GUIDANCE LAWS FOR n-MOBILE AGENTS TO GRIP A MOVING TARGET

By
Pranay Gadiya

Thesis

Submitted in Partial fulfillment of the requirements
for the degree of Master of Science at
The University of Texas at Arlington
December 2021

Arlington, TX

Supervising Committee:

Chakravarthy Animesh; Supervising Professor
Wang, Shuo Linda
Bowling, Alan
Mullins, Baxter

Acknowledgments

Sincere thanks to my thesis advisor professor Dr. Animesh Chakravarthy for providing a defined path and objective needed to complete my master's thesis. His guidance and support helped a long way in completing this thesis. A mention to Abhishek Kashyap (pursuing PhD Aerospace Engineering from University of Texas at Arlington) for bridging the gap between theoretical understanding of EKF and actual representation of EKF in MATLAB code. Support from colleagues Utsav Mudgal for backing up the theoretical concepts towards collision cone and Hridesh Shah for supporting in coding.

Dedication

I dedicate this thesis to my lovely family and friends, those that I received and those whom I found. I want to take this opportunity to thank my lovely parents Mom and Dad for always supporting in my all decisions and always praying for me for my success. A big thank you to all my friends for standing beside me all the time, these two and a half years would not have been the same without them.

List of Figures

ABSTRACT

GUIDANCE LAWS FOR n-MOBILE AGENTS TO GRIP A MOVING TARGET

Pranay Gadiya, M.S.

Aerospace Engineering

The University of Texas at Arlington, 2021

Supervising Professor: **Dr. Animesh Chakravarthy**

In this thesis, we consider the problem of n mobile agents carrying a gripper, with the objective of gripping a moving circular target. The target may move with constant velocity, or even maneuver. We develop guidance laws by which the agents can first cage the target and subsequently grip it. These guidance laws are developed based on a collision cone framework, using which the mobile agents cooperatively steer their velocity vectors in an appropriate fashion to meet their objective. It is assumed that each of the mobile agents are equipped with a lidar sensor, using which they obtain the range and bearing to the target. These measurements are incorporated into an Extended Kalman Filter whose outputs are fed back to the guidance laws. Simulations are performed using kinematic models of the engagement, and they demonstrate that the pursuing agents can successfully grip the target.

Contents

Chapter 1 Introduction

1.1 Background

In a pursuit evasion game, one or more evaders try to avoid being captured and one or more pursuers try to capture those evaders. Here capture is said to be successful when pursuer approach the evader or target and catches or grips it. Examples of single pursuer, single evader, pursuit evasion differential games include the homicidal chauffer game. These are similar to childhood game such as 'Like to Catch Up' where one player tries to catch another person and the goal of the other person is to not get caught by using various skills. Other examples are the game of two cars, the lion and the man, the lady in the lake, games with visibility constraints and so on. A source of inspiration for cooperative pursuit strategies is the fact that some animals form groups as they hunt their prey. Success is achieved if any one of the pursuers manages to come within a given distance of the evader or intercepts the evader and catches it [1]. In the literature, pursuit problems involving multiple pursuers and a single target are considered in, for example, Refs. [17-24], whereas pursuit problems involving multiple pursuers and multiple targets are considered in Refs. [25–27], among others.

1.2 Literature Survey for Collision Cone

Rendezvous cone is cone concept is used to calculate and employ the trajectory for a mobile agent. It is a concept based on the collision cone theory. In [3], the authors have worked on the development of collision cone theory for obstacle avoidance. Researchers in [1] have developed the collision cone theory so that it can be applied to various shaped objects such as circular, elliptical, or arbitrarily shaped objects and have shown conditions for collision avoidance and rendezvous between these objects. The research community is utilizing the concept of shapes and predictive algorithms for collision avoidance with the help of collision cones.

This thesis will use the concept of collision cone to surround and grip a target instead of avoiding it. We can utilize the relative position and velocity of various objects in 2D, and 3D space and provide a guidance law that would create conditions for this to occur. In this thesis we would concern ourselves with the development of the guidance laws that guides the movement and trajectory that n mobile agents with a gripper would take to surround and grip the target.

1.3 Literature Survey for Gripping

Arm mounted unmanned aerial vehicles (UAV) provide more feasible and attractive solution to manipulate objects in remote areas where access to arm mounted ground vehicles is not possible. In [6], an UAV model equipped with gripper is utilized to grab objects from inaccessible locations. The main purpose was that the aerospace robot should be able to grasp and carry a payload from remote locations to desired destination. UAVs equipped with a gripper are designed with capability of lifting and picking fruits from an elevated position [11]. The inspiration for this concept in [11] was obtained from the swarm robots which have a gripper to pick small cubes, cans to even collecting experimental samples in case of space explorations.

