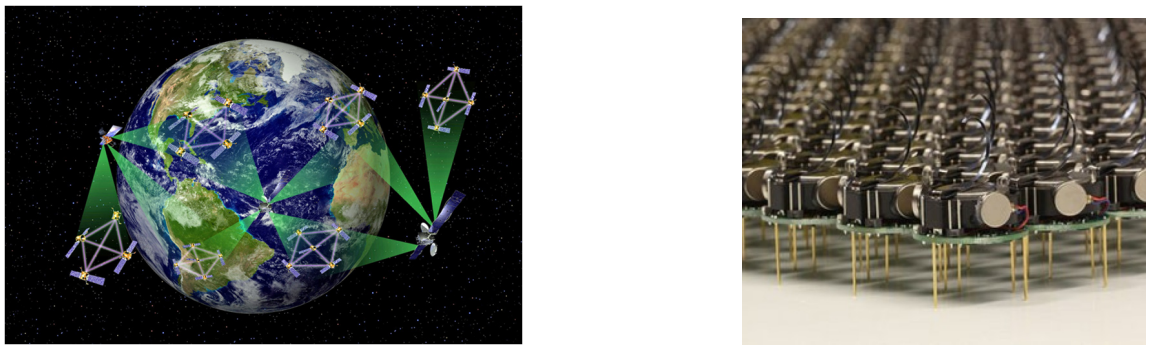
The coordinated and cooperative control of multi-agent systems is increasingly receiving attention in many applications since it offers significant advantages over a single system in terms of cost, efficiency, and robustness. Applications include formation control of robots [24–26], aerospace vehicles [27–30], constellation of satellites [31, underwater vehicles [32–34], distributed sensor networks [35], flocking [36, 37], swarming [38] etc. Some examples are shown in Fig. 1.1, and Fig. 1.2.

There are two main broad categories of information sharing in cooperative path planning application: 1) centralized information - where each agent sends information to centralized agent which plan all waypoints for all agents and 2) Distributed information sharing -where each agents shares the information between the neighboring agents and reach to a common agreement which is called consensus. Distributed and decentralized control strategies are used in many scenarios to obtain a common objective in a cooperative way. One of the prerequisites is cooperation and coordination of the agents. The consensus based strategies for cooperative control of multiple agents has been widely discussed in [39–43]. Algebraic graph theoretic tools



(a) (b)

Figure 1.1. Multi-agent systems (a) Flock of birds (Courtesy: Nature), and (b) Blue Angels (Courtesy: US NAVY).



(a) (b)

Figure 1.2. Multi-agent systems (a) F-6 System (Courtesy: DARPA), and (b) Swarm of Robots.

are used to describe the inter vehicle communication. Graph Laplacian matrices are used to define the communication structure, known as algebraic graph connectivity. The communication between the agents or vehicles may be fixed or time varying. In [40], the consensus algorithms are developed in the context of linear continuous and discrete-time vehicle systems. A method for the decentralized information flow between the vehicles is developed in [42] and formation stability is also proved.

In systems and control problems, the properties of Graph Laplacian matrix are very important, notably the second smallest eigenvalue of Laplacian is a critical parameter for systems stability requirements. [44] investigated to obtain the best posi-

tional configuration of the network so to maximize this eigenvalue for state-dependent Laplacian.

In real missions, the communication exchange between the agents may be dynamic for example, the communication between the vehicles may drop or new communication links may appear. The control strategies should be developed so they adapt to account for the effects of such dynamically changing communication topologies. The communication networks with switching topologies and time delays are addressed in [41] and introduce disagreement function into the Lyapunov function and showed the convergence analysis. In [45], the cooperative control laws for the case of static and time-varying communication topology between the neighboring agents are proposed. They also investigate the quantized communication between the agents such as uniform and logarithmic quantization and find the convergence characteristics. The concept of graph theory and Lyapunov stability is used for the formation control of nonholonomic mobile robots in [24]. A decentralized control methodology is developed in [33] for stable synchronization of rigid body networks. State feedback and output feedback controllers are proposed which ensure the team to move asymptotically along the desired trajectory.

However, most of these works are applied to linear problems and there exist significant complexities when applied to nonlinear problems. Owing to the promising application of cooperative multi-agent system in many scenarios, we investigate and develop consensus based control strategies for a cooperative group of multiple UAVs with nonlinear dynamics and limited state information.

1.3 Measurement Uncertainties

There are many uncertainties involved with UAV system such as wind gusts, measurement noises, weight, center of gravity locations and airspeed variations, and

uncertainty in UAV model parameters (for example stability and control derivatives). The accuracy of measurement sensors is very important factor which directly influences the controller tracking performance. The true state information about the target is not possible to be known precisely in practice. There are two situation arises and need to be considered for successful extraction of available information: 1) the measurements are corrupted by external disturbances, and 2) limited no. of sensors are available or restricted by limited bandwidth. Estimation tools are often required for state estimation of the system when the system or measurements are corrupted by noise and when complete information is unavailable. Successful estimation depends on how effectively it can extract the information from noisy measurements. Estimation techniques have been widely studied by numerous researchers in many applications such as vision based estimation and target tracking [46], multi-agent consensus problem [47–50], aerodynamic parameter estimation [51], relative position and attitude estimation for docking mission of spacecraft [52], and orbital rendezvous [53].

A rich literature on different types of estimation techniques can be found such as: consensus filter and distributed Kalman filter used for sensors network [35,54], particle filter in ballistic target tracking [44], Unscented Kalman Filter (UKF) and Distributed Particle Filter (DPF) for autonomous navigation of UAVs [55], Monte Carlo filters used in target tracking and wireless communications [56], and Extended Kalman Filter (EKF) used for vision based state estimation and target tracking for cooperative UAVs [46]. The estimator we used in this paper is an Extended Kalman Filter (EKF), a widely used nonlinear Kalman filter which shows reasonably good performance for Gaussian noise statistics. It provides approximately an optimum Kalman filter which estimates the state closer to the true value. The detailed mathematical description of this type of EKF can be found in [49]. The nonlinear dynamics is usually linearized

about some known priori state and Kalman filter equations are developed to estimate the state with reasonable accuracy.

For ballistic target tracking [44], a particle filter based estimation is designed under radar glint noise, which arises from interference between two or more reflections from the target surface and results to noisy angle measurements. The effect of glint noise can be significant when it closes to the target. The characteristics of noise distribution is modeled as non-Gaussian type in this case. A comparative result is shown with the EKF, and it is concluded that the higher accuracy can be obtained with increasing the particle numbers used in the simulation. However, it increases the computational cost significantly, and a therefore a trade-off is necessary in real-time implementation.

An effective controller and estimator design is essential for successful target tracking. In the present study, we design the controller for cooperative UAVs and then combine with the estimation technique (EKF) while adapting the system to measurement uncertainties and finally accomplish the cooperative target tracking.

1.4 Target Tracking by UAVs

Target tracking is one of the important tasks performed by UAVs in many applications. The use of UAVs for commercial applications is still a big challenge complicated by constraints it has during flights to maintain coordination with air traffic control, its effective operation during take off, and landing, and to avoid collisions with other planes in air or other environmental obstacles. In real scenarios, it is not always possible to obtain the full information of the target trajectory. Our primary motivation is to design guidance laws and control algorithms for UAVs so as to continuously track a target for cases with complete or partial knowledge of the

target UAV. We employ a backstepping like technique to design the control laws for UAVs which offers a simplified way to achieve tracking for general defined trajectory.

1.5 Target Tracking by Cooperative UAVs

The benefits of using cooperative UAVs are many, can be used to accomplish tasks in dangerous environments instead of sending out human operators. For multiple UAVs mission, although it requires to have extra sensors, extra communication devices, synthesize information from different sources, and recognize trusted source, and redundancy, a greater efficiency, and accuracy can be achieved if we can utilize and coordinate information effectively between the UAVs. It is very much possible to reduce operational costs by implementing tasks cooperatively and using a coordination by using smaller and cheaper UAVs rather than larger more expensive ones. Using more than one UAV in many cases help us to achieve greater benefits in real missions. We employ cooperating UAVs for applications such as border patrol, search and rescue, surveillance, communications relaying, and mapping of hostile territory.

1.6 Organization of the Dissertation

The ultimate objective of this study is to design and implement control laws for cooperative UAVs and show robust performance in simulation. The overall picture of the proposed approach is shown in Fig. 1.3.

The governing equations of motion considered for Chaser UAVs, and Target UAV are described in Chapter 2. Chapter 2 also includes mathematical preliminaries of control theory, probability, and statistics which will be utilized throughout this dissertation.

The initial approach of this study is to design backstepping like control laws for target tracking with a single UAV. The detailed derivation of the mathematical formulation is described in Chapter 3. Three control laws for speed, flight path angle, and heading angle are designed and the simulation results are shown for the case with complete target information as it is obtained by sensors. A investigation is also performed when only partial target information is known, and an indirected information is available from on board sensors or using on board seeker antenna. An extended Kalman filter is then utilized to extract the information to estimate the full state information of target UAV, and a separate Kalman filter is utilized to estimate its own states if the state measurements are corrupted by uncertainties. The control system is then synthesized using the estimated state information of the target and UAV's own state and we investigate the target tracking performance. We consider two different cases when the measurements are corrupted by: 1) White Noise uncertainties, 2) Colored noise uncertainties which are described in section 3.2, and 3.3, respectively. To include more accurate measurement information available for UAVs, the measurement noise covariance is described as a function of the pseudo range distance between the target and chaser which integrates target information in a way to get better target measurements as it approaches the target. The mathematical formulation is discussed in section 3.4, and the simulation results are shown for measurements with these varying colored noise uncertainties.

In Chapter 4, we discuss the cooperative target tracking problem. For a cooperative group of N UAVs, the control laws for target tracking are designed. The cooperative behavior is highly restricted to the communication structure or information exchange between the UAVs and target information is passed externally into the network by pinning it at least one node of the network. An EKF estimator is designed to estimate the state of the target UAV with the assumption of having only range,

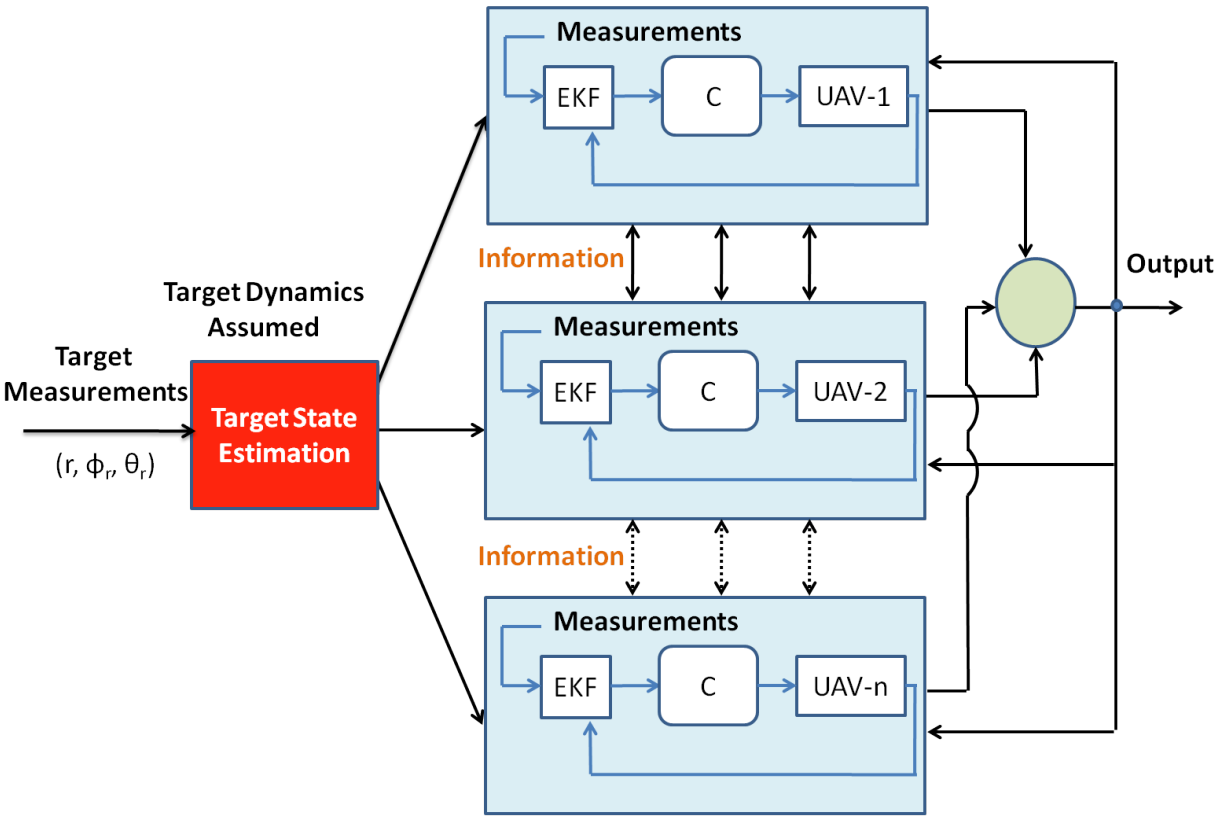


Figure 1.3. Flow Chart of the problem.

azimuth, and elevation angle measurements from a ground based sensor or from on-board seeker antenna. There are N EKF estimators incorporated for N UAVs to estimate their own states. For simplicity, we also ignore the uncertainties involved with the communication information between the UAVs. The mathematical development and implementation is discussed in detail for both complete target information in section 4.1 and partial target information with fixed and switching topologies in section 4.2.

Finally, Chapter 5 summarizes this research findings and concludes with the directions for future work.

1.7 Limitations to the Present Framework

The present study addresses the solution using backstepping-like advanced-nonlinear control technique for target tracking with UAVs. The proposed framework is limited to the assumption of point-mass UAV model. A general framework of the controller is integrated with the navigation system for point-mass UAVs. The full benefits of using backstepping approach can be realized by implementing a fully described UAV model including vehicle geometric and environmental constraints. A practical implementation on real hardware platform is also essential to understand and extend our theoretical innovations and discussions to the next level.

One broad aspect of this approach can be seen from the present study, we can utilize the benefits of using distributed control approach by synthesizing information from the neighboring UAVs to track the target coordinately and cooperatively. In the present study, all UAVs are sharing all state information to the neighboring UAVs. For cooperative case, the sensors involved for sending information to neighboring UAVs are considered to be perfect i.e. without losing any information. For a more realistic case, all UAVs may not share or send same information to other UAVs. A sensor fusion approach can be utilized by fusing information from different sensors with different sensing capabilities to track the moving target. The work presented in this dissertation has not addressed sensor fusion aspect.

The proposed framework can also be easily extended to the development of formation control strategies where multiple UAVs can maintain a specific formation while tracking the target. Finally, the present study brings the ideas of using back-stepping like approach and distributed control algorithms together and proposes a novel architecture for nonlinear systems which can be viewed as a basic foundation for target tracking application in $3D$ space.

1.8 Summary of Contributions

This research aims to develop and verify a cooperative controller and estimator for multiple UAVs in a target tracking application. Vehicle cooperation is constructed based on the graph theoretic approach and backstepping technique is then applied to develop control commands for the guidance controller.

The significant contributions of this dissertation are summarized below:

- Mathematical formulation and numerical implementation of a $3-D$ guidance controller for target tracking with a single UAV (complete target information)
- Mathematical formulation and implementation of estimation based guidance laws, and control for target tracking with a single UAV (Partial Target Information)
- Mathematical formulation and numerical implementation of $3-D$ guidance laws and control for a cooperative group of UAVs for different type communication structure: fixed (time-invariant) and switching (dynamically changing) topologies where complete target information is available
- Mathematical formulation and numerical implementation of estimation based guidance laws, and control for target tracking with a cooperative group of UAVs (Partial Target Information) for fixed and switching topologies

List of Publications:

The publications are listed as below:

Journals

1. Mousumi Ahmed, and Kamesh Subbarao, *"Nonlinear Guidance and Control Laws for* 3*-D Target Tracking applied to Unmanned Aerial Vehicles"*, Journal of Aerospace Engineering, ASCE, February, 2012.

Conferences

1. Mousumi Ahmed, and Kamesh Subbarao, *"Estimation based Cooperative Guidance Controller for 3D Target Tracking with multiple UAVs"*, American Control Conference, Montreal, Canada, June 2012.
2. Kamesh Subbarao, and, Mousumi Ahmed, *"3D Target Tracking by UAVs subject to Measurement Uncertainties",* IEEE Multi-conference on Systems and Control, Denver, Colorado, September, 2011.
3. Mousumi Ahmed, and Kamesh Subbarao, *"Nonlinear Guidance and Consensus for Unmanned Vehicles with Time Varying Connection Topologies",* 49th Aerospace Science and Meeting Exhibit, AIAA, 4-7 January, 2011.
4. Mousumi Ahmed, and Kamesh Subbarao, *"Nonlinear 3-D Trajectory Guidance for Unmanned Aerial Vehicles",* ICARCV (International Conference on Control, Automation, Robotics and Vision), Singapore, 7-10 December, 2010.

CHAPTER 2

MATHEMATICAL DESCRIPTION OF THE SYSTEM

2.1 Governing Equations of Motion

We consider point mass assumptions for UAV model: flat earth approximation, the velocity vector, reference line of the UAV, drag and thrust forces are collinear, and the lift force is orthogonal to the velocity vector $\mathbf{v}_g$. Therefore, the states of the model are: vehicle position (x, y, z) in Inertial reference frame, magnitude of the velocity vector v_g, course or ground heading angle χ and flight path angle γ. Ground Heading angle, χ is the angle between the projection of the velocity vector onto the $x-y$ plane and the x axis whereas flight path angle, γ is the angle between the velocity vector $\mathbf{v}_g$ and its projection onto the $x-y$ plane. Therefore, we get the following kinematic equations for point mass UAV model:

$$\begin{aligned}
\dot{x} &= v_g \cos\gamma \cos\chi \\
\dot{y} &= v_g \cos\gamma \sin\chi \\
\dot{z} &= v_g \sin\gamma \\
\dot{v} &= c_1(v_g^c - v_g) \\
\dot{\gamma} &= c_2(\gamma^c - \gamma) \\
\dot{\chi} &= c_3(\chi^c - \chi)
\end{aligned} \tag{2.1}$$

For simplicity, the first order dynamics models are assumed for the speed, flight path angle and the course angle dynamics. c_1, c_2, and c_3 (> 0) are the time constants associated with the dynamics. The magnitude of the speed is : $v_g = \sqrt{(\dot{x}^2 + \dot{y}^2 + \dot{z}^2)}$ There are three control inputs into the model: v_g^c, γ^c, χ^c are the commanded speed,

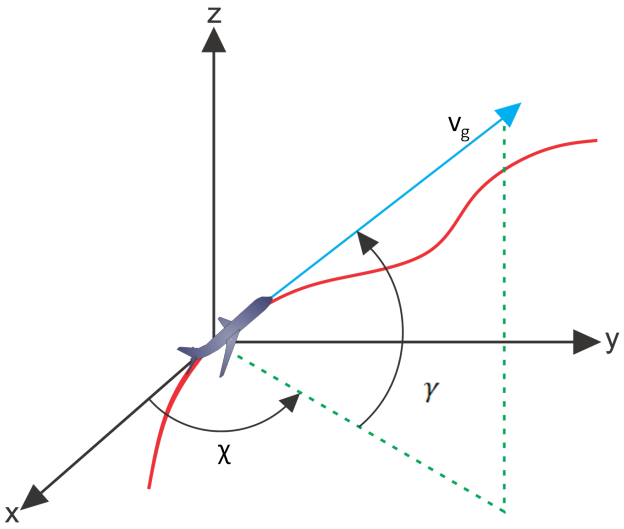


Figure 2.1. 3D Representation of UAV.

flight path angle and course angle that are computed by the controller.

We consider another representation of the kinematic equation for a point mass chaser UAV taken from [57] where a five state model is described and the velocity is kept constant. But in our case, we include first order dynamics for velocity the same way as described for UAV model in case 1. Therefore, the kinematic equations for UAV can be written as:

$$\begin{aligned}
\dot{x} &= v_g \cos\gamma \cos\chi \\
\dot{y} &= v_g \cos\gamma \sin\chi \\
\dot{z} &= v_g \sin\gamma \\
\dot{v}_g &= c_1(v_g^c - v_g) \\
\dot{\chi} &= \frac{g}{v_g} \tan\phi\eta \\
\dot{\gamma} &= \frac{g\cos\gamma}{v_g}(\eta - 1)
\end{aligned} \tag{2.2}$$

where the magnitude of velocity $v_g = \sqrt{(\dot{x}^2 + \dot{y}^2 + \dot{z}^2)}$, χ is the heading angle, and γ is the flight path angle or pitch angle, g is the gravitational acceleration, ϕ is the roll angle and η is the load factor.

We can write the following equations from three dimensional missile engagement geometry [57]:

$$\dot{\chi} = \frac{g}{v_g}\tan\phi\eta = \frac{a_h}{v_g\cos\gamma} \tag{2.3}$$

$$\dot{\gamma} = \frac{g\cos\gamma}{v_g}(\eta - 1) = \frac{a_v}{v_g} \tag{2.4}$$

Using the above two equations, we can determine the roll angle ϕ, and load factor η if we know the horizontal and vertical acceleration components of UAV.

$$\begin{aligned} \eta &= \frac{a_v}{g\cos\gamma} + 1 \\ \phi &= \tan^{-1}\left(\frac{a_h}{\eta g\cos\gamma}\right) \end{aligned} \tag{2.5}$$

For simplicity, we consider the following expression for heading angle, and the flight path angle dynamics. In this research, the particular objective is to design control laws for commanded velocity v_g^c, horizontal acceleration a_h^c, and vertical acceleration a_v^c so as to track a moving target.

$$\begin{aligned} \dot{x} &= v_g\cos\gamma\cos\chi \\ \dot{y} &= v_g\cos\gamma\sin\chi \\ \dot{z} &= v_g\sin\gamma \\ \dot{v}_g &= c_1(v_g^c - v_g) \\ \dot{\gamma} &= \frac{a_v^c}{v_g} \\ \dot{\chi} &= \frac{a_h^c}{v_g\cos\gamma} \end{aligned} \tag{2.6}$$

2.2 Target UAV Model

We assume that the desired reference trajectory is generated from dynamics of a 'virtual reference vehicle', as summarized below:

$$\begin{aligned} \dot{x}_r &= v_{gr} \cos\gamma_r \cos\chi_r \\ \dot{y}_r &= v_{gr} \cos\gamma_r \sin\chi_r \\ \dot{z}_r &= v_{gr} \sin\gamma_r \end{aligned} \tag{2.7}$$

where, $(x_r,\ y_r,\ z_r)$, v_{gr}, γ_r and χ_r denote the reference position, velocity, flight path angle and course (ground track) angle respectively. v_{gr}, γ_r and χ_r are specified function of time $\in C^\infty$. The reference ground speed can be expressed as $v_{gr} = \sqrt{(\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2)}$. For a rich range of trajectories, v_{gr}, χ_r and γ_r could be arbitrary functions of time. In this dissertation, for target trajectory, we will use the similar kind of dynamics as chaser either Eq. (2.1) or Eq. (2.6) with specified known control inputs.

2.3 Measurement Model

The measurement model for the target UAV in $3D$ space is taken as of the following form:

$$\tilde{\mathbf{y}}_{rk} = \mathbf{h}_{rk}(\mathbf{X}_{rk}) + \mathbf{v}_{rk} \tag{2.8}$$

where $\mathbf{h}_r(\mathbf{X}_r(t),\ t) = [r_{rk},\ \phi_{rk},\ \theta_{rk}]^T$ is the discrete function vector and $\mathbf{v}_{rk}$ is the noise vector associated with the measurement uncertainties. The discrete-time measurements for range r_{rk}, azimuth ϕ_{rk}, and elevation angle θ_{rk} are assumed to be

available from a ground based sensor or from on-board seeker antenna and are of the following form:

$$\begin{aligned} r_{rk} &= \sqrt{x_{rk}^2 + y_{rk}^2 + z_{rk}^2} + v_{rrk} \\ \phi_{rk} &= \tan^{-1} \frac{y_{rk}}{x_{rk}} + v_{\phi rk} \\ \theta_{rk} &= \sin^{-1} \frac{z_{rk}}{r_{rk}} + v_{\theta rk} \end{aligned} \tag{2.9}$$

For the chaser UAV, we consider 6 states discrete-time measurements are available from on-board sensors, namely GPS + IMU.

$$\tilde{\mathbf{y}}_k = \mathbf{h}(\mathbf{X}_k) + \mathbf{v}_k \tag{2.10}$$

where $\mathbf{h}(\mathbf{X}_k) = [x_k,\ y_k,\ z_k,\ v_{gk},\ \gamma_k,\ \chi_k]^T$

2.4 Mathematical Preliminaries

In this section, we briefly discuss and summarize some mathematical concepts and tools which will be required to develop the theoretical content of this dissertation. The mathematical definitions, theorems (without proof), and lemma are discussed based on [49, 58–61].

Definition. Norm of Vector:

Norm is defined as the measure of the length of a vector. The norm of vector $\mathbf{x}$ denoted by $||\mathbf{x}||$ which can be defined as a norm if it has the following properties [60]:

1. $||\mathbf{x}|| \geq 0\ \forall\ \mathbf{x}$ and $||\mathbf{x}|| = 0$ if and only if $\mathbf{x} = \mathbf{0}$
2. $||\mathbf{x}|| = |\alpha|||\mathbf{x}||$, for any real α
3. $||\mathbf{x}_1 + \mathbf{x}_2|| \leq ||\mathbf{x}_1|| + ||\mathbf{x}_2||$ for every $\mathbf{x}_1$, and $\mathbf{x}_2$ (Triangular Inequality)

Definition. Induced (Matrix) Norm

The norm of a Matrix $A \in \mathbb{R}^{m \times n}$ is defined as $||A|| = sup_{\mathbf{x} \neq 0} \frac{||A\mathbf{x}||}{||\mathbf{x}||}$ (Induced Norm).

Table 2.1. Commonly used Vector and Matrix Norms

Type	Vector Norm	Matrix Norm
1-norm	$\|\|\mathbf{x}\|\|_1 = \sum_{i=1}^{n} \|x_i\|$	$\|\|A\|\|_1 = \max_j \sum_{i=1}^{n} \|a_{ij}\|$
2-norm	$\|\|\mathbf{x}\|\|_2 = [\sum_{i=1}^{n} \|x_i^2\|]^{1/2}$	$\|\|A\|\|_2 =$ max singular value of A
Infinity-norm	$\|\|\mathbf{x}\|\|_\infty = \max_i \|x_i\|$	$\|\|A\|\|_\infty = \max_i \sum_{j=1}^{n} \|a_{ij}\|$

Definition. $\mathcal{L}_p$ Norm

For functions of time, define the $\mathcal{L}_p$ norm as:

$$||x||_p \triangleq \left(\int_0^\infty |x(\tau)|^p d\tau \right)^{1/p}$$

for $p \in [1, \infty)$ and say that $x \in \mathcal{L}_p$ when $||x||_p$ exists (i.e. when $||x||_p$ is finite). The $\mathcal{L}_\infty$ is defined as

$$||x||_\infty \triangleq \sup_{t \geq 0} |x(t)|$$

where $x \in \mathcal{L}_\infty$ when $||x||_\infty$ exists. $x(t)$ can be a scalar or a vector function. If x is a scalar function, then |.| denotes the absolute value. If x is a vector function in $\mathcal{R}^n$ then |.| denotes any norm in $\mathcal{R}^n$.

Lemma. Hölder's Inequality

If $p,\ q \in [1, \infty]$, and $\frac{1}{p} + \frac{1}{q} = 1$, then $f \in \mathcal{L}_p$, $g \in \mathcal{L}_q$ imply that $fg \in \mathcal{L}_1$ and

$$||fg||_p \leq ||f||_p ||q||_q$$

When p = q = 2, the Hölder's inequality becomes the Schwartz inequality, i.e.,

$$||fg||_2 \leq ||f||_2 ||q||_2$$

Lemma. Minkowski Inequality

For $p \in [1, \infty]$ and $f,\ g \in \mathcal{L}_p$ imply that $f + g \in \mathcal{L}_p$ and

$$||f + g||_p \leq ||f||_p + ||g||_p$$

Lemma. Barbălat's Lemma

If f, $\dot{f} \in \mathcal{L}_\infty$ and $f \in \mathcal{L}_p$ for some $p \in [1, \infty)$, then $f(t) \to 0$ as $t \to \infty$. If $\lim_{t\to\infty} \int_0^t f(\tau)d\tau$ exists and is finite, and f(t) is a uniformly continuous function, then $\lim_{t\to\infty} f(t) = 0$.

Lemma. Schwartz Inequality

$|\mathbf{x}^T\mathbf{y}| \leq ||\mathbf{x}||||\mathbf{y}||$

The triangle inequlity is the direct consequence of the Schwartz inequality.

Matrix definiteness

A real symmetric matrix $A \in \mathbb{R}^{n\times n}$ is said to be positive semi-definite if $\mathbf{x}^T A\mathbf{x} \geq 0$ for all $\mathbf{x}$ and positive definite if $\mathbf{x}^T A\mathbf{x} > 0$ for all $\mathbf{x} \neq \mathbf{0}$. Matrix A is said to be negative semi-definite if $-A$ is positive semi-definite, and negative definite if $-A$ is positive definite. It is called indefinite when no definiteness can be determined. For symmetric positive semi-definite matrix A the eigenvalues and singular values are the same. For any symmetric positive definite matrix A, the following inequality holds true:

$$\sigma_{\min}(A)||\mathbf{x}||^2 \leq \mathbf{x}^T A\mathbf{x} \leq \sigma_{\max}(A)||\mathbf{x}||^2$$

where $\sigma_{\max}(A)$, $\sigma_{\min}(A)$ are the maximum and minimum eigenvalues of A respectively.

2.5 Stability of Systems

Lyapunov stability analysis is the most useful and well known technique for nonlinear systems. There are two methods to analyze the systems: 1) Lyapunov's Direct Method, and 2) Lyapunov Indirect Method. Some important definitions are need to be described first [58]:

Definition. Stability

We consider systems described by ordinary differential equations of the form

$$\dot{x} \;=\; f(t,x), x(t_0) = x_0 \tag{2.11}$$

where $x \in R^n$, $f : \tau \times B(r) \;\rightarrow\; R$, $\tau = [0, \infty)$ and $B(r) = \{x \in R^n |\ |x| < r$ where $r \in R^+\}$. We assume that f is of such nature that for every $x_0 \in B(r)$ and every $t_0 \in R^+$ Eq. 2.11 possesses one and only one solution $x(t; t_0, x_0)$.

Definition. Equilibrium Points

A state x_e is said to be an equilibrium state (or equilibrium point) of the system if once $x(t)$ is equal to x_e it remains at x_e for all future time which implies that $f(t, x^*) \equiv 0$, $\forall t \geq t_0$.

Definition. Stable in the sense of Lyapunov

The equilibrium state x_e is said to be stable (in the sense of Lyapunov) if for arbitrary t_0 and $\epsilon > 0$, there exists a $\delta(\epsilon, t_0)$ such that $|x_0 - x_e| < \delta$ implies $|x(t; t_0, x_0) - x_e| < \epsilon$ for all $t \geq t_0$.

Definition. Asymptotically Stable

The equilibrium state x_e is said to be asymptotically stable if (1) it is stable, and (2) there exists a $\delta(t_0)$ such that $|x_0 - x_e| < \delta(t_0)$ implies $\lim_{t\rightarrow\infty} |x(t; t_0, x_0) - x_e| = 0$.

Definition. Exponentially Stable

The equilibrium state x_e is exponentially stable if there exists an $\alpha > 0$, and for every $\epsilon > 0$ there exists a $\delta(\epsilon) > 0$ such that $|x(t; t_0; x_0) - x_e| \leq \epsilon e^{-\alpha(t-t_0)}$ for all $t \geq t_0$ whenever $|x_0 - x_e| < \delta(\epsilon)$.

Definition. Uniformly Bounded

The solutions of 2.11 are uniformly bounded if for any $\alpha > 0$ and $t_0 \in R^+$, there exists a $\beta = \beta(\alpha)$ independent of t_0 such that if $|x_0| < \alpha$, then $|x(t; t0; x0)| < \beta$ for all $t \geq t_0$.

Definition. Ultimately Uniformly Bounded

The solutions of 2.11 are uniformly ultimately bounded (with bound B) if there exists a $B > 0$ and if corresponding to any $\alpha > 0$ and $t_0 \in R^+$, there exists a $T = T(\alpha) > 0$ (independent of t_0) such that $|x_0| < \alpha$ implies $|x(t; t0; x0)| < B$ for all $t \geq t_0 + T$.

Definition. Class $\mathcal{K}$ function

A continuous function $\phi : [0, r] \to \mathcal{R}^+$ (or a continuous function $\phi : [0, \infty) \to \mathcal{R}^+$) is said to belong **class** $\mathcal{K}$, i.e. $\phi \in \mathcal{K}$ if

1. $\phi(0) = 0$
2. ϕ is strictly increasing on $[0, r]$ (or on $[0, \infty)$)

Definition. Class $\mathcal{KR}$ function

A continuous function $\phi : [0, \infty) \to \mathcal{R}^+$ is said to belong **class** $\mathcal{KR}$, i.e. $\phi \in \mathcal{KR}$ if

1. $\phi(0) = 0$
2. ϕ is strictly increasing on $[0, \infty)$
3. $\lim_{r\to\infty} \phi(r) = \infty$

Definition. Positive/Negative Definite Function

A function $V(t, x) : \mathcal{R}^+ \times \mathcal{B}(r) \to \mathcal{R}$ with $V(t, 0) = 0$, $\forall\ t \in \mathcal{R}^+$ is positive definite if there exist a continuous function $\phi \in \mathcal{K}$ such that $V(t, x) \geq \phi(|x|)$, $\forall\ t \in \mathcal{R}^+$, $x \in \mathcal{B}(r)$ and some $r > 0$. $V(t, x)$ is called negative definite if $-V(t, x)$ is positive definite.

Definition. Positive/Negative Semidefinite Function

A function $V(t, x) : \mathcal{R}^+ \times \mathcal{B}(r) \to \mathcal{R}$ with $V(t, 0) = 0$, $\forall\, t \in \mathcal{R}^+$ is positive semidefinite if there exist a continuous function $\phi \in \mathcal{K}$ such that $V(t, x) \geq 0$, $\forall\ t \in \mathcal{R}^+$, $x \in \mathcal{B}(r)$ and some $r > 0$. $V(t, x)$ is called negative semidefinite if $-V(t, x)$ is positive semidefinite.

Definition. Decrescent Function

A function $V(t, x) : \mathcal{R}^+ \times \mathcal{B}(r) \to \mathcal{R}$ with $V(t, 0) = 0$, $\forall\ t \in \mathcal{R}^+$ is said to be

decrescent if there exist a continuous function $\phi \in \mathcal{K}$ such that $|V(t,x)| \leq \phi(|x|)$, $\forall\ t \in \mathcal{R}^+$, $x \in \mathcal{B}(r)$ and some $r > 0$.

Definition. Radially Unbounded Function

A function $V(t,x) : \mathcal{R}^+ \times \mathcal{R}^n \to \mathcal{R}$ with $V(t,0) = 0$, $\forall\ t \in \mathcal{R}^+$ is said to be radially bounded function if there exist a continuous function $\phi \in \mathcal{KR}$ such that $|V(t,x)| \geq \phi(|x|)$ $\forall x \in \mathcal{R}^n$ and $t \in \mathcal{R}^n$.

Lyapunov Direct Method

Theorem. Suppose there exists a positive definite function $V(t;x) : R \times B(r) \to R$ for some $r > 0$ with continuous first-order partial derivatives with respect to x, t, and $V(t;0) = 0$ $\forall t \in R^+$. Then the following statements are true:

1. If $\dot{V} \leq 0$, then $x_e = 0$ is stable.
2. If V is decrescent and $V \leq 0$, then $x_e = 0$ is u.s.
3. If V is decrescent and $V < 0$, then x_e is u.a.s.
4. If V is decrescent and there exist φ_1, φ_2,$\varphi_3 \in K$ of the same order of magnitude such that

$$\varphi_1(|x|) \leq V(t,x) \leq \varphi_2(|x|), \quad \dot{V}(t,x) \leq -\varphi(|x|)$$

for all $x \in B(r)$ and $t \in R^+$, then $x_e = 0$ is exponentially stable.

In the above theorem, the state x is restricted to be inside the ball $B(r)$ for some $r > 0$.

Theorem. Assume that Eq. 2.11 possesses unique solutions for all $x_0 \in R^n$. Suppose there exists a positive definite, decrescent and radially unbounded function $V(t,x) : R^+ \times R^n \to R^+$ with continuous first-order partial derivatives with respect to t, x and $V(t,0) = 0$ $\forall t \in R^+$. Then the following statements are true:

1. If $\dot{V} < 0$, then $x_e = 0$ is u.a.s. in the large.

2. If there exist φ_1, φ_2, $\varphi_3 \in KR$ of the same order of magnitude such that

$$\varphi_1(|x|) \leq V(t,x) \leq \varphi_2(|x|), \quad \dot{V}(t,x) \leq -\varphi(|x|)$$

then $x_e = 0$ is e.s. in the large. Statement (i) of Theorem is also equivalent to that there exist φ_1, $\varphi_2 \in K$, and $\varphi_3 \in KR$ such that

$$\varphi_1(|x|) \leq V(t,x) \leq \varphi_2(|x|), \quad \dot{V}(t,x) \leq -\varphi(|x|), \quad \forall x \in R^n$$

Theorem. Assume that Eq. 2.11 possesses unique solutions for all $x_0 \in R^n$. If there exists a function $V(t,x)$ defined on $|x| \geq R$ (where R may be large) and $t \in [0,\infty)$ with continuous first-order partial derivatives with respect to x, t and if there exist φ_1, $\varphi_2 \in KR$ such that

1. $\varphi_1(|x|) \leq V(t,x) \leq \varphi_2(|x|)$
2. $\dot{V} \leq 0$ for all $|x| \geq R$ and $t \in [0;\infty)$ then the solutions are bounded. If in addition there exists $\varphi_3 \in K$ defined on $[0,\infty)$ and
3. $\dot{V} \leq -\varphi_3(|x|)$ for all $|x| \geq R$, and $t \in [0;\infty)$ then the solutions are u.u.b.

Lemma. Let, f, $V : [0,\infty) \to \mathcal{R}$, then

$$\dot{V} = -\alpha V + f, \qquad \forall\ t \geq t_0 \geq 0$$

implies that

$$V(t) = e^{-\alpha(t-t_0)}V(t_0) + \int_t^{t_0} e^{-\alpha(t-t_0)} f(\tau) d\tau, \qquad \forall\ t \geq t_0 \geq 0$$

for any finite constant α

Lyapunov Indirect Method Let $x_e = 0$ be an equilibrium state of Eq. 2.11 and assume that $f(t,x)$ is continuously differentiable with respect to x for each $t \geq 0$. Then in the neighborhood of $x_e = 0$, a Taylor series expansion of function f can be written as

$$\begin{aligned} \dot{x} &= f(t,x) \\ &= A(t)x + f_1(t,x) \end{aligned} \tag{2.12}$$

where $A(t) = \Delta f|_{x=0}$ is referred to as Jacobian matrix of f evaluated at $x = 0$ and $f_1(t, x)$ represents the remaining terms in the series expansion.

Theorem. Assume that $A(t)$ is uniformly bounded and that

$$\lim_{x \to 0} \sup_{t \geq 0} \frac{|f_1(t, x)|}{|x|}$$

Let z_e be the equilibrium of

$$\dot{z}(t) \quad = \quad A(t)z(t) \tag{2.13}$$

The following statements are true for the equilibrium $x_e = 0$ of Eq. 2.12

1. If $z_e = 0$ is u.a.s then $x_e = 0$ is u.a.s
2. If $z_e = 0$ is unstable then $x_e = 0$ is unstable
3. If $z_e = 0$ is u.s or stable, no conclusions can be drawn about the stability of $x_e = 0$.

Lemma. Convergence: If a real function W(t) satisfies the inequality [61]

$$\dot{W} + \alpha W(t) \quad \leq \quad 0 \tag{2.14}$$

where α is a real number. Then

$$W(t) \quad \leq \quad W(0)e^{-\alpha t}$$

Definition. Invariant Set

A set $\Omega \in \mathcal{R}^n$ is invariant with respect to Eq. 2.11 if every solution of Eq. 2.11 starting in Ω remains in Ω for all t.

Theorem. LaSalle's Invariance Principle

Assume that Eq. (2.11) possesses unique solutions for all $x_0 \in \mathcal{R}^n$. Suppose there exists a positive definite and radially unbounded function $V(x) : \mathcal{R}^n \to \mathcal{R}^+$ with continuous first-order partial derivative with respect to x and $V(0) = 0$. If

1. $\dot{V} \leq 0 \; \forall x \in \mathcal{R}^n$

2. The origin $x = 0$ is the only invariant subset of the set $\omega = x \in \mathcal{R}^n | \dot{V} = 0$

then the equilibrium $x_e = 0$ of Eq. 2.11 is asymptotically stable in the large.

2.6 Probability Concepts and Statistics

The preliminary concepts of random process, stochastic random process, and probability concepts are summerized in this section based on [62, 63].

The outcomes of a random experiment can not be defined in advance. The set of all possible outcomes of an experiment is called sample space denoted by S of the experiment. An event E is a subset of sample space S and is said to occur if the outcome is the element of an element of that subset. We can define the Probability on events. For each event E of the sample space S, $P(E)$ is the probability of event E if it satisfies the following three conditions:

1. $0 \leq P(E) \leq 1$
2. $P(S) = 1$
3. For any sequence of mutually exclusive events $E_1, \; E_2, \; \ldots$ i.e. that events $E_n E_m = \phi$ when $n \neq m$, then $P\left(\bigcup_{n=1}^{\infty} E_n\right) = \sum_{n=1}^{\infty} P(E_n)$

Stochastic Random Process

A stochastic random process is simply a collection of random vectors defined on the same probability space. It is denoted as $\mathbf{X} = X(t), t \in T$ where $X(t)$ is the state of the process at time t and T is the index set. If T is countable then it is called discrete-time stochastic process, and if T is continuum, it is called as continuous-time process [62].

Stationary and Non-stationary Random Processes

A random process is called stationary if the parameters of the probability model of

the process such as mean, variance, power spectral composition, and higher order moments of the process are time invariant. If a random process doesn't meet the time-invariant property of parameters, it is called non-stationary.