Recent advances have opened the door to UAVs that can interact with their environment by manipulating objects. Researchers in [9], seek to investigate and develop the tools that will be necessary to perform manipulation tasks. That research paper presents progress and results toward a design and physical system to emulate mobile manipulation by an unmanned aerial vehicle with dexterous arms and end effectors. Taking the newly designed gripping mechanism into account, a control strategy for autonomous perching with a quadrotor is further proposed, implemented, and validated in [10]. This research paper focuses on the step to engage with the target during perching, latch on to the target after engagement and release the target when de-perching.

Researchers in [12] knew that it is very difficult for the system to perform manipulation tasks due to the complexity of the environment. So, they designed a manipulator together with an UAV which can perform the necessary manipulation tasks which can grip the sample objects in disaster sites where it is too dangerous for a human to enter. In [13], the author presents a UAV-mounted system for grasping corrugated boxes which can be carried to locations that are unreachable by a ground vehicle using a gripper. Here, the proposed gripping system uses two pairs of end effectors which are arranged perpendicular to each other. The mechanism is designed to handle position and orientation errors of the box relative to the gripper. [13]

The gripping concept was also used in [14] to improve automatic harvesting for fresh-eating tomato where a new picking end-effector was designed which mainly consisted of three parts, for fruit holding, stem holding and fruit separating. A gripper was so designed that the holding and separating force parameters were calculated based on the measured result of the fruit material characteristics, so that the holding operation is reliable and nondestructive [14]. In [15] and [16], the authors developed a quadcopter for delivering the medicines in peak traffic hours.

1.4 Literature Survey for Lidar & EKF

Position estimation and tracking of objects has attracted much attention in the robotics community. To achieve this a Lidar sensor has been used, which uses eye-safe laser beams to create a 3D representation of the surveyed environment. A typical lidar sensor emits pulsed light waves into the surrounding environment. These pulses bounce off surrounding objects and return to the sensor. The sensor uses the time it took for each pulse to return to the sensor to calculate the distance it traveled. These pulses are raw data which are extracted for estimating distance and angle to the object. Use of lidar enables a robot to navigate in an environment shared with other robots by obtaining relative positions of the other robots. It relaxes the dependence on network reliability to deliver positioning data. The estimate of the relative robot position with reduced uncertainty is obtained by feeding the lidar system dynamics measurements into a Kalman filter, resulting in an accurate tracking system. [5]

Extended Kalman Filter is introduced to solve the problem of non-linearity in system dynamics. Extended Kalman Filter makes the nonlinear function into a linear function using Taylor Series and employs the linear approximation of a nonlinear function.

1.5 Motivation

The motivation for the class of problems considered in this thesis is the requirement to cage and grip a circular shaped moving object. Toward this end, this paper considers the problem of four mobile agents carrying a gripper on a two-dimensional (2-D) plane pursuing a moving circular object with the objective of eventually surrounding and gripping the object. The actual position of the target is determined using a lidar sensor whose measurements are fed into an Extended Kalman Filter to estimate the dynamic states of the system. We then feedback these states to guidance laws to close the loop.

1.6 Thesis Organization

In this thesis, we will go through the mathematical model of the cooperative guidance laws which will help the mobile agents to approach the target while maintaining the formation, and gripping guidance laws which will guide the specific mobile agents to grip the target, in chapter 2. Chapter 3 discusses the use of lidar sensors and the Extended Kalman filter to determine the estimated dynamic states of the target. Chapter 4 presents simulation results in MATLAB in various scenarios. Finally, Chapter 5 presents the conclusions.

Chapter 2 Cooperative Pursuit and Gripping Guidance Laws

2.1. Collision Cone Concept

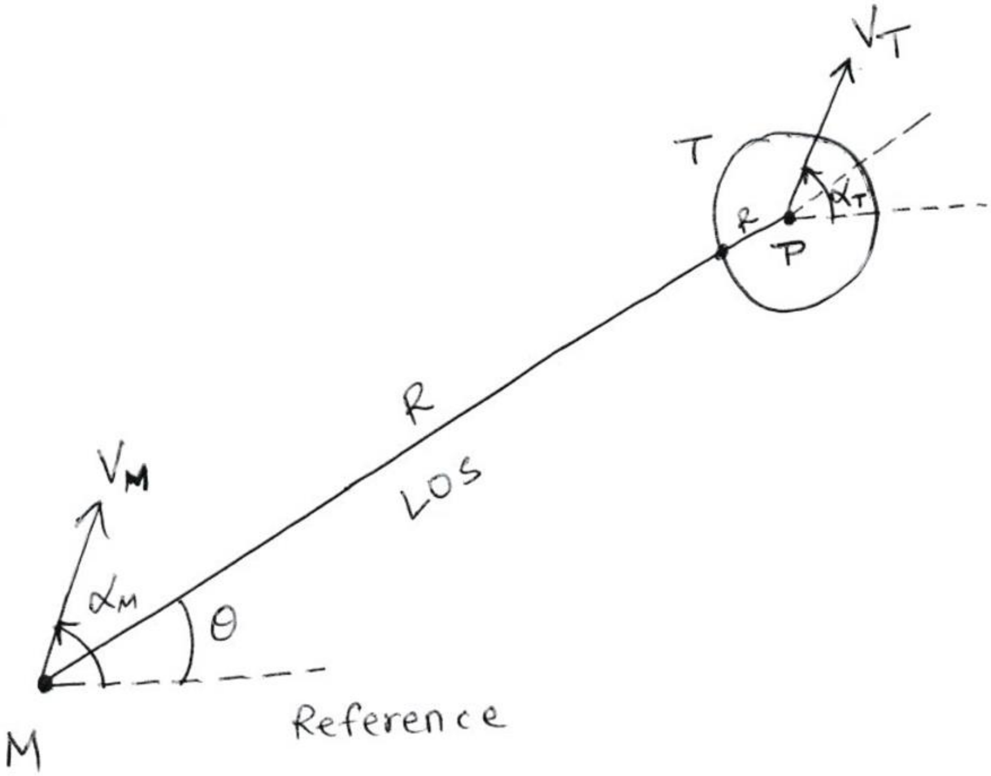

Figure 1. Engagement Geometry between a point and circle

This thesis is based on collision cone concept so we will review some basic results from the collision cones literature from reference [8]. Consider Fig.1 showing an example of an engagement between two objects M and T. Here, the target T is a circle with radius R and center P. V_M and V_T are speeds with which M and T moves respectively whereas α_M and α_T are their respective heading angles. r represents the distance between $M\ and\ P$. θ represents the bearing angle of MP. V_θ is the component of the relative velocity (of T with respect to M) normal to MP and V_r is the component along MP. The engagement geometry can be visualized in the (V_θ, V_r) space. [8]

$$V_r = \dot{r} = V_T \cos(\alpha_T - \theta) - V_M \cos(\alpha_M - \theta) \tag{2.1.1}$$

$$V_\theta = r\dot{\theta} = V_T \sin(\alpha_T - \theta) - V_M \sin(\alpha_M - \theta) \tag{2.1.2}$$

When the M and T both move with constant velocities, the state equations governing the engagement are: [8]

$$\dot{r} = V_r \tag{2.1.3}$$

$$\dot{\theta} = \frac{V_\theta}{r} \tag{2.1.4}$$

$$\dot{V}_\theta = \frac{-V_\theta V_r}{r} \tag{2.1.5}$$

$$\dot{V}_r = \frac{V_\theta^2}{r} \tag{2.1.6}$$

As per reference [1], the miss distance which is the predicted distance between the point M and the center P is given by

$$r_m^2 = r_0 \sqrt{\frac{V_{\theta 0}^2}{V_{\theta 0}^2 + V_{r0}^2}} \tag{2.1.7}$$

The miss time which is the time at which the two objects are at their point of closest approach is given from reference [1].

$$t_m = \frac{-rV_r}{V_\theta^2 + V_r^2} \tag{2.1.8}$$

When the radius of the target is R, then the capture condition with non-zero miss distance is said to be occur when the miss distance r_m is less than R. We define a miss-distance function y as follows: [8]

$$y = \frac{r^2 V_\theta^2}{V_\theta^2 + V_r^2} - R^2 \tag{2.1.9}$$

According to Ref. [1],[8]; the conditions $y < 0\ and\ V_r < 0$ are both necessary and sufficient conditions for one object to intercept the other object if both objects continue to move with constant velocities for all future time. This is represented by a cone in the relative velocity space as shown in Fig.2 which depicts that if the instantaneous value of (V_θ, V_r) lie inside this cone, then the two objects are on a collision course with one another. Whereas if both objects move with varying velocities for all future time, then the condition $y < 0\ and\ V_r \leq 0$ is a sufficient condition for interception.