Definition. Probability Distribution Function, F(x)

The distribution function or cumulative distribution function of the random variable X is defined for any real number x by

$$\begin{aligned} F(x) &= P(X \leq x) \\ &= P\{X \in (-\infty, x)\} \end{aligned}$$

For discrete random variable X,

$$F(x) = \sum_{y \leq x} P(X = y)$$

Definition. Probability Density Function, f(x)

For continuous-time process, it is defined by

$$f(x) = \frac{dF}{dx}$$

and it is non-negative $f(x) \geq 0$, and the integral of the pdf in the range $\pm\infty$ is unity i.e.

$$\int_{-\infty}^{\infty} f(x)dx = 1$$

For any set B,

$$P\{X \in B\} = \int_B f(x)dx$$

Definition. Probability Mass Function, p(x)

For a discrete random variable X, probability mass function $p(x)$ of X is defined by:

$$p(x) = P(X = x)$$

The $p(x)$ is positive for all countable numbers of values of x. Therefore, we can wrtie

$p(x_i) > 0$, $i = 1,\ 2,\ 3, \ldots$

$p(x) = 0$, all other values of x

The sum of $p(x_i)$ all values of x_i is 1 i.e.

$$\sum_{i=1}^{\infty} \ = \ 1$$

Definition. Expected value or Mean

The expected value or average value of a discrete random variable x is described by the first moments of x denoted by

$$\mu \ = \ \sum_{j} x(j)p(x(j))$$

Similarly we can write for discrete random vector **x**

Definition. Variance

The expected value or average value of a discrete random variable x is described by the second moments of x

$$\sigma^2 \ = \ \sum_{j} (x(j) - \mu)^2 p(x(j))$$

where $p(x)$ is the probability mass function, and σ is the standard deviation i.e. the square root of the variance.

Definition. Autocorrelation Function

The autocorrelation, or autocovariance is another characteristic of a stochastic process which describes the general dependency of $x(t)$ with its value at a short time later, $x(t + \tau)$.

$$R_{xx}(\tau) \ = \ E[x_k(t)x_k(t + \tau)]$$

Definition. Power Spectral Density Function

Power spectral density describes how the energy of fluctuations is distributed as function of frequency. The PSD can written in terms of autocorrelation function as:

$$S_{xx}(f) = \int_{-\infty}^{\infty} R_{xx} e^{-j2\pi f\tau} d\tau$$

where $S_{xx}(f)$ is two-sided PSD function (since the frequency range is $-\infty \leq f \leq \infty$) One-sided PSD function can be written as:

$$G_{xx}(f) = 2S_{xx}(f)$$

The unit of $G_{xx}(f)$ is (unit of $x)^2$/unit of f.

Gaussian Random Variables

A most commonly used distribution for state estimation is Gaussian noise distribution. The probability density function for Gaussian distribution is

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

White Noise

A white noise is defined as an uncorrelated random noise process with equal power at all frequencies. Theoretically, a random white noise has infinite power at all frequencies in the range of $-\infty$ to $+\infty$. In practice, we consider band limited white noise with a flat power spectral density which covers the frequency range of a limited band width.

The autocorrelation function of a continuous-time white noise with variance σ^2 is a dirac delta function which is only nonzero at $\tau = 0$ ($x_k(t)$ and $x_k(t+\tau)$ are totally uncorrelated $\forall\tau > 0$) i.e.

$$\begin{aligned} R_{xx}(\tau) &= E\left[x_k(t)x_k(t+\tau)\right] \\ &= A\delta(\tau) \end{aligned}$$

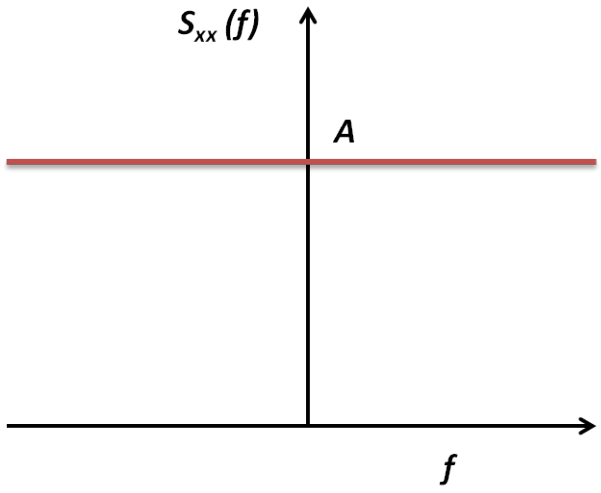


Figure 2.2. PSD function for white noise.

The two-sided power spectral density (PSD) of a white noise can be obtained by taking the Fourier Transform of the following equation:

$$S_{xx}(f) = \int_{-\infty}^{\infty} Rxx(\tau)e^{-j2\pi f\tau}d\tau = A$$

The PSD function for white noise is constant in terms of frequency as shown in Fig. 2.2.

Colored Noise

We can generate the colored noise by propagating the first order differential equation with band limited white noise input.

$$\dot{x} + Bx \quad = \quad \sqrt{B}w \tag{2.15}$$

The autocorrelation function for white noise input is $R_{ww}(\tau) = A\delta(\tau)$ and the single sided PSD for w is $G_{ww}(f) = 2A$. For piecewise constant w with time increment T denoted by w_k, we can write the solution of the above differential equation as:

$$x_{k+1} = e^{-BT}x_k + \frac{1 - e^{-BT}}{\sqrt{B}}w_k$$

The autocorrelation function can be shown for this case:

$$R_{xx}(\tau) = \frac{A}{2} e^{-B|\tau|}$$

which is exponentially correlated autocorrelation function. $R_{xx}(\tau)$ decays faster with smaller time constant $(1/B)$ i.e. larger value of B. If $R_{xx}(\tau) \to 0$ as $\tau \to \infty$, it implies that the noises are uncorrelated i.e. zero mean.

The one-sided PSD function can be derived as:

$$G_{xx}(f) = \frac{2AB}{B^2 + (2\pi f)^2}$$

2.7 Numerical Integration Techniques

The most commonly used numerical integration technique is Runge-Kutta (R-K) method and many variants of R-K methods are available in the literature. We used different solvers to simulate the first order shaping filter equation and generate the colored noise. For all cases, we get very much consistent results. Therefore, we will be used ODE45 solver in MATLAB for the rest of simulation of this dissertation.

To be consistent with the general notation of first order differential equation (ODE), consider the following form of ODE:

$$\dot{x} = f(x, t) \tag{2.16}$$

with the initial condition of $x(t_0) = x_0$. We can use either of these approaches to solve the above equation.

n-order Runge-Kutta (RK) Method

The general solution form for n-order RK method is as follows:

$$\begin{aligned} x_{k+1} &= x_k + \alpha_1 k_1 + \alpha_2 k_2 + \quad \ldots \quad + \alpha_n k_n \\ k_1 &= h f(t_k, x_k) \\ k_j &= h f(t_k + c_j h, x_k + \sum_{i=1}^{j-1} a_{ji} k_i) \end{aligned}$$

where h is the step size, and the coefficients of α_j, c_j, a_{ji} are chosen so that x_k simulates its solution to order h^{n+1} i.e. $x(t_k) = x_k + O(h^{n+1})$.

Classical Fourth-Order RK Method

The most widely used solution for 4^{th} order $R-K$ method is of the following form [64]:

$$\begin{aligned} x_{k+1} &= x_k + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 &= hf(t_k, x_k) \\ k_2 &= hf(t_k + \frac{h}{2}, x_k + \frac{k_1}{2}h) \\ k_3 &= hf(t_k + \frac{h}{2}, x_k + \frac{k_2}{2}h) \\ k_4 &= hf(t_k + h, x_k + k_3 h) \end{aligned}$$

For stochastic differential equation, we simulate the first order shaping filter equation for piecewise constant white noise w at a certain time step.

R-K Integrator for Time Varying Stochastic Differential Equations

Kasdin [65] proposed a 4^{th} order R-K solution for more general case of Ito nonlinear stochastic differential equation:

$$\frac{dX}{dt} = F(X(t,\zeta),t) + G(X(t,\zeta),t)w(t,\zeta) \tag{2.17}$$

The solution of x_{k+1} for linearized version of the above equation is proposed as [66]:

$$\begin{aligned} x_{k+1} &= x_k + \alpha_1 k_1 + \alpha_2 k_2 + \quad \dots \quad + \alpha_n k_n \\ k_1 &= hF\left(t_k, x_k\right) + hG\left(t_k, x_k\right) w_1 \\ k_j &= hF\left(t_k + c_j h, x_k + \sum_{i=1}^{j-1} a_{ji} k_i\right) + hG\left(t_k + c_j h, x_k + \sum_{i=1}^{j-1} a_{ji} k_i\right) w_j \end{aligned}$$

where w_i identically distributed noise with variance $q_i Q/h$, where Q is the spectral density of the input white noise w of Eq. 2.17. The details of determining the coefficients α_i, a_j, c_i, and q_i can be found in [66].

ODE45 Solver in MATLAB

In MATLAB, integration solver ode45 mainly combines the 4^{th} order method and 5^{th} order RK method [67] and it is very similar to classical 4^{th} order RK method which is mentioned above. It uses the adaptive step size and at each step it chooses the step size to achieve the desired accuracy. In many practical applications, the ODE45 routine is quite a good solver to apply as a first try.

lsim in MATLAB

In MATLAB, lsim is used to simulate the time response of continuous or discrete Linear Time Invariant (LTI) Systems to arbitrary inputs. We provide the input as the random white noise vector w which is equally space of time steps with the first order system state space representation or transfer functions.

Simulation Results Using Different Solvers

The solution of Eq. (2.15) using different solvers as mentioned above are provided in this section. The band limited white noise w is generated for one-sided PSD with amplitude $A = 5$ for $N = 1024$ sample points and w is then used as input into Eq. (2.15). The simulation is performed for $B = 2$ in each case. The simulation results using lsim/ode45 solver, classical 4^{th} RK/ode45 solver, classical RK/RK-solution proposed by Kasdin [65] are shown in Fig. 2.3, Fig. 2.4, and Fig. 2.5 respectively. The band limited white noise and colored noise using lsim is shown in Fig. 2.8 and the corresponding PSD function is shown in Fig. 2.6. The associated colored noise PSD function plots from using different solvers are shown in Fig. 2.7, 2.8, and 2.9.

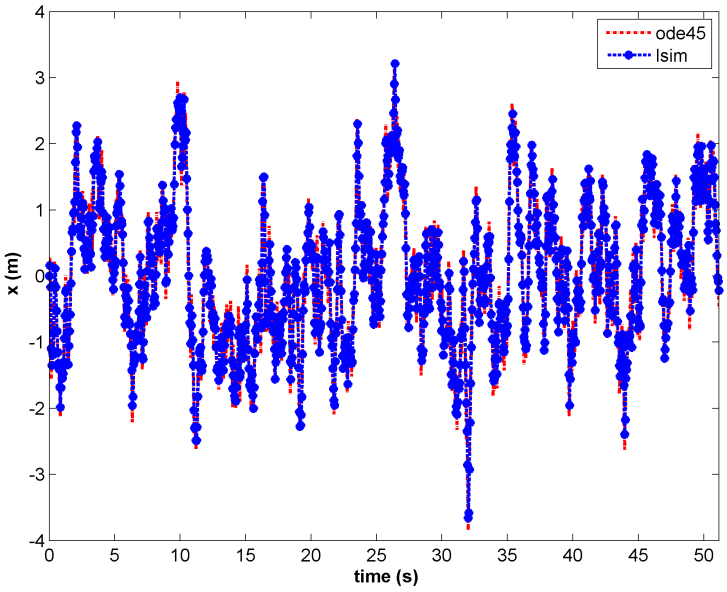


Figure 2.3. Solution of First Order Shaping Filter Equation using lsim and ODE45 solver.

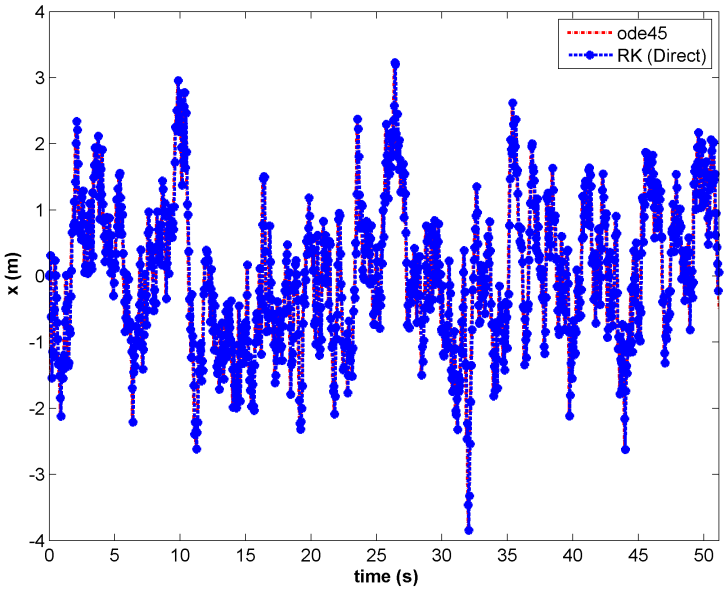


Figure 2.4. Solution of First Order Shaping Filter Equation using ODE45 solver and RK-direct.

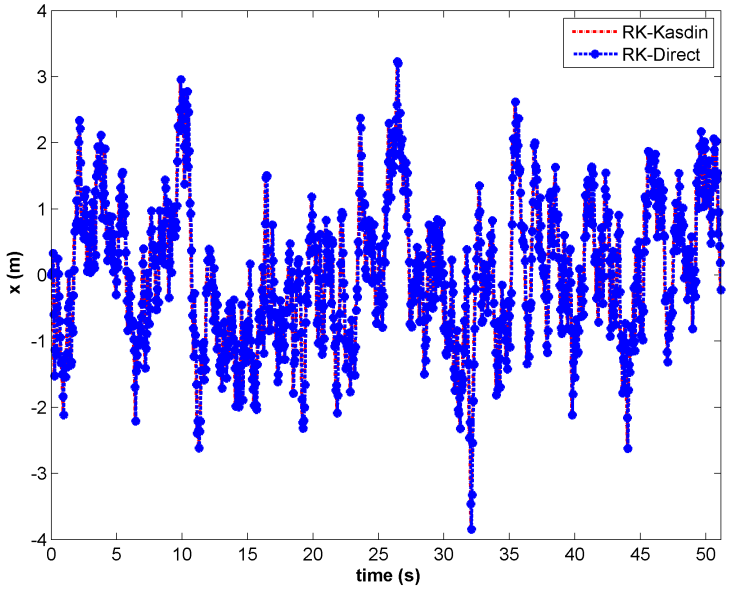


Figure 2.5. Solution of First Order Shaping Filter Equation using RK-Kasdin and RK-Direct.

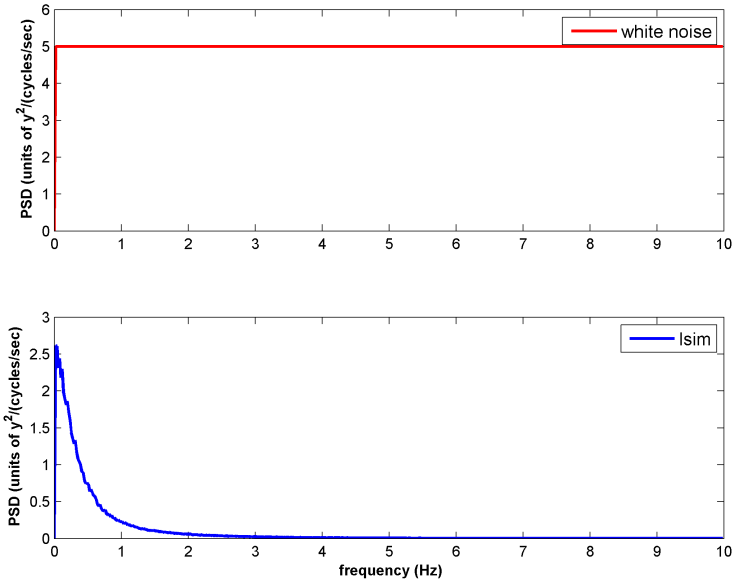


Figure 2.6. PSD for white noise and colored noise (using lsim solver).

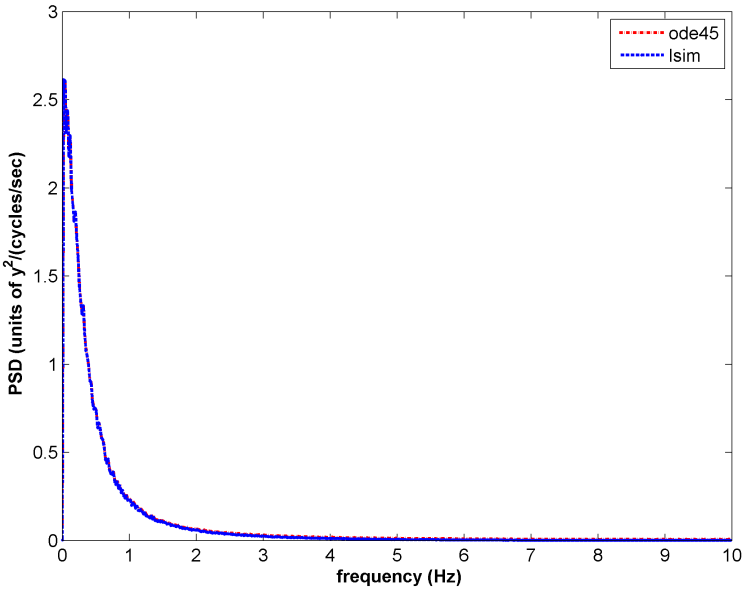


Figure 2.7. PSD for colored noise (using lsim and ode45 solver).

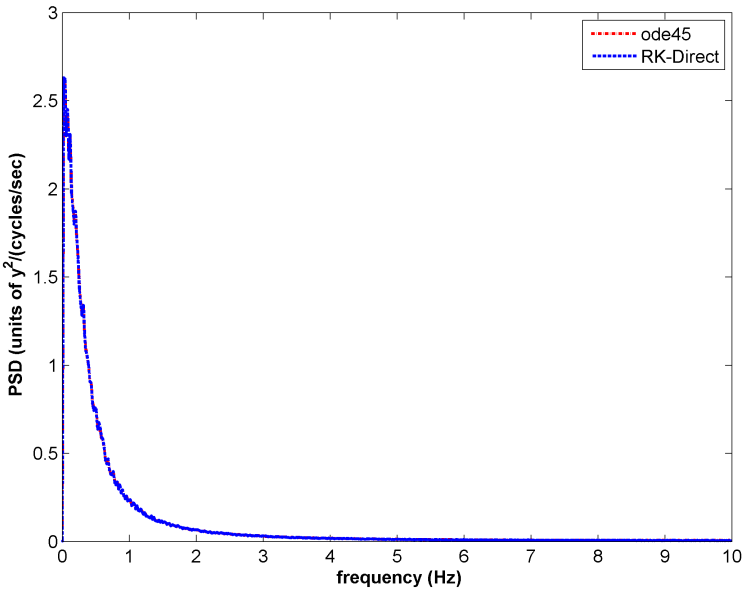


Figure 2.8. PSD for colored noise (using ode45 and RK-Direct).

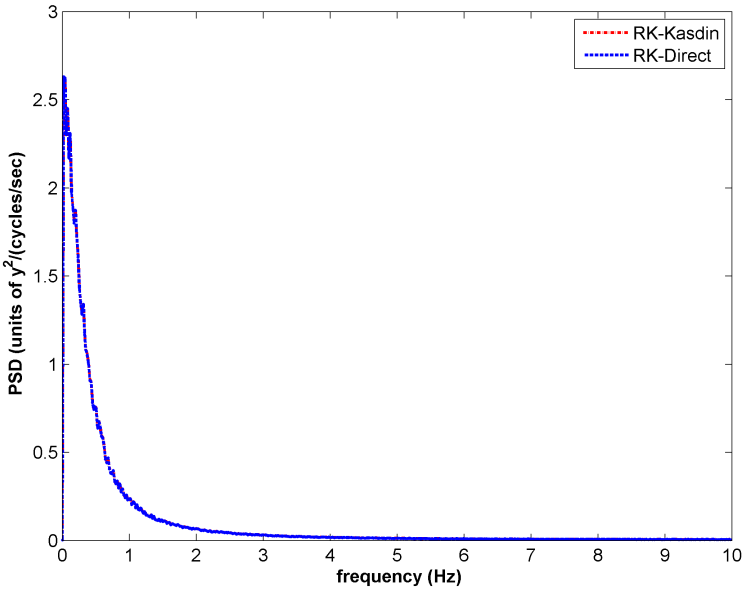


Figure 2.9. PSD for colored noise (using RK-Direct and RK-Kasdin).

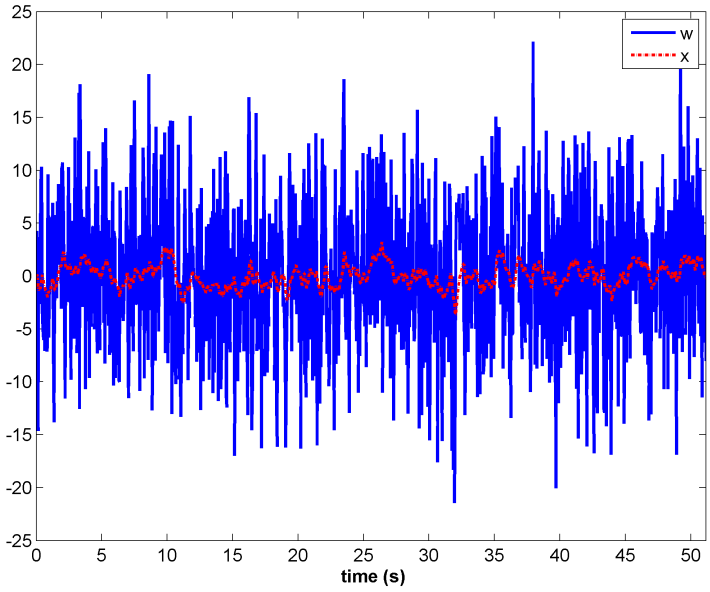


Figure 2.10. Colored noise and low band width white noise with time.

CHAPTER 3

CONTROL LAWS FOR TARGET TRACKING WITH A SINGLE UAV

In this chapter, we describe the mathematical formulation of backstepping like control laws for a single UAV for the following cases (a) perfect target state information in section 3.1; (b) partial target state information with measurements corrupted by white noises in section 3.2; (c) partial target state information with colored noise measurement uncertainties in section 3.3; and (d) varying measurement error covariance as the UAV approaches the target in section 3.4.

3.1 Backstepping based Control Laws (Complete Target Information)

A backstepping like recursive control technique is applied to design the control laws for UAV. First, the stable position error dynamics are specified and the desired virtual control laws are designed for groundspeed v_g^d, flight path angle γ^d, and ground heading angle χ^d so as to make the position error zero. In the next step, the same technique is recursively applied for synthesizing the control laws for ground-speed v_g^c, flight path angle γ^c, and ground heading angle χ^c. In this section, two different sets of combination of target and chaser UAV model are considered and the corresponding control laws are derived. The detailed mathematical formulation is shown only for set 1, and the results listed for set 2.

Case 1: Target Model Eq.(2.7) - Chaser Model Eq. (2.1)

In this case, we specify similar type kinematic model used for target and chaser UAV. For the target model we used Eq. (2.7), and for the chaser model Eq. (2.1). Control

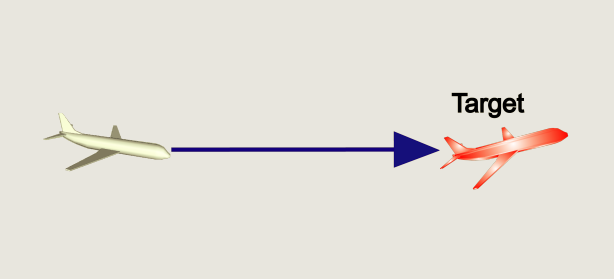


Figure 3.1. Target Tracking for a single UAV.

laws are designed for a single UAV by specifying the stable error dynamics and the control commands are backed out for velocity, flight angle, and course angle.

Theorem (Main Result)**.** Given the reference trajectory obtained from Eq. (2.7) and the actual dynamics of the UAV in Eq. (2.1), the following guidance laws ensure that $\|x - x_r\|$, $\|y - y_r\|$ and $\|z - z_r\| \to 0$ and all other signals are bounded.

$$\begin{aligned} v_g^c &= v_g - \frac{1}{c_1}\left[-\cos\gamma(-\dot{v}_1^d + \lambda_1\tilde{v}_1) - \sin\gamma(-\dot{v}_2^d + \lambda_2\tilde{v}_2)\right] \\ \gamma^c &= \gamma - \frac{1}{c_2 v_g}\left[\sin\gamma(-\dot{v}_1^d + \lambda_1\tilde{v}_1) - \cos\gamma(-\dot{v}_2^d + \lambda_2\tilde{v}_2)\right] \\ \chi^c &= \chi - \frac{1}{c_3}(-\dot{\chi}^d + \lambda_3\tilde{\chi}) \end{aligned} \tag{3.1}$$

where v_g^c, γ^c and χ^c represent the commanded ground speed, flight path angle and ground heading angle of the UAV, respectively, v_1, v_2, v_1^d, $\dot{v}_1^d$, v_2^d, $\dot{v}_2^d$, $\dot{\chi}^d$ are non-linear functions that depend upon the reference target states and the state tracking errors and $\tilde{v}_1 = v_1 - v_1^d$, $\tilde{v}_2 = v_2 - v_2^d$ and $\tilde{\chi} = \chi - \chi^d$.

Proof. We derive the control laws from a Lyapunov construction using a backstepping like approach. Let, $v_1 = v_g\cos\gamma$ and $v_2 = v_g\sin\gamma$. Thus, $v_g = \sqrt{v_1^2 + v_2^2}$ and the

flight path angle $\gamma = \tan^{-1}(\frac{v_2}{v_1})$. Using these equations, the actual dynamics is then recast as follows,

$$\begin{aligned} \dot{x} &= v_1 \cos\chi \\ \dot{y} &= v_1 \sin\chi \\ \dot{z} &= v_2 \end{aligned} \tag{3.2}$$

Now we introduce virtual control signals namely, v_1^d, χ^d and γ^d that will enable exponential tracking of the target trajectory. The position tracking errors are defined as, $e_x = x - x_r$, $e_y = y - y_r$ and $e_z = z - z_r$. The time derivative of these position tracking errors are $\dot{e}_x = \dot{x} - \dot{x}_r$, $\dot{e}_y = \dot{y} - \dot{y}_r$ and $\dot{e}_z = \dot{z} - \dot{z}_r$ accordingly. For exponentially stable position tracking error dynamics i.e,

$$\begin{aligned} \dot{e}_x &= -\alpha_1 e_x \\ \dot{e}_y &= -\alpha_2 e_y \\ \dot{e}_z &= -\alpha_3 e_z \end{aligned}$$

where $\alpha_1 > 0$, $\alpha_2 > 0$ and $\alpha_3 > 0$ user specified constants. We can then derive the desired virtual control signals v_1^d, v_2^d and χ^d to be,

$$\begin{aligned} v_1^d \cos\chi^d &= -\alpha_1 e_x + \dot{x}_r \\ v_1^d \sin\chi^d &= -\alpha_2 e_y + \dot{y}_r \\ v_2^d &= -\alpha_3 e_z + \dot{z}_r \end{aligned}$$

From the above we obtain the following desired virtual control signals for exponentially stable position tracking error dynamics,

$$\begin{aligned} v_1^d &= \sqrt{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2} \\ v_2^d &= -\alpha_3 e_z + \dot{z}_r \\ \chi^d &= \tan^{-1}\left(\frac{-\alpha_2 e_y + \dot{y}_r}{-\alpha_1 e_x + \dot{x}_r}\right) \end{aligned}$$

The commanded vehicle speed, flight path angle and the course angle are then derived by ensuring asymptotically stable off-manifold dynamics as shown later. Having obtained the desired virtual control signals, we then denote the off-manifold errors as $\tilde{v}_1 = v_1 - v_1^d$, $\tilde{v}_2 = v_2 - v_2^d$ and $\tilde{\chi} = \chi - \chi^d$. The off-manifold dynamics for the above is then obtained as follows:

$$\begin{aligned}
\dot{\tilde{v}}_1 &= \dot{v}_1 - \dot{v}_1^d \\
\dot{\tilde{v}}_2 &= \dot{v}_2 - \dot{v}_2^d \\
\dot{\tilde{\chi}} &= \dot{\chi} - \dot{\chi}^d
\end{aligned}$$

Now,

$$\begin{aligned}
\dot{v}_1 &= (-v_g \sin\gamma)\dot{\gamma} + \dot{v}_g \cos\gamma \\
&= -v_2\dot{\gamma} + \dot{v}_g \cos\gamma \qquad (3.3) \\
\dot{v}_2 &= (v_g \cos\gamma)\dot{\gamma} + \dot{v}_g \sin\gamma \\
&= v_1\dot{\gamma} + \dot{v}_g \sin\gamma \qquad (3.4)
\end{aligned}$$

and thus,

$$\begin{aligned}
\dot{\tilde{v}}_1 &= \dot{v}_1 - \dot{v}_1^d \\
&= -v_2 c_2(\gamma^c - \gamma) + c_1(v_g^c - v_g)\cos\gamma - \dot{v}_1^d \\
&= -c_2\ v_2\delta\gamma + c_1\ v_1\frac{\delta v_g}{v_g} - \dot{v}_1^d
\end{aligned}$$

where $\dot{\gamma}$ and $\dot{\chi}$ are substituted from Eq. (2.1) and $\delta\gamma = \gamma^c - \gamma$ and $\delta v_g = v_g^c - v_g$. Similarly we obtain,

$$\begin{aligned}
\dot{\tilde{v}}_2 &= \dot{v}_2 - \dot{v}_2^d \\
&= c_2\ v_1(\gamma^c - \gamma) + c_1(v_g^c - v_g)\sin\gamma - \dot{v}_2^d \\
&= c_2\ v_1\delta\gamma + c_1\ v_2\frac{\delta v_g}{v_g} - \dot{v}_2^d
\end{aligned}$$

Finally the off-manifold error dynamics for $\tilde{v}_1$ and $\tilde{v}_2$ can be written as,

$$\begin{pmatrix} \dot{\tilde{v}}_1 \\ \\ \dot{\tilde{v}}_2 \end{pmatrix} = \begin{pmatrix} -v_2 & v_1 \\ \\ v_1 & v_2 \end{pmatrix} \begin{pmatrix} c_2\delta\gamma \\ \\ c_1\dfrac{\delta v_g}{v_g} \end{pmatrix} + \begin{pmatrix} -\dot{v}_1^d \\ \\ -\dot{v}_2^d \end{pmatrix} \tag{3.5}$$

The control objective is then to ensure that $\tilde{v}_1$ and $\tilde{v}_2 \to 0$ as $t \to \infty$. Notice the guidance objective based on this off-manifold dynamics is achieved by appropriate assignment of γ^c, v_g^c. Breaking away from the strict backstepping paradigm, we prescribe the desired off-manifold error dynamics to be, $\dot{\tilde{v}}_1 = -\lambda_1\tilde{v}_1$ and $\dot{\tilde{v}}_2 = -\lambda_2\tilde{v}_2$. Now the control laws are obtained from Eq. (3.5) as follows,

$$\begin{pmatrix} c_2\delta\gamma \\ \\ c_1\dfrac{\delta v_g}{v_g} \end{pmatrix} = -\frac{1}{v_g} \begin{pmatrix} -\sin\gamma & \cos\gamma \\ \\ \cos\gamma & \sin\gamma \end{pmatrix} \begin{pmatrix} -\dot{v}_1^d + \lambda_1\tilde{v}_1 \\ \\ -\dot{v}_2^d + \lambda_2\tilde{v}_2 \end{pmatrix}$$

Therefore, the control signal for the ground speed tracking is

$$v_g^c = v_g - \frac{1}{c_1}\left[\cos\gamma(-\dot{v}_1^d + \lambda_1\tilde{v_1}) + \sin\gamma(-\dot{v}_2^d + \lambda_2\tilde{v_2})\right] \tag{3.6}$$

and the control signal for the flight path angle tracking is

$$\gamma^c = \gamma - \frac{1}{c_2 v_g}\left[-\sin\gamma(-\dot{v}_1^d + \lambda_1\tilde{v_1}) + \cos\gamma(-\dot{v}_2^d + \lambda_2\tilde{v_2})\right] \tag{3.7}$$

Similarly, for the course angle (ground track) tracking,

$$\chi^c = \chi - \frac{1}{c_3}(-\dot{\chi}^d + \lambda_3\tilde{\chi}) \tag{3.8}$$

Note, $\lambda_1 > 0$, $\lambda_2 > 0$ and $\lambda_3 > 0$ are user chosen control gains that govern the off-manifold error dynamics.

The expressions for the $\dot{v_1}^d$,$\dot{v_2}^d$ and $\dot{\chi}^d$ can be derived from the following:

$$\begin{aligned} v_1^d &= \sqrt{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2} \\ v_2^d &= -\alpha_3 e_z + \dot{z}_r \\ \chi^d &= \tan^{-1}\left(\frac{-\alpha_2 e_y + \dot{y}_r}{-\alpha_1 e_x + \dot{x}_r}\right) \end{aligned}$$

and thus,

$$\begin{aligned}
\dot{v}_1^d &= (-\alpha_1 e_x + \dot{x}_r)(-\alpha_1 v_1 \cos\chi + \ddot{x}_r + \alpha_1 \dot{x}_r) + (-\alpha_2 e_y + \dot{y}_r)(-\alpha_2 v_1 \sin\chi + \ddot{y}_r + \alpha_2 \dot{y}_r) \\
\dot{v}_2^d &= -\alpha_3 v_2 + \ddot{z}_r + \alpha_3 \dot{z}_r \\
\dot{\chi}^d &= \frac{1}{v_1^d}\big[\cos\chi^d(-\alpha_2 v_1 \sin\chi + \ddot{y}_r + \alpha_2 \dot{y}_r) - \sin\chi^d(-\alpha_1 v_1 \cos\chi + \ddot{x}_r + \alpha_1 \dot{x}_r)\big] \qquad (3.9)
\end{aligned}$$

Note, when $v_1^d = 0$, $\dot{\chi}^d$ is not defined. However, $v_1^d = 0 \Rightarrow (-\alpha_1 e_x + \dot{x}_r) = 0$ and $(-\alpha_2 e_y + \dot{y}_r) = 0$. Note, if the target maneuver is along the z axis (a straight line trajectory), then $\dot{\chi}^d$ can reach unreasonably high values when the position errors in x and $y \rightarrow 0$. To avoid these singularities from affecting the control law, we modify the $\dot{\chi}^d$ signal as follows,

$$\begin{aligned}
\dot{\chi}^d &= \frac{v_g}{v_g\ v_1^d}\big[\cos\chi^d(-\alpha_2 v_1 \sin\chi + \ddot{y}_r + \alpha_2 \dot{y}_r) - \sin\chi^d(-\alpha_1 v_1 \cos\chi + \ddot{x}_r + \alpha_1 \dot{x}_r)\big] \\
&= \frac{v_g}{v_1^d}\Big(\frac{1}{v_g}\big[\cos\chi^d(-\alpha_2 v_1 \sin\chi + \ddot{y}_r + \alpha_2 \dot{y}_r) - \sin\chi^d(-\alpha_1 v_1 \cos\chi + \ddot{x}_r + \alpha_1 \dot{x}_r)\big]\Big) \\
&= \frac{1}{\kappa}\dot{\chi}_m^d \qquad (3.10)
\end{aligned}$$

where $\kappa = \frac{v_1^d}{v_g}$ and $\dot{\chi}_m^d = \frac{1}{v_g}\big[\cos\chi^d(-\alpha_2 v_1 \sin\chi + \ddot{y}_r + \alpha_2 \dot{y}_r) - \sin\chi^d(-\alpha_1 v_1 \cos\chi + \ddot{x}_r + \alpha_1 \dot{x}_r)\big]$. Thus, $\dot{\chi}^d$ is implemented as $\dot{\chi}^d = \dot{\chi}_m^d$ when $\|v_1^d\| \leq \epsilon$ ($\epsilon << 1$ is a specified tolerance). Note, $\dot{\chi}_m^d$ doesn't contain any singularity. Thus the control law for course angle tracking is modified to,

$$\chi^c = \chi - \frac{1}{c_3}(-\dot{\chi}_m^d + \lambda_3 \tilde{\chi}) \qquad (3.11)$$

when $\|v_1^d\| \leq \epsilon$ which is in turn translated to a maximum desired course angle rate constraint. Note, when $\|v_1^d\| = \epsilon$, $(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2 = \epsilon^2$. Thus the implementation of the control laws checks this condition to determine the maximum desired course angle rate constraint. The condition is quite easy to check since it is based only on the reference trajectory and the position errors in the horizontal plane.

The closed loop stability of the controller is verified through a Lyapunov stability analysis. To that effect, let's choose a candidate Lyapunov function as below.

$$W = \frac{1}{2}(\tilde{v}_1^2 + \tilde{v}_2^2 + \tilde{\chi}^2 + {e_x}^2 + {e_y}^2 + {e_z}^2)$$

The time derivative of W is

$$\dot{W} = \tilde{v}_1\dot{\tilde{v}}_1 + \tilde{v}_2\dot{\tilde{v}}_2 + \tilde{\chi}\dot{\tilde{\chi}} + e_x\dot{e}_x + e_y\dot{e}_y + e_z\dot{e}_z$$

When $\|v_1^d(0)\| > \epsilon$, using the control laws described above, it can easily be shown that the time derivative of the Lyapunov function candidate is

$$\dot{W} \;=\; -\lambda_1\tilde{v}_1^2 - \lambda_2\tilde{v}_2^2 - \lambda_3\tilde{\chi}^2 - \alpha_1{e_x}^2 - \alpha_2{e_y}^2 - \alpha_3{e_z}^2$$

Thus the errors asymptotically converge to the residual set $(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2 = \epsilon^2$. If the reference trajectory is a straight line along z direction, $\dot{x}_r = 0$ and $\dot{y}_r = 0$ and the errors are bounded in $\frac{e_x^2}{a^2} + \frac{e_y^2}{b^2} = 1$ where $a = \frac{\epsilon}{\alpha_1}$ and $b = \frac{\epsilon}{\alpha_2}$. Note, that the errors can be made arbitrarily small by using sufficiently large values of α_1 and α_2.

From above, it is clear that except for the special case mentioned above exponential stability can be obtained using the designed guidance laws. Thus, $\tilde{v}_1$, $\tilde{v}_2$, $\tilde{\chi}$, e_x, e_y and e_z all go to zero as $t \to \infty$. Thus, $x \to x_r$, $y \to y_r$, $z \to z_r$, $v_1 \to v_1^d$, $v_2 \to v_2^d$ and $\chi \to \chi^d$. Since e_x and $e_y \to 0$, $\chi^d \to \tan^{-1}\left(\frac{\dot{y}_r}{\dot{x}_r}\right)$ i.e. $\chi^d \to \chi_r$ and hence $\chi \to \chi_r$. Notice, that since e_x, $e_y \to 0$, $v_1^d \to \sqrt{\dot{x}_r^2 + \dot{y}_r^2}$, i.e. $v_1^d \to v_{gr}\sin\gamma_r$. Also, since $e_z \to 0$, $v_2^d \to v_{gr}\cos\gamma_r$. Thus, $v_1 \to v_{gr}\sin\gamma_r$ and $v_2 \to v_{gr}\cos\gamma_r$. It can now be trivially shown that $v_g \to v_{gr}$ and $\gamma \to \gamma_r$. Thus the guidance objective is achieved.