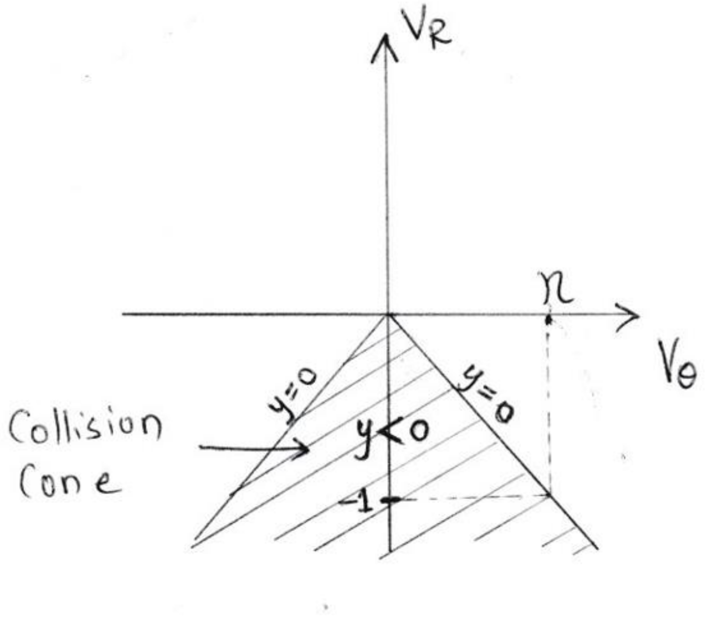

Figure 2. Representation of the collision cone in a relative velocity space

2.2. Caging

Consider $n = 4$ mobile agents $A_1, \dots, A_4$ carrying a gripper with an objective of pursuing and gripping a circular moving target. The mobile agents and the target are all moving on a plane. Here, circular target of radius R may move with constant velocity (i.e., non-maneuvering) or may even maneuver where the acceleration vector acts normal to velocity vector. A scenario comprising four mobile agents performing a cooperative pursuit for caging and gripping the target is illustrated in Fig.3.

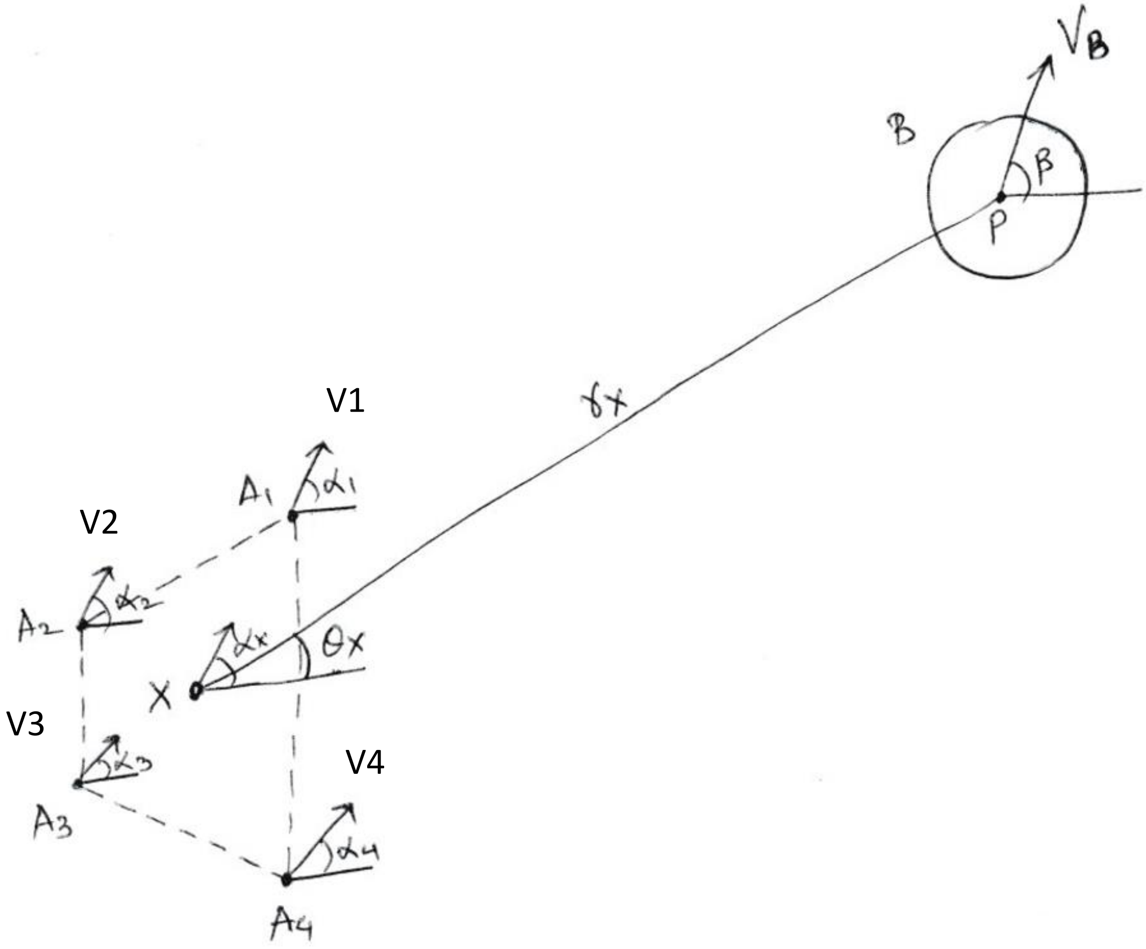

Figure 3. Schematic of 4 mobile agents pursuing a target

Here $V_1, \dots, V_n$ represent velocity vectors and $r_1, \dots, r_n$ represent their respective position vectors of the pursuing mobile agents $A_1, \dots, A_n$. The angles of the associated velocity vectors (with respect to a reference line) are $\alpha_1, \dots, \alpha_{n,}$ respectively. V_B represents the velocity vector of the circular target and β denote the angle made by $V_{\boldsymbol{B}}$ respect to the horizontal. X is a virtual point that lies in the convex hull of $A_1, \dots, A_n$ defined by,

$$X = \sum_{i=1}^{n} \mathrm{\lambda_i r_i}, \quad \sum_{i=1}^{n} \mathrm{\lambda_i} = 1, \quad \lambda_i > 0, \;\; i = 1, \dots, n \tag{2.2.1.1}$$

An algorithm to compute $(\lambda_1, \dots, \lambda_n)$ is discussed in ref. [1]. When the chosen point X lies at the centroid of the convex hull, we choose $\lambda_1 \; = \; \dots \; = \; \lambda_n \; = \; (1/n) \; = \; (1/4) \; = \; 0.25$. Then the chosen $(\lambda_1, \dots, \lambda_n)$ combination is used to compute magnitude of the velocity vector V_X of point X as follows from ref. [1]:

$$V_X = \sqrt{(\lambda_1 V_1 \cos\alpha_1 + \ldots + \lambda_n V_n \cos\alpha_n)^2 + (\lambda_1 V_1 \sin\alpha_1 + \cdots + \lambda_n V_n \sin\alpha_n)^2} \tag{2.2.1.2}$$

The angle made by V_X with respect to the reference line is denoted by α_X and is given by

$$\alpha_X = tan^{-1}\left(\frac{\lambda_1 V_1 \sin\alpha_1 + \cdots + \lambda_n V_n \sin\alpha_n}{\lambda_1 V_1 \cos\alpha_1 + \cdots + \lambda_n V_n \cos\alpha_n}\right) \tag{2.2.1.3}$$

Defining $r_X \; = \; XP$, and θ_X as the angular bearing of the line XP, the relative velocity components of B with respect to X are given by

$$V_{\theta,X} = V_B \sin(\beta - \theta x) - V_X \sin(\alpha_X - \theta x) \tag{2.2.1.4}$$

$$V_{r,X} = V_B \cos(\beta - \theta x) - V_X \cos(\alpha_X - \theta x) \tag{2.2.1.5}$$

Let $a_1, \dots, a_n$ represent the magnitudes of the applied accelerations of $A_1, \dots, A_n$ and these are applied at angles $\delta_1, \dots, \delta_n$, respectively, with these angles measured with respect to a reference line. Here we explicitly account the target’s lateral acceleration a_B.