Also for the special case mentioned above, it is shown that the errors can be made arbitrarily small with an appropriate choice of the control gains. The trajectories are uniformly ultimately bounded in this case. □

Case 2: Target Model Eq. (2.7) - Chaser Model Eq. (2.6)

Theorem (Main Result)**.** Given the reference trajectory obtained from Eq. (2.7) and the actual dynamics of the UAV in Eq. (2.6), the following guidance laws ensure that $\|x - x_r\|$, $\|y - y_r\|$ and $\|z - z_r\| \to 0$ and all other signals are bounded.

$$\begin{aligned} v_g^c &= v_g - \frac{1}{c_1}\left[-\dot{v}_g^d + \lambda_{vg}\ \tilde{v}_g\right] \\ a_h^c &= v_g \cos\gamma(-\lambda_\chi\ \tilde{\chi} + \dot{\chi}^d) \\ a_v^c &= v_g(-\lambda_\gamma\ \tilde{\gamma} + \dot{\gamma}^d) \end{aligned} \tag{3.12}$$

We apply the similar technique as described for case 1 to get the following desired virtual control inputs for v_g^d, γ^d, and χ^d:

$$\begin{aligned} v_g^d &= \sqrt{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2 + (-\alpha_3 e_z + \dot{z}_r)^2} \\ \chi^d &= \tan^{-1}\left(\frac{-\alpha_2 e_y + \dot{y}_r}{-\alpha_1 e_x + \dot{x}_r}\right) \\ \gamma^d &= \tan^{-1}\left(\frac{-\alpha_3 e_z + \dot{z}_r}{\sqrt{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2}}\right) \\ &= \tan^{-1}\left(\frac{P}{Q}\right) \end{aligned} \tag{3.13}$$

where $P \triangleq -\alpha_3 e_z + \dot{z}_r$, and $Q \triangleq \sqrt{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2}$. We can determine the derivatives of v_g^d, χ^d and γ^d from Eq. (3.13) as required by the actual inputs:

$$\begin{aligned} \dot{v}_g^d &= \frac{1}{v_g^d}\left((-\alpha_1 e_x + \dot{x}_r)(-\alpha_1 v_g \cos\gamma\cos\chi + \alpha_1 \dot{x}_r + \ddot{x}_r)\right. \\ &\quad +(-\alpha_2 e_y + \dot{y}_r)(-\alpha_2 v_g \cos\gamma \sin\chi + \alpha_2 \dot{y}_r + \ddot{y}_r) \\ &\quad \left.+(-\alpha_3 e_z + \dot{z}_r)(-\alpha_3 v_g \sin\gamma + \alpha_3 \dot{z}_r + \ddot{z}_r)\right) \\ \dot{\chi}^d &= \frac{(-\alpha_2 v_g \cos\gamma \sin\chi + \alpha_2 \dot{y}_r + \ddot{y}_r)(-\alpha_1 e_x + \dot{x}_r)}{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2} \\ &\quad -\frac{(-\alpha_2 e_y + \dot{y}_r)(-\alpha_1 v_g \cos\gamma\cos\chi + \alpha_1 \dot{x}_r + \ddot{x}_r)}{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2} \end{aligned}$$

Table 3.1. Parameters used in simulation

Parameter	α_1	α_2	α_3	c_1	c_2	c_3	λ_1	λ_2	λ_3
Case 1	1	1	1	10	10	10	2	2	2
Case 2	2	2	2	10	10	10	2	2	2

$$\dot{\gamma}^d = \frac{\dot{P}Q - P\dot{Q}}{P^2 + Q^2}$$

where

$$\begin{aligned}
\dot{P} &= -\alpha_3 v_g \sin\gamma + \alpha_3 \dot{z}_r + \ddot{z}_r \\
\dot{Q} &= \frac{(-\alpha_1 e_x + \dot{x}_r)(-\alpha_1 v_g \cos\gamma\cos\chi + \alpha_1 \dot{x}_r + \ddot{x}_r)}{\sqrt{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2}} \\
&+ \frac{(-\alpha_2 e_y + \dot{y}_r)(-\alpha_2 v_g \cos\gamma\sin\chi + \alpha_2 \dot{y}_r + \ddot{y}_r)}{\sqrt{(-\alpha_1 e_x + \dot{x}_r)^2 + (-\alpha_2 e_y + \dot{y}_r)^2}}
\end{aligned}$$

Simulation Case Studies

A curved path trajectory for the target UAV is generated. The initial conditions for the target trajectory are: $x_r(0) = 100\ m$, $y_r(0) = 100\ m$, $z_r(0) = 100\ m$,$v_{gr} = 10\ m/s$, $\gamma_r(0) = 8°$, $\chi_r(0) = 8°$. The control inputs for target UAV: case 1 : $v_{gr}^c = 12\ m/s$, $a_{hr}^c = 0.2\ m/s^2$, and $a_{vr}^c = 0.2\ m/s^2$, and case 2: $v_{gr}^c = 10\ m/s$, $\gamma_r^c = 0\ deg/s$, and $\chi_r^c = 20\ deg/s$. The initial conditions of the chaser actual vehicle are: $x(0) = 50\ m$, $y(0) = 50\ m$, $z(0) = 50\ m$, $v_g(0) = 8\ m/s$, $\gamma(0) = 10°$, $\chi(0) = 10°$. The actuator constants and other gain parameters are listed in Table 3.1 for both cases. The chaser UAV successfully achieves tracking of target UAV for both cases as we can see from Fig. 3.2, and 3.7. All the tracking errors reach to zero with time as we see from Fig. 3.3, Fig. 3.4, Fig. 3.5, and 3.6 for case 1, and Fig. 3.8, Fig. 3.9, Fig. 3.10, and 3.11 for case 2.

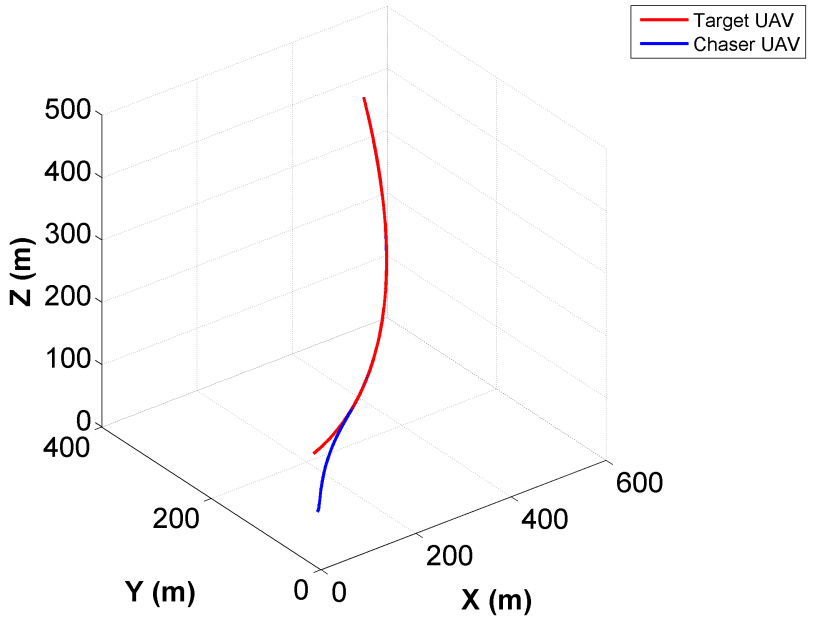


Figure 3.2. XYZ motion with time in $3D$ space (Case 1).

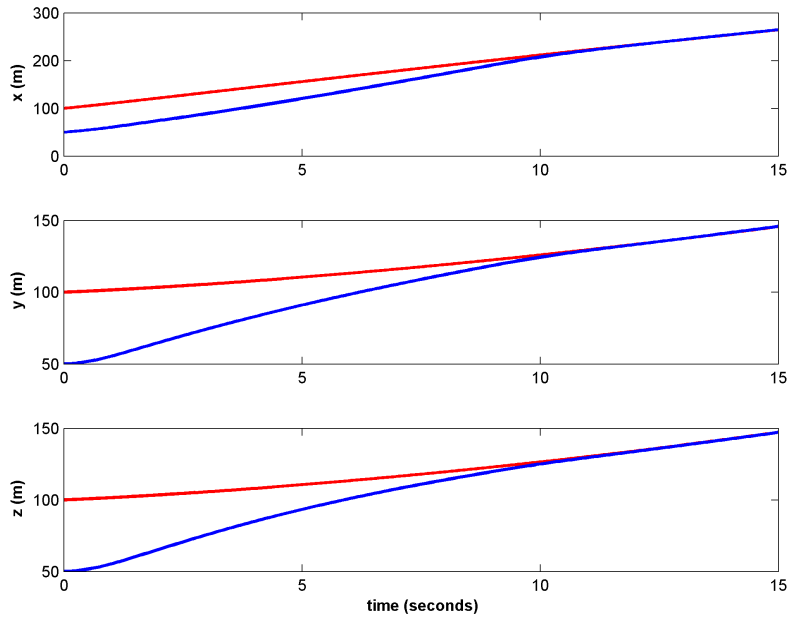


Figure 3.3. X, Y, and Z Position with time (Case 1).

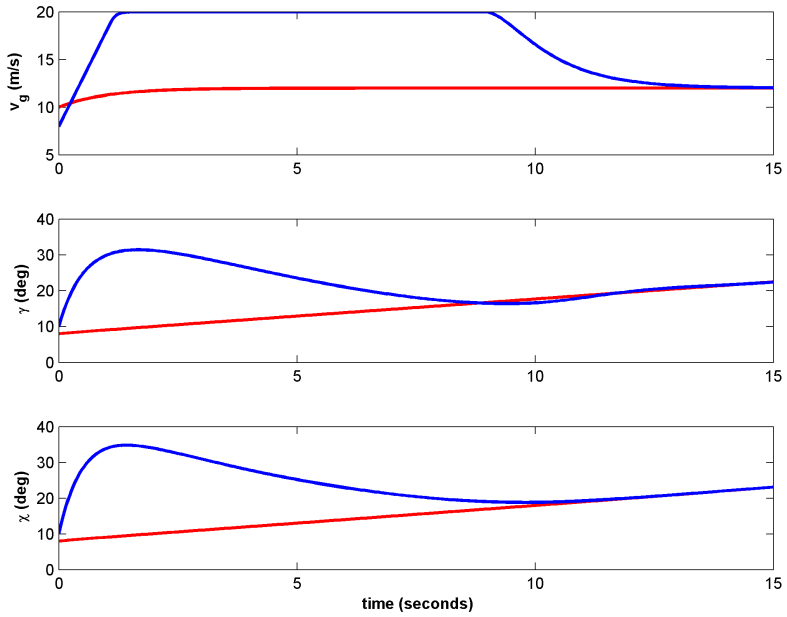


Figure 3.4. Velocity, Flight path, and Heading angle with time (Case 1).

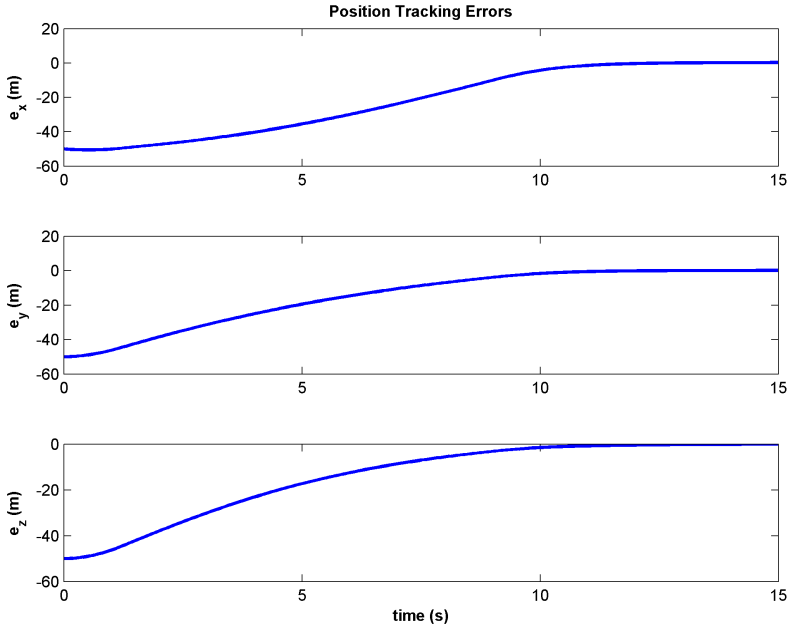


Figure 3.5. Position Tracking Error with time (Case 1).

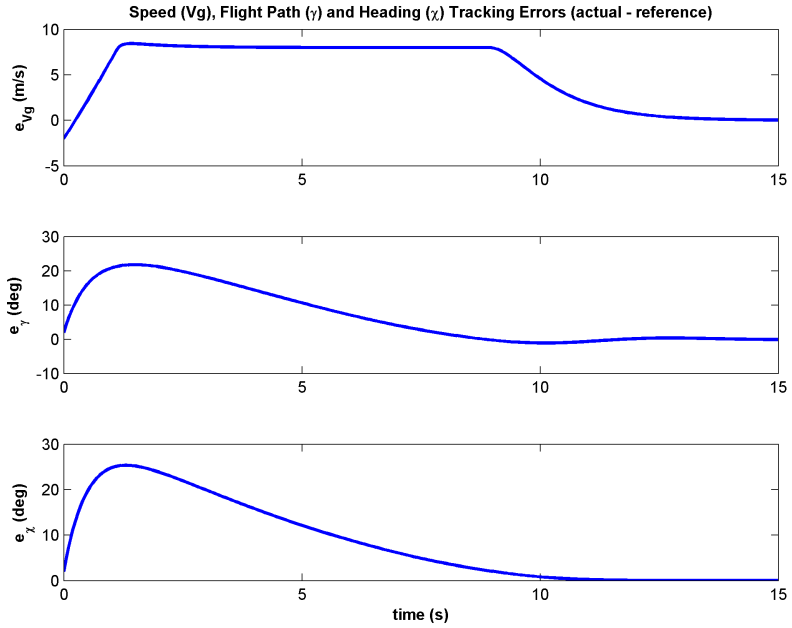


Figure 3.6. Velocity, Flight path, and Heading angle error with time (Case 1).

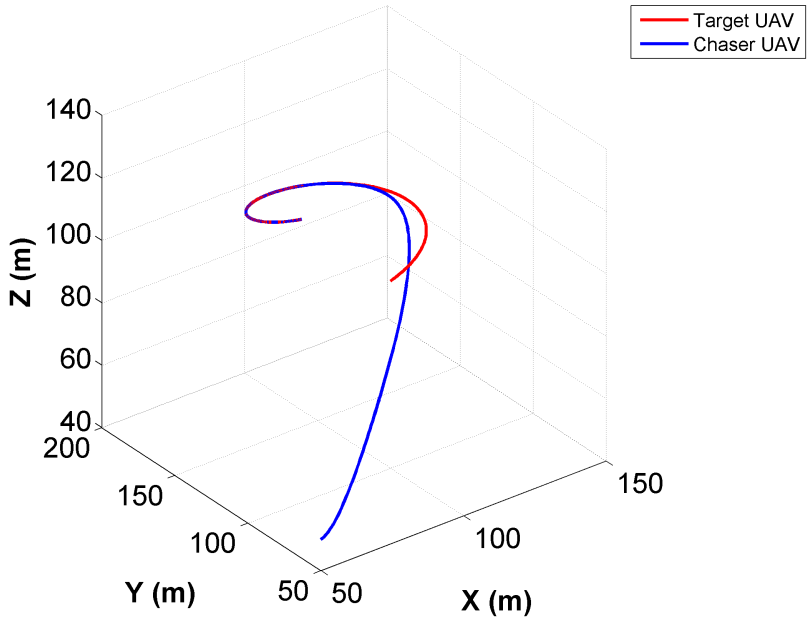


Figure 3.7. XYZ motion with time in $3D$ space (Case 2).

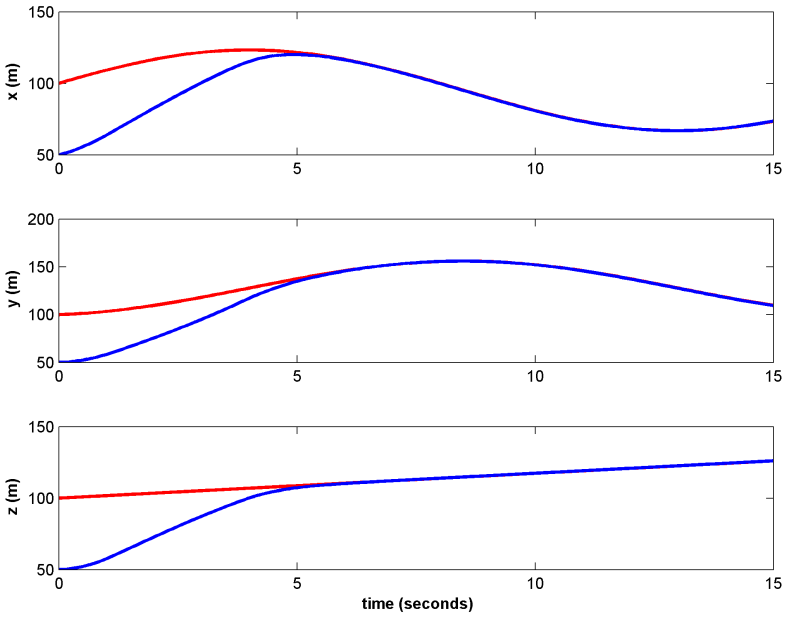


Figure 3.8. X, Y, and Z Position with time (Case 2).

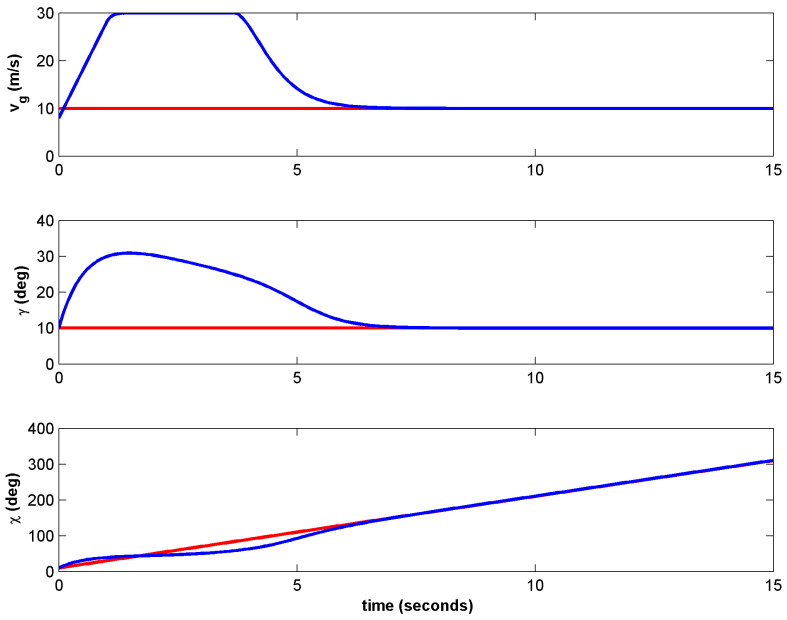


Figure 3.9. Velocity, Flight path, and Heading angle with time (Case 2).

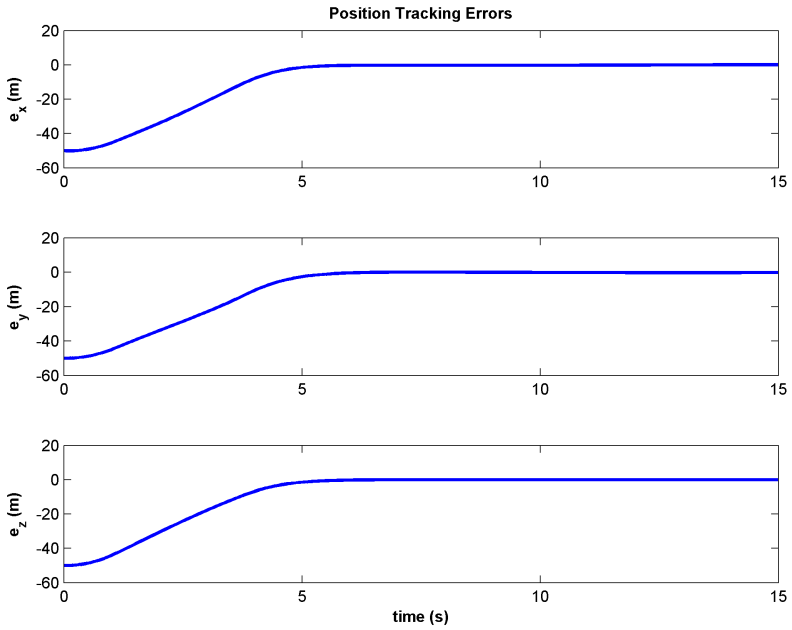


Figure 3.10. Position Tracking Error with time (Case 2).

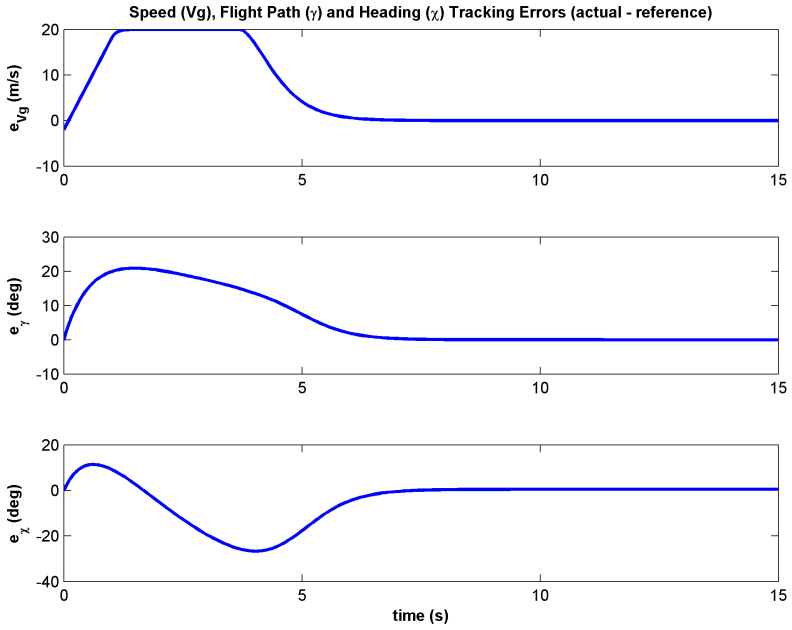


Figure 3.11. Velocity, Flight path, and Heading angle error with time (Case 2).

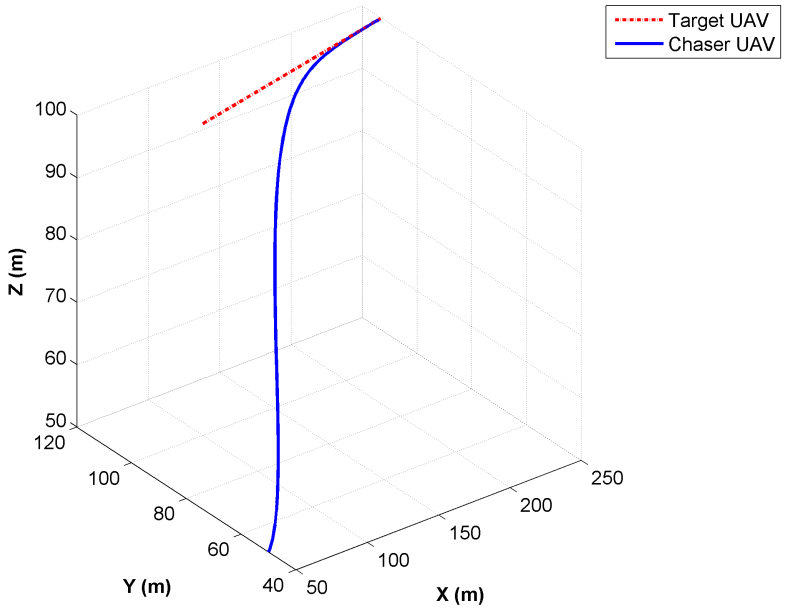


Figure 3.12. Tracking of a straight line trajectory.

We generate the results for some other general defined target trajectories with chaser dynamics described in Eq. (2.1), and target dynamics Eq. (2.7) is similar to chaser Eq. (2.1) with specified known control inputs. Target tracking is shown for the trajectories such as straight path at constant altitude in Fig. 3.12, circular path in Fig. 3.13, and, helical path as shown in Fig. 3.14. We can easily see that the controller works well for all these cases but tuning of the gains are required accordingly.

For the rest of the dissertation, we consider the chaser dynamics as described in Eq. (2.1) and the target dynamics in Eq. (2.7) as similar to chaser dynamics in Eq. (2.6) with specified known control inputs for v_{gr}^c, a_{hr}^c, and a_{vr}^c.

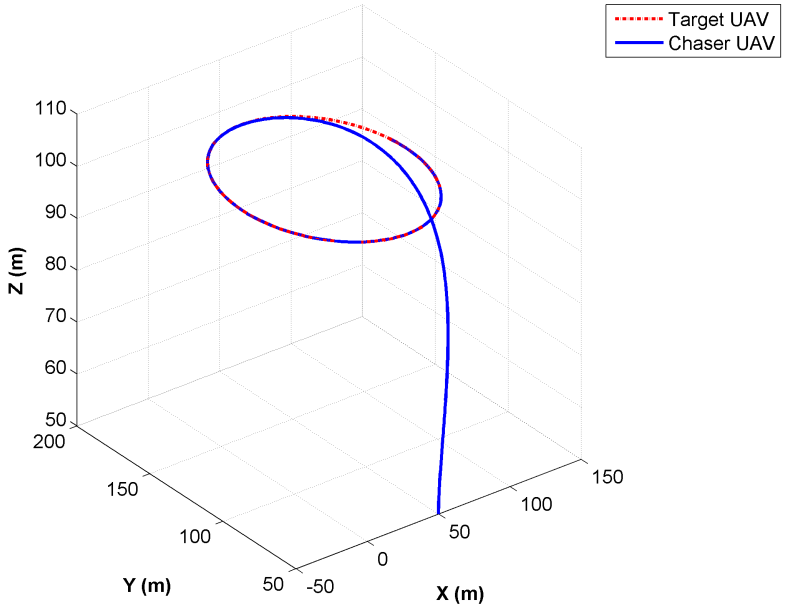


Figure 3.13. Tracking of a circular trajectory.

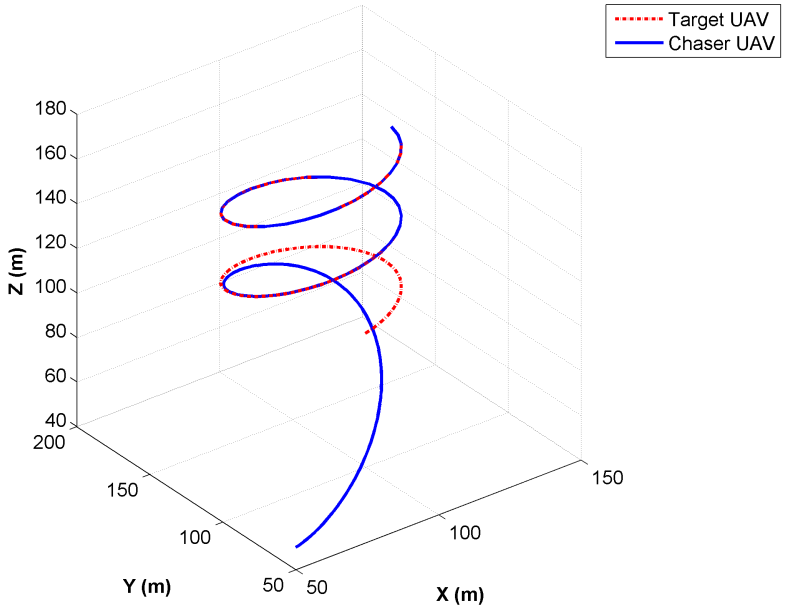


Figure 3.14. Tracking of a helical trajectory.

3.2 Estimation based Control laws for a single UAV (Partial Target Information with White Noise Uncertainties)

In this section, an estimation based control laws for a single UAV is developed and implemented in MATLAB to show robust performance of the controller in target tracking. We assume that the target full state information is not available to the chaser. However, an indirect measurements for range, azimuth angle, and elevation angle are available from on-board sensors. We can synthesize these state measurements in an Extended Kalman Filter (EKF) and estimate the target's 6 state measurements. An EKF provides the optimal estimates of the system states synthesizing available measurements assuming a priori known statistical models for the system and measurement noises. Fig. 3.15 shows an overview of our design approach to develop an estimation based control laws for UAV. The standard EKF equations for both cases are developed in this section based on reference [49]:

Continuous-Discrete EKF for Target state estimates

We consider the discrete-time measurement model Eq. (2.8) which is affected by the measurement noise uncertainties assumed to be zero-mean Gaussian white noise with known covariance. Therefore, the measurements for range, azimuth, and elevation angle are:

$$\begin{aligned} r_{rk} &= \sqrt{x_{rk}^2 + y_{rk}^2 + z_{rk}^2} + v_{rrk} \\ \phi_{rk} &= \tan^{-1}\left(\frac{y_{rk}}{x_{rk}} + v_{\phi rk}\right) \\ \theta_{rk} &= \sin^{-1}\left(\frac{z_{rk}}{r_{rk}} + v_{\theta rk}\right) \end{aligned} \tag{3.14}$$

where $v_{rrk},\ v_{\phi rk},\ v_{\theta rk}$ are the measurement noises in range, azimuth, and elevation angle measurements respectively at k time step. We also consider that the dynamic model for target UAV is corrupted by white noise uncertainties with known covari-

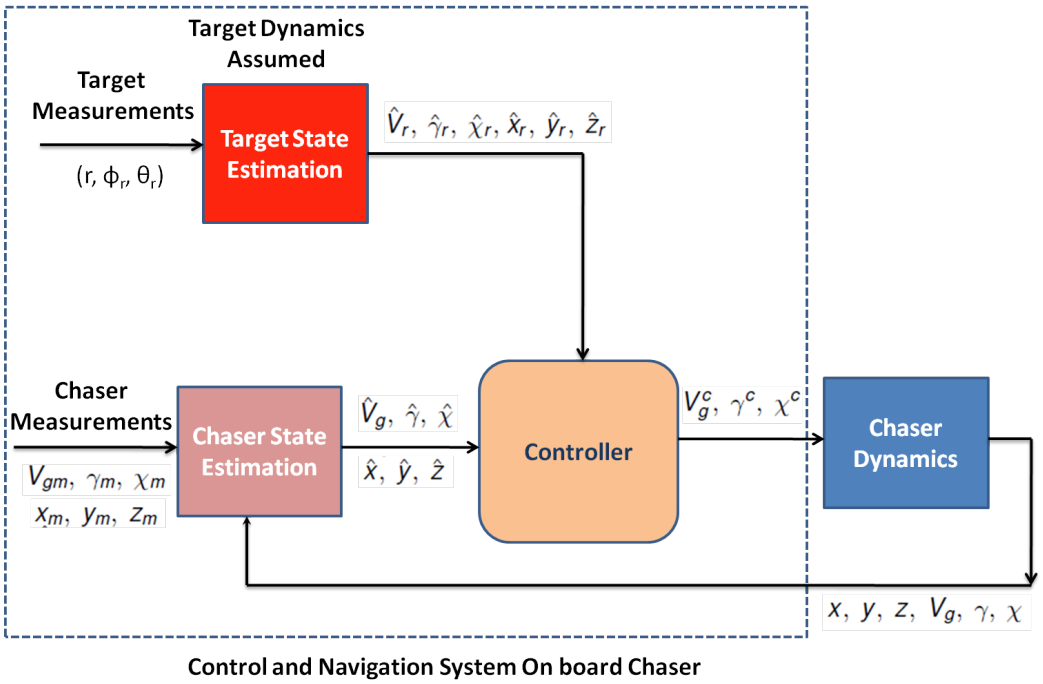


Figure 3.15. Estimation based Target Tracking.

ance. The continuous-time state model and the discrete-time measurements for reference UAV can be written as follows :

$$\begin{aligned} \dot{\mathbf{X}}_r(t) &= \mathbf{f}_r(\mathbf{X}_r(t), \mathbf{u}_r(t), t) + \mathbf{G}_r(t)\mathbf{w}_r(t) \\ \tilde{\mathbf{y}}_{rk} &= \mathbf{h}_r(\mathbf{X}_{rk}) + \mathbf{v}_{rk} \end{aligned} \tag{3.15}$$

where $\mathbf{X}_r = [x_r,\ y_r,\ z_r,\ v_{gr},\ \gamma_r,\ \chi_r]^T$ is the state vector; $\tilde{\mathbf{y}}_{rk}$ is the measurement vector; $\mathbf{w}_r = [0,\ 0,\ 0,\ \omega_{vgr},\ \omega_{\gamma r},\ \omega_{\chi r}]^T$ is $\sim N(0, \mathbf{Q}_r(t))$ which means that it is zero-mean normally distributed white noise vector with the process noise covariance $\mathbf{Q}_r(t) = E\{\mathbf{w}_r(t)\mathbf{w}_r^T(t)\}$; $\mathbf{v}_{rk} = [v_{rrk},\ v_{\phi rk},\ v_{\theta rk}]^T$ is $\sim N(0, \mathbf{R}_{rk})$ with measurement noise covariance $\mathbf{R}_{rk}$ as defined by $\mathbf{R}_{rk} = E\{\mathbf{v}_{rk}\mathbf{v}_{rk}^T\}$. We further assume that the process noise and the measurement noise are uncorrelated i.e. $E\{\mathbf{w_r}(t)\mathbf{v}_{rk}^T\} = 0$.

The nonlinear continuous-time vector function $\mathbf{f}_r(\mathbf{X}_r(t), \mathbf{u}_r(t), t)$ is expressed as:

$$\mathbf{f}_r(\mathbf{X}_r(t), \mathbf{u}_r(t), t) = \begin{bmatrix} v_{gr} \cos\gamma_r \cos\chi_r \\ v_{gr} \cos\gamma_r \sin\chi_r \\ v_{gr} \sin\gamma_r \\ v^c_{gr} - v_{gr} \\ \frac{a^c_{vr}}{v_{gr}} \\ \frac{a^c_{hr}}{v_{gr} \cos\gamma_r} \end{bmatrix} \tag{3.16}$$

where $\mathbf{u}_r(t) = \left[v^c_{gr},\ a^c_{hr},\ a^c_{vr}\right]^T$. The nonlinear discrete-time function vector in the measurement model is: $\mathbf{h}_r(\mathbf{X}_k) = [r_{rk},\ \phi_{rk},\ \theta_{rk}]^T$.

The Kalman gain of the EKF is:

$$\mathbf{K}_{rk} = \mathbf{P}^-_{rk}\mathbf{H}^T_{rk}(\hat{\mathbf{X}}^-_{rk}) \left[\mathbf{H}_{rk}(\hat{\mathbf{X}}^-_{rk})\mathbf{P}^-_{rk}\mathbf{H}^T_{rk}(\hat{\mathbf{X}}^-_{rk}) + \mathbf{R}_{rk}\right]^{-1}$$

The update equations for the state and covariance are:

$$\hat{\mathbf{X}}^+_{rk} = \hat{\mathbf{X}}^-_{rk} + \mathbf{K}_{rk}[\tilde{\mathbf{y}}_{rk} - \mathbf{h}_r(\hat{\mathbf{X}}^-_{rk})] \tag{3.17}$$

$$\mathbf{P}^+_{rk} = [\mathbf{I} - \mathbf{K}_{rk}\mathbf{H}_{rk}(\hat{\mathbf{X}}^-_{rk})]\mathbf{P}^-_{rk} \tag{3.18}$$

The estimated state propagation is governed by:

$$\dot{\hat{\mathbf{X}}}(t) = \mathbf{f}(\hat{\mathbf{X}}_r(t), \mathbf{u}_r(t), t)$$

and the estimation error covariance is propagated using:

$$\begin{aligned} \dot{\mathbf{P}}_r(t) &= \mathbf{F}_r(\hat{\mathbf{X}}_r(t), t)\mathbf{P}_r(t) + \mathbf{P}_r(t)\mathbf{F}_r(\hat{\mathbf{X}}_r(t), t)^T \\ &\quad + \mathbf{G}_r(t)\mathbf{Q}_r(t)\mathbf{G}_r(t)^T \end{aligned} \tag{3.19}$$

where $\mathbf{G}_r(t) = [0_{3\times3} \quad \mathbf{I}_{3\times3}]^T$, and $\mathbf{I}$ is the Identity matrix.

Assume, $\mathbf{f}_r$, and $\mathbf{h}_r$ function vectors are locally differentiable. We can determine the state Jacobian matrix $\mathbf{F}_r(\hat{\mathbf{X}}_r(t), t) = \frac{\partial \mathbf{f}_r}{\partial \mathbf{X}_r}|_{\hat{\mathbf{X}}_r(t)}$ as:

$$\mathbf{F}_r(\hat{\mathbf{X}}_r(t), t) = \begin{bmatrix} 0_{3\times 3} & \mathbf{F}_{rd1} \\ 0_{3\times 3} & \mathbf{F}_{rd2} \end{bmatrix}$$

where

$$\mathbf{F}_{rd1} = \begin{bmatrix} \cos\hat{\gamma}_r \cos\hat{\chi}_r & -\hat{v}_{gr}\sin\hat{\gamma}_r\cos\hat{\chi}_r & -\hat{v}_{gr}\cos\hat{\gamma}_r\sin\hat{\chi}_r \\ \cos\hat{\gamma}_r \sin\hat{\chi}_r & -\hat{v}_{gr}\sin\hat{\gamma}_r\sin\hat{\chi}_r & \hat{v}_{gr}\cos\hat{\gamma}_r\cos\hat{\chi}_r \\ \sin\hat{\gamma}_r & \hat{v}_{gr}\cos\hat{\gamma}_r & 0 \end{bmatrix}$$

$$\mathbf{F}_{rd2} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

and the measurement Jacobian matrix $\mathbf{H}_k(\hat{\mathbf{X}}_r(k)) = \frac{\partial \mathbf{h}_r}{\partial \mathbf{X}_r}$ evaluated at $\hat{\mathbf{X}}_{rk}$ as:

$$\mathbf{H}_{rk}(\hat{\mathbf{X}}_{rk}^-) = \begin{bmatrix} \frac{x_r}{r_r} & \frac{y_r}{r_r} & \frac{z_r}{r_r} & 0 & 0 & 0 \\ -\frac{y_r}{\sqrt{x_r^2+y_r^2}} & \frac{x_r}{\sqrt{x_r^2+y_r^2}} & 0 & 0 & 0 & 0 \\ -\frac{x_r z_r}{r_r^2\sqrt{x_r^2+y_r^2}} & -\frac{y_r z_r}{r_r^2\sqrt{x_r^2+y_r^2}} & \frac{\sqrt{x_r^2+y_r^2}}{r_r^2} & 0 & 0 & 0 \end{bmatrix}$$

We can write the output estimate equation as: $\hat{\mathbf{y}}_{rk} = \mathbf{h}_r(\hat{\mathbf{X}}_{rk})$.

The estimator above will synthesize, $\hat{x}_r$, $\hat{y}_r$, $\hat{z}_r$, $\hat{v}_{gr}$, $\hat{\gamma}_r$, $\hat{\chi}_r$. Additional derivatives that are needed are obtained by simply augmenting the filter state vector or using a "dirty-derivative" approximation [68].

The following state equations are used to determine the target state estimates and its derivatives:

$$\begin{aligned} \ddot{z}_d &= a\dot{z}_d + bz \\ \dot{z} &= c\dot{z}_d + dz \end{aligned}$$

The transfer function relates the output $\dot{z}(s)$ to input $z(s)$ as:

$$H(s) = \frac{qs + r}{s + p}$$

where $q \triangleq d$, $r \triangleq (cb - ad)$, and $p \triangleq -a$, and $a < 0$, $b > 0$, $c > 0$, and $d > 0$.

Continuous-Discrete EKF for chaser UAV state estimation

For the chaser UAV state estimation, we consider the nonlinear continuous state model Eq. (2.1) which is affected by the process noises and a discrete measurement model Eq. (2.10) which is affected by the measurement white noises as follows:

$$\begin{aligned} \dot{\mathbf{X}}(t) &= \mathbf{f}(\mathbf{X}(t), \mathbf{u}(t), t) + \mathbf{G}(t)\mathbf{w}(t), \\ \tilde{\mathbf{y}}_k &= \mathbf{h}(\mathbf{X}_k) + \mathbf{v}_k,\ \mathbf{v}_k \sim N(0, \mathbf{R}_k) \end{aligned} \tag{3.20}$$

where $\mathbf{w}(t) \sim N(0, \mathbf{Q}(t))$, $\mathbf{X} = [x,\ y,\ z,\ v_g,\ \gamma,\ \chi]^T$ is the state vector of the chaser, $\mathbf{w}_k = [0,\ 0,\ 0,\ \omega_{vg},\ \omega_\gamma,\ \omega_\chi]^T$ is the process noise vector and $\mathbf{v}_k = [v_x,\ v_y,\ v_z,\ v_{vg},\ v_\gamma,\ v_\chi]^T$ is the measurement noise vector, associated with the known covariances $\mathbf{Q}(t)$ and $\mathbf{R}_k$ respectively.

$\mathbf{f}(\mathbf{X}(t), \mathbf{u}(t), t)$ for the chaser UAV is expressed as:

$$\mathbf{f}(\mathbf{X}(t), \mathbf{u}(t), t) = \begin{bmatrix} v_g \cos\gamma \cos\chi \\ v_g \cos\gamma \sin\chi \\ v_g \sin\gamma \\ c_1(v_g^c - v_g) \\ c_2(\gamma^c - \gamma) \\ c_3(\chi^c - \chi) \end{bmatrix}$$

The measurements for the chaser are assumed to be available from an on-board GPS + IMU + Air speed sensor unit: $\mathbf{h}(\mathbf{X}_k) = [x,\ y,\ z,\ v_g,\ \gamma,\ \chi]_k^T$. The matrix $\mathbf{G}(t)$ is the similar to $\mathbf{G}_r(t)$. The Extended Kalman Filter for the chaser is derived in a similar way as the target. The Kalman gain is given by

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T(\hat{\mathbf{X}}_k^-) \left[\mathbf{H}_k^T(\hat{\mathbf{X}}_k^-)\mathbf{P}_k^- \mathbf{H}_k(\hat{\mathbf{X}}_k^-) + \mathbf{R}_k\right]^{-1}$$

The update equations for the state and covariance:

$$\begin{aligned} \hat{\mathbf{X}}_k^+ &= \hat{\mathbf{X}}_k^- + \mathbf{K}_k(\tilde{\mathbf{y}}_k - \mathbf{h}(\hat{\mathbf{X}}_k^-)) \\ \mathbf{P}_k^+ &= [\mathbf{I} - \mathbf{K}_k\mathbf{H}_k(\hat{\mathbf{X}}_k^-)]\mathbf{P}_k^- \end{aligned}$$

The predicted estimate state propagation is:

$$\dot{\hat{\mathbf{X}}}(t) = \mathbf{f}(\hat{\mathbf{X}}(t), \mathbf{u}(t), t)$$

The error covariance propagation is:

$$\dot{\mathbf{P}}(t) = \mathbf{F}(\hat{\mathbf{X}}(t), t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}^T(\hat{\mathbf{X}}(t), t) + \mathbf{G}(t)\mathbf{Q}(t)\mathbf{G}^T(t)$$

The Jacobian matrix $\mathbf{F}(\hat{\mathbf{X}}(t), t) = \frac{\partial \mathbf{f}}{\partial \mathbf{X}}|_{\hat{\mathbf{X}}(t)}$ is given as:

$$\mathbf{F}(\hat{\mathbf{X}}(t), t) = \begin{bmatrix} 0_{3\times3} & \mathbf{F}_d \\ 0_{3\times3} & \mathbf{diag}([-c_1,\ -c_2,\ -c_3]) \end{bmatrix}$$

wherein

$$\mathbf{F}_d = \begin{bmatrix} \cos\hat{\gamma}\cos\hat{\chi} & -\hat{v}_g\sin\hat{\gamma}\cos\hat{\chi} & -\hat{v}_g\cos\hat{\gamma}\sin\hat{\chi} \\ \cos\hat{\gamma}\sin\hat{\chi} & -\hat{v}_g\sin\hat{\gamma}\sin\hat{\chi} & \hat{v}_g\cos\hat{\gamma}\cos\hat{\chi} \\ \sin\hat{\gamma} & \hat{v}_g\cos\hat{\gamma} & 0 \end{bmatrix}$$

The jacobian matrix can be written as: $H_k(\hat{\mathbf{X}}_k^-) = \mathbf{I}_{6\times 6}$. We can get the estimated states of the target using the output estimate equation $\hat{\mathbf{y}}_k = \hat{\mathbf{X}}_k$.