We assume that the target maneuvers such that its acceleration vector acts normal to its velocity vector; that is target may change its direction but otherwise moves with constant speed. When B has an acceleration of magnitude a_B, it maneuvers and when $a_B = 0$, target doesn't maneuver stating that target moves with constant velocity. The nonlinear state equations governing the kinematics between the point X and center P of the circle, represented in a polar coordinate frame, are as follows: [1]

$$\dot{r_X} = V_{rx} \tag{2.2.1.6}$$

$$\dot{\theta_X} = \frac{V_{\theta x}}{r_X} \tag{2.2.1.7}$$

$$\dot{V}_{\theta,X} = -\frac{V_{\theta X}V_{rX}}{r_X} - \dot{V}_X \sin(\alpha_X - \theta_X) - \dot{\alpha}_X V_X \cos(\alpha_X - \theta_X) + a_B \cos(\beta - \theta) \tag{2.2.1.8}$$

$$\dot{V}_{r,X} = \frac{V_{\theta X}^2}{r_X} - \dot{V}_X \cos(\alpha_X - \theta_X) + \dot{\alpha}_X V_X \sin(\alpha_X - \theta_X) - a_B \sin(\beta - \theta) \tag{2.2.1.9}$$

$$\dot{\beta} = \frac{a_B}{V_B} \tag{2.2.1.10}$$

$$\begin{aligned} \dot{V_X} = \frac{1}{V_X} [& (-\lambda_X \sin\alpha_1 + \lambda_Y \cos\alpha_1)\lambda_1 V_1 \dot{\alpha}_1 + \cdots \\ & + (-\lambda_X \sin\alpha_n + \lambda_Y \cos\alpha_n)\lambda_n V_n \dot{\alpha}_n \\ & + (\lambda_X \cos\alpha_1 + \lambda_Y \sin\alpha_1)\lambda_1 \dot{V}_1 + \cdots \\ & + (\lambda_X \cos\alpha_n + \lambda_Y \sin\alpha_n)\lambda_n \dot{V}_n] \end{aligned} \tag{2.2.1.11}$$

$$\begin{aligned} \dot{\alpha}_X = \frac{1}{\lambda_X^2 + \lambda_Y^2} [& (\lambda_X \cos\alpha_1 + \lambda_Y \sin\alpha_1)\lambda_1 V_1 \dot{\alpha}_1 + \cdots \\ & + (\lambda_X \cos\alpha_n + \lambda_Y \sin\alpha_n)\lambda_n V_n \dot{\alpha}_n + (\lambda_X \sin\alpha_1 - \lambda_Y \cos\alpha_1)\lambda_1 \dot{V}_1 \\ & + \cdots + (\lambda_X \sin\alpha_n - \lambda_Y \cos\alpha_n)\lambda_n \dot{V}_n] \end{aligned} \tag{2.2.1.12}$$

$$\dot{\alpha}_i = \frac{a_1 \sin(\delta_i - \alpha_i)}{V_i}, i = 1, \dots, n \tag{2.2.1.13}$$

$$\dot{V}_i = a_1 \cos(\delta_i - \alpha_i), i = 1, \dots, n \tag{2.2.1.14}$$

Note that Eq. (2.2.1.11) and Eq. (2.2.1.12) are obtained by differentiating Eq. (2.2.1.2) and Eq. (2.2.1.3), respectively, with respect to time. In these equations, $\lambda_X \ and \ \lambda_Y$ are defined as

$$\lambda_X = \lambda_1 V_1 \cos\alpha_1 + \cdots + \lambda_n V_n \cos\alpha_n \tag{2.2.1.15}$$

$$\lambda_Y = \lambda_1 V_1 \sin\alpha_1 + \cdots + \lambda_n V_n \sin\alpha_n \tag{2.2.1.16}$$

Equations (2.2.1.13) and (2.2.1.14) represent the lateral and longitudinal acceleration components, respectively, of the i^{th} mobile agent. Therefore, resultant acceleration will be as follows:

$$a_i = \sqrt{alat_i^2 + along_i^2} \tag{2.2.1.17}$$

$$where, alat_i = \dot{\alpha}_i V_i \tag{2.2.1.18}$$

$$along_i = \dot{V}_i \tag{2.2.1.19}$$

Thus, resultant acceleration will be applied at respective angles.

$$\delta_i = \tan^{-1}\left(\frac{alat_i}{along_i}\right) + \alpha_i \tag{2.2.1.20}$$

Equations (2.2.1.6) - (2.2.1.14) thus govern the kinematics of the relative velocity between any point X residing in the convex hull of the pursuing mobile agents $A_1, \dots, A_4$ and the center of the target circle B when B is maneuvering or non-maneuvering. The guidance problem we address in this section is to determine suitable accelerations of the pursuing agents so that they cooperatively steer the velocity vector of X such that X intercepts the target circle.

2.2.1 Cooperative Pursuit & Guidance Laws with Rigid Links

We consider our example where the pursuing mobile agents have connected each other with rigid links which act as a gripper. As depicted in Fig. 4, the solid lines show three rigid links between the four mobile agents i.e., $A_1A_2, A_2A_3\ and\ A_3A_4$. From Fig. 4, dashed line shows the open end of the link which is A_1A_4. In this thesis, we must make sure that the mobile agents need to reorient the formation appropriately so that the formation approaches the target from the direction of open ends of the rigid links. Thus, we need to calculate the required accelerations which are to be provided to every agent that will enable them to move towards the target while simultaneously rotating and maintaining formation to orient $\hat{h}$ which represents the unit normal to the line joining the open end of the links. So, we will follow the theory from reference [1].

Two assumptions have been made on the sizes of the target circle and quadrilateral formed by the mobile agents; they are as follows:

(i) the open edge of the quadrilateral is of length greater than the diameter of the circle, and

(ii) the perimeter of the quadrilateral is greater than the circumference of the circle B.

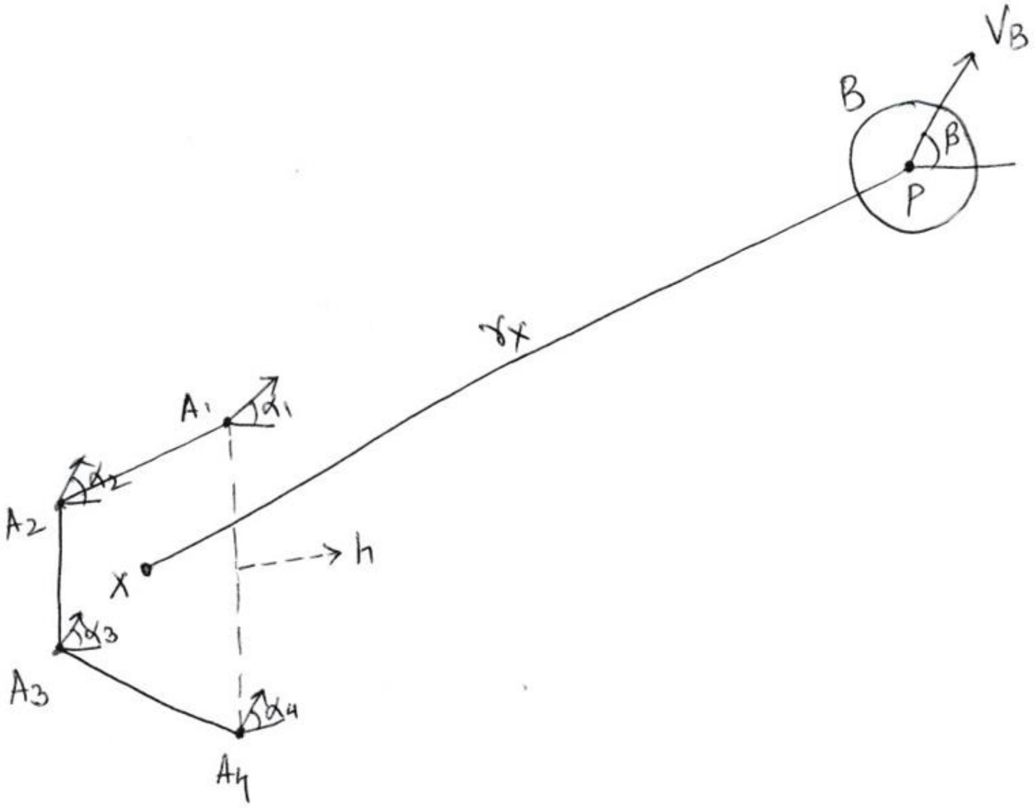

Figure 4. Schematic of 4 mobile agents pursuing a target with rigid links

From reference [1], guidance laws have been derived for governing the longitudinal and lateral accelerations of the pursuing mobile agents with which they can achieve the following objectives:

Objective a) Drive the velocity vector of the centroid of their convex hull into the collision cone to the target B.

Objective b) Rotate their formation appropriately so that they approach the target along the direction of the open end of the chain, and

Objective c) Maintain the distances between the mobile agents to be constant.