Simulation Case Study

We simulate a scenario for following design parameters: $c_1 = 100,\ c_2 = 100,\ c_3 = 100,\ \alpha_1 = 1,\ \alpha_2 = 1,\ \alpha_3 = 1,\ \lambda = 50,\ \lambda_2 = 50,\ \lambda_3 = 50$. The simulation was run for 20 seconds and the data were updated $\Delta t = 0.05$ seconds interval i.e. the update rate is $20Hz$. The controller continuously updates three control commands based on the estimated target and chaser UAV states. The transfer function used to calculate estimated target state derivatives is:

$$H(s) \quad = \quad \frac{0.1s + 55}{s + 50}$$

The simulation results shows that tracking is achieved with quite good approximation.

The parameters used for the target UAV and chaser UAV are as follows: The truth model of target UAV is obtained for the following initial conditions: $v_{gr}(0) = 10\ m/s,\ \gamma_r(0) = 12^\circ,\ \chi_r(0) = 12^\circ, x_r(0) = 100\ m,\ y_r(0) = 100\ m,\ z_r(0) = 100\ m$, and for specified known control inputs: $v_{gr}^c = 12\ m/s,\ a_{hr}^c = 0.2\ m/s^2,\ a_{vr}^c = 0.2\ m/s^2$. The variances and standard deviation of noises associated with the sensors for target and chaser measurements are listed in Table 3.2, and 3.3.

The initial guess of the estimated states of reference trajectory are: $v_{gre}(0) = 11\ m/s,\ \gamma_{re}(0) = 17^\circ,\ \chi_{re}(0) = 17^\circ, x_{re}(0) = 105\ m,\ y_{re}(0) = 105\ m,\ z_{re}(0) = 105\ m.$

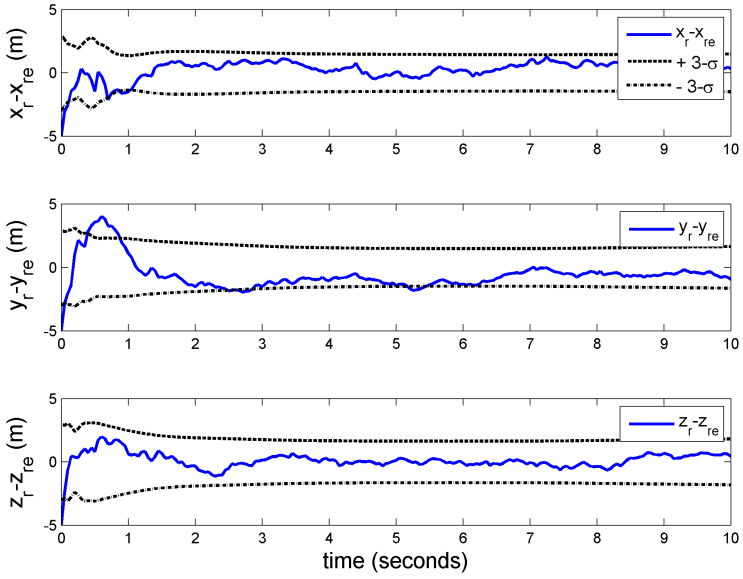


Figure 3.16. Position Tracking Errors (true-estimated) with $3-\sigma$ bounds for Target UAV.

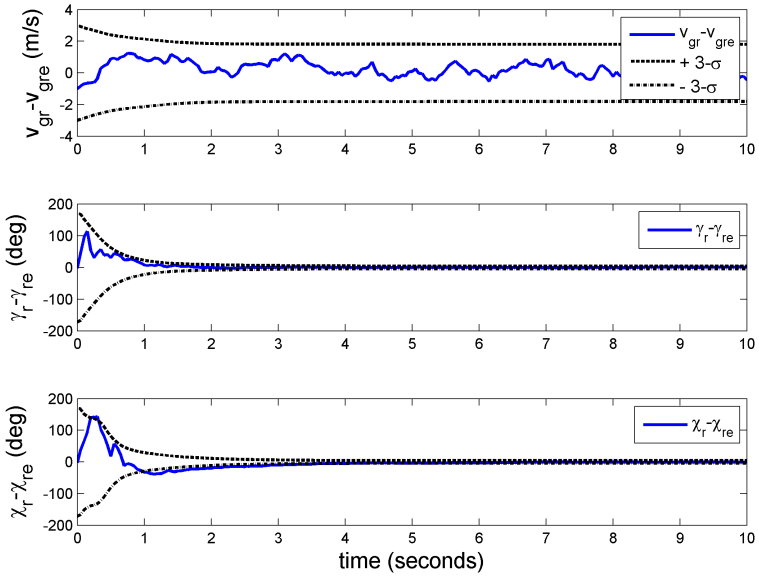


Figure 3.17. Velocity, Flight Path Angle, and Heading Angle Errors (true-estimated) with $3-\sigma$ bounds for Target UAV.

Table 3.2. Variances and Standard Deviations of measurement noises used in simulation (Target)

Parameter	r	ϕ	θ
σ^2	$1\ m^2$	$2e^{-4}\ rad^2$	$2e^{-4}\ rad^2$
σ	$\pm 1\ m$	$\pm 0.01\ rad\ (\pm 0.57\,\text{deg})$	$\pm 0.01\ rad\ (\pm 0.57\,\text{deg})$

Table 3.3. Variances and Standard Deviations of measurement noises used in simulation (Chaser)

Parameter	x	y	z	v_g	γ	χ
σ^2	$1\ m^2$	$1\ m^2$	$1\ m^2$	$1\ (m/s)^2$	$2e^{-4}\ rad^2$	$2e^{-4}\ rad^2$
σ	$\pm 1\ m$	$\pm 1\ m$	$\pm 1\ m$	$\pm 1\ m/s$	$\pm 0.01\ rad$ $(\pm 0.57\,\text{deg})$	$\pm 0.01\ rad$ $(\pm 0.57\,\text{deg})$

The true model of chaser UAV is obtained for the following conditions: $v_g(0) = 8\ m/s,\ \gamma(0) = 10^\circ,\ \chi(0) = 10^\circ,\ x(0) = 50\ m,\ y(0) = 50\ m,\ z(0) = 50\ m$, and for the estimated state: $v_{ge}(0) = 9\ m/s,\ \gamma_e(0) = 15^\circ,\ \chi_e(0) = 15^\circ,\ x_e(0) = 55\ m,\ y_e(0) = 55\ m,\ z_e(0) = 55\ m$. The initial covariance for the reference and chaser vehicle is chosen to be the same i.e. $P_0 = I_{6\times 6}$ where I is an identity matrix. The covariance of process noise for the target: $Q_r(t) = diag([1,\ (0.57\pi/180)^2,\ (0.57\pi/180)^2])$ and for chaser: $Q(t)$ is the same as target. The covariance matrix of the measurement noises for target and chaser are: $R_{rk} = diag([1,\ (0.57\pi/180)^2,\ (0.57\pi/180)^2])$ and $R_k = diag([1,\ 1,\ 1,\ 1,\ (0.57\pi/180)^2,\ (0.57\pi/180)^2])$, respectively.

Fig. 3.16, 3.17 and 3.18 shows the performance of the EKF which gives the best estimate of target true state. Fig. 3.19 shows the true and estimated target measurements - it validates the filter performance to estimate target states. Fig. 3.20, and Fig. 3.21 shows the performance of the filter with $3-\sigma$ error bounds and from

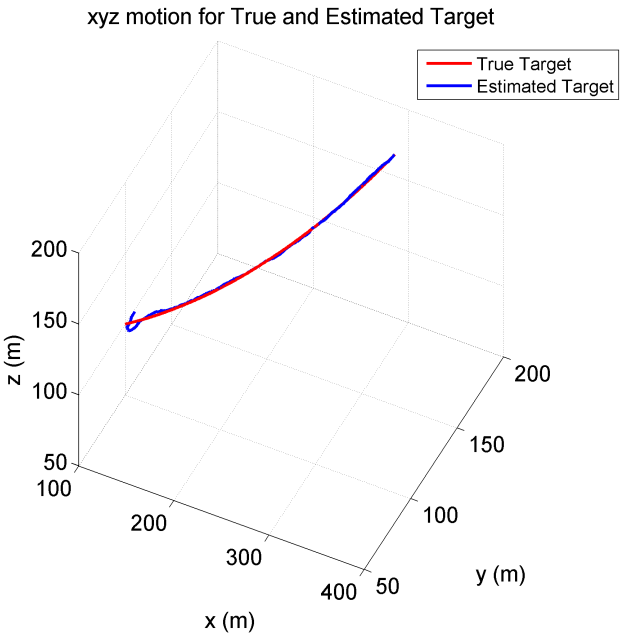


Figure 3.18. The true and estimated state of the reference vehicle.

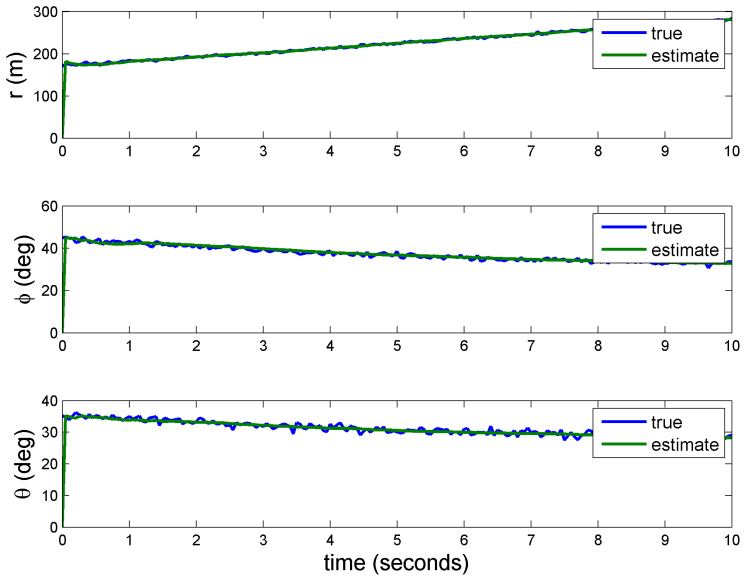


Figure 3.19. True and Estimated Target Measurements.

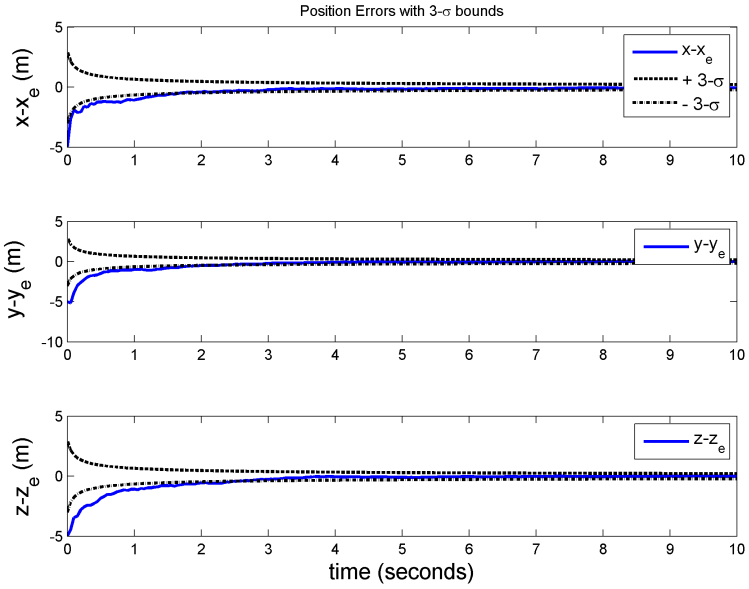


Figure 3.20. Position Tracking Errors (True-Estimated) with $3-\sigma$ bounds for Chaser.

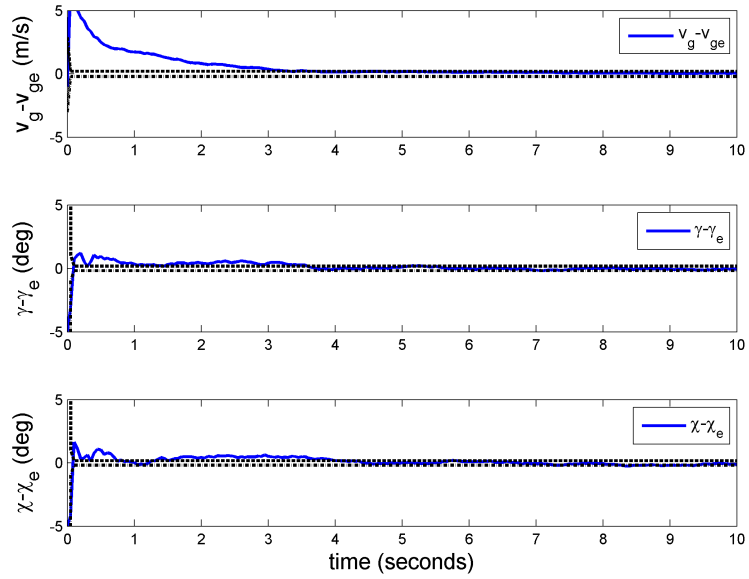


Figure 3.21. Velocity, Flight path, and Heading angle Errors (True-Estimated) with $3-\sigma$ bounds for Chaser.

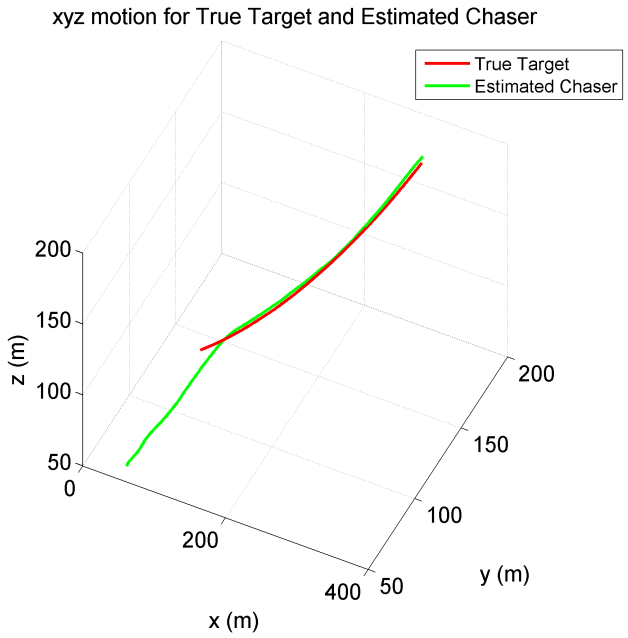


Figure 3.22. The estimated chaser trajectory tracks the true target trajectory.

Fig. 3.22, we see that the estimated XYZ motion of the chaser trajectory tracks the target UAV as expected.

3.3 Estimation based Control laws for a single UAV (Partial Target Information with Colored Noise Uncertainties)

In this section, a continuous-discrete EKF is implemented when the measurements and process noises are corrupted by colored noise uncertainties. Colored noise is propagated by solving first order differential equation with band limited white noise as an input into it. We can write the augmented system including the colored noise dynamics and design the EKF to estimate the augmented state.

EKF for Target State Estimation with Colored Noise Uncertainties

For target state estimation, an EKF is designed which can estimate the states when the measurements are corrupted by non-white noise measurement uncertainties. Three measurements (range (r), azimuth angle (ϕ), and elevation angle (θ)) of target are available to the chaser and the associated noise vector is $\mathbf{v}_{rk}$. The non-white measurement noise vector $\mathbf{v}_{rk}$ can be generated by the following first-order shaping filter equation which is driven by zero-mean band limited white noise:

$$\begin{aligned} \dot{\mathbf{Z}}_r &= \mathbf{f}_{rf}(\mathbf{Z}_r) + \boldsymbol{\vartheta}_{rf}\mathbf{w}_{rf}, \quad \mathbf{w}_{rf}(t) \sim N(0, \mathbf{Q}_{rf}(t)) \\ \mathbf{v}_{rk} &= \mathbf{h}_{rf}(\mathbf{Z}_{rk}) + \boldsymbol{\nu}_{rfk}, \ \boldsymbol{\nu}_{rfk} \ \sim N(0, \mathbf{R}_{rfk}) \end{aligned} \tag{3.21}$$

where, $Z_r = [z_{rr}, \ z_{\phi r}. \ z_{\theta r}]^T$, $\mathbf{f}_{zr} = [-b_{rr}, \ -b_{\phi r}, \ -b_{\theta r}]^T$, $\boldsymbol{\vartheta}_{rf} = \mathbf{diag}\left([\sqrt{b_{rr}}, \ \sqrt{b_{\phi r}}, \ \sqrt{b_{\theta r}}]\right)$, $h_{rf} = \mathbf{Z}_{rk}$, all b's are positive constants.

The state and measurement equations for the augmented system are as follows:

$$\begin{aligned} \dot{\mathbf{X}}_{ra} &= f_{ra}(\mathbf{X}_{ra}(t), \mathbf{u}_r(t), t) + \mathbf{G}_{ra}(t)\mathbf{w}_{ra}(t), \ \mathbf{w}_{ra}(t) \sim N(0, \mathbf{Q}_{ra}(t)) \\ \tilde{\mathbf{y}}_{rk} &= \mathbf{h}_{rk}(\mathbf{X}_{rk}) + \mathbf{v}_{rk} \\ &= \mathbf{h}_r(\mathbf{X}_{rk}) + \mathbf{h}_{rf}(\mathbf{Z}_{rk}) + \boldsymbol{\nu}_{rfk} \\ &= \mathbf{h}_{ra}(\mathbf{X}_{rak}) + \boldsymbol{\nu}_{rfk} \end{aligned} \tag{3.22}$$

where, $\mathbf{X}_{ra} = [\mathbf{X}_r \quad \mathbf{Z}_r]^T$, $\mathbf{f}_{ra} = [\mathbf{f}_r \quad \mathbf{f}_{rf}]^T$, $\mathbf{G}_{ra} = \begin{bmatrix} \mathbf{G}_r & \underline{0}_{6\times 3} \\ \underline{0}_{3\times 3} & \vartheta_{rf} \end{bmatrix}$, $\mathbf{w}_{ra} = [\mathbf{w}_r \quad \mathbf{w}_{rf}]^T$, and $\mathbf{h}_{ra}(\mathbf{X}_{rak}) = \mathbf{h}_{rk}(\mathbf{X}_{rk}) + \mathbf{h}_{rf}(\mathbf{Z}_{rk})$

Assume that $\mathbf{w}_r$, and $\mathbf{w}_{rf}$ are uncorrelated i.e. the process noise covariance matrix $\mathbf{Q}_r a(t)$ for the augmented system can be written by

$$\begin{aligned} \mathbf{Q}_{ra}(t) &= E\begin{bmatrix} \mathbf{w}_r \\ \mathbf{w}_{rf} \end{bmatrix} \begin{bmatrix} \mathbf{w}_r^T & \mathbf{w}_{rf}^T \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{Q}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{rf} \end{bmatrix} \end{aligned} \tag{3.23}$$

The measurement covariance matrix associated with the augmented measurement model:

$$E\left[\boldsymbol{\nu}_{rfk}\boldsymbol{\nu}_{rfk}^T\right] = \mathbf{R}_{rfk} \tag{3.24}$$

We can write the Kalman Filter equations in a similar way as mentioned in Eqn. (3.5-3.9) for the augmented system i.e.

$$\mathbf{K}_{rak} = \mathbf{P}_{rak}^- \, \mathbf{H}_{rak}^T(\hat{\mathbf{X}}_{rak}^-) \, [\mathbf{H}_{rak}^T(\hat{\mathbf{X}}_{rak}^-) \, \mathbf{P}_{rak}^- \, \mathbf{H}_{rak}(\hat{\mathbf{X}}_{rak}^-) + \mathbf{R}_{rak}]^{-1} \tag{3.25}$$

The update equations for the state and covariance:

$$\hat{\mathbf{X}}_{rak}^+ = \hat{\mathbf{X}}_{rak}^- + \mathbf{K}_{rak}(\tilde{\mathbf{y}}_{rk} - \mathbf{h}_{ra}(\hat{\mathbf{X}}_{rak}^-)) \tag{3.26}$$

$$\mathbf{P}_{rak}^+ = \left[\mathbf{I} - \mathbf{K}_{rak}\mathbf{H}_{rak}(\hat{\mathbf{X}}_{rak}^-)\right] \mathbf{P}_{rak}^- \tag{3.27}$$

The state estimate propagation is:

$$\dot{\hat{\mathbf{X}}}_{ra}(t) = \mathbf{f}_{ra}(\hat{\mathbf{X}}_{ra}(t), \mathbf{u}_r(t), t) \tag{3.28}$$

The error covariance propagation is:

$$\begin{aligned}\dot{\mathbf{P}}_{ra}(t) &= \mathbf{F}_{ra}(\hat{\mathbf{X}}_{ra}(t),t)\mathbf{P}_{ra}(t) + \mathbf{P}_{ra}(t)\mathbf{F}_{ra}^{T}(\hat{\mathbf{X}}_{ra}(t),t) \\ &\quad +\mathbf{G}_{ra}(t)\mathbf{Q}_{ra}(t)\mathbf{G}_{ra}^{T}(t) \end{aligned} \tag{3.29}$$

For the augmented system, the Jacobian matrices are given below:

The state Jacobian matrix $\mathbf{F}_{ra}(\hat{\mathbf{X}}_{ra}(t),t) = \frac{\partial \mathbf{f}_{ra}}{\partial \mathbf{X}_{ra}}|_{\hat{\mathbf{X}}_{ra}(t)}$ which becomes as follows:

$$\mathbf{F}_{ra} = \begin{bmatrix} \mathbf{F}_r & \underline{0}_{3\times 3} \\ \underline{0}_{6\times 6} & \mathbf{F}_{rf} \end{bmatrix}$$

where, $\mathbf{F}_{rf} = \mathbf{diag}\left([-b_{rr}, \quad -b_{\phi r}, \quad -b_{\theta r}]\right)$

The measurement Jacobian matrix $\mathbf{H}_{ak}(\hat{\mathbf{X}}_{rak}) = \frac{\partial \mathbf{h}_{ra}}{\partial \mathbf{X}_{rak}}|_{\hat{\mathbf{X}}_{rak}}$ can be expressed as: $\mathbf{H}_{rak}(\hat{\mathbf{X}}_{rk}^{-}) = [\mathbf{H}_{rk} \;\; \mathbf{H}_{rfk}]$ where, $\mathbf{H}_{rfk} = \mathbf{I}_{3\times 3}$.

EKF for Chaser State Estimation with Colored Noise Measurement Uncertainties

The chaser state measurements are considered to be corrupted by colored noise uncertainties. For the chaser state, six state measurements are available from sensors, and six first order shaping filters are utilized to generate the colored noise uncertainties and added these in the measurement model.

$$\begin{aligned}\dot{\mathbf{Z}} &= \mathbf{f}_f(\mathbf{Z}) + \boldsymbol{\vartheta}_f \mathbf{w}_f, \quad \mathbf{w}_f(t) \sim N(0, \mathbf{Q}_f(t)) \\ \mathbf{v}_k &= \mathbf{h}_f(\mathbf{Z}_k) + \boldsymbol{\nu}_{fk}, \; \boldsymbol{\nu}_{fk} \sim N(0, \mathbf{R}_{fk}) \end{aligned} \tag{3.30}$$

where, $Z = [z_x, \;\; z_y, \;\; z_z, \;\; z_{vg}, \;\; z_\gamma, \;\; z_\chi]^T$, $\mathbf{f}_f = [-b_x, \;\; -b_y, \;\; -b_z, \;\; -b_{vg}, \;\; -b_\gamma, \;\; -b_\chi]^T$, $\boldsymbol{\vartheta}_f = \mathbf{diag}\left([\sqrt{b_x}, \;\; \sqrt{b_y}, \;\; \sqrt{b_z}, \sqrt{b_{vg}}, \;\; \sqrt{b_\gamma}, \;\; \sqrt{b_\chi}]\right)$, $\mathbf{h}_f = \mathbf{Z}_k$, and all b's are positive constants.

We can write the augmented state and measurement equations for the chaser are as follows:

$$\begin{aligned}\dot{\mathbf{X}}_a &= f_a(\mathbf{X}_a(t), \mathbf{u}(t), t) + \mathbf{G}_a(t)\mathbf{w}_a(t),\ \mathbf{w}_a(t) \sim N(0, \mathbf{Q}_a(t)) \\ \tilde{\mathbf{y}}_k &= \mathbf{h}_a(\mathbf{X}_{ak}) + \boldsymbol{\nu}_{fk} \end{aligned} \tag{3.31}$$

where, $\mathbf{X}_a = [\mathbf{X},\ \ \mathbf{Z}]^T$, $\mathbf{f}_a = [\mathbf{f},\ \ \mathbf{f}_f]^T$, $\mathbf{G}_{ra} = \begin{bmatrix} \mathbf{G}_r & \underline{0}_{6\times 3} \\ \underline{0}_{3\times 3} & \boldsymbol{\vartheta}_f \end{bmatrix}$, $\mathbf{w}_a = [\mathbf{w},\ \ \mathbf{w}_f]^T$, and $\mathbf{h}_a(\mathbf{X}_{ak}) = \mathbf{h}(\mathbf{X}_k) + \mathbf{h}_f(\mathbf{Z}_{fk})$.

The Kalman Filter equations for chaser can be written as follows:

$$\mathbf{K}_{ak} = \mathbf{P}^-_{ak}\ \mathbf{H}^T_{ak}(\hat{\mathbf{X}}^-_{ak})\ [\mathbf{H}^T_{ak}(\hat{\mathbf{X}}^-_{ak})\ \mathbf{P}^-_{ak}\ \mathbf{H}_{ak}(\hat{\mathbf{X}}^-_{ak}) + \mathbf{R}_{ak}]^{-1} \tag{3.32}$$

Update equations for the state and covariance are:

$$\hat{\mathbf{X}}^+_{ak} = \hat{\mathbf{X}}^-_{ak} + \mathbf{K}_{ak}(\tilde{\mathbf{y}}_k - \mathbf{h}_a(\hat{\mathbf{X}}^-_{ak})) \tag{3.33}$$

$$\mathbf{P}^+_{ak} = [\mathbf{I} - \mathbf{K}_{ak}\mathbf{H}_{ak}(\hat{\mathbf{X}}^-_{ak})]\mathbf{P}^-_{ak} \tag{3.34}$$

State estimate propagation is:

$$\dot{\hat{\mathbf{X}}}_a(t) = \mathbf{f}_a(\hat{\mathbf{X}}_a(t), \mathbf{u}(t), t) \tag{3.35}$$

Error covariance propagation is:

$$\begin{aligned}\dot{\mathbf{P}}_a(t) &= \mathbf{F}_a(\hat{\mathbf{X}}_a(t), t)\mathbf{P}_a(t) + \mathbf{P}_a(t)\mathbf{F}^T_a(\hat{\mathbf{X}}_a(t), t) \\ &\quad + \mathbf{G}_a(t)\mathbf{Q}_a(t)\mathbf{G}^T_a(t)\end{aligned} \tag{3.36}$$

The state Jacobian matrix is: $\mathbf{F}_a(\hat{\mathbf{X}}_a(t), t) = \frac{\partial \mathbf{f}_a}{\partial \mathbf{X}_a}|_{\hat{\mathbf{X}}_a(t)}$ which becomes as follows:

$$\mathbf{F}_a = \begin{bmatrix} \mathbf{F} & \underline{0}_{3\times 3} \\ \underline{0}_{6\times 6} & \mathbf{F}_f \end{bmatrix}$$

Table 3.4. Variances of process noises used in simulation (Target)

Parameter	Value	Unit
σ^2_{vgr}	1	$(m/s)^2$
$\sigma^2_{\gamma r}$	$(0.65\pi/180)^2$	rad^2
$\sigma^2_{\chi_r}$	$(0.65\pi/180)^2$	rad^2
σ^2_{zrr}	4	m^2
$\sigma^2_{z\phi r}$	$(0.65\pi/180)^2$	rad^2
$\sigma^2_{z\theta r}$	$(0.65\pi/180)^2$	rad^2

where, $\mathbf{F}_f = \mathbf{diag}\left([-b_x,\ -b_y,\ -b_z,\ -b_{vg},\ -b_\gamma,\ -b_\chi]\right)$.

The measurement Jacobian matrix is: $\mathbf{H}_{ak}(\hat{\mathbf{X}}_{ak}) = \frac{\partial \mathbf{h}_{ak}}{\partial \mathbf{X}_{ak}}|_{\hat{\mathbf{X}}_{ak}}$ which can be written as:

$$\mathbf{H}_{ak}(\hat{\mathbf{X}}_k^-) = [\mathbf{H}_k \ \ \mathbf{H}_{fk}], \text{ where } \mathbf{H}_{fk} = \mathbf{I}_{6\times 6}.$$

Simulation Results

The simulation results for target tracking with a single UAV are shown in figures 3.23-3.35. All tracking errors are within $3-\sigma$ bounds. The controller gains used in simulation: $\alpha_1 = 1,\ \alpha_2 = 1,\ \alpha_3 = 1,\ \lambda_{vg} = 50,\ \lambda_\chi = 50,\ \lambda_\gamma = 50$. The actuator constants are: $c_1 = 100,\ c_2 = 100,\ c_3 = 100$. The variances associated with target measurements noises are: $\sigma_r^2 = 4\ m^2$, $\sigma_\phi^2 = (0.65\pi/180)^2\ rad^2$, and $\sigma_\theta^2 = (0.65\pi/180)^2\ rad^2$. The initial error covariance for target is $P_{r0} = 10\ I_{9\times 9}$, and for chaser $P_0 = 10\ I_{12\times 12}$. The process noise covariance matrix $Q_{ra}(t)$ for target is calculated using Eq. (3.23). The following matrices are needed to calculate $Q_{ra}(t)$, and the simulation is executed for the listed parameters in Table 3.4:

$$\begin{aligned} Q_r(t) &= 2\ \mathbf{diag}\left([\sigma^2_{vgr},\ \sigma^2_{\gamma r},\ \sigma^2_{\chi r}]\right) \\ Q_{rf}(t) &= 2\ \mathbf{diag}\left([\sigma^2_{zrr},\ \sigma^2_{z\phi r},\ \sigma^2_{z\theta r}]\right) \end{aligned}$$

For chaser, the parameters used in simulation are tabulated in Table 3.5. The

Table 3.5. Variances of process noises used in simulation (Chaser)

Parameter	Value	Unit
σ_{vg}^2	1	$(m/s)^2$
σ_{γ}^2	$(0.65\pi/180)^2$	rad^2
σ_{χ}^2	$(0.65\pi/180)^2$	rad^2
σ_{zx}^2	2	m^2
σ_{zy}^2	2	m^2
σ_{zz}^2	2	m^2
σ_{zvg}^2	1	$(m/s)^2$
$\sigma_{z\gamma}^2$	$(0.65\pi/180)^2$	rad^2
$\sigma_{z\chi}^2$	$(0.65\pi/180)^2$	rad^2

following matrices are needed to calculate the process noise covariance $\mathbf{Q}_a$.

$$\begin{aligned} \mathbf{Q}(t) &= \mathbf{diag}\left([\sigma_{vg}^2,\ \sigma_{\gamma}^2,\ \sigma_{\chi}^2]\right) \\ \mathbf{Q}_f(t) &= \mathbf{diag}\left([\sigma_{zx}^2,\ \sigma_{zy}^2, \sigma_{zz}^2,\ \sigma_{zvg}^2,\ \sigma_{z\gamma}^2,\ \sigma_{z\chi}^2]\right) \end{aligned}$$

The measurement noise covariance matrices used in EKF for target and chaser:

$$\begin{aligned} R_{rk} &= 0.75\ diag\left([4,\ (0.65\pi/180)^2,\ (0.65\pi/180)^2]\right) \\ R_k &= 0.75\ diag\left([2,\ 2,\ 2,\ 1,\ (0.65\pi/180)^2,\ (0.65\pi/180)^2]\right) \end{aligned}$$

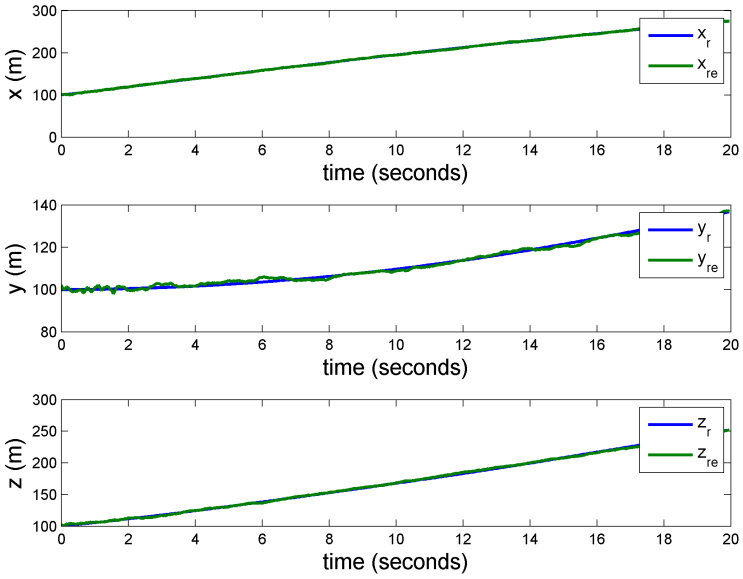


Figure 3.23. The true and estimated state of target vehicle.

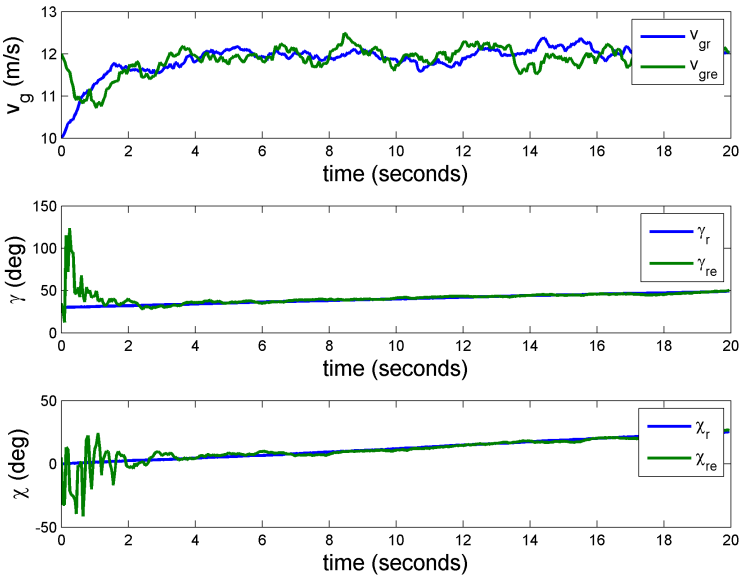


Figure 3.24. The true and estimated state of target vehicle.

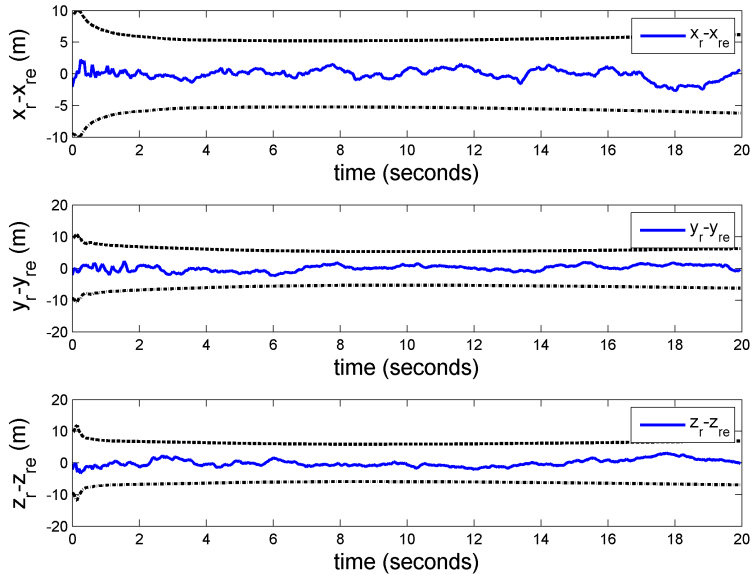


Figure 3.25. Position Errors with $3-\sigma$ bounds.

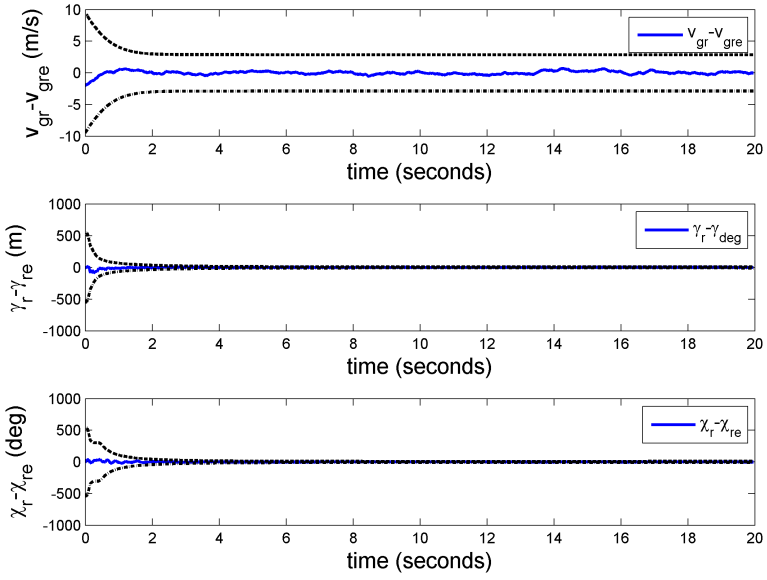


Figure 3.26. Velocity, flight path, and heading angle error with $3-\sigma$ bounds.

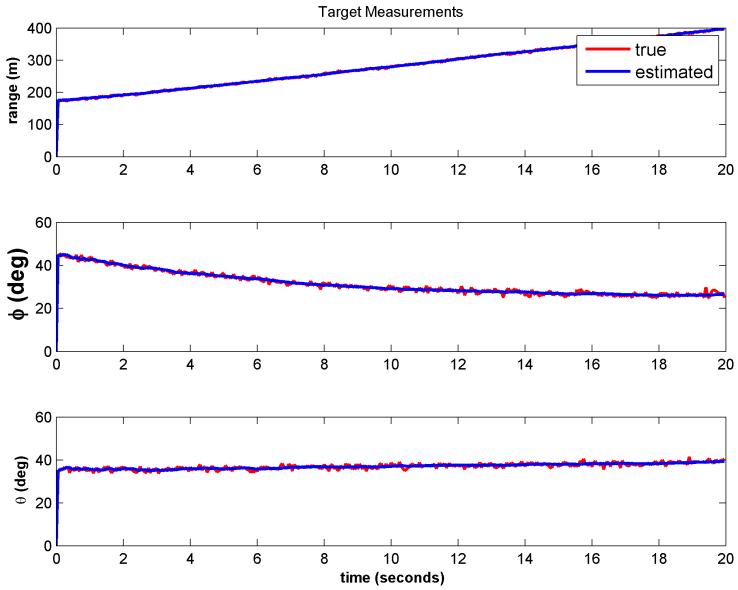


Figure 3.27. Target True, and Estimated Measurements.

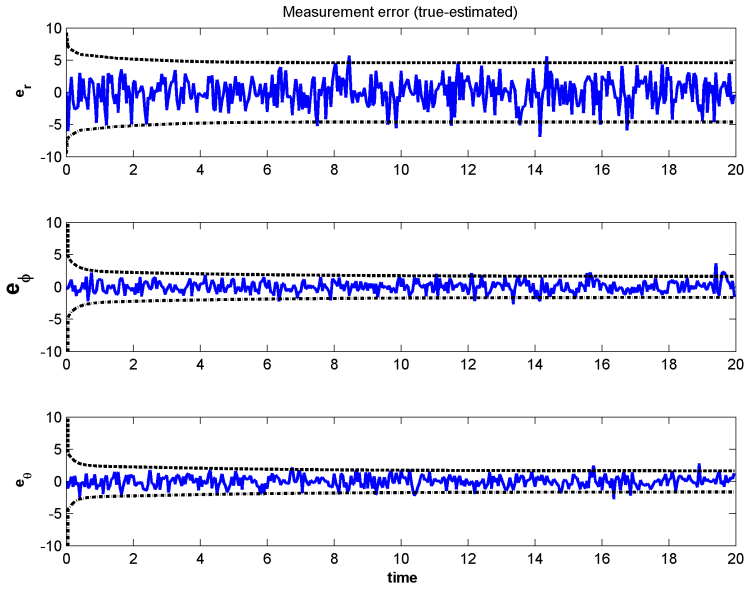


Figure 3.28. Measurements Error (true-estimated) with $3-\sigma$ bounds.

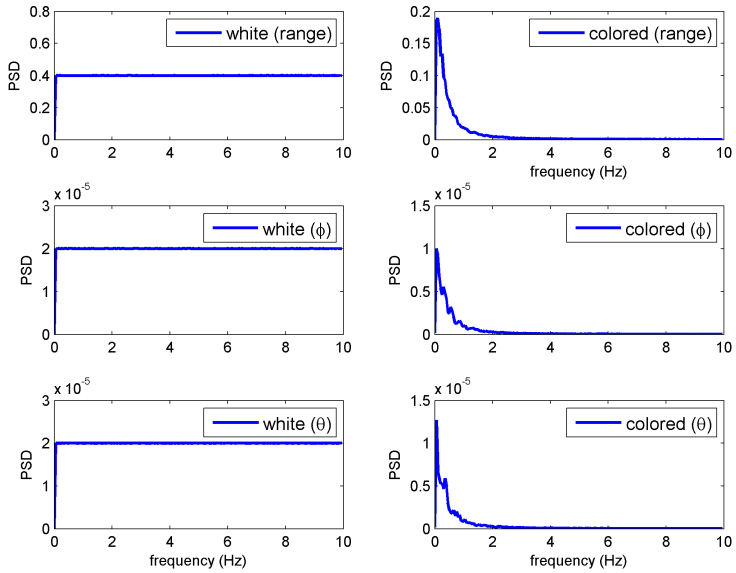


Figure 3.29. Measurements Error (true-estimated) with $3-\sigma$ bounds.

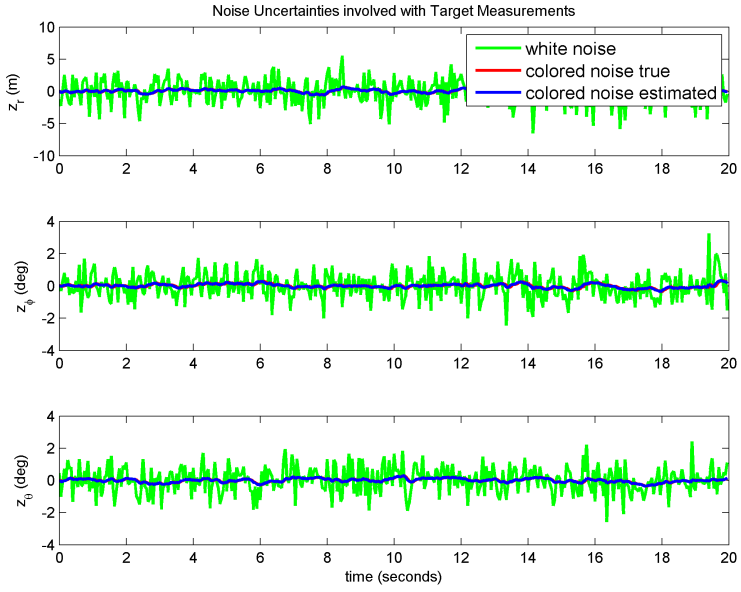


Figure 3.30. Measurements Noise.

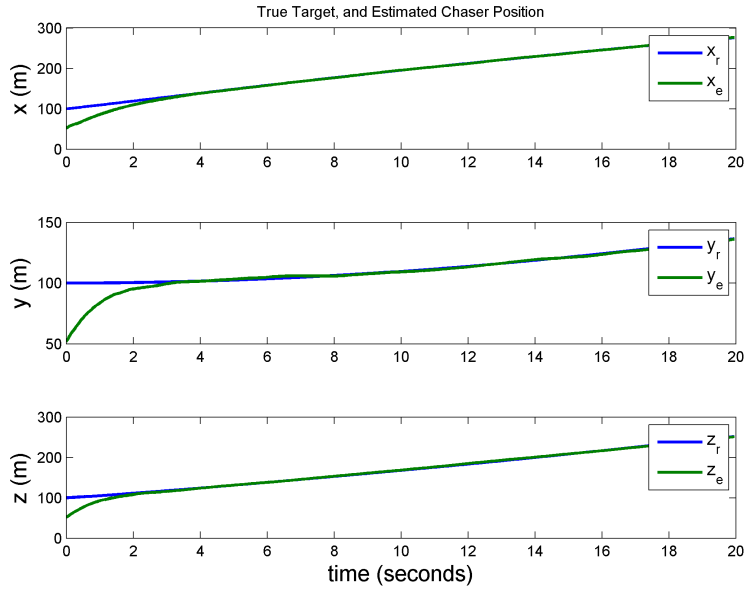


Figure 3.31. True Target and Estimated Chaser (Position).

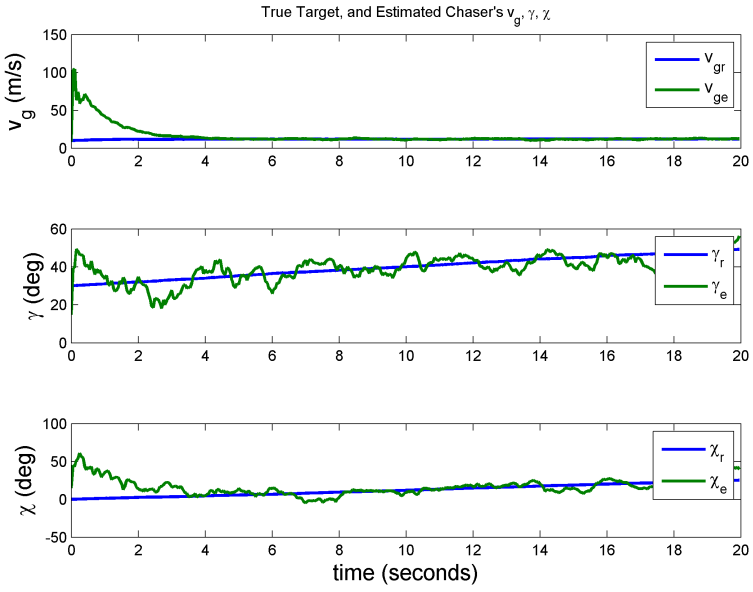


Figure 3.32. True Target and Estimated Chaser (v_g, γ, χ).

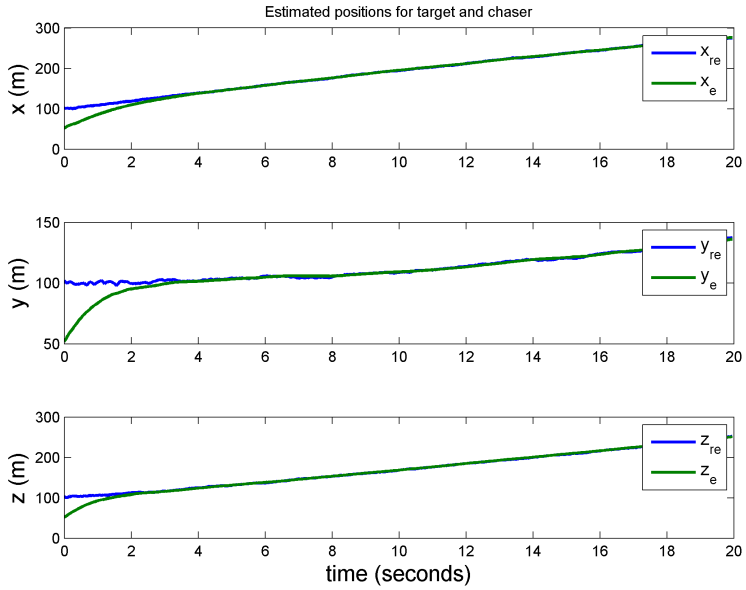


Figure 3.33. Estimated Target and Estimated Chaser (Position).

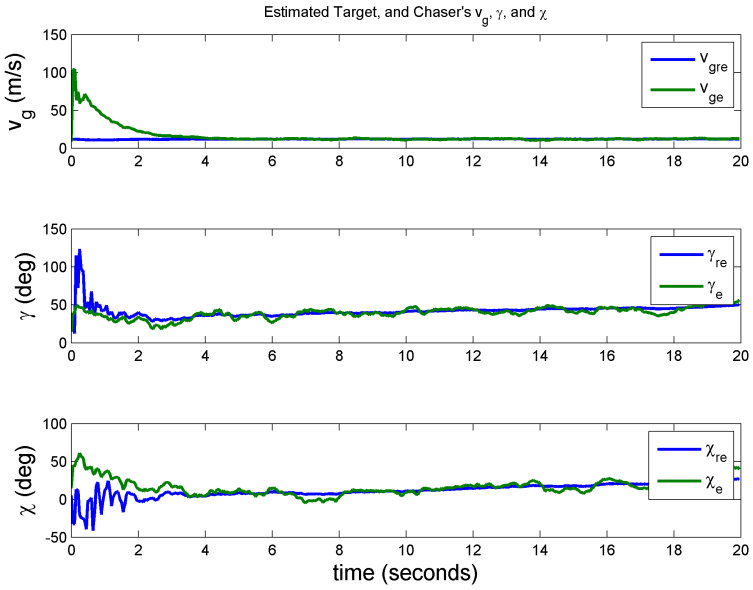


Figure 3.34. Estimated Target and Estimated Chaser (v_g, γ, χ).

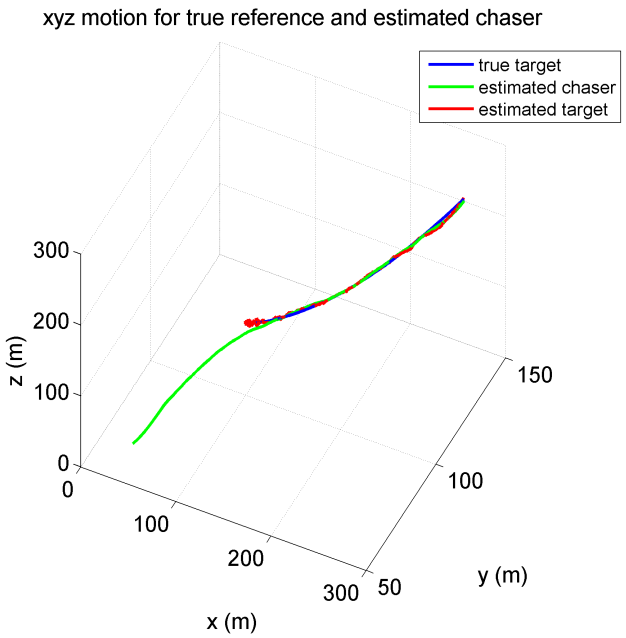


Figure 3.35. Estimation based Target Tracking in $3D$.

3.4 Range Dependent Covariance for Target Measurements

In practice, the error covariance matrix for measurements is not constant all over time. To implement a more realistic scenario for target tracking, we include the following variance function for target range measurement:

$$\sigma_r^2 = a_2(r - a_1)^2 + a_0 \tag{3.37}$$

where $a_0 > 0$, $a_1 > 0$ $a_2 > 0$, and r is the distance between the target and chaser.

The above function in Eq. (3.37) is utilized in [69] to represent the variance of range measurement noise. It implies that the error variance is getting smaller as the chaser closes to the target and when it reaches to the target it has minimum error variance a_0 (known as "*sweet spot*"). The variances for azimuth angle, and elevation angle can also be affected and it is more feasible assumption that the measurements affected by noises will reduce as it comes closer to the target. Likewise, the variances for azimuth, and elevation angle can be written as follows:

$$\begin{aligned} \sigma_\phi^2 &= \alpha_\phi \left(a_2(r - a_1)^2 + a_0\right) \\ \sigma_\theta^2 &= \alpha_\theta \left(a_2(r - a_1)^2 + a_0\right) \end{aligned} \tag{3.38}$$

where $\alpha_\phi > 0$, and $\alpha_\theta > 0$. The covariance matrix for target measurement becomes $R = diag([\sigma_r^2 \ \sigma_\phi^2 \ \sigma_\theta^2])$. If the parameters are chosen as $a_0 = 1\ m^2$, $a_1 = 1\ m^2$, $a_2 = 0.0024\ m^2$, $\alpha_\phi = 0.01(rad/m)^2$, $\alpha_\phi = 0.01(rad/m)^2$, and r varies from 0 to 100 m, we can easily see from Fig. 3.36 that the variances for range, azimuth angle, and elevation angle become minimum at $r = 1\ m$.

Simulation Results

The simulation is performed for the case when the measurements are corrupted by white noise uncertainties. The parameters are chosen to calculate the variances for range, azimuth angle, and elevation angle as: $a_0 = 1\ m^2$, $a_1 = 1\ m^2$,

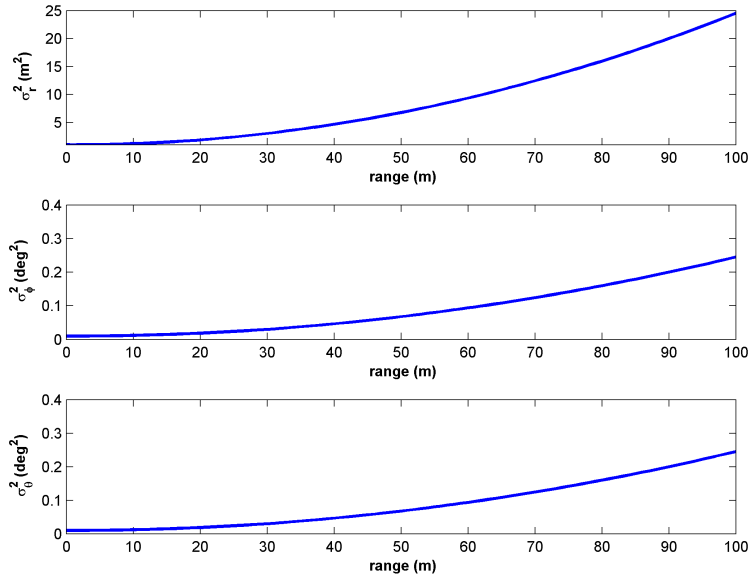


Figure 3.36. Variances with range.

$a_2 = 0.0024\ m^2$, $\alpha_\phi = 1e^{-5}\ (rad/m)^2$, $\alpha_\phi = 1e^{-5}\ (rad/m)^2$. The initial position of target UAV and chaser UAV: $X_r(0) = [100\ m,\ 100\ m,\ 100m]^T$,$X(0) = [50\ m,\ 50\ m,\ 50\ m]^T$. Therefore, the initial range vector: $\vec{r_0} = [50\ 50\ 50]^T$, and its magnitude is $|r_0| = 86.6\ m$. The initial variances for range, azimuth angle, and elevation angle measurement noise: $\sigma_r^2 = 18.9\ m^2$, $\sigma_\phi^2 = 0.62\ deg^2$, and $\sigma_\theta^2 = 0.62\ deg^2$ i.e. the standard deviations are $\sigma_r = \pm 4.34\ m$, $\sigma_\phi = \pm 0.79\ deg$, and $\sigma_\theta = \pm 0.79\ deg$. The simulation runs for 30 seconds. The process noise covariances for target and chaser are: $Q_r(t) = diag([1\ (m/s)^2,\ 3.28\ deg^2,\ 3.28 deg^2])$, and $Q(t) = diag([1e^{-3}\ (m/s)^2,\ 3.28\ deg^2,\ 3.28\ deg^2])$ respectively. From Fig. 3.37-3.47, it is seen that the chaser achieves satisfactory tracking performance with varying target measurement noise variances. Fig. 3.48 shows the target measurement noise variances with time, and we can easily see that the variances are decreasing with time. At 30^{th} seconds the variances become: $\sigma_r^2 = 1.02\ m^2$, $\sigma_\phi^2 = 0.03\ deg^2$, and $\sigma_\theta^2 = 0.03\ deg^2$.

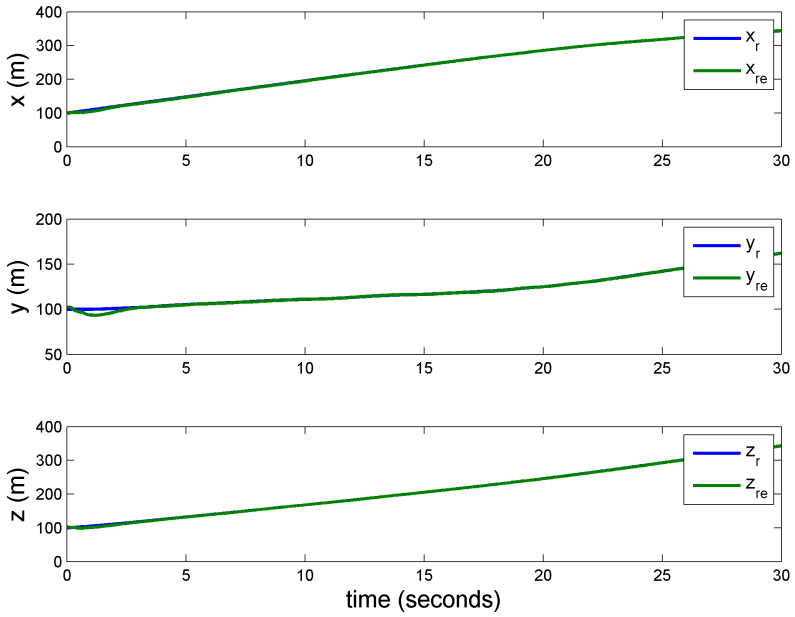


Figure 3.37. The true and estimated position state of target vehicle.

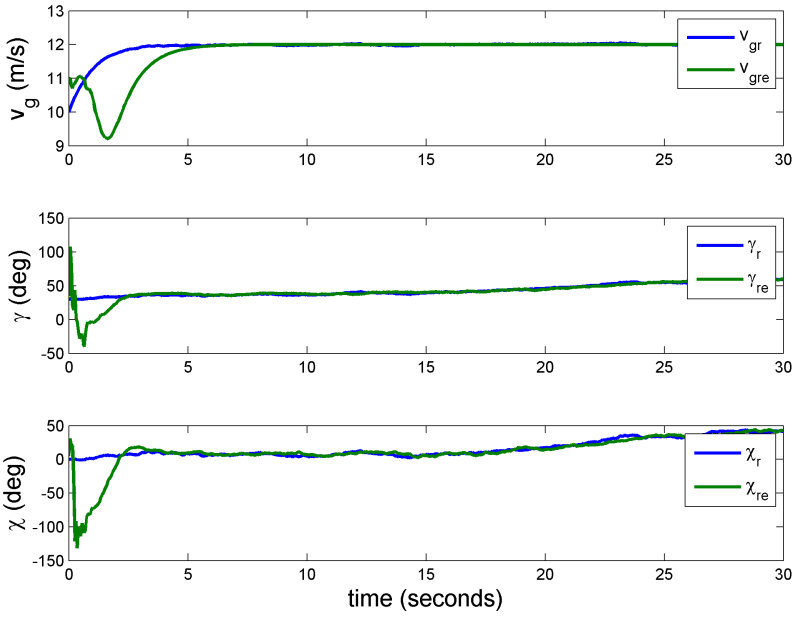


Figure 3.38. The true and estimated speed, flight path, and heading angle of target vehicle.

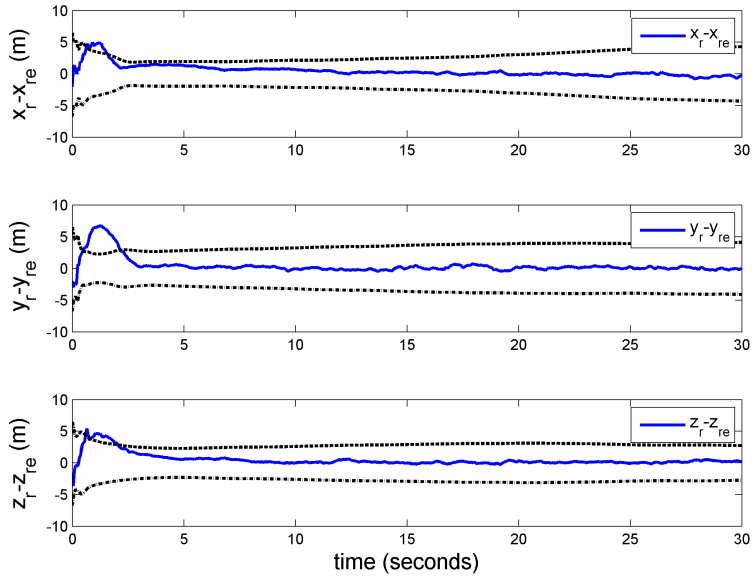


Figure 3.39. Position Errors with $3-\sigma$ bounds.

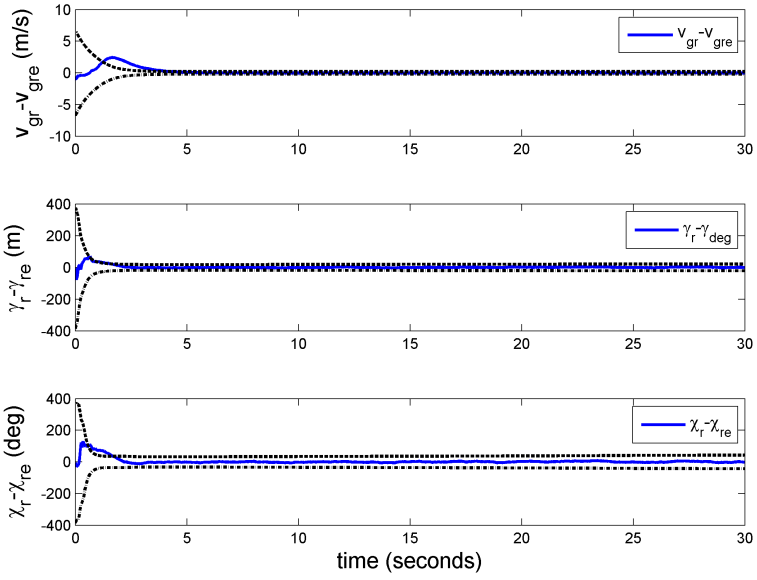


Figure 3.40. velocity, flight path, and heading angle error with $3-\sigma$ bounds.

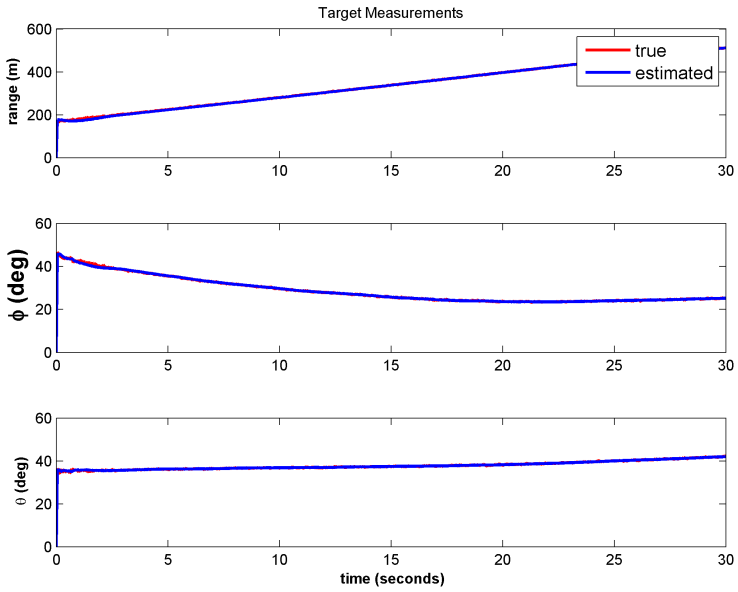


Figure 3.41. Target True, and Estimated Measurements.

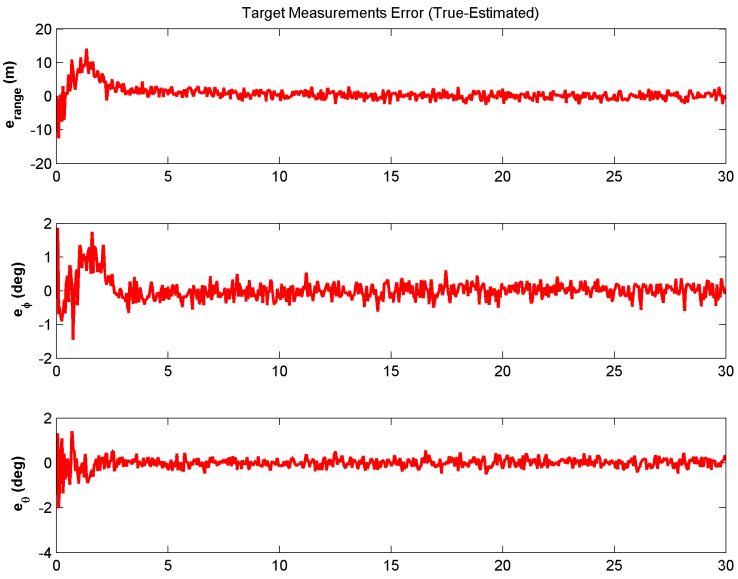


Figure 3.42. Target Measurements Error.

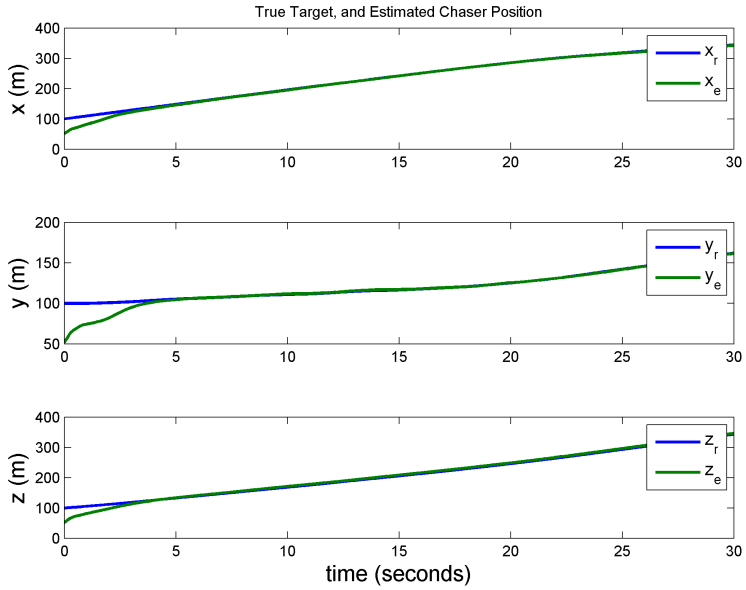


Figure 3.43. True Target and Estimated Chaser (Position).

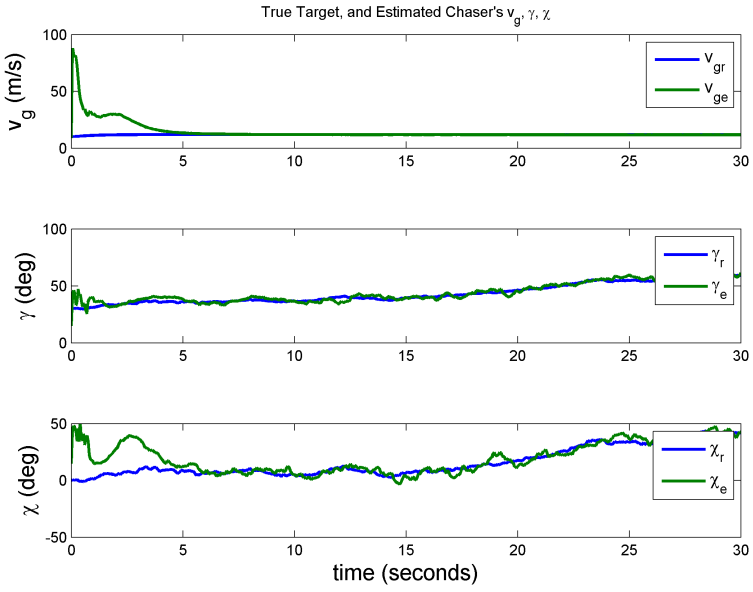


Figure 3.44. True Target and Estimated Chaser (v_g, γ, χ).

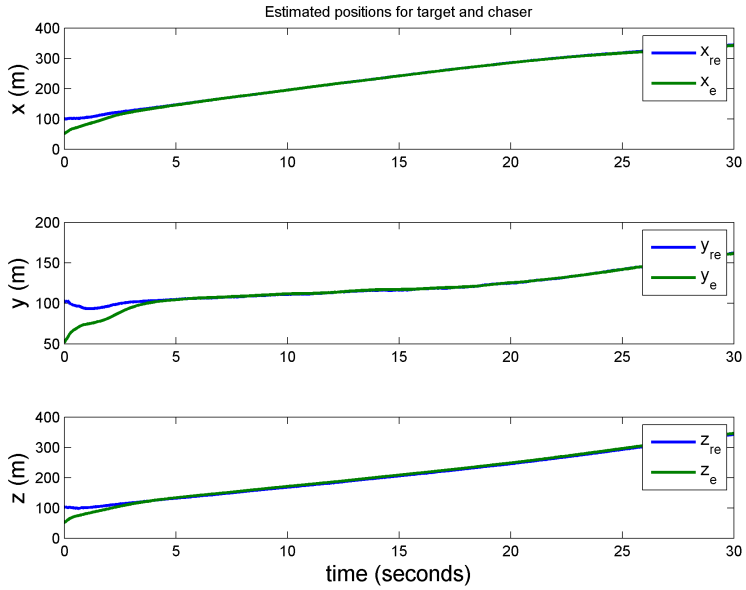


Figure 3.45. Estimated Target and Estimated Chaser (Position).

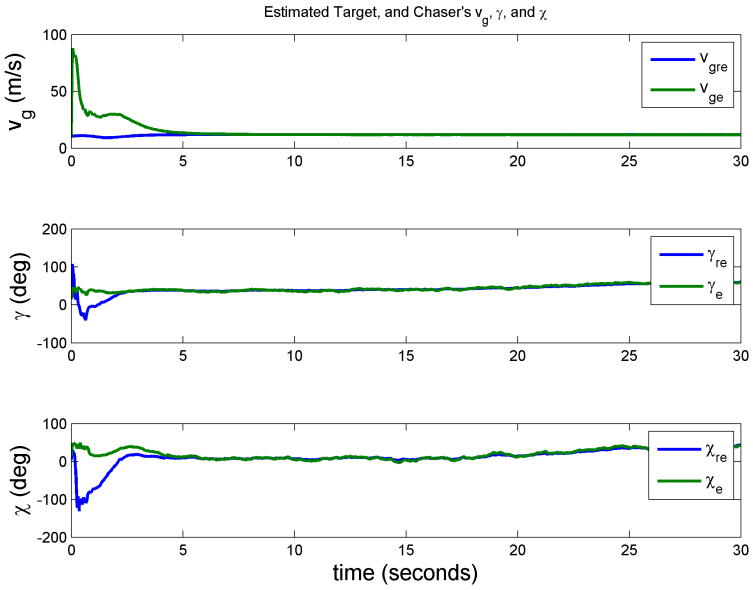


Figure 3.46. Estimated Target and Estimated Chaser (v_g, γ, χ).

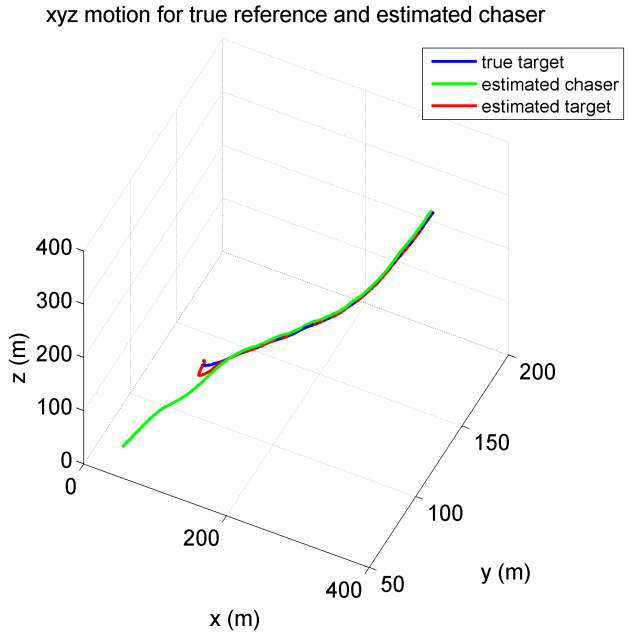


Figure 3.47. Estimation based Target Tracking in $3D$.

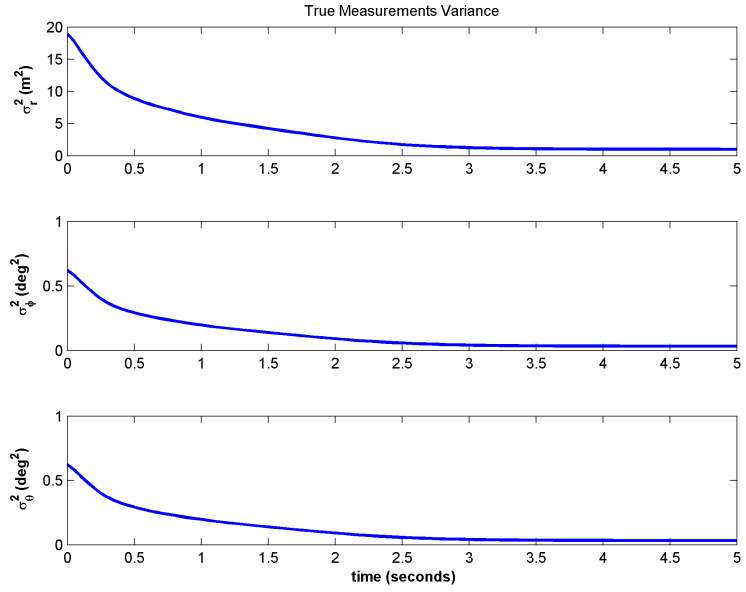


Figure 3.48. Target Measurement Noise Variance with time.

CHAPTER 4

CONTROL LAWS FOR TARGET TRACKING WITH COOPERATIVE UAVS

In this section, we describe general terminologies of algebraic graph theory in section 4.1 [70]; derive the control laws for a cooperative group of UAVs with complete target information and the simulation results are shown in section 4.2 for fixed and switching topology case; and derive and implementation of estimation based control laws with partial target information for fixed and switching topology case in section 4.3.

4.1 General Description of Dynamic Graph

Consider a network of multiple UAVs which are connected through a communication topology. The local communication between the vehicles is represented by a dynamic graph $G(t)$. Consider a dynamic graph $G(t) = (V,\ E(t))$ where V is the nonempty finite set of nodes $V = \{1,\ 2,\ \ldots\ldots,\ n\}$ and E(t) are a set of edges $E \subseteq V \times V$. Each of the vehicles is represented by a node and their interaction behavior is represented by an edge. The edges are represented by an adjacency matrix which represents the interactions among the vehicles. The adjacency matrix of the graph G is expressed as, $\mathbf{A} = [a_{ij}]$ where a_{ij} is graph edge weight, defined by $a_{ij} = 1$ if $(\nu_i,\ \nu_j)\ \in\ E$ and $a_{ij} = 0$ otherwise. Here, edge $e_{ij} = (\nu_i,\ \nu_j)$ means that node ν_j can obtain information from node ν_i. Therefore, node ν_i is considered as the parent node whereas ν_j is the child node. The set of neighbors of an agent or a node ν_i is represented by $N_i = \{\nu_j : (\nu_j,\ \nu_i) \in E\}$ i.e. the set of nodes with arcs which enter to ν_i as can be shown via Fig. 4.1.

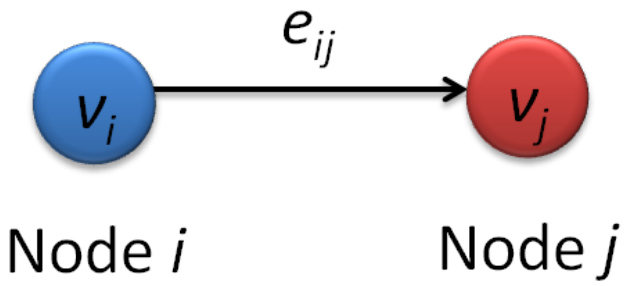


Figure 4.1. A Dynamic Graph.

A directed graph (or digraph) is for more specific cases where the flow of information may be unidirectional. A graph is said to be undirected if there exists both ways information exchange from ν_i to ν_j and ν_j to ν_i for which A is symmetric. A directed tree graph is said to have a spanning tree if and only if there exists a node having a path to every other node i.e. a leader node. A graph is strongly connected if and only if ν_i, and ν_j are connected for all distinct nodes ν_i, $\nu_j \in G$ irrespective of the orientation of the edges which implies that it has a directed spanning tree.

Graph Laplacian matrix L is a widely used term to represent the graph structure of a network in the literature which holds important properties depending on the communication. It is defined as, $L = D - A$ where D is a valency matrix. The diagonal elements of $D = [d_{ij}]$ are the row sums of adjacency matrix A expressed as $d_{ii} = \sum_j^{N_i} a_{ij}$. Each diagonal term d_{ii} represents the number of neighbors of node ν_i which is its in-degree. L has exact one zero eigenvalue with associated right eigenvector 1_n if and only if the associated directed graph has a directed spanning tree and rest of the eigenvalues are in the open right half plane. L is symmetric and positive semi-definite for balanced graph i.e. in-degree and out-degree for all nodes are the same, for example an undirected graph.

The communication topology may be fixed and time varying. For fixed or time-invariant communication, the adjacency matrix A is constant i.e. a_{ij} are constant over time. If the topology changes to a new topology after a certain time interval, it is said to be dynamically changing or switching topology. These situations may arise in cases where limited information or new information may be available from the neighboring vehicles after a certain time interval. A group of graphs is said to be strongly connected if the union of graphs is strongly connected [71]. The group has a spanning tree if the union of graphs has a spanning tree.

The following matrix definitions will be needed throughout the present study:

Definition. Irreducible and Reducible Matrix

A nonnegative matrix $A \in \mathcal{R}^{n\times n}$ is said to be irreducible if there does not exist a permutation matrix P such that PAP^T is block triangular, otherwise it is reducible matrix. For example, a reducible Matrix can be formed as.

$$PAP^T = \begin{bmatrix} * & 0 \\ * & * \end{bmatrix}$$

A graph G is strongly connected if and only if the adjacency matrix A is irreducible.

Definition. Row Stochastic Matrix

A nonnegative matrix $A \in \mathcal{R}^{n\times n}$ $(A \geq 0)$ with all its row sums equal to $+\mathbf{1}$ is called a row stochastic matrix.

Definition. Doubly Stochastic Matrix

A nonnegative matrix $A \in \mathcal{R}^{n\times n}$ $(A \geq 0)$ with all its row and column sums equal to $+\mathbf{1}$ is called a doubly stochastic matrix.

The maximum eigenvalue of a stochastic matrix is 1.

Definition. SIA

A row stochastic matrix is indecomposable and aperiodic (SIA) [72] if

$$Q = \lim_{n\to\infty} A^n$$

exists and all the rows are the same.

According to Lemma 2.6 in [70], for the directed graph with having a directed spanning tree, e^{-Lt} for all $t \geq 0$ is row-stochastic matrix with positive diagonal entries and it also satisfies $\mathbf{1}_n^T\mathbf{v} = 1$ and $L^T\mathbf{v} = \mathbf{0}$, then $e^{-Lt} \to \mathbf{1}_n\mathbf{v}$ as $t \to \infty$, where $\mathbf{v}$ is the left eigen vector associated with zero eigenvalue of L. For the strongly connected graph, L is a irreducible matrix. According to the corollary mentioned in [40], if the directed graph has a directed spanning tree, the row stochastic A matrix is indecomposable and aperiodic (SIA), that is, $lim_{m\to\infty}\ A^m \to v$, where v satisfies $A^T\ v = v$ and $\mathbf{1}^T v = \mathbf{1}$, where v is nonnegative. Let, $S_A = \{A_1,\ A_2,\ \ \ldots\ \ A_l\}$ be a set of row stochastic matrices with positive diagonal entries. If the union of the directed graphs of A_i has a spanning tree, the matrix product $\prod_{i=1}^{l}\ A_i$ is SIA [70]. According to Lemma 2.27 [70], if the union of directed graphs $\bar{G} = \{G_1,\ G_2,\ \ldots,\ G_n\}$ has a directed spanning tree, and L_i is the graph Laplacian matrix associated with each directed graph G_i where each L_i is a row stochastic matrix, then matrix product $e^{-L_1\Delta_1}e^{-L_2\Delta_2}\ \ldots\ e^{-L_n\Delta_n}$ is SIA.

4.2 Control Laws for Cooperative UAVs (Complete Target Information)

In this section, cooperative control laws are initially designed for multiple UAVs with complete target state information. For each UAV, all state information are available from its on-board sensors (GPS+IMU). The present study is addressed for different types of communication topologies (fixed, and time varying case).

Cooperative Control Laws with fixed topology (Complete Target Information)

Consider a team of N UAVs which are connected through a fixed interaction topology (G). We consider the dynamics Eq. (2.1) for each vehicle i:

$$\begin{aligned}
\dot{x}_i &= v_{gi}\cos\gamma_i\cos\chi_i \\
\dot{y}_i &= v_{gi}\cos\gamma_i\sin\chi_i \\
\dot{z}_i &= v_{gi}\sin\gamma_i \\
\dot{v}_{gi} &= c_{1i}(v_g^{ic} - v_{gi}) \\
\dot{\gamma}_i &= c_{2i}(\gamma^{ic} - \gamma_i) \\
\dot{\chi}_i &= c_{3i}(\chi^{ic} - \chi_i)
\end{aligned} \tag{4.1}$$

A pinning control technique introduced in [73], is used whereby the target vehicle is pinned into a subset of the network. In this study, the reference trajectory information is made available into the group by pinning it into at least one node of the network. The connectivity in this case is represented by matrix $\mathbf{B}$. This is a diagonal matrix defined by $\mathbf{B} = [b_{ij}]$ where the diagonal element b_{ii} is the pinning gain and $b_{ii} > 0$ for at least one node i. For the target UAV, the kinematic equations described in Eq. (2.6) is used in this study.

Define the errors for each vehicle as follows:

$$\begin{aligned} e_{xi} &= \sum_{j=1}^{N_i} a_{ij}(x_j - x_i) + b_i(x_r - x_i) \\ e_{yi} &= \sum_{j=1}^{N_i} a_{ij}(y_j - y_i) + b_i(y_r - y_i) \\ e_{zi} &= \sum_{j=1}^{N_i} a_{ij}(z_j - z_i) + b_i(z_r - z_i) \end{aligned} \tag{4.2}$$

where $i = \{1,\ 2,\ \ldots\ ,\ N\}$. It can also be expressed in the following form:

$$\begin{aligned} \mathbf{e}_x &= -(\mathbf{L} + \mathbf{B})(\mathbf{x} - x_r\underline{\mathbf{1}}) \\ \mathbf{e}_y &= -(\mathbf{L} + \mathbf{B})(\mathbf{y} - y_r\underline{\mathbf{1}}) \\ \mathbf{e}_z &= -(\mathbf{L} + \mathbf{B})(\mathbf{z} - z_r\underline{\mathbf{1}}) \end{aligned} \tag{4.3}$$

where $\mathbf{L} = [l_{ij}]$ is the graph Laplacian matrix expressed as: $\mathbf{L} = (\mathbf{D} - \mathbf{A})$, and $\mathbf{1}_{N\times 1}$ is a vector where each element is 1. The matrix $\mathbf{D} = [d_{ij}]$ is the diagonal matrix. The diagonal elements of $\mathbf{D}$ are the row sums of adjacency matrix $\mathbf{A}$ expressed as $d_{ii} = \sum_j^{N_i} a_{ij}$. Each diagonal term d_{ii} represents the number of neighbors of node ν_i which is its in-degree. Position vectors are: $\mathbf{x}_{N\times 1} = [x_1,\ x_2,\ \ \ldots\ \ ,\ x_N]^T$, $\mathbf{y}_{N\times 1} = [y_1, y_2,\ \ \ldots\ \ ,\ y_N]^T$, and $\mathbf{z}_{N\times 1} = [z_1, z_2,\ \ \ldots\ \ ,\ z_N]^T$.

Theorem (Main Result)**.** Given the reference trajectory in Eq. (2.6) and the actual dynamics for each of the UAVs in Eq. (4.1) and a fixed time-invariant communication topology (G) among the vehicles, the following guidance laws ensure that the tracking

errors, $\|x_i - x_r\|$, $\|y_i - y_r\|$, $\|z_i - z_r\|$ and $\|v_{gi} - v_{gr}\|$, $\|\gamma_i - \gamma_r\|$, $\|\chi_i - \chi_r\|$ are all uniformly ultimately bounded (UUB), where $i = \{1, 2, \ldots\ldots, N\}$.

$$\begin{aligned} v_g^{ic} &= v_{gi} - \frac{1}{c_{1i}}\left[-\dot{v}_{gi}^d + \lambda_{vgi}\ e_{vgi} - b_{ii}\ v_{gr}\right] \\ \gamma^{ic} &= \gamma_i - \frac{1}{c_{2i}}\left[-\dot{\gamma}_i^d + \lambda_{\gamma i}\ e_{\gamma i} - b_{ii}\ \gamma_r\right] \\ \chi^{ic} &= \chi_i - \frac{1}{c_{3i}}\left[-\dot{\chi}_i^d + \lambda_{\chi i}\ e_{\chi i} - b_{ii}\ \chi_r\right] \end{aligned} \tag{4.4}$$

When the reference states are constants, the tracking errors asymptotically converge to zero for all initial conditions if G is strongly connected i.e. the graph has a directed spanning tree.

Proof. The desired vehicle dynamics for each vehicle is as follows:

$$\begin{aligned} \dot{x}_i &= v_{gi}^d \cos\gamma_i^d \cos\chi_i^d \\ \dot{y}_i &= v_{gi}^d \cos\gamma_i^d \sin\chi_i^d \\ \dot{z}_i &= v_{gi}^d \sin\gamma_i^d \end{aligned} \tag{4.5}$$

where $i = \{1,\ 2,\ \ldots\ ,\ N\}$, v_{gi}^d, γ_i^d, and χ_i^d are three control commands (to be determined later) that will be used to generate the desired speed, flight path angle and heading angle.

The error dynamics can be written in compact form as:

$$\begin{aligned} \dot{\mathbf{e}}_x &= -(\mathbf{L} + \mathbf{B})(\dot{\mathbf{x}} - \dot{x}_r\underline{\mathbf{1}}) \\ \dot{\mathbf{e}}_y &= -(\mathbf{L} + \mathbf{B})(\dot{\mathbf{y}} - \dot{y}_r\underline{\mathbf{1}}) \\ \dot{\mathbf{e}}_z &= -(\mathbf{L} + \mathbf{B})(\dot{\mathbf{z}} - \dot{z}_r\underline{\mathbf{1}}) \end{aligned} \tag{4.6}$$

Assuming a single integrator like dynamics in each "position" channel, i.e. $\dot{\mathbf{x}} = \mathbf{u}_x$, $\dot{\mathbf{y}} = \mathbf{u}_y$, and $\dot{\mathbf{z}} = \mathbf{u}_z$, we arrive at the following control law for position tracking based on the local neighborhood feedback:

$$\begin{aligned} \mathbf{u}_x &= \alpha_x \mathbf{e}_x + (\mathbf{L} + \mathbf{B})\dot{x}_r \underline{\mathbf{1}} \\ \mathbf{u}_y &= \alpha_y \mathbf{e}_y + (\mathbf{L} + \mathbf{B})\dot{y}_r \underline{\mathbf{1}} \\ \mathbf{u}_z &= \alpha_z \mathbf{e}_z + (\mathbf{L} + \mathbf{B})\dot{z}_r \underline{\mathbf{1}} \end{aligned} \tag{4.7}$$

where $\alpha_x,\ \alpha_y$ and α_z are diagonal positive definite gain matrices of appropriate dimension. It is to be mentioned that if the graph is strongly connected then $(\mathbf{L} + \mathbf{B})$ is positive definite. We can derive the following desired values:

$$\begin{aligned} v_{gi}^d &= \sqrt{(u_{xi})^2 + (u_{yi})^2 + (u_{zi})^2} \\ \gamma_i^d &= \tan^{-1}\left(\frac{u_{zi}}{\sqrt{(u_{xi})^2 + (u_{yi})^2}}\right) \\ \chi_i^d &= \sin^{-1}\left(\frac{u_{yi}}{u_{xi}}\right) \end{aligned} \tag{4.8}$$

Now the errors in the $\mathbf{v}_g,\ \boldsymbol{\gamma},$ and $\boldsymbol{\chi}$ are defined as,

$$\begin{aligned} \mathbf{e}_{vg} &= \mathbf{v}_g - \mathbf{v}_g^d \\ \mathbf{e}_\gamma &= \gamma - \gamma^d \\ \mathbf{e}_\chi &= \chi - \chi^d \end{aligned} \tag{4.9}$$

where $\mathbf{v}_g = [v_{g1},\ v_{g2},\ \ \ldots\ ,\ v_{gN}]^T$, $\boldsymbol{\gamma} = [\gamma_1,\ \gamma_2,\ \ \ldots\ ,\ \gamma_N]^T$, and $\boldsymbol{\chi} = [\chi_1,\ \chi_2,\ \ \ldots\ ,\ \chi_N]^T$. The dynamics for the off-manifold variables that guarantee exponential stability is assumed as:

$$\begin{aligned} \dot{e}_{vgi} &= -\lambda_{vgi}\ e_{vgi} \\ \dot{e}_{\gamma i} &= -\lambda_{\gamma i}\ e_{\gamma i} \\ \dot{e}_{\chi i} &= -\lambda_{\chi i}\ e_{\chi i} \end{aligned} \tag{4.10}$$

The gains, $\lambda_{vgi},\ \lambda_{\gamma i},\ \lambda_{\chi i}$ are positive user specified values. The derivatives of v_{gi}^d, γ_i^d, and χ_i^d are then derived as:

$$\begin{aligned}
\dot{v}_{gi}^d &= \frac{\dot{u}_{xi}u_{xi} + \dot{u}_{yi}u_{yi} + \dot{u}_{zi}u_{zi}}{v_{gi}^d} \\
\dot{\gamma}_i^d &= \frac{1}{(v_{gi}^d)^2}\left[\frac{\dot{u}_{zi}\left((u_{xi})^2 + (u_{yi})^2\right) - u_{zi}\left(u_{xi}\dot{u}_{xi} + u_{yi}\dot{u}_{yi}\right)}{\sqrt{(u_{xi})^2 + (u_{yi})^2}}\right] \\
\dot{\chi}_i^d &= \frac{\dot{u}_{yi}\cos\chi_i^d}{\sqrt{(u_{xi})^2 + (u_{yi})^2}} - \frac{\dot{u}_{xi}\sin\chi_i^d}{\sqrt{(u_{xi})^2 + (u_{yi})^2}}
\end{aligned} \tag{4.11}$$

Note, the computation of the derivatives of v_{gi}^d, γ_i^d, and χ_i^d requires us to compute the derivatives of the control signals from the position control loop namely derivatives of $u_{xi},\ u_{yi}$ and u_{zi}. From Eq. (4.7) we calculate these derivatives as follows,

$$\begin{aligned}
\dot{\mathbf{u}}_x &= \alpha_x\dot{\mathbf{e}}_x + (\mathbf{L}+\mathbf{B})\ddot{x}_r\underline{\mathbf{1}} \\
\dot{\mathbf{u}}_y &= \alpha_y\dot{\mathbf{e}}_y + (\mathbf{L}+\mathbf{B})\ddot{y}_r\underline{\mathbf{1}} \\
\dot{\mathbf{u}}_z &= \alpha_z\dot{\mathbf{e}}_z + (\mathbf{L}+\mathbf{B})\ddot{z}_r\underline{\mathbf{1}}
\end{aligned} \tag{4.12}$$

Note, the implementation of the control loops for position tracking requires u_{xi}, u_{yi}, and u_{zi} and the velocity level tracking is achieved through the derivatives of $u_{xi},\ u_{yi}$ and u_{zi}. Also, the control laws are derived for tracking of a reference trajectory prescribed in terms of the position variables only. In this formulation, the consensus for the speed, flight path angle and heading variables is not included explicitly. If we desire the latter, the desired error dynamics in Eq. (4.10) is modified as follows,

$$\begin{aligned}
\dot{\mathbf{e}}_{vg} &= -\boldsymbol{\lambda}_{vg}\ \mathbf{e}_{vg} + (\mathbf{L}+\mathbf{B})v_{gr}\underline{\mathbf{1}} \\
\dot{\mathbf{e}}_{\gamma} &= -\boldsymbol{\lambda}_{\gamma}\ \mathbf{e}_{\gamma} + (\mathbf{L}+\mathbf{B})\gamma_r\underline{\mathbf{1}} \\
\dot{\mathbf{e}}_{\chi} &= -\boldsymbol{\lambda}_{\chi}\ \mathbf{e}_{\chi} + (\mathbf{L}+\mathbf{B})\chi_r\underline{\mathbf{1}}
\end{aligned} \tag{4.13}$$

where $\boldsymbol{\lambda}_{vg}$, $\boldsymbol{\lambda}_{\gamma}$, and $\boldsymbol{\lambda}_{\chi}$ are diagonal positive definite control gain matrices of appropriate dimension. Using Eq. (4.1), and Eq. (4.13) we can finally derive the control laws as follows:

$$\begin{aligned} v_g^{ic} &= v_{gi} - \frac{1}{c_{1i}}\left[-\dot{v}_{gi}^d + \lambda_{vgi}\ e_{vgi} - b_{ii}\ v_{gr}\right] \\ \gamma^{ic} &= \gamma_i - \frac{1}{c_{2i}}\left[-\dot{\gamma}_i^d + \lambda_{\gamma i}\ e_{\gamma i} - b_{ii}\ \gamma_r\right] \\ \chi^{ic} &= \chi_i - \frac{1}{c_{3i}}\left[-\dot{\chi}_i^d + \lambda_{\chi i}\ e_{\chi i} - b_{ii}\ \chi_r\right] \end{aligned} \tag{4.14}$$

To prove the closed loop stability of the entire system, we select a Lyapunov function candidate as follows:

$$W = \frac{1}{2}\left(\mathbf{e}_{vg}^T\mathbf{e}_{vg} + \mathbf{e}_{\gamma}^T\mathbf{e}_{\gamma} + \mathbf{e}_{\chi}^T\mathbf{e}_{\chi} + \mathbf{e}_x^T\mathbf{e}_x + \mathbf{e}_y^T\mathbf{e}_y + \mathbf{e}_z^T\mathbf{e}_z\right) \tag{4.15}$$

Then, the time derivative of W is

$$\dot{W} = \mathbf{e}_{vg}^T\dot{\mathbf{e}}_{vg} + \mathbf{e}_{\gamma}^T\dot{\mathbf{e}}_{\gamma} + \mathbf{e}_{\chi}^T\dot{\mathbf{e}}_{\chi} + \mathbf{e}_x{}^T\dot{\mathbf{e}}_x + \mathbf{e}_y{}^T\dot{\mathbf{e}}_y + \mathbf{e}_z{}^T\dot{\mathbf{e}}_z \tag{4.16}$$

Using the guidance laws described above in Eq. (4.13), and Eqs. (4.6), (4.7), the time derivative of the Lyapunov function candidate can be written as:

$$\begin{aligned} \dot{W} &= \mathbf{e}_{vg}^T\dot{\mathbf{e}}_{vg} + \mathbf{e}_{\gamma}^T\dot{\mathbf{e}}_{\gamma} + \mathbf{e}_{\chi}^T\dot{\mathbf{e}}_{\chi} + \mathbf{e}_x{}^T\dot{\mathbf{e}}_x + \mathbf{e}_y{}^T\dot{\mathbf{e}}_y + \mathbf{e}_z{}^T\dot{\mathbf{e}}_z \\ &= \mathbf{e}_{vg}^T\left(-\lambda_{vg}\mathbf{e}_{vg} + (L+B)v_{gr}\underline{\mathbf{1}}\right) + \mathbf{e}_{\gamma}^T\left(-\lambda_{\gamma}\mathbf{e}_{\gamma} + (L+B)\gamma_r\underline{\mathbf{1}}\right) \\ &\quad + \mathbf{e}_{\chi}^T\left(-\lambda_{\chi}\mathbf{e}_{\chi} + (L+B)\chi_r\underline{\mathbf{1}}\right) + \mathbf{e}_x{}^T\left(-\alpha_x(L+B)\mathbf{e}_x - (L+B)(L+B-I)\dot{x}_r\underline{\mathbf{1}}\right) \\ &\quad + \mathbf{e}_y{}^T\left(-\alpha_y(L+B)\mathbf{e}_y - (L+B)(L+B-I)\dot{y}_r\underline{\mathbf{1}}\right) \\ &\quad + \mathbf{e}_z{}^T\left(-\alpha_z(L+B)\mathbf{e}_z - (L+B)(L+B-I)\dot{z}_r\underline{\mathbf{1}}\right) \end{aligned}$$

As mentioned earlier, if the graph has a directed spanning tree, $(L+B)$ is positive definite. If we denote $\underline{\sigma}$ as the minimum singular value of $(L+B)$, $\overline{\sigma}$ as the maximum

singular value of $(L+B)$, $\underline{\lambda}_{vg}$, $\underline{\lambda}_{\gamma}$, $\underline{\lambda}_{\chi}$,$\underline{\alpha}_{x}$, $\underline{\alpha}_{y}$, and $\underline{\alpha}_{z}$ as the smallest eigenvalue of λ_{vg}, λ_{γ}, λ_{χ}, α_{x}, α_{y}, and α_{z} respectively, we can derive as follows:

$$\begin{aligned}
\dot{W} \;\leq\; & -\underline{\lambda}_{vg}\,\|\mathbf{e}_{vg}\|^{2} + \overline{\sigma}\|\,\mathbf{e}_{vg}\|\,\|v_{gr}\underline{\mathbf{1}}\| - \underline{\lambda}_{\gamma}\,\|\mathbf{e}_{\gamma}\|^{2} + \overline{\sigma}\|\,\mathbf{e}_{\gamma}\|\,\|\gamma_{r}\underline{\mathbf{1}}\| \\
& -\underline{\lambda}_{\chi}\,\|\mathbf{e}_{\chi}\|^{2} + \overline{\sigma}\|\,\mathbf{e}_{\chi}\|\,\|\chi_{r}\underline{\mathbf{1}}\| - \underline{\alpha}_{x}\,\underline{\sigma}\,\|\mathbf{e}_{x}\|^{2} + \overline{\sigma}\,\overline{\sigma_{i}}\,\|\mathbf{e}_{x}\|\,\|\dot{x}_{r}\underline{\mathbf{1}}\| \\
& -\underline{\alpha}_{y}\,\underline{\sigma}\,\|\mathbf{e}_{y}\|^{2} + \overline{\sigma}\,\overline{\sigma_{i}}\,\|\mathbf{e}_{y}\|\,\|\dot{y}_{r}\underline{\mathbf{1}}\| - \underline{\alpha}_{z}\,\underline{\sigma}\,\|\mathbf{e}_{z}\|^{2} + \overline{\sigma}\,\overline{\sigma_{i}}\,\|\mathbf{e}_{z}\|\,\|\dot{z}_{r}\underline{\mathbf{1}}\|
\end{aligned}$$

Where, $\overline{\sigma_{i}}$ is the maximum singular value of $(L+B-I)$. We also note that the reference trajectory (target trajectory) is bounded, i.e $\|v_{gr}\underline{\mathbf{1}}\| \leq v_{gr,\max}$,$\|\gamma_{r}\underline{\mathbf{1}}\| \leq \gamma_{r,\max}$, $\|\chi_{r}\underline{\mathbf{1}}\| \leq \chi_{r,\max}$, $\|\dot{x}_{r}\underline{\mathbf{1}}\| \leq x_{dr}$,$\|\dot{y}_{r}\underline{\mathbf{1}}\| \leq y_{dr}$, $\|\dot{z}_{r}\underline{\mathbf{1}}\| \leq z_{dr}$ then

$$\begin{aligned}
\dot{W} \;\leq\; & -\underline{\lambda}_{vg}\,\|\mathbf{e}_{vg}\|^{2} + \overline{\sigma}\;v_{gr,max}\,\|\mathbf{e}_{vg}\| - \underline{\lambda}_{\gamma}\,\|\mathbf{e}_{\gamma}\|^{2} + \overline{\sigma}\;\gamma_{r,max}\,\|\mathbf{e}_{\gamma}\| \\
& -\underline{\lambda}_{\chi}\,\|\mathbf{e}_{\chi}\|^{2} + \overline{\sigma}\;\chi_{r,max}\,\|\mathbf{e}_{\chi}\| - \underline{\alpha}_{x}\,\underline{\sigma}\,\|\mathbf{e}_{x}\|^{2} \\
& +\overline{\sigma}\;\overline{\sigma_{i}}\;x_{dr}\,\|\mathbf{e}_{x}\| - \underline{\alpha}_{y}\,\underline{\sigma}\,\|\mathbf{e}_{y}\|^{2} + \overline{\sigma}\;\overline{\sigma_{i}}\;y_{dr}\,\|\mathbf{e}_{y}\| \\
& -\underline{\alpha}_{z}\,\underline{\sigma}\,\|\mathbf{e}_{z}\|^{2} + \overline{\sigma}\;\overline{\sigma_{i}}\;z_{dr}\,\|\mathbf{e}_{z}\| \\
\leq\; & -\beta\|\mathbf{s}\|^{2} + r\;\|\mathbf{s}\|
\end{aligned}$$

Define, $P = \mathbf{diag}\left(\underline{\lambda}_{vg},\quad \underline{\lambda}_{\gamma},\quad \underline{\lambda}_{\chi},\quad \underline{\alpha}_{x}\,\underline{\sigma},\quad \underline{\alpha}_{y}\,\underline{\sigma},\quad \underline{\alpha}_{z}\,\underline{\sigma}\right)$ where β is the minimum singular value of P.

Also, $r = \|\,[\overline{\sigma}\;v_{gr,max},\quad \overline{\sigma}\;\gamma_{r,max},\quad \overline{\sigma}\;\chi_{r,max},\quad \overline{\sigma_{i}}\;x_{dr},\quad \overline{\sigma_{i}}\;y_{dr},\quad \overline{\sigma_{i}}\;z_{dr}]\,\|$ and $\mathbf{s} = [\|\mathbf{e}_{vg}\|\;\;\|\mathbf{e}_{\gamma}\|\;\;\|\mathbf{e}_{\chi}\|\;\;\|\mathbf{e}_{x}\|\;\;\|\mathbf{e}_{y}\|\;\;\|\mathbf{e}_{z}\|]^{T}$. Clearly the Lyapunov function derivative is of the form,

$$\dot{W} \leq -2\beta\;W + r\;\sqrt{2W}$$

Therefore,

$$\sqrt{W} \leq \sqrt{W(0)}\exp\left(-\beta\;t\right) + \frac{r}{\beta}\left(1 - \exp\left(-\beta\;t\right)\right) \leq \sqrt{W(0)} + \frac{r}{\sqrt{2}\beta}$$

Eq. (4.15) can also be written in the following form:

$$W = \frac{1}{2}||\mathbf{s}||^{2}$$

Thus,

$$\|\mathbf{s}\| \leq \|\mathbf{s}(0)\| + \frac{r}{\beta}$$

Remarks:

- When the consensus states are constant, i.e. $x_r\ ,\ y_r, \ldots =$ constant, $r = 0$ and we see that for the graph which has a directed spanning tree, all vehicles in the formation reach the same consensus (desired state).
- When the desired reference states are not constant but have bounded derivatives up to order 2, the trajectory tracking errors are bounded. The tracking errors can be made arbitrarily small using a high values of the control gains β.
- For the directed graph having a directed spanning tree, the system is asymptotically stable (i.e. bounded-input and bounded tracking errors) which implies that, if the target states and its derivatives (up to 2 order) are uniformly bounded, so all tracking errors are uniformly bounded.

□

Cooperative Control Laws with Switching Topology (Complete Target Information)

We consider a scenario where the communication between the UAVs switches to a new topology as the information updates over certain time instants. It may create new or lose connections with the neighboring UAVs within an effective range of detection. We investigate the consensus of UAVs in cases for which, all UAVs meet the target tracking objective.

For this purpose, we consider the following graph connectivity over certain switching instants $t = \{t_0,\ t_1,\ \ldots\ ,\ t_k\}$ as shown in Fig. 4.2 are: $\bar{G} = \{G_1,\ G_2,\ \ldots\ ,\ G_k\}$. Therefore, the graph Laplacian matrix is: $\bar{\mathbf{L}} = \{\mathbf{L}_1,\ \mathbf{L}_2,\ \ldots\ ,\ \mathbf{L}_k\}$ and the gain matrix is: $\bar{\mathbf{B}} = \{\mathbf{B}_1,\ \mathbf{B}_2,\ \ldots\ ,\ \mathbf{B}_k\}$ throughout the consecutive switching instants. We

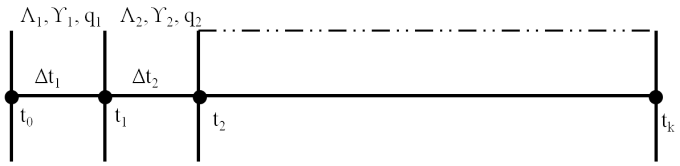


Figure 4.2. Switching Instants.

can prove the following theorem for the consensus in case of switching topology using the similar arguments as in [40]:

Theorem (Main Result)**.** The control laws as described in Eq. (3.1) achieves target tracking asymptotically with uniformly ultimately bounded (UUB) tracking errors for the graph connectivity of $\bar{G} = \{G_1,\ G_2,\ \ \dots\ ,\ G_k\}$ over infinitely uniformly bounded non-overlapping switching time instants $t = \{t_0,\ t_1,\ t_2,\ \ \dots\ ,\ t_k\}$ provided that the union of the graphs over the switching instants has directed information flow having a spanning tree.

Proof. We can write the total error for a particular switching instant say, k^{th} instant $\boldsymbol{\xi}_k = [\mathbf{e}_x,\ \mathbf{e}_y,\ \mathbf{e}_z,\ \mathbf{e}_{vg},\ \mathbf{e}_\gamma,\ \mathbf{e}_\chi]_k^T$

We consider the total error dynamics for the entire system at k^{th} instant:

$$\dot{\boldsymbol{\xi}}_k \ = \ -\boldsymbol{\Lambda}_k\boldsymbol{\xi}_k + \boldsymbol{\Upsilon}_k\mathbf{q}_k \tag{4.17}$$

where $\boldsymbol{\Lambda}_k$, and $\boldsymbol{\Upsilon}_k$ are the block diagonal matrices in the following form:

$$\begin{aligned}
\boldsymbol{\Lambda}_k \ &= \ \mathbf{diag}\,[\boldsymbol{\alpha}_{xk}(\mathbf{L}_k+\mathbf{B}_k) \quad \boldsymbol{\alpha}_{yk}(\mathbf{L}_k+\mathbf{B}_k) \quad \boldsymbol{\alpha}_{zk}(\mathbf{L}_k+\mathbf{B}_k) \quad \boldsymbol{\lambda}_{vgk} \quad \boldsymbol{\lambda}_{\gamma k} \quad \boldsymbol{\lambda}_{\chi k}] \\
\boldsymbol{\Upsilon}_k \ &= \ \mathbf{diag}\,[(\mathbf{L}_k+\mathbf{B}_k)(\mathbf{L}_k+\mathbf{B}_k-\mathbf{I}) \quad (\mathbf{L}_k+\mathbf{B}_k)(\mathbf{L}_k+\mathbf{B}_k-\mathbf{I}) \quad (\mathbf{L}_k+\mathbf{B}_k)(\mathbf{L}_k+\mathbf{B}_k-\mathbf{I}) \\
&\qquad\qquad (\mathbf{L}_k+\mathbf{B}_k) \quad (\mathbf{L}_k+\mathbf{B}_k) \quad (\mathbf{L}_k+\mathbf{B}_k)]
\end{aligned}$$

where each diagonal block is a square matrix having dimension $N \times N$, $\mathbf{L}_k + \mathbf{B}_k$ represents the communication topology for the graph connection at k^{th} instant. We

consider the same graph connection topology in x, y and z position states. $\boldsymbol{\lambda}_{vgk}$ is a diagonal matrix having speed control gains in each diagonal terms for each vehicle at k^{th} instant. Similarly, we can define the controls gain matrices $\boldsymbol{\lambda}_{\gamma k}$, and $\boldsymbol{\lambda}_{\chi k}$ for flight path angle, and heading angle at k^{th} instant. Furthermore, define $\mathbf{q}_k = [\dot{x}_r\underline{1} \quad \dot{y}_r\underline{1} \quad \dot{z}_r\underline{1} \quad v_{gr}\underline{1} \quad \gamma_r\underline{1} \quad \chi_r\underline{1}]^T$.

For a particular switching interval $\Delta t_1 = t_1 - t_0$ as shown in Fig. 4.2, we get from Eq. (4.17),

$$\boldsymbol{\xi}(t_1) \;=\; e^{(-\boldsymbol{\Lambda}_1\Delta t_1)}\boldsymbol{\xi}(t_0) + \int_{t_0}^{t_1} \boldsymbol{\Upsilon}_1\; e^{(-\boldsymbol{\Lambda}_1\tau)}\; \mathbf{q}_1 d\tau$$

For the interval $\Delta t_2 = t_2 - t_1$, we can get:

$$\begin{aligned}\boldsymbol{\xi}(t_2) &= e^{(-\boldsymbol{\Lambda}_2\Delta t_2)}\boldsymbol{\xi}(t_1) + \int_{t_1}^{t_2} \boldsymbol{\Upsilon}_2\; e^{(-\boldsymbol{\Lambda}_2\tau)}\; \mathbf{q}_2 d\tau \\ &= e^{(-\boldsymbol{\Lambda}_2\Delta t_2)}e^{(-\boldsymbol{\Lambda}_1\Delta t_1)}\boldsymbol{\xi}(t_0) + e^{(-\boldsymbol{\Lambda}_2\Delta t_2)}\int_{t_0}^{t_1} \boldsymbol{\Upsilon}_1\; e^{(-\boldsymbol{\Lambda}_1\tau)}\; \mathbf{q}_1 d\tau + \int_{t_1}^{t_2} \boldsymbol{\Upsilon}_2\; e^{(-\boldsymbol{\Lambda}_2\tau)}\; \mathbf{q}_2 d\tau\end{aligned}$$

For the interval $\Delta t_3 = t_3 - t_2$, we can get:

$$\begin{aligned}\boldsymbol{\xi}(t_3) &= e^{(-\boldsymbol{\Lambda}_3\Delta t_3)}e^{(-\boldsymbol{\Lambda}_2\Delta t_2)}e^{(-\boldsymbol{\Lambda}_1\Delta t_1)}\boldsymbol{\xi}(t_0) + e^{(-\boldsymbol{\Lambda}_3\Delta t_3)}e^{(-\boldsymbol{\Lambda}_2\Delta t_2)}\int_{t_0}^{t_1} \boldsymbol{\Upsilon}_1\; e^{(-\boldsymbol{\Lambda_1}\tau)}\; \mathbf{q}_1 d\tau \\ &\quad + e^{(-\boldsymbol{\Lambda}_3\Delta t_3)}\int_{t_1}^{t_2} \boldsymbol{\Upsilon}_2\; e^{(-\boldsymbol{\Lambda}_2\tau)}\; \mathbf{q}_2 d\tau + \int_{t_2}^{t_3} \boldsymbol{\Upsilon}_3\; e^{(-\boldsymbol{\Lambda}_3\tau)}\; \mathbf{q}_3 d\tau\end{aligned}$$

Similarly, we can easily show that the error becomes at k^{th} time instant as,

$$\begin{aligned}\boldsymbol{\xi}(t_k) &= e^{(-\boldsymbol{\Lambda}_k\Delta t_k)}e^{(-\boldsymbol{\Lambda}_{k-1}\Delta t_{k-1})} \quad \ldots\ldots \quad e^{(-\boldsymbol{\Lambda}_1\Delta t_1)}\boldsymbol{\xi}(t_0) \\ &\quad + e^{(-\boldsymbol{\Lambda}_k\Delta t_k)}e^{(-\boldsymbol{\Lambda}_{k-1}\Delta t_{k-1})} \quad \ldots\ldots \quad e^{(-\boldsymbol{\Lambda}_2\Delta t_2)}\int_{t_0}^{t_1} \boldsymbol{\Upsilon}_1\; e^{(-\boldsymbol{\Lambda}_1\tau)}\; \mathbf{q}_1 d\tau \\ &\quad + e^{(-\boldsymbol{\Lambda}_k\Delta t_k)}e^{(-\boldsymbol{\Lambda}_{k-1}\Delta t_{k-1})} \quad \ldots\ldots \quad e^{(-\boldsymbol{\Lambda}_3\Delta t_3)}\int_{t_1}^{t_2} \boldsymbol{\Upsilon}_2\; e^{(-\boldsymbol{\Lambda}_2\tau)}\; \mathbf{q}_2 d\tau + \quad \ldots\ldots \\ &\quad + e^{(-\boldsymbol{\Lambda}_k\Delta t_k)}\int_{t_{k-2}}^{t_{k-1}} \boldsymbol{\Upsilon}_{k-1}\; e^{(-\boldsymbol{\Lambda}_{k-1}\tau)}\; \mathbf{q}_{k-1} d\tau + \int_{t_{k-1}}^{t_k} \boldsymbol{\Upsilon}_k\; e^{(-\boldsymbol{\Lambda}_k\tau)}\; \mathbf{q}_k d\tau\end{aligned}$$

$$\begin{aligned}
\boldsymbol{\xi}(t_k) &= e^{-\sum_{j=1}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\boldsymbol{\xi}(t_0) + e^{-\sum_{j=2}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\int_{t_0}^{t_1}\boldsymbol{\Upsilon}_1\, e^{(-\boldsymbol{\Lambda}_1\tau)}\,\mathbf{q}_1 d\tau \\
&\quad + e^{-\sum_{j=3}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\int_{t_1}^{t_2}\boldsymbol{\Upsilon}_2\, e^{(-\boldsymbol{\Lambda}_2\tau)}\,\mathbf{q}_2 d\tau + \quad \ldots\ldots\ldots \\
&\quad + e^{(-\boldsymbol{\Lambda}_k\Delta t_k)}\int_{t_{k-2}}^{t_{k-1}}\boldsymbol{\Upsilon}_{k-1}\, e^{(-\boldsymbol{\Lambda}_{k-1}\tau)}\,\mathbf{q}_{k-1} d\tau + \int_{t_{k-1}}^{t_k}\boldsymbol{\Upsilon}_k\, e^{(-\boldsymbol{\Lambda}_k\tau)}\,\mathbf{q}_k d\tau
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{\xi}(t_k) &\leq \|e^{-\sum_{j=1}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\boldsymbol{\xi}(t_0)\| + \|e^{-\sum_{j=2}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\int_{t_0}^{t_1}\boldsymbol{\Upsilon}_1\, e^{(-\boldsymbol{\Lambda}_1\tau)}\,\mathbf{q}_1 d\tau\| \\
&\quad + \|e^{-\sum_{j=3}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\int_{t_1}^{t_2}\boldsymbol{\Upsilon}_2\, e^{(-\boldsymbol{\Lambda}_2\tau)}\,\mathbf{q}_2 d\tau\| + \quad \ldots\ldots\ldots \\
&\quad + \|e^{(-\boldsymbol{\Lambda}_k\Delta t_k)}\int_{t_{k-2}}^{t_{k-1}}\boldsymbol{\Upsilon}_{k-1}\, e^{(-\boldsymbol{\Lambda}_{k-1}\tau)}\,\mathbf{q}_{k-1} d\tau\| \\
&\quad + \|\int_{t_{k-1}}^{t_k}\boldsymbol{\Upsilon}_k\, e^{(-\boldsymbol{\Lambda}_k\tau)}\,\mathbf{q}_k d\tau\|
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{\xi}(t_k) &\leq \|e^{-\sum_{j=1}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\|\|\boldsymbol{\xi}(t_0)\| + \|\int_{t_0}^{t_1} e^{-\sum_{j=2}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\boldsymbol{\Upsilon}_1\, e^{(-\boldsymbol{\Lambda}_1\tau)}d\tau\|\|\mathbf{q}_1\| \\
&\quad + \|\int_{t_1}^{t_2} e^{-\sum_{j=3}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\boldsymbol{\Upsilon}_2\, e^{(-\boldsymbol{\Lambda}_2\tau)}d\tau\|\|\mathbf{q}_2\| + \quad \ldots\ldots\ldots \\
&\quad + \|\int_{t_{k-2}}^{t_{k-1}} e^{(-\boldsymbol{\Lambda}_k\Delta t_k)}\boldsymbol{\Upsilon}_{k-1}\, e^{(-\boldsymbol{\Lambda}_{k-1}\tau)}d\tau\|\|\mathbf{q}_{k-1}\| + \|\int_{t_{k-1}}^{t_k}\boldsymbol{\Upsilon}_k\, e^{(-\boldsymbol{\Lambda}_k\tau)}d\tau\|\|\mathbf{q}_k\|
\end{aligned}$$

For each switching interval, the reference trajectory is bounded, i.e. $\|\mathbf{q}_1\| \leq \mathbf{q}_{1r,max}$, $\|\mathbf{q}_2\| \leq \mathbf{q}_{2r,max}$ $\ldots\ldots\ldots$ $\|\mathbf{q}_k\| \leq \mathbf{q}_{kr,max}$. If we denote, $\underline{\sigma}_1$ as the minimum singular value of $\boldsymbol{\Lambda}_1$, $\underline{\sigma}_2$ as the minimum singular value of $\boldsymbol{\Lambda}_2$, $\ldots\ldots\ldots$, $\underline{\sigma}_k$ for $\boldsymbol{\Lambda}_k$, we can then denote $\underline{\kappa}_{1k}$ as the minimum singular value of $e^{-\sum_{j=1}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}$, $\underline{\kappa}_{2k}$ as the minimum singular value of $e^{-\sum_{j=2}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}$, $\ldots\ldots\ldots$, $\underline{\kappa}_k$ for $e^{-(\boldsymbol{\Lambda}_k\Delta t_k)}$. Furthermore,

we denote, $\overline{\rho}_1$ as maximum singular value of $\boldsymbol{\Upsilon}_1$, $\overline{\rho}_2$ is the maximum singular value of $\boldsymbol{\Upsilon}_2$, ..., $\overline{\rho}_k$ for $\boldsymbol{\Upsilon}_k$.

$$\begin{aligned}
\|\boldsymbol{\xi}(t_k)\| &\leq \|e^{-\sum_{j=1}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\|\|\boldsymbol{\xi}(t_0)\| + \|e^{-\sum_{j=2}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\|\|\boldsymbol{\Upsilon}_1\| \parallel \int_{t_0}^{t_1} e^{(-\boldsymbol{\Lambda}_1\tau)}d\tau\|\|\mathbf{q}_1\| \\
&\quad + \|e^{-\sum_{j=3}^{k}(\boldsymbol{\Lambda}_j\Delta t_j)}\|\|\boldsymbol{\Upsilon}_2\| \parallel \int_{t_1}^{t_2} e^{(-\boldsymbol{\Lambda}_2\tau)}d\tau\|\|\mathbf{q}_2\| + \ldots\ldots\ldots \\
&\quad + \|e^{(-\boldsymbol{\Lambda}_k\Delta t_k)}\|\|\boldsymbol{\Upsilon}_{k-1}\|\| \int_{t_{k-2}}^{t_{k-1}} e^{(-\boldsymbol{\Lambda}_{k-1}\tau)}d\tau\|\|\mathbf{q}_{k-1}\| \\
&\quad + \|\boldsymbol{\Upsilon}_k\| \parallel \int_{t_{k-1}}^{t_k} e^{(-\boldsymbol{\Lambda}_k\tau)}d\tau\|\|\mathbf{q}_k\| \\
&\leq \underline{\kappa}_{1k}\|\boldsymbol{\xi}(t_0)\| + \underline{\kappa}_{2k}\ \overline{\rho}_1\ \overline{\mu}_1\ q_{1r,max} + \underline{\kappa}_{3k}\ \overline{\rho}_2\ \overline{\mu}_2\ q_{2r,max} + \quad \ldots\ldots\ldots \\
&\quad + \underline{\kappa}_k\ \overline{\rho}_{k-1}\ \overline{\mu}_{k-1}\ q_{(k-1)r,max} + \overline{\rho}_k\ \overline{\mu}_k\ q_{kr,max} \\
&\leq \underline{\kappa}_{1k}\|\boldsymbol{\xi}(t_0)\| + \overline{S}_1 + \overline{S}_2 + \quad \ldots\ldots\ldots \quad + \overline{S}_k \\
&\leq \underline{\kappa}_{1k}\|\boldsymbol{\xi}(t_0)\| + \sum_{j=1}^{k}\overline{S}_j
\end{aligned}$$

where $\overline{\mu}_1$ is the maximum singular value of the integrated term $\int_{t_0}^{t_1} e^{(-\boldsymbol{\Lambda}_1\tau)}d\tau$, and so on. Since the reference trajectory states and its derivative up to second order in each particular switching instant are bounded, each term of $\sum_{j=1}^{k}\overline{S}_j$ can be significantly reduced at the expense of higher control gains (i.e. maximum eigenvalue of $\underline{\Lambda_i}$). For the group having a directed spanning tree (i.e. the union of these directed graphs across each interval has a directed spanning tree), $\underline{\kappa}_{1k}$ is an exponentially reducing term. Therefore, we can easily show that the tracking errors are uniformly ultimately bounded (UUB) if the target states and the derivatives up to order 2 are bounded. □

Simulation Case Studies

The simulation results are shown for two cases: 1) Fixed Topology, and 2) Switching Topology.

Case 1: Fixed Topology (Complete Target Information)

We simulate for the following two scenarios of graph connections as shown in Fig. 4.3. The target trajectory is pinned into $UAV1$ for both cases.

Topology 1: A strongly connected digraph G_1 with 3 UAVs is considered as shown in Fig. 4.3*a*. The adjacency matrix A_1 for this case is:

$$A_1 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

The graph Laplacian matrix is obtained as:

$$L_1 = \begin{bmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \\ -1 & -1 & 2 \end{bmatrix}$$

Since the target UAV is pinned into the UAV 1, we can write the matrix B as:

$$B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The actuator constants and gains for each UAV are:
$c_1 = 10,\ c_2 = 10,\ c_3 = 10,\ \alpha_1 = 2,\ \alpha_2 = 2,\ \alpha_3 = 2,\ \lambda_{vg} = 2,\ \lambda_\gamma = 2,\ \lambda_\chi = 2.$

Figure 4.3. Cooperative Tracking for Multiple UAVs: Different fixed Communication Topology (a) Graph G_1 , and (b) Graph G_2.

The parameters for the target trajectory: $v_{gr}(0) = 15\ m/s,\ v_{gr}^c = 12\ m/s,\ a_{hr}^c = 0.2\ m/s^2,\ a_{vr}^c = 0.2\ m/s^2,\ \chi_r(0) = 12°,\ x_r(0) = 100\ m,\ y_r(0) = 100\ m,\ z_r(0) = 100\ m.$ The actual vehicles are initialized with the following values: for UAV1: $x_1(0) = 40\ m,\ y_1(0) = 60\ m,\ z_1(0) = 70\ m,\ v_{g1}(0) = 8\ m/s,\ \gamma_1(0) = 8°,\ \chi_1(0) = 8°$; for UAV2: $x_2(0) = 50\ m,\ y_2(0) = 80\ m,\ z_2(0) = 90\ m,\ v_{g2}(0) = 10\ m/s,\ \gamma_2(0) = 13°, \chi_2(0) = 13°$; and for UAV3: $x_3(0) = 60\ m,\ y_3(0) = 100\ m,\ z_3(0) = 110\ m,\ v_{g3}(0) = 12\ m/s,\ \gamma_3(0) = 18°,\ \chi_3(0) = 18°.$

The simulation results for the communication topology 1 are shown in Fig. 4.4-4.8 where all UAVs reach consensus and track successfully the target trajectory.

Topology 2: Consider, a group of 4 UAVs having the digraph G_2, which is strongly connected as shown in Fig. 4.3*b*. The adjacency and graph Laplacian matrices can be written as:

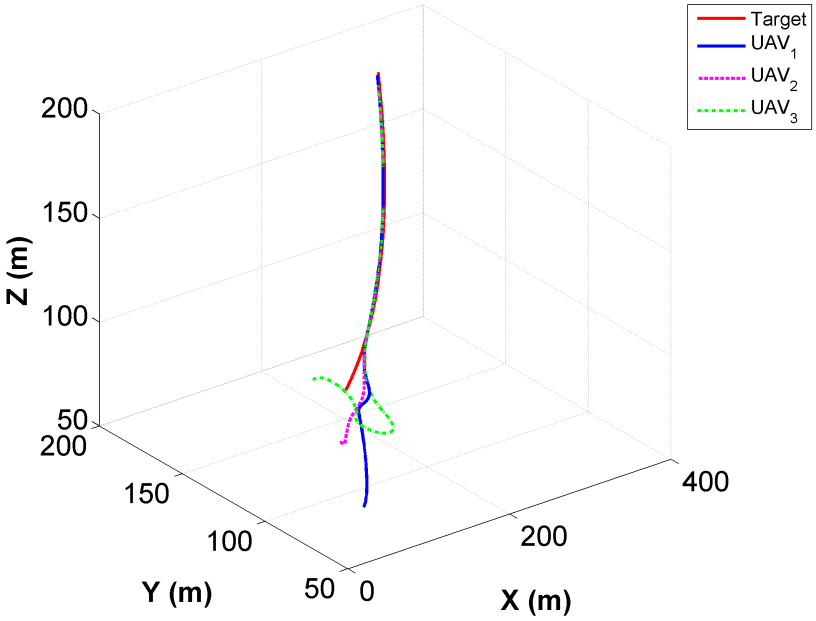


Figure 4.4. Target and 3 Chaser UAVs trajectories for Topology 1.

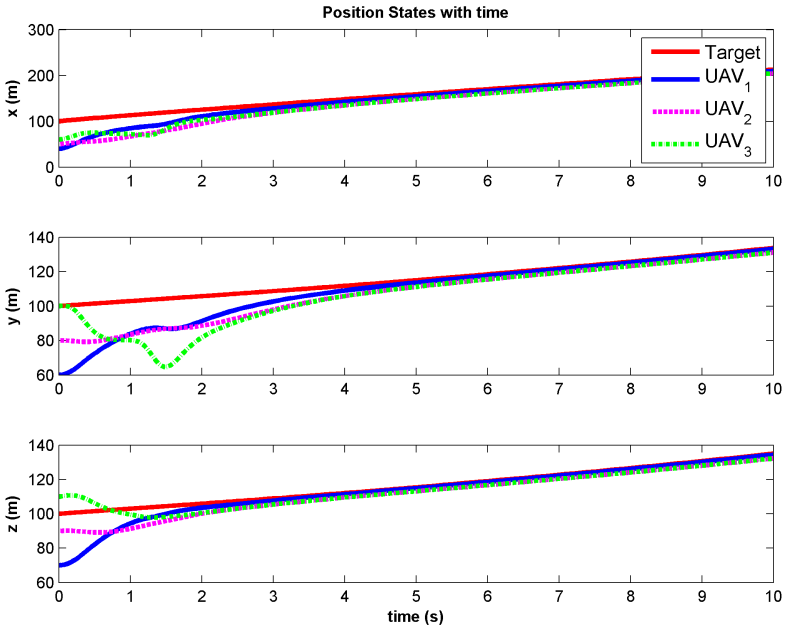


Figure 4.5. Position States with time for 3 UAVs with Topology 1.

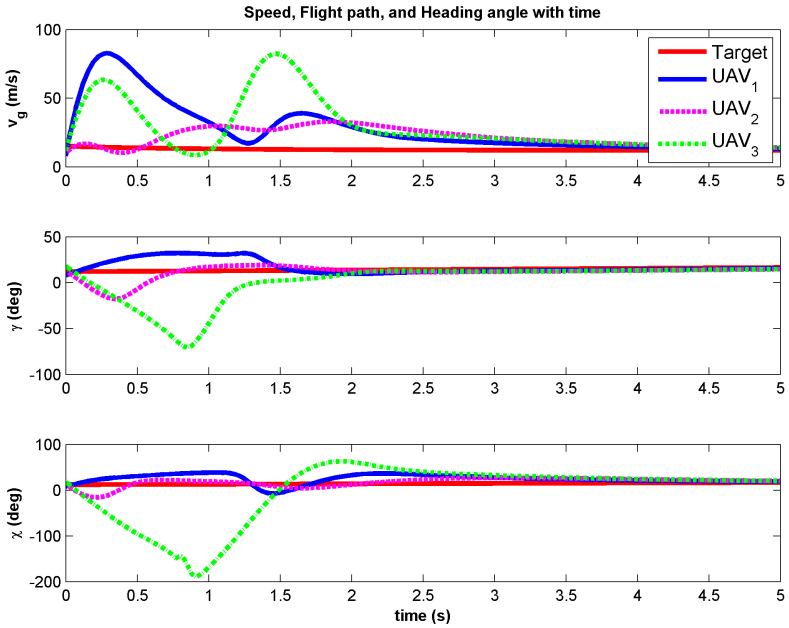


Figure 4.6. Speed, Flight Path, and Heading angle for 3 UAVs with Topology 1.

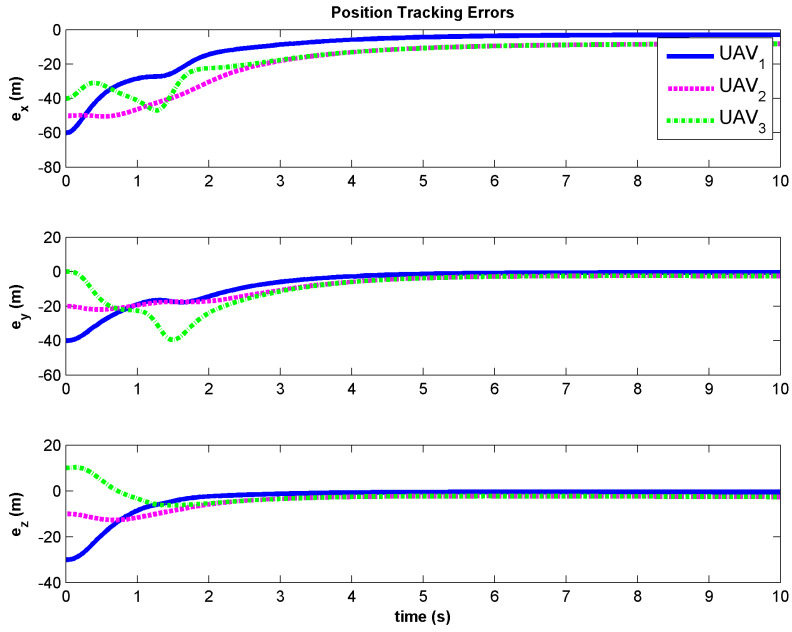


Figure 4.7. Position Tracking Errors for 3 UAVs with Topology 1.

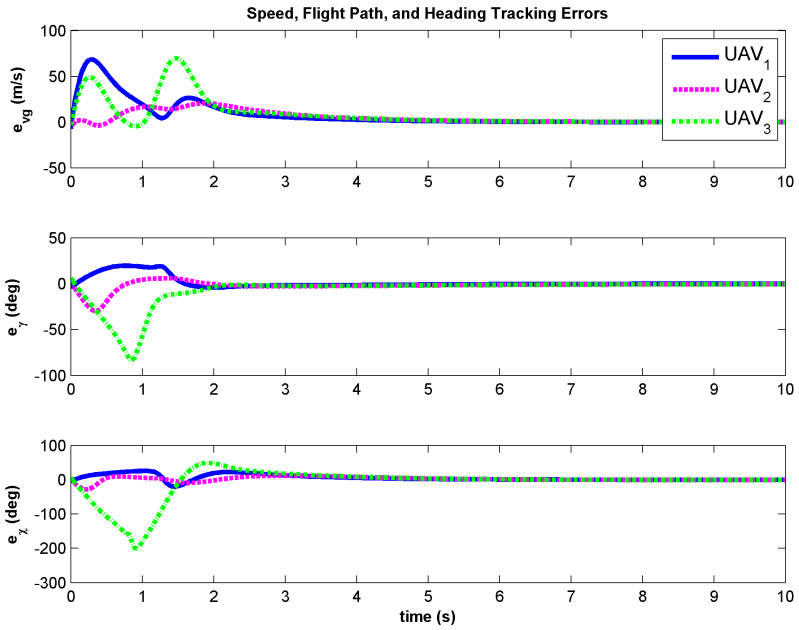


Figure 4.8. Speed, Flight Path, and Heading Errors for 3 UAVs: Topology 1.

$$A_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad L_2 = \begin{bmatrix} 1 & 0 & -1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & 1 \end{bmatrix}$$

The simulation results for the communication topology 2 are shown in Fig. 4.9-4.13. We can see that 4 UAVs achieve cooperative target tracking successfully as expected.

Case 2: Switching Topology (Complete Target Information)

First, we consider for 3 UAVs where the communication topology switches to a new topology at every 5 seconds and these are: $G_1 \rightarrow G_2 \rightarrow G_3$. The simulation run time is 15 seconds. The adjacency matrices for the graphs $\{G1,\ G2,\ G3\}$ are

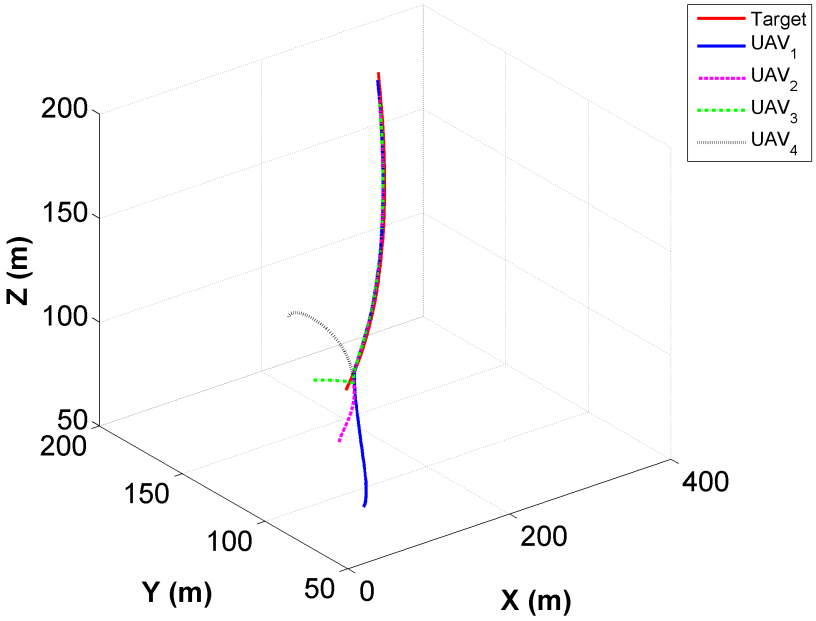


Figure 4.9. Target and 4 Chaser UAVs trajectories for Topology 2.

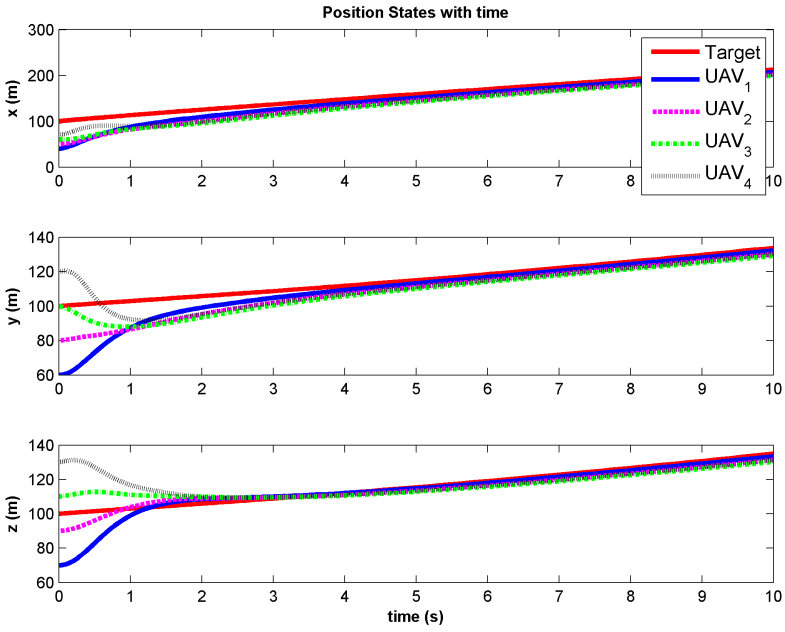


Figure 4.10. Position States with time for 4 UAVs with Topology 2.

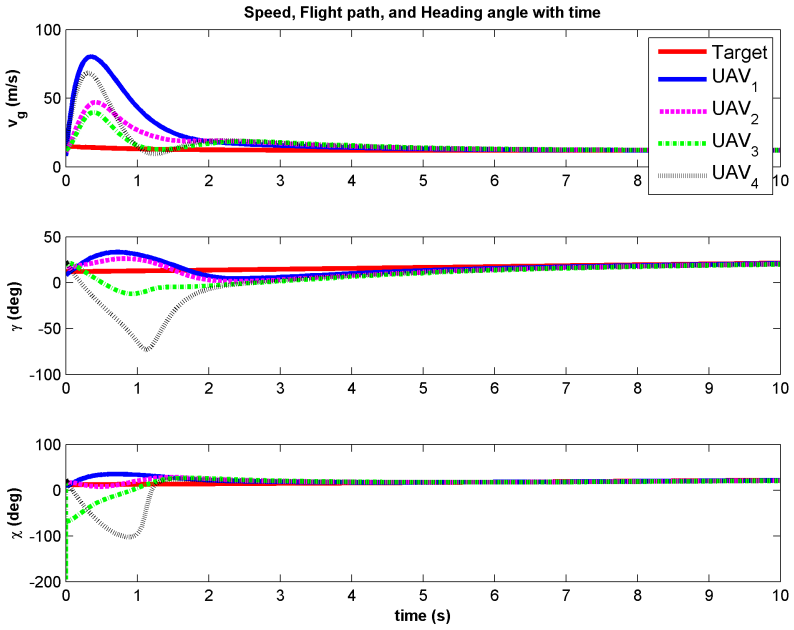


Figure 4.11. Speed, Flight Path, and Heading angle for 4 UAVs with Topology 2.

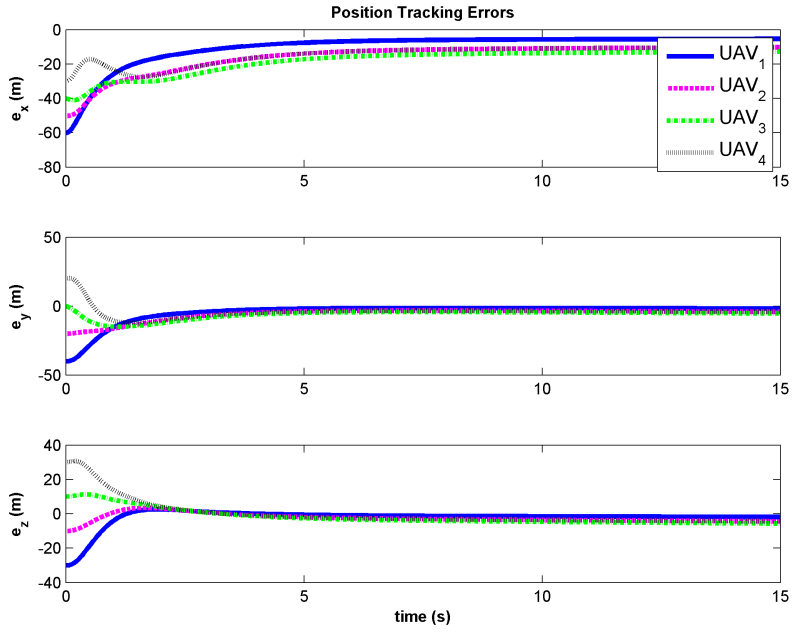


Figure 4.12. Position Tracking Errors for 4 UAVs with Topology 2.

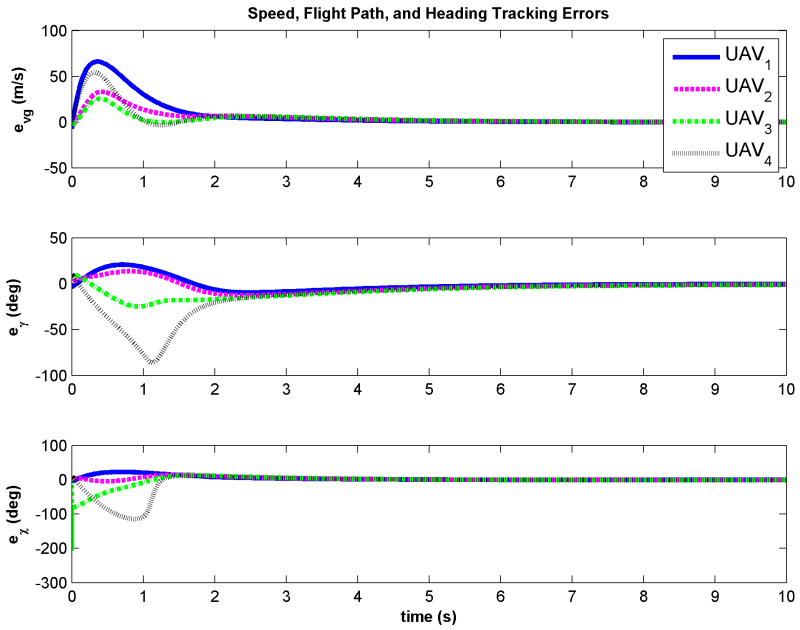
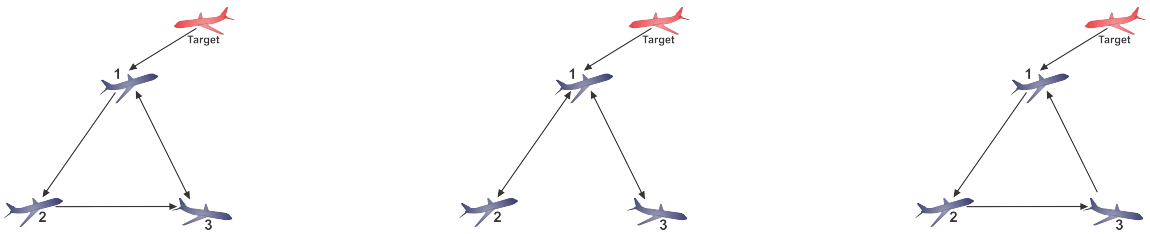


Figure 4.13. Speed, Flight Path, and Heading Errors for 4 UAVs with Topology 2.

respectively:

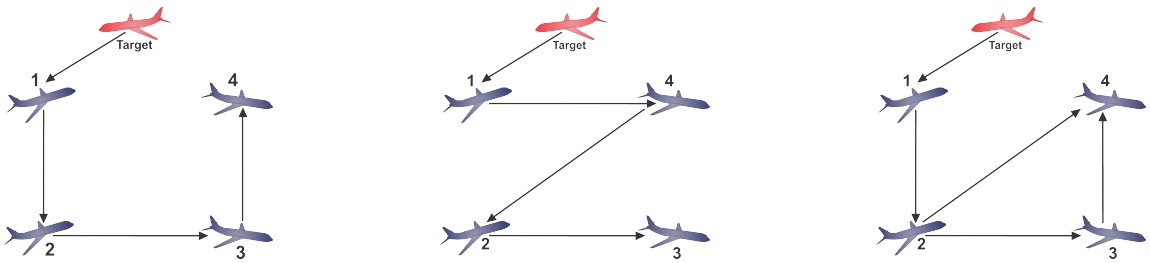
$$A_1 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

For all switching instants, the target vehicle is pinned into the vehicle 1 for all networks i.e. the gain matrices are the same for all cases. For this particular case, we consider the graphs at each interval as strongly connected graphs i.e. at each interval it has a directed spanning tree. From Fig. 4.16, 4.17, and 4.18, we can easily see that all UAVs reach consensus of cooperative tracking since the union of three graphs has a directed spanning tree.



(a) (b) (c)

Figure 4.14. Cooperative Tracking for 3 UAVs with Switching Topologies (a) Graph G_1, (b) Graph G_2, and (c) Graph G_3.



(a) (b) (c)

Figure 4.15. Cooperative Tracking for 4 UAVs with Switching Topologies (a) Graph G_1, (b) Graph G_2, and (c) Graph G_3.

We also consider a group of 4 UAVs as shown in Fig. 4.14 where the adjacency matrices for each switching interval are considered as follows:

$$A_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Each of the topologies at each interval has a directed spanning tree. We can easily verify with the simulation results as shown in Fig. 4.19, 4.20, and 4.21 that the consensus of target tracking is achieved successfully as long as the union of the graphs over the interval has a directed spanning tree.

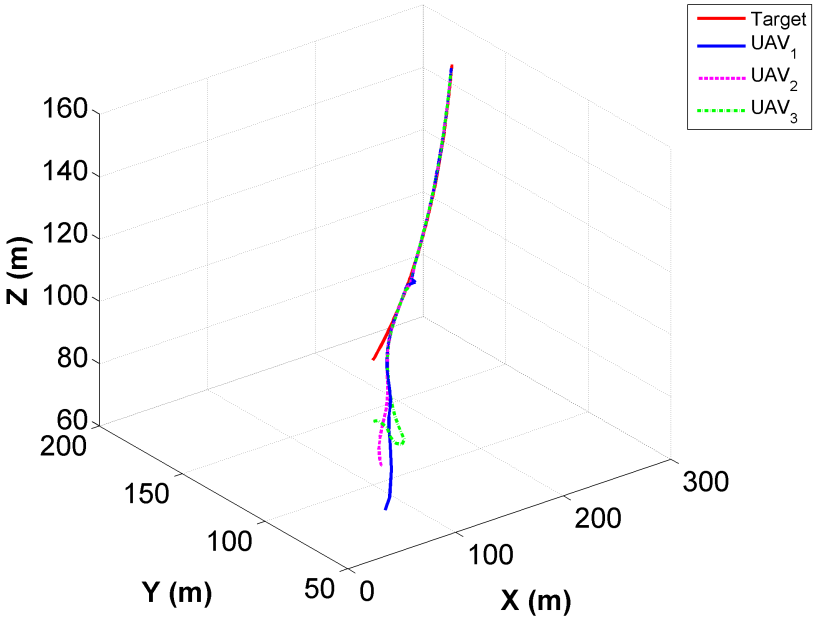


Figure 4.16. XYZ motion for 3 UAVs with switching topology.

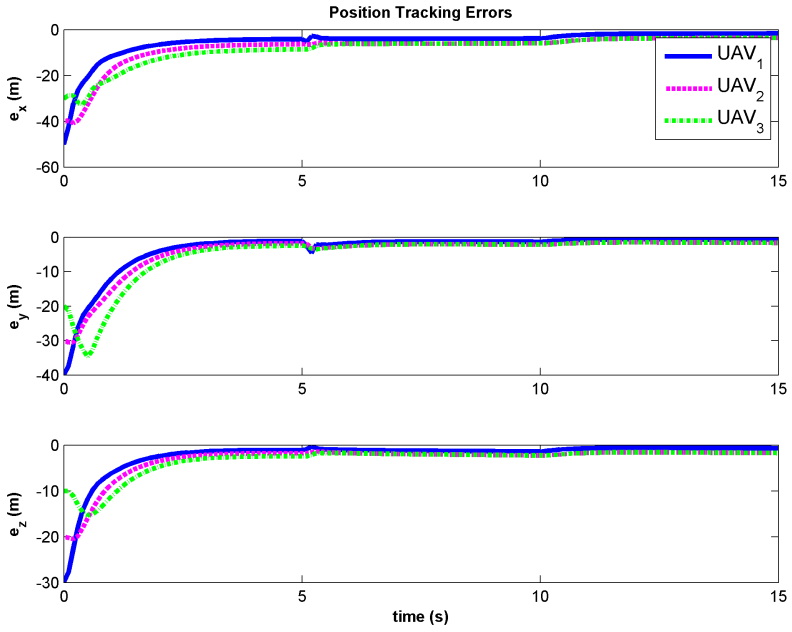


Figure 4.17. Position Tracking Errors with switching topology for 3 UAVs.

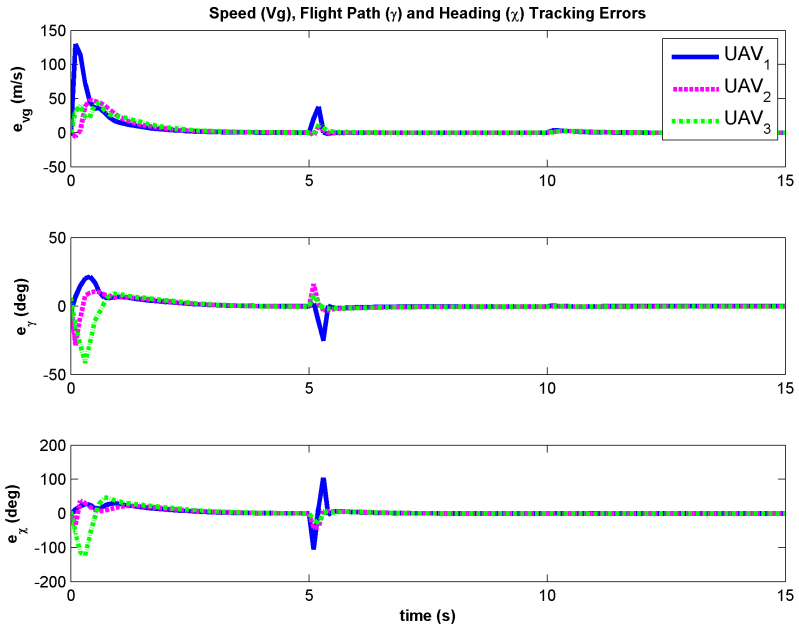


Figure 4.18. Speed, Flight Path, and Heading tracking errors with switching topology for 3 UAVs.

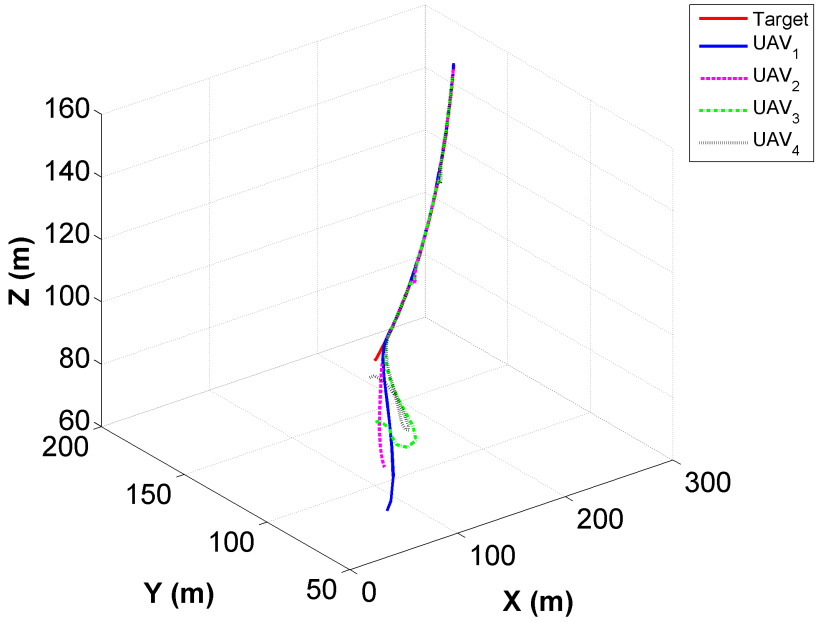


Figure 4.19. XYZ motion for 4 UAVs with switching topology.

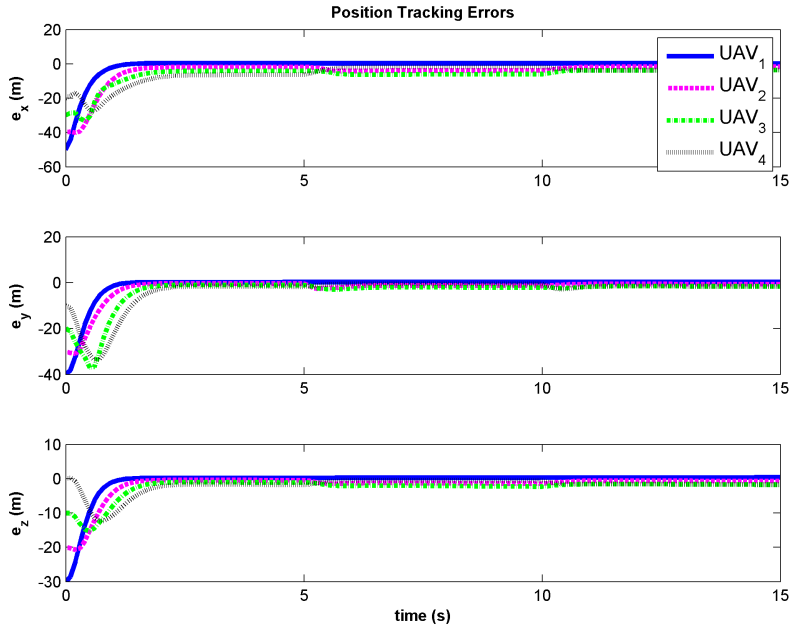


Figure 4.20. Position Tracking Errors with switching topology for 4 UAVs.

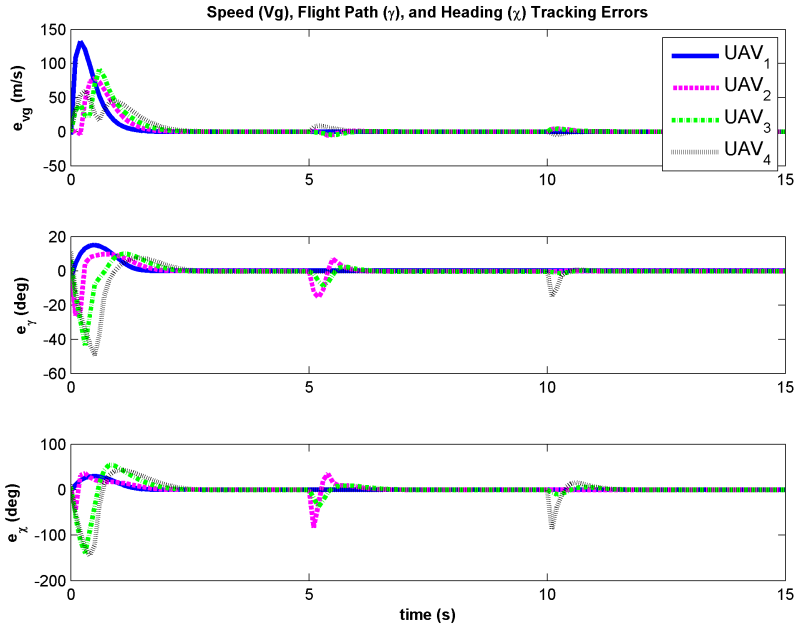


Figure 4.21. Speed, Flight Path, and Heading tracking errors with switching topology for 4 UAVs.

4.3 Estimation based Control Laws for Cooperative UAVs (Partial Target Information)

In this section, an estimation based cooperative control laws is designed for fixed topology and switching topologies when the target full information is not available. Assume, each UAV sends it's own full state information to the neighboring UAVs depending on its connectivity to other UAVs, and perfect communication is available (i.e. there is no communication drop out/corruption by uncertainties between the UAVs). The simulation results are shown for 3 UAVs.

Estimation Based Cooperative Control Laws with Fixed Topology

A similar EKF is designed to estimate the target estimates as described in Chapter 3. For N cooperative UAVs, there are N no. additional EKF needed to calculate its own states.

Continuous-Discrete EKF for Each Chaser State Estimation

We consider N individual estimators for N chaser UAVs of the network. The measurements for each chaser are assumed to be available from its own on-board (GPS + IMU + Air speed) sensor unit and these are: position $x_i,\ y_i,\ z_i$, velocity v_{gi}, flight path angle γ_i, course angle χ_i. For each chaser state estimation, we take the nonlinear continuous-time state model which is affected by the process noise, and the discrete measurement model which is affected by the measurement noise, where all these noises are assumed to be zero-mean Gaussian with known covariances:

$$\begin{aligned} \dot{\mathbf{X}}_i(t) &= \mathbf{f}_i(\mathbf{X}_i(t), \mathbf{u}_i(t), t) + \mathbf{G}_i(t)\mathbf{w}_i(t),\ \mathbf{w}_i(t) \sim N(0, \mathbf{Q}_i(t)) \\ \tilde{\mathbf{y}}_{ik} &= \mathbf{h}_i(\mathbf{X}_{ik}) + \mathbf{v}_{ik},\ \mathbf{v}_{ik} \sim N(0, \mathbf{R}_{ik}) \end{aligned} \tag{4.18}$$

where $i = 1,\ 2,\ 3, \ldots N$, $\mathbf{X}_i = [x_i,\ y_i,\ z_i,\ v_{gi},\ \gamma_i,\ \chi_i]^T$ is the state vector of chaser i, $\mathbf{w}_{ik} = [0, 0, 0,\ \omega_{vgi},\ \omega_{\gamma i},\ \omega_{\chi i}]^T$ is the process noise vector with known covariance $\mathbf{Q}_i(t)$ and $\mathbf{v}_{ik} = [v_{xi},\ v_{yi},\ v_{zi},\ v_{vgi},\ v_{\gamma i},\ v_{\chi i}]^T$ is the measurement noise vector with known

covariance $\mathbf{R}_{ik}$. The expression of function vector $\mathbf{f}_i(\mathbf{X}_i(t), \mathbf{u}_i(t), t)$ used in Eq. 4.18 is:

$$\mathbf{f}_i(\mathbf{X}_i(t), \mathbf{u}_i(t), t) = \begin{bmatrix} v_{gi} \cos\gamma_i \cos\chi_i \\ v_{gi} \cos\gamma_i \sin\chi_i \\ v_{gi} \sin\gamma_i \\ c_{1i}(v_{gi}^c - v_{gi}) \\ c_{2i}(\gamma_i^c - \gamma_i) \\ c_{3i}(\chi_i^c - \chi_i) \end{bmatrix} \tag{4.19}$$

The measurement function vector for chaser i is: $\mathbf{h}_i(\mathbf{X}_{ik}) = [x_i,\ y_i,\ z_i,\ v_{gi},\ \gamma_i,\ \chi_i]_k^T$. For simplification, we consider the matrix $G_i(t)$ is the same as the target.

The EKF equations are developed for each chaser we derived in a similar way as described in Chapter 3 which are:

Kalman gain:

$$\mathbf{K}_{ik} = \mathbf{P}_{ik}^- \mathbf{H}_{ik}^T(\hat{\mathbf{X}}_{ik}^-) \left[\mathbf{H}_{ik}^T(\hat{\mathbf{X}}_{ik}^-) \mathbf{P}_{ik}^- \mathbf{H}_{ik}(\hat{\mathbf{X}}_{ik}^-) + \mathbf{R}_{ik}\right]^{-1} \tag{4.20}$$

The update equations for the state and covariance:

$$\hat{\mathbf{X}}_{ik}^+ = \hat{\mathbf{X}}_{ik}^- + \mathbf{K}_{ik}(\tilde{\mathbf{y}}_{ik} - \mathbf{h}_i(\hat{\mathbf{X}}_{ik}^-)) \tag{4.21}$$

$$\mathbf{P}_{ik}^+ = [\mathbf{I} - \mathbf{K}_{ik}\mathbf{H}_{ik}(\hat{\mathbf{X}}_{ik}^-)]\mathbf{P}_{ik}^- \tag{4.22}$$

The predicted estimate state propagation is:

$$\dot{\hat{\mathbf{X}}}_i(t) = \mathbf{f}_i(\hat{\mathbf{X}}_i(t), \mathbf{u}_i(t), t) \tag{4.23}$$

The error covariance propagation is:

$$\dot{\mathbf{P}}_i(t) = \mathbf{F}_i(\hat{\mathbf{X}}_i(t), t)\mathbf{P}_i(t) + \mathbf{P}_i(t)\mathbf{F}_i^T(\hat{\mathbf{X}}_i(t), t) + \mathbf{G}_i(t)\mathbf{Q}_i(t)\mathbf{G}_i^T(t) \tag{4.24}$$

The Jacobian matrices are:

$$\mathbf{F}_i(\hat{\mathbf{X}}_i(t), t) = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{F}_{1i} \\ \mathbf{0}_{3\times3} & \mathbf{F}_{2i} \end{bmatrix}$$

where

$$\mathbf{F}_{1i} = \begin{bmatrix} \cos\hat{\gamma}_i \cos\hat{\chi}_i & -\hat{v}_{gi} \sin\hat{\gamma}_i \cos\hat{\chi}_i & -\hat{v}_{gi} \cos\hat{\gamma}_i \sin\hat{\chi}_i \\ \cos\hat{\gamma}_i \sin\hat{\chi}_i & -\hat{v}_{gi} \sin\hat{\gamma}_i \sin\hat{\chi}_i & \hat{v}_{gi} \cos\hat{\gamma}_i \cos\hat{\chi}_i \\ \sin\hat{\gamma}_i & \hat{v}_{gi} \cos\hat{\gamma}_i & 0 \end{bmatrix}$$

and $\mathbf{F}_{2i} = diag\left([-c_{1i} \ -c_{2i} \ -c_{3i}]\right)$

$$H_{ik}(\hat{\mathbf{X}}_{ik}^{-}) = I_{6\times6}$$

The output estimate equation is:

$$\hat{\mathbf{y}}_{ik} = \mathbf{h}_i(\hat{\mathbf{X}}_{ik}) \tag{4.25}$$

The controller will exploit the estimation of target state estimator and chasers state estimators and update the control commands for each chaser at each time step based on their connectivity.

Simulation Results

We consider a cooperative group of 3 UAVs and the communication of the UAVs used in the simulation is shown in Figure 4.22. The target UAV is pinned into UAV 1. The adjacency matrix of the network is:

$$A_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

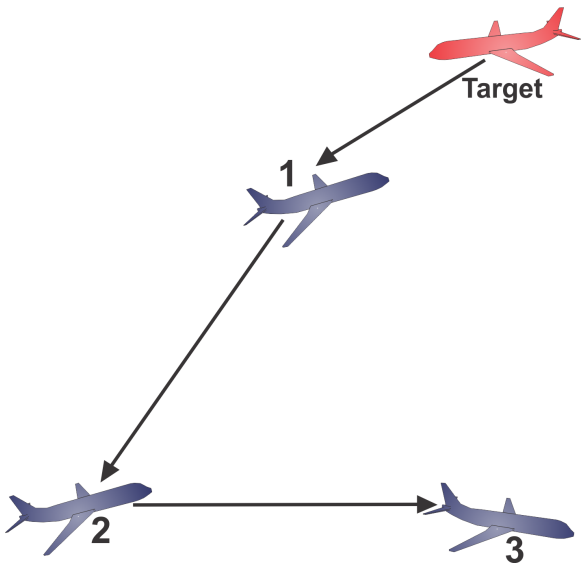


Figure 4.22. Graph Topology for Cooperative UAVs.

We simulate the model for 20 seconds with time step of $T = 0.05$ *sec*. The simulation results show that the estimated states of chaser UAVs track the estimated state of the target UAV successfully. The following design parameters are used in the simulation to obtain the good results: $c_{1i} = 100,\ c_{2i} = 100,\ c_{3i} = 100,\ \alpha_{1i} = 5,\ \alpha_{2i} = 5,\ \alpha_{3i} = 5,\ \lambda_{vgi} = 5,\ \lambda_{\gamma i} = 5,\ \lambda_{\chi i} = 5$ where $i = 1,\ 2,\ 3$.

We used the following parameters for generating the truth model of Target UAV: $v_{gr}(0) = 10\ m/s,\ \gamma_r(0) = 10°,\ \chi_r(0) = 12°,\ x_r(0) = 100\ m,\ y_r(0) = 100\ m,\ z_r(0) = 100\ m,\ v_{gr}^c = 12\ m/s$,$a_{hr}^c = 0.2\ m/s^2$,$a_{vr}^c = 0.2\ m/s^2$.
The initial guess of the estimated states of target trajectory are: $v_{gre}(0) = 11\ m/s,\ \gamma_{re}(0) = 35°,\ \chi_{re}(0) = 5°,\ x_{re}(0) = 102\ m,\ y_{re}(0) = 102\ m,\ z_{re}(0) = 102\ m$. The performance of EKF for Target UAV are shown in Fig. 4.25 and 4.26. The states errors are within $3 - \sigma$ bounds as shown in Fig. 4.25.

We consider the following initial conditions for chaser UAVs to obtain the truth model: $x_1(0) = 90\ m,\ y_1(0) = 90\ m,\ z_1(0) = 90\ m,\ v_{g1}(0) = 11\ m/s,\ \gamma_1(0) = 12°,\ \chi_1(0) = 12°,\ x_2(0) = 50\,m,\ y_2(0) = 50\,m,\ z_2(0) = 50\,m,\ v_{g2}(0) = 11\,m/s,\ \gamma_2(0) =$

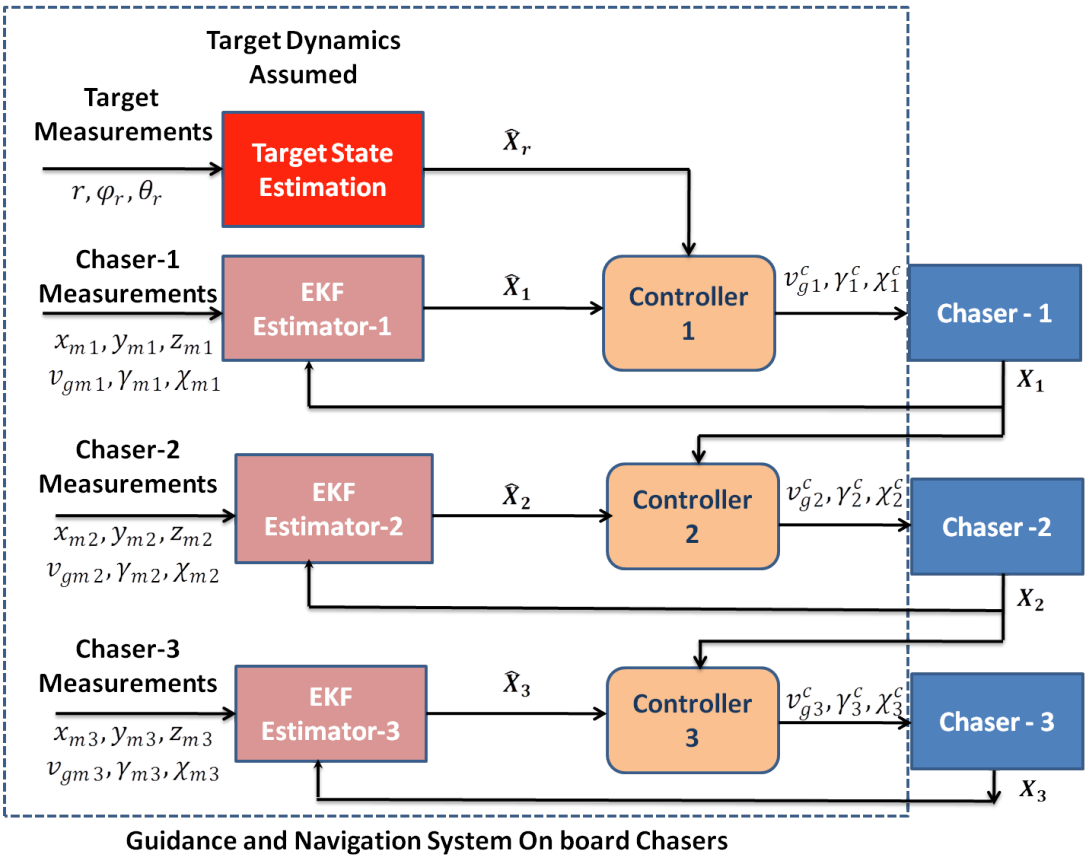


Figure 4.23. Flow Chart for Cooperative Target Tracking of 3 UAVs.

$10°,\ \chi_2(0) = 10°,\ x_3(0) = 60\,m,\ y_3(0) = 30\,m,\ z_3(0) = 30\,m,\ v_{g3}(0) = 12\,m/s,\ \gamma_3(0) = 12°,\ \chi_3(0) = 12°$.

The initial conditions for the estimated states: $x_{e1}(0) = 102\ m,\ y_{e1}(0) = 102\ m,\ z_{e1}(0) = 102\ m,\ v_{ge1}(0) = 12\ m/s,\ \gamma_{e1}(0) = 17°,\ \chi_{e1}(0) = 17°,\ x_{e2}(0) = 52\ m,\ y_{e2}(0) = 52\ m,\ z_{e2}(0) = 52\ m,\ v_{ge2}(0) = 12\ m/s,\ \gamma_{e2}(0) = 15°,\ \chi_{e2}(0) = 15°,\ x_{e3}(0) = 62\ m,\ y_{e3}(0) = 32\ m,\ z_{e3}(0) = 32\ m,\ v_{ge3}(0) = 13\ m/s,\ \gamma_{e3}(0) = 17°,\ \chi_{e3}(0) = 17°$. The initial covariance for the target UAV is $\mathbf{P}_0 = I_{6\times6}$ and for chaser UAVs, it is considered to be the same for simplicity i.e. $\mathbf{P}_{i0} = (I_{6\times6})$, where I is an identity matrix.

The covariance of process noise for the target and chaser UAV are:

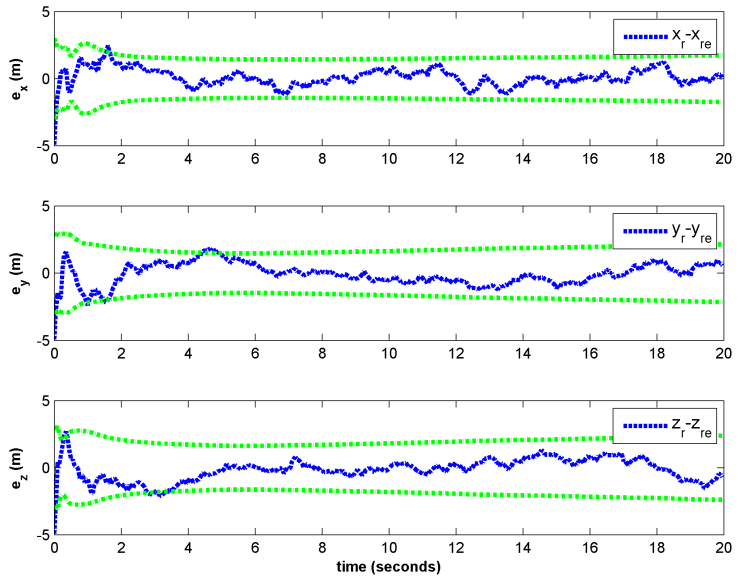


Figure 4.24. Errors (true-estimated) with $3-\sigma$ bounds for Target UAV.

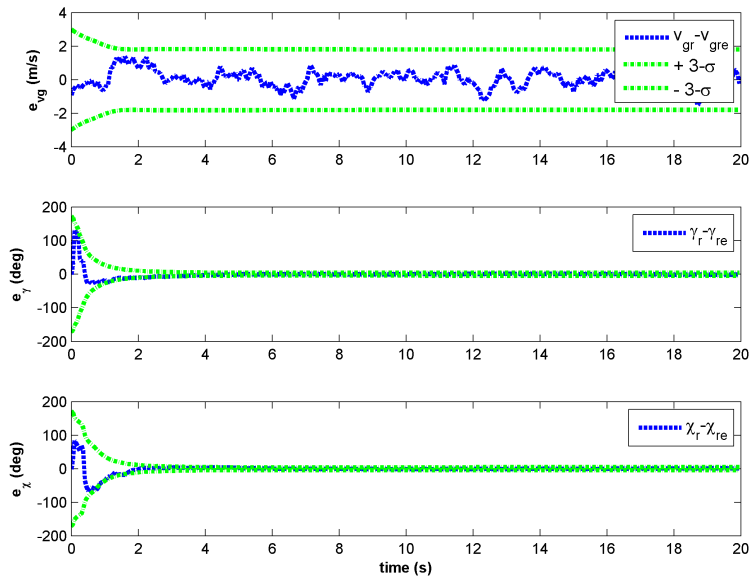


Figure 4.25. Errors (true-estimated) with $3-\sigma$ bounds for Target UAV.

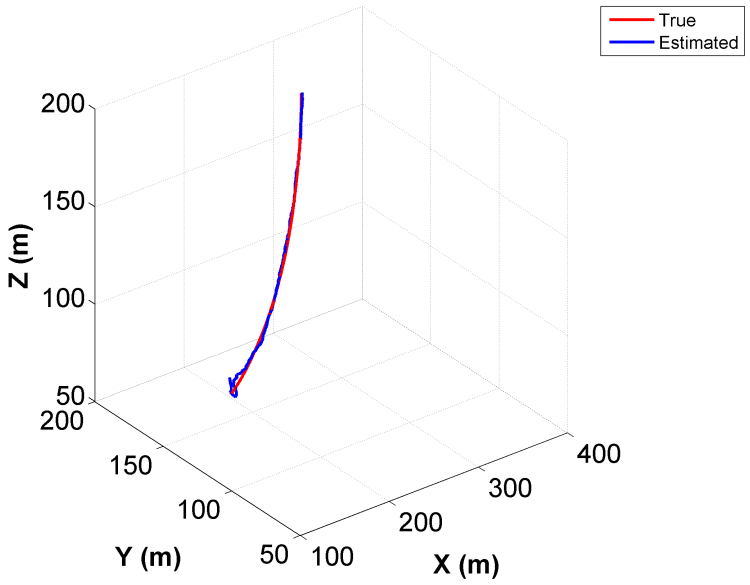


Figure 4.26. True and Estimated Target Trajectory.

$\mathbf{Q}_r(t) = \mathbf{diag}([1\ (m/s)^2,\ 0.65\ deg^2,\ 0.65\ deg^2])$ and $\mathbf{Q}(t)$ is considered to be the same for all chasers. The covariance matrix of the measurement noises for the target and chaser UAV are:

$R_{rk} = \mathbf{diag}([1\ m^2,\ 0.65\ deg^2,\ 0.65\ deg^2])$ and

$R_ik = \mathbf{diag}([1\ m^2,\ 1\ m^2,\ 1\ m^2,\ 1\ (m/s)^2,\ 0.65\ deg^2,\ 0.65\ deg^2])$.

The performance of EKF for target state estimation is shown in Fig. 4.25 and 4.26 which provides the best estimate of target true states. The estimated target states are within $3-\sigma$ error bounds as shown in Fig. 4.25. Fig. 4.27 and 4.29 show the performance of the controller and estimators for cooperative UAVs. All UAVs achieve consensus of target tracking under noisy environments. The estimated states of each chaser are successfully tracking the estimated states of target UAV as shown in Fig. 4.27. The $3-\sigma$ error bounds are shown for 3 chasers in Fig. 4.30, Fig. 4.31, and Fig. 4.32 respectively.

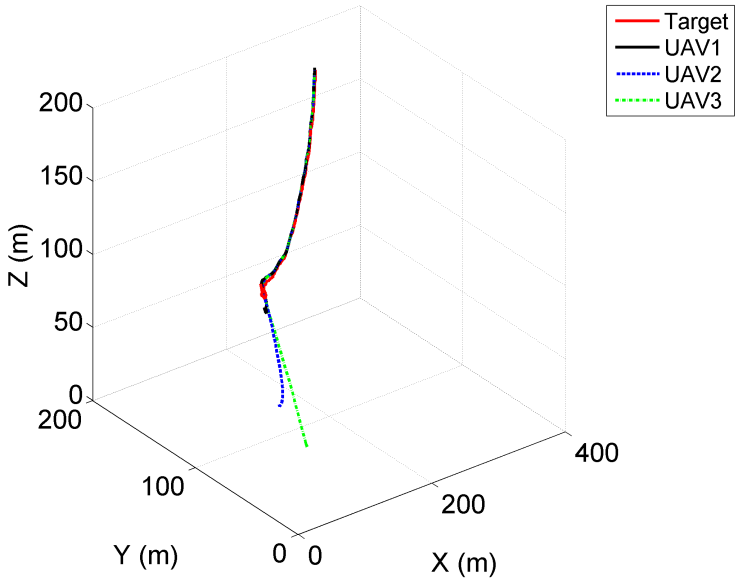


Figure 4.27. Estimated Cooperative Chasers track the estimated Target UAV.

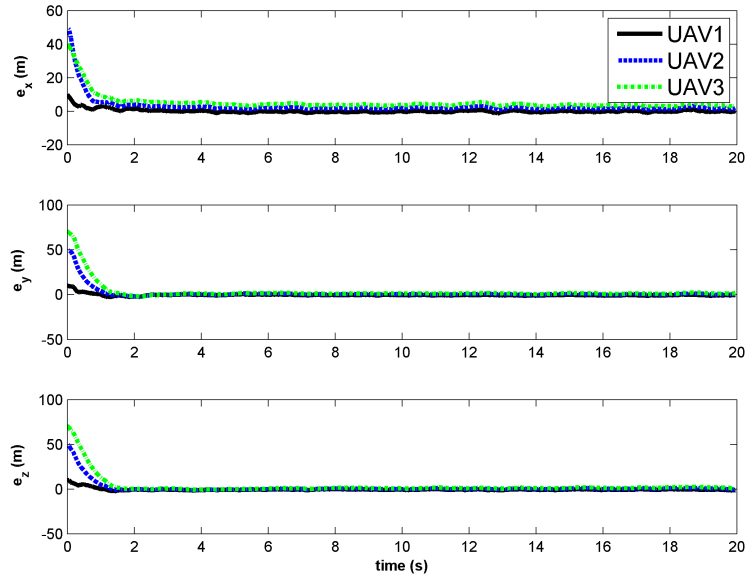


Figure 4.28. Tracking Errors (estimated target-estimated chaser).

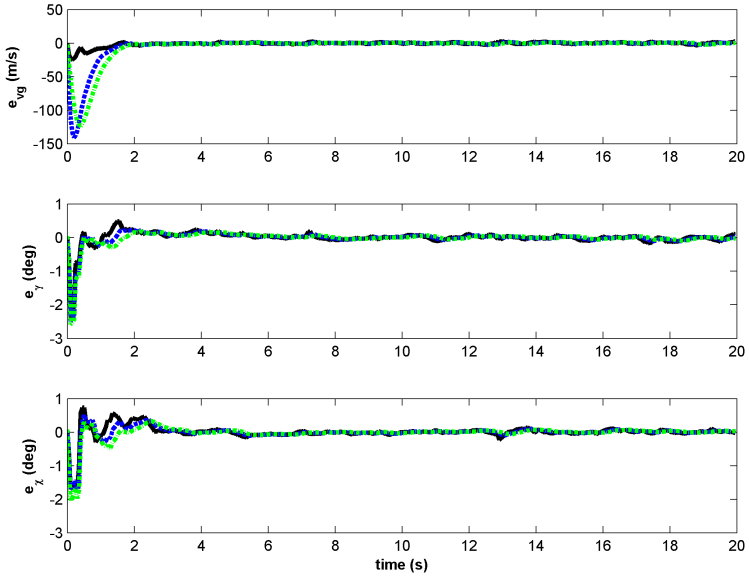


Figure 4.29. Tracking Errors (estimated target-estimated chaser).

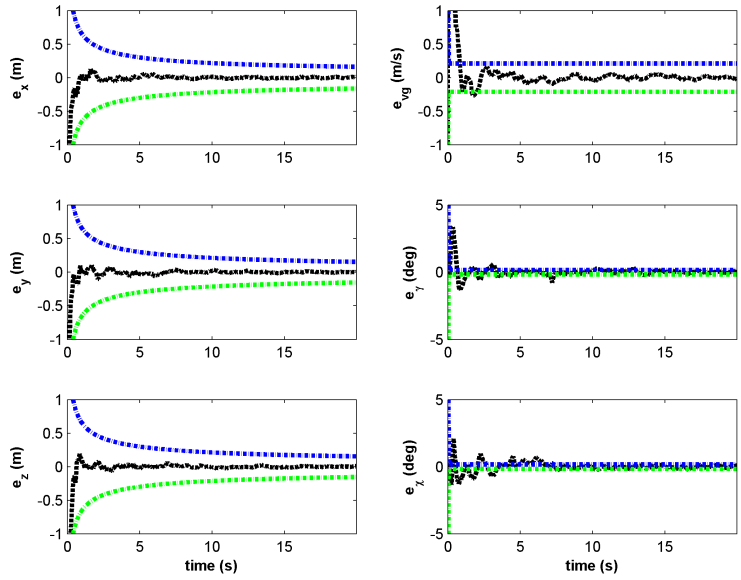


Figure 4.30. Errors (true-estimated) with $3-\sigma$ bounds for chaser 1.

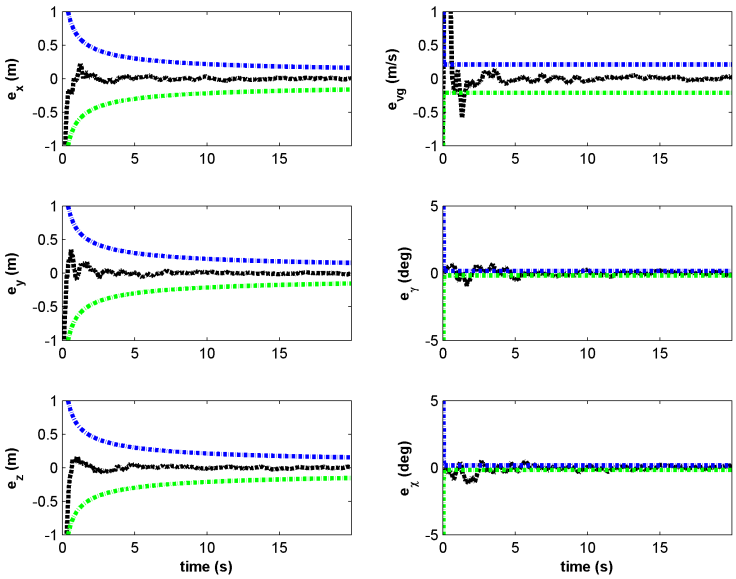


Figure 4.31. Errors (true-estimated) with $3-\sigma$ bounds for chaser 2.

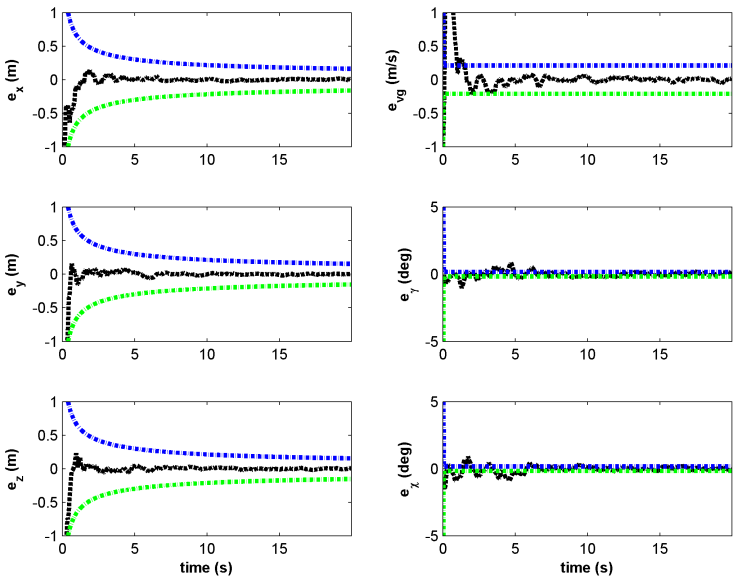


Figure 4.32. Errors (true-estimated) with $3-\sigma$ bounds for chaser 3.

Target Tracking for Cooperative UAVs with Switching Topology (Partial Target Information)

In this section, an estimation based cooperative target tracking for the switching topology scenario is simulated when the partial target information is available to the UAVs. The simulation is performed for the same scenario as it is considered for 3 cooperative UAVs in Chapter 3 for switching topology case. Consider, partial target information is available to at least one node of the network. The simulation results are shown in Fig. 4.33-4.38. The filter performance provides reasonably good approximation about the target state, and the controller provides satisfactory tracking performance as long as the union of the graphs has a directed spanning tree.

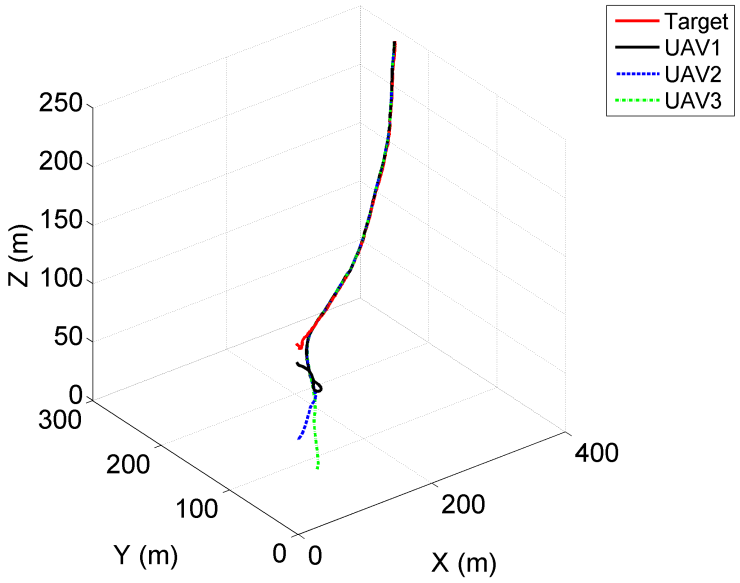


Figure 4.33. Estimated Cooperative Chasers track the estimated Target UAV.

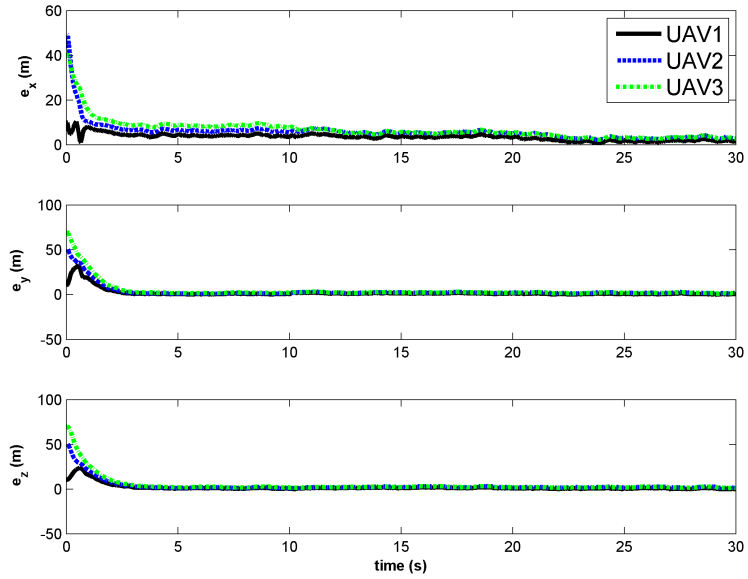


Figure 4.34. Position Tracking Errors (estimated target-estimated chaser).

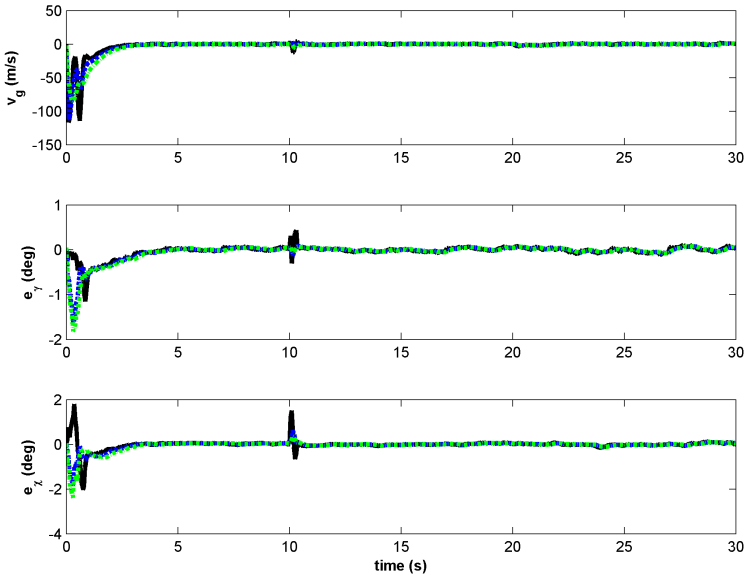


Figure 4.35. Speed, Flight Path, and Heading angle Tracking Errors (estimated target-estimated chaser).

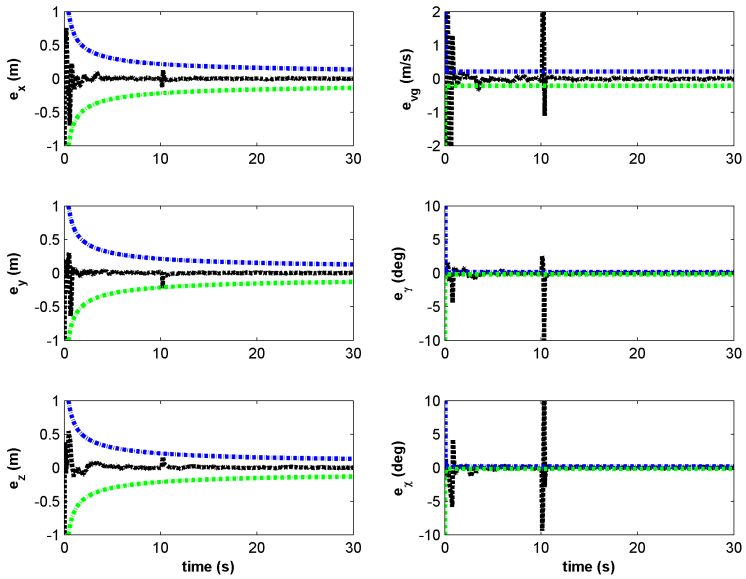


Figure 4.36. Errors (true-estimated) with $3-\sigma$ bounds for chaser 1.

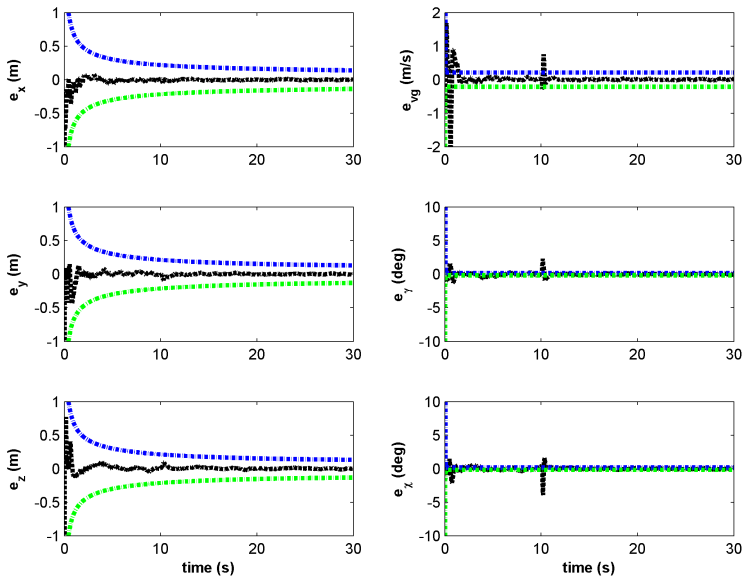


Figure 4.37. Errors (true-estimated) with $3-\sigma$ bounds for chaser 2.

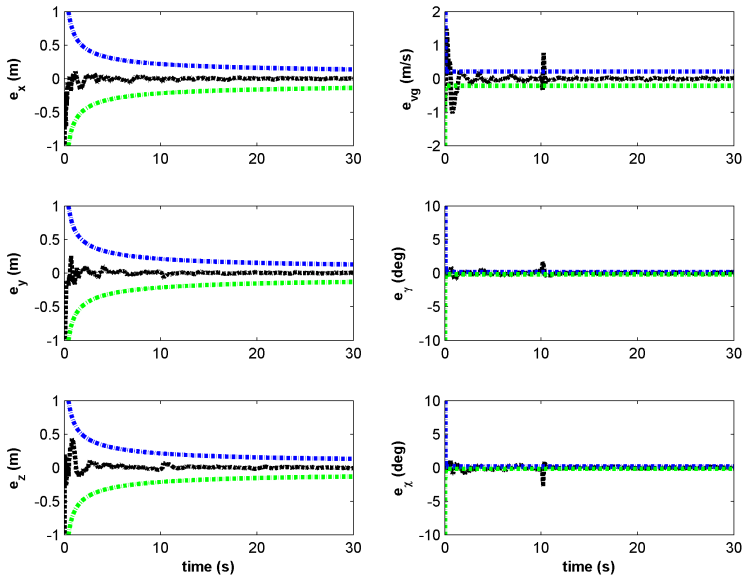


Figure 4.38. Errors (true-estimated) with $3-\sigma$ bounds for chaser 3.

CHAPTER 5

SUMMARY, CONCLUSIONS, AND FUTURE WORK

5.1 Summary, and Conclusions

The major contribution of this research is to design a simplified cooperative control algorithms for multiple UAVs from the dynamics and control point of view, that can be applied to a target tracking application in $3D$ environment. A primary foundation of the control strategies for a point mass UAV with nonlinear dynamics is developed based on backstepping like approach and then extended for cooperative group of UAVs. Inter-vehicle communication, accuracy of sensor devices, and group control strategies are critical factors for cooperative control problem. We address these three critical factors for UAVs in the context of target tracking application. The cooperation between the UAVs is represented using algebraic graph tools, and a cooperative control technique is designed for multiple UAVs which computes three control commands (velocity, flight path angle, and course angle) while guaranteeing asymptotic tracking of a target UAV. An estimation based controller is then integrated to achieve robust performance with measurements or sensor uncertainties. We summarize the main research findings and results as below:

1. A backstepping based controller is derived for target tracking with a single UAV and demonstrated with the numerical simulation results for several defined target trajectory. The controller is derived for two different UAV models and the tracking performance is shown for each case. The control laws proposed in this dissertation guarantees the asymptotic stability, the convergence of tracking-error is ultimately uniformly bounded if the target trajectory and its

derivatives are bounded upto order 2. The controller shows satisfactory tracking performance which can be used to track any arbitrary defined trajectory.

2. Estimation based controller is designed for the case when the limited state information about the target is available. A nonlinear estimator is included to estimate the target states by synthesizing the target range, azimuth angle, and elevation angle measurements; and a separate estimator is integrated to estimate UAV states. Estimation based controller is designed for white and colored noise measurement uncertainties. For both cases, the controller achieves tracking successfully and all tracking errors are bounded within $3-\sigma$. For colored noise uncertainties, the tracking is achieved at the expense of computational cost since it increases the system's state description. An investigation is also performed when the noise variances for target measurements are function of range.
3. A cooperative controller is derived for target tracking with multiple UAVs. A consensus-like cooperation is developed based on the graph theoretic formulation, and then backstepping like technique is utilized to design the controller. Lyapunov stability analysis is performed and it shows that the controller shows asymptotic tracking convergence with bounded target. Simulation results are provided to demonstrate the efficacy of the proposed approach for different types of graph topologies with perfect state information about the target: fixed time-invariant and switching topology
4. Estimation based cooperative controller is derived and implemented for multiple UAVs when the partial target information available to the network. The estimator is designed for white noise uncertainties and simulate for two different graph topology: fixed time-invariant and switching topology. The nonlinear estimator performs well with limited or corrupted state information knowledge.

5.2 Future Work

Several potential extensions of this research can be possible based on the contributions stated in this dissertation. We implemented control strategies for a point mass UAV model; further extension of this research can be applied for a more realistic UAV model which is well described and design the controller with including vehicle and environmental constraints into it.

The communication between the UAVs may be interrupted/lost due to external disturbances/potential threats which can be fixed or dynamically changing. In some cases, the target behavior may become very wild for example environmental hazardous- cyclone, hurricane. An investigation can be performed in this context by including highly maneuvering target with highly unpredictable target state. We may also include switched target model, and design the controller for UAV, and suggest the feasible technique to track the target in such cases. Careful design is required to address the proper coordination of UAVs so that it can achieve the objective even if some communications fail.

Additionally, there are some other issues such as communication time delays, collision avoidance techniques should be accounted for to maximize group capabilities. A further investigation can also be directed to design the appropriate estimator with the more realistic UAV model including non-Gaussian noise characteristics.

REFERENCES

[1] G. L. Stephens, R. G. Ellingson, J. V. Jr., W. Bolton, T. P. Tooman, F. P. J. Valero, P. Minnis, P. Pilewskie, G. S. Phipps, S. Sekelsky, J. R. Carswell, S. D. Miller, A. Benedetti, R. B. McCoy, R. F. M. Jr., L. A., and R. Bambhaj, "The department of energys atmospheric radiation measurement (arm) unmanned aerospace vehicle (uav) program," American Meteorological Society, vol. 81, pp. 2915–2938, 2000.

[2] D. Lawrence, E. Frew, and W. Pisano, "Lyapunov vector fields for autonomous unmanned aircraft flight control," Journal of Guidance, Control, and Dynamics, vol. 31, no. 5, pp. 1220–1229, September-October 2008.

[3] E. W. Frew, D. A. Lawrence, C. Dixon, J. Elston, and W. J. Pisano, "Lyapunov guidance vector fields for unmanned aircraft applications," American Control Conference, 2007.

[4] H. S. Tyler and M. R. Akella, "Coordinated standoff tracking of moving targets: Control laws and information architectures," Journal of Guidance, Control, and Dynamics, vol. 32, no. 1, pp. 56–69, January-February 2009.

[5] D. Kingston, R. Beard, T. McLain, M. Larsen, and W. Ren, "Autonomous vehicle technologies for small fixed wing uavs," Proc. AIAA Unmanned and Unlimited Systems, Technologies, and Operations, no. Paper No. AIAA 2003-6559., September 2003, san Diego, California.

[6] D. R. Nelson, D. B. Barber, T. W. McLain, and R. W. Beard, "Vector field path following for miniature air vehicles," IEEE Transactions on Robotics, vol. 23, no. 3, pp. 519–529, June 2007.

[7] S. R. Griffiths, "Vector field approach for curved path following for miniature aerial vehicles," AIAA Guidance, Navigation, and Control Conference and Exhibit, no. AIAA 2006-6467, 21 - 24 August 2006.

[8] S. Park, J. Deyst, and J. P. How, "A new nonlinear guidance logic for trajectory tracking," AIAA Guidance, Navigation and Control Conference and Exhibit, August 2004.

[9] W. Ren, "Formation keeping and attitude allignment for multiple spacecraft through local interactions," AIAA, Journal of Guidance,Control and Dynamics, vol. 30, no. 2, pp. 633–638, March-April 2007.

[10] H. Chen, K. Chang, and C. S. Agate, "Tracking with uav using tangent-plus-lyapunov vector field guidance," 12th International Conference on Information Fusion, 2009.

[11] M. Guelman, "The closed-form solution of true proportional navigation," IEEE Transactions on Aerospace and Electronic Systems, vol. AES-12, pp. 472 – 482, 1976.

[12] C.-D. Yang and C.-C. Yang, "A unified approach to proportional navigation," IEEE Transactions on Aerospace and Electronic Systems, 1997.

[13] T. Yamasaki, H. Sakaida, and K. Enomoto, "Robust trajectory-tracking method for uav guidance using proportional navigation," International Conference on Control, Automation and Systems, 2007.

[14] G. Ambrosino, M. Ariola, F. Ciniglio, F. Corraro, E. D. Lellis, and A. Pironti, "Path generation and tracking in 3-d for uavs," IEEE Transactions on Control Sytems Technology, vol. 17, no. 4, pp. 980–988, July 2009.

[15] C.-D. Yang and H.-Y. Chen, "Three-dimensional nonlinear h_∞ guidance law," International Journal of Robust and Nonlinear Control, vol. 11, pp. 109–129, 2001.

[16] S. Bharadwaj, A. V. Rao, and K. D. Mease, "Entry trajectory tracking law via feedback linearization," Journal of Guidance, Control and Dynamics, vol. 21, no. 5, September-October 1998.

[17] R. Srivastava, A. K. Sarkar, D. Ghose, and S. Gollakota, "Nonlinear three dimensional composite guidance law based on feeback linearization," AIAA Guidance, Navigation and Control Conference and Exhibit, August 2004.

[18] A. K. Sarkar, Po-Kanchanbagh, and D. Ghose, "Realistic pursuer evader engagement with feedback linearization based nonlinear guidance law," AIAA Gu, August 2006.

[19] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, Nonlinear and adaptive control design. Wiley-Interscience, May 1995.

[20] W. Ren and E. Atkins, "Nonlinear trajectory tracking for fixed wing uavs via backstepping and parameter adaptation," AIAA Guidance, Navigation, and Control Conference and Exhibit, no. AIAA-2005-6196, August 2005.

[21] D. Jung and P. Tsiotras, "Bank-to-turn control for a small uav using backstepping and parameter adaptation," 17th World Congress, The International Federation of Automatic Control, July 2008.

[22] N. Lechevin and C. A. Rabbath, "Backstepping guidance for missile modeled as uncertain time-varying first-order systems," American Control Conference, 2007.

[23] L. Lapierre and D. Soetanto, "Nonlinear path-following control of an auv," Ocean Engineering, Elsevier, vol. 34, pp. 1734–1744, 2007.

[24] W. J. Dong, Y. Guo, and J. A. Farrell, "Formation control of nonholonomic mobile robots," American Control Conference, pp. 5602–5607, 14-16 June 2006.

[25] S.-J. Chung and J.-J. Slotine, "Cooperative robot control and concurrent synchronization of lagrangian systems," IEEE transactions on Robotics, vol. 25, no. 3, pp. 686–700, 2009.

[26] D. Gu, "A differential game approach to formation control," IEEE Transactions on Control System Technology, vol. 16, no. 1, pp. 85–93, 2008.

[27] M. Mesbahi and F. Y. Hadaegh, "Formation flying control of multiple spacecraft via graphs,matrix inequalities and switiching," AIAA, Journal of Guidance,Control and Dynamics, vol. 24, no. 2, pp. 369–377, March-April 2001.

[28] C. Peterson and D. A. Paley, "Cooperative control of unmanned vehicles in a time-varying flowfield," AIAA Guidance, Navigation and Control Conference and Exhibit, 2009.

[29] A. Bester, P. A. Vela, G. Pryor, and A. Tannenbaum, "Flying in formation using pursuit guidance algorithm," American Control Conference, June 2005.

[30] M. A. Anderson and A. C. Robbins, "Formation flight as a cooperative game," AIAA, 1998.

[31] C. R. Mclnnes, "Autonomous ring formation for a planar constellation of satellites," AIAA Journal of Guidance, Navigation and Control, vol. 18, no. 5, pp. 1215–1217, 1995.

[32] E. Fiorelli, N. Leonard, P. Bhatta, D. Paley, R. Bachmayer, and D. Fratantoni, "Multi-auv control and adaptive sampling in monterey bay," IEEE Journal of Oceanic Engineering, vol. 31, no. 4, pp. 935–948, 2006.

[33] S. Nair and N. E. Leonard, "Stable synchronization of rigid body networks," American Institute of Mathematical Sciences, vol. 2, no. 4, p. 595624, 2007.

[34] I. I. Kaminer, O. A. Yakimenko, V. N. Dobrokhodov, and M. I. Lizarraga, "Cooperative control of small uavs for naval applications," IEEE Conference on Decision and Control, December 2004.

[35] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filters," 44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05, pp. 8179 – 8184, December 2005.

[36] C. W. Reynolds, "Flocks, herds, and schools: A distributed behavioral model," Computer Graphics, vol. 21, no. 4, pp. 25–34, 1987.

[37] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: algorithms and theory," IEEE Transactions on Auto, vol. 51, pp. 410–420, March 2006.

[38] V. Gazi and K. M. Passino, "Stability analysis of social foraging swarms," IEEE Transactions on Systems, Man and Cybernetics, vol. 34, no. 1, February 2004.

[39] W. Ren, R. W. Beard, and E. M.Atkins, "Information consensus in multivehicle cooperative control," IEEE Control Systems Magazine, vol. 27, no. 2, pp. 71–82, April 2007.

[40] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," IEEE Transactions on Automatic Control, vol. 50, no. 5, pp. 655–661, May 2005.

[41] R. Olfati-Saber and R. M.Murray, "Consensus problems in networks of agents with swtiching topology and time delays," IEEE Transactions on Automatic Control, vol. 49, no. 9, pp. 1520 – 1533, September 2004.

[42] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," IEEE Transactions on Automatic Control, vol. 49, no. 9, pp. 1465–1476, September 2004.

[43] R. Olfati-Saber and R. M. Murray, "Agreement problems in networks with directed graphs and switiching topology," IEEE Conference on Decision and Control, 2003.

[44] J. Kim, M. Tandale, P. K. Menon, and E. Ohlmeyer, "Particle filter for ballistic target tracking with glint noise," AIAA Guidance, Navigation and Control Conference and Exhibit, 2010.

[45] D. V. Dimarogonas and K. H. Johansson, "Quantized agreement under time-varying communication topology," American Control Conference, 2008.

[46] B. Bethke, M. Valenti, and J. How, "Cooperative vision based estimation and tracking using mulitple uavs," in Lecture Notes in Control and Information Sciences, Advances in Cooperative Control and Optimization, vol. 369, 2007.

[47] W. Ren, R. W. Beard, and D. B. Kingston, "Multi-agent kalman consensus with relative uncertainity," American Control Conference, June 2005.

[48] R. Olfati-Saber, "Distributed kalman filtering for sensor networks," IEEE Conference on Decision and Control, pp. 5492 – 5498, 2007.

[49] J. L. Crassidis and J. L. Junkins, Optimal Estimation of Dynamic Systems. Chapman and Hall/CRC, 2004.

[50] M. Alighanbari and J. P. Ho, "An unbiased kalman consensus algorithm," American Control Conference, pp. 3519–3524, June 2006.

[51] J. Garcia-Velo and B. Walker, "Aerodynamic parameter estimation for higher performance aircraft using extended kalman filtering," Journal . Guidance, Control and Dynamics, vol. 20, pp. 1257–1259, 1997.

[52] N. K. Philip and M. R. Ananthasayanam, "Relative position and attitude estimation and control schemes for the final phase of an autonomous docking mission of spacecraft," Acta Astronautica, vol. 52, Issue 7, pp. 511–522, 2003.

[53] D. C. Woffinden and D. K. Geller, "Relative angles-only navigation and pose estimation for autonomous orbital rendezvous," AIAA/AAS Astrodynamics Specialist Conference and Exhibit, 2006.

[54] R. Olfati-Saber and J. S. Shamma, "Consensus fliters for sensor networks and distributed sensor fusion," IEEE Conference on Decision and Control, 2005.

[55] G. G. Rigatos, "Distributed filtering over sensor networks for autonomous navigation of uavs," IEEE Vehicular Technology Conference Fall, 2010.

[56] R. Chen, "Monte carlo fliters and its applications in target tracking and wireless communications," Sixth U.S. Army Conference on Applied Statistics (ACAS), 2000.

[57] P. B. Sujit and R. Beard, "Multiple uav path planning using anytime algorithms," American Control Conference, pp. 2978–2983, June 2009.

[58] P. A. Ioannou and J. Sun, Robust Adaptive Control. Prentice Hall, 1995.

[59] K. Subbarao, "Structured adaptive model inversion (sami): Theory and applications to trajectory tracking for nonlinear dynamical systems," Ph.D. dissertation, Texas A&M University, 2001.

[60] C.-T. Chen, Linear System Theory and Design, 3rd, Ed. Oxford University Press, 1999.

[61] J.-J. E. Slotine and W. Li, Applied Nonlinear Control. Prentice-Hall, Inc., 1991.

[62] S. M. Ross, Stochastic Processes. John Wiley & Sons, Inc., 1996.

[63] ——, Introduction to Probablity Models, 9th ed., Ninth, Ed. Academic Press, 2006.

[64] S. Chapra and R. Canale, Numerical Methods for Engineers. McGraw-Hill Science/Engineering/Math, 2005, iSBN-13: 978-0073101569.

[65] N. J. Kasdin, "Runge-kutta algorithm for the numerical integration of stochastic differential equations," Journal of Guidance, Control, and Dynamics, AIAA, vol. 18, no. 1, pp. 114–120, 1995.

[66] ——, "Discrete simulation of colored noise and stochastic processes and $1/f^{\alpha}$ power law noise generation," Proceedings of the IEEE, vol. 3, May 1995.

[67] J. R. Dormand and P. J. Prince, "A family of embedded runge-kutta formulae," Journal of Computational and Applied Mathematics, vol. 6, pp. 19–26, 1980.

[68] A. Tayebi and S. McGilvray, "Attitude stabilization of a vtol quadrotor aircraft," IEEE Transactions on Control System Technology, vol. 14, pp. 562–571, 2006.

[69] F. Morbidi, C. Ray, and G. L. Mariottini, "Cooperative active target tracking for heterogeneous robots with application to gait monitoring," IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3608 – 3613, 2011.

[70] W. Ren and R. Beard, Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications. Springer, 2007.

[71] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," IEEE Transactions on Automatic Control, vol. 48, pp. 988 – 1001, June 2003.

[72] J. Wolfowitz, "Products of indecomposable, aperiodic, stochastic matrices," Proc. American Mathematical Society, pp. 733–737, 1963.

[73] X. F. Wan and G. Chen, "Pinning control of scale-free dynamical network," Physica, pp. 521–531, 2002.

BIOGRAPHICAL STATEMENT

Mousumi Ahmed was born in Sylhet, Bangladesh in 1981. She received her B.S. degree in Mechanical Engineering from Bangladesh University of Engineering and Technology in 2005. After graduation, Mousumi joined as a Lecturer in Department of Mechanical Engineering at Bangladesh University of Engineering and Technology and taught mechanical engineering subjects to undergraduate students from 2005-2008. She obtained her M.S. degree in Mechanical Engineering from Bangladesh University of Engineering and Technology in 2008.

Mousumi started her Ph.D. studies at University of Texas at Arlington (UTA) in Fall 2008. At UTA, she worked with Dr. Kamesh Subbarao in areas of cooperative guidance laws and control for Unmanned Aerial Vehicles and received her Ph.D. degree in Mechanical Engineering in 2012. Mousumi also worked as a Graduate Teaching Assistant for Dynamics, Continuum Mechanics, Introduction to Manufacturing Engineering, and Mechanical and Structural Properties of Materials from 2008-2012. She was awarded with Amelia Earhart Fellowship (Zonta International) in 2011, University Scholar Award for the year 2010-2011 at UTA. Mousumi is an active member of AIAA, IEEE, IEEE Control System society, Tau Beta Pi, and Phi Kappa Phi.

After Ph.D., Mousumi will join as a Postdoctoral Researcher in Satellite Technology Laboratory, Department of Mechanical and Aerospace Engineering at UTA, and will work with Dr. Ben Harris on the development of chip-scale spacecrafts technology.