Guidance laws for achieving the above objectives are given in the following matrix form $AU = Y$, where U represents the governing lateral and longitudinal accelerations of each agent which is shown below: [1]

$$\begin{bmatrix} \lambda_1 N_{\alpha,1} & \lambda_1 N_{V,1} & \lambda_2 N_{\alpha,2} & \lambda_2 N_{V,2} & \cdots & \cdots & \lambda_{n-1} N_{\alpha,n-1} & \lambda_{n-1} N_{V,n-1} & \lambda_n N_{\alpha,n} & \lambda_n N_{V,n} \\ -V_1 s_1 & c_1 & V_2 s_2 & c_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & 0 & 0 & -V_{n-1} s_{n-1} & c_{n-1} & V_n s_n & -c_n \\ -V_1 c_1 & -s_1 & 0 & 0 & 0 & 0 & 0 & 0 & V_n c_n & s_n \end{bmatrix} \begin{bmatrix} u_{\alpha,1} \\ u_{V,1} \\ u_{\alpha,2} \\ u_{V,2} \\ . \\ . \\ . \\ u_{\alpha,n-1} \\ u_{V,n-1} \\ u_{\alpha,n} \\ u_{V,n} \end{bmatrix}$$

$$= \begin{bmatrix} -K(y_X - w) + a_B \left[\frac{\partial y_X}{\partial V_{\theta,X}} \cos(\beta - \theta_X) - \frac{\partial y_X}{\partial V_{r,X}} \sin(\beta - \theta_X) \right] \\ \frac{V_{\theta,12}^2}{r_{12}} \\ . \\ . \\ . \\ \frac{V_{\theta,n-1,n}^2}{r_{n-1,n}} \\ V_{\theta,1n} \left(\frac{2V_{r,1n}}{r_{1n}} - K_1 \right) - r_{1n} K_2 \left(\theta_{1n} - \theta_{1n,d} \right) + r_{1n} K_1 \dot{\theta}_{1,nd} \end{bmatrix} \qquad (2.2.2.1)$$

where, we have used the shorthand notation that for every pair of adjacent mobile agents i and j, $s_i \equiv \sin(\alpha_i - \theta_{ij})$, $s_j \equiv \sin(\alpha_j - \theta_{ij})$ and $c_i \equiv \cos(\alpha_i - \theta_{ij})$, $c_j \equiv \cos(\alpha_j - \theta_{ij})$.

First row equation of above matrix equation (2.2.2.1) makes sure to follow objective (a) which helps to determine longitudinal and lateral accelerations of the pursuing mobile agents that will drive the velocity vector of X inside the collision cone to the target. This equation has been derived from collision cone function y_X (shown below) which is a function of the predicted actual miss distance. y_X is constant in time when target B is non maneuvering whereas y_X varies with time when B is maneuvering.

$$y_x = \frac{r_X^2 V_{\theta X}^2}{V_{\theta X}^2 + V_{rX}^2} - R^2 \qquad (2.2.2.2)$$

We define an error function as $e = y_X - w$; where w is a reference value and value of w can lie between $-R^2\ and\ 0$. This error function can be used to design controls for systems with non-linear dynamics and they may provide global asymptotic stability properties. Driving error function to zero means forcing predicted miss distance to the reference value w where the decay in error is exponential in nature. So, to achieve this, we differentiate error function along the trajectories of the system defined by Equations (2.2.1.6) - (2.2.1.14), we also see that we can ensure that error function follows the dynamics $\dot{e} = -Ke,$ where $K > 0$, when the control inputs of the pursuing mobile agents satisfy the equation

$$\lambda_1\left(N_{1\alpha}u_{\alpha,1} + N_{1v}u_{V,1}\right) + \cdots + \lambda_n\left(N_{n\alpha}u_{\alpha,n} + N_{nv}u_{V,n}\right) \qquad (2.2.2.3)$$

$$= -K(y_X - w) + a_B\left[\frac{\partial y_X}{\partial V_{\theta,X}}\cos(\beta - \theta_X) - \frac{\partial y_X}{\partial V_{r,X}}\sin(\beta - \theta_X)\right]$$

$where\ N_{i\alpha}\ and\ N_{iv}$ are as follows:

$$N_{i\alpha} = \frac{V_i}{V_X}\left[\frac{\partial y_X}{\partial V_{r,X}}\cos(\alpha_X - \theta_X) + \frac{\partial y_X}{\partial V_{\theta X}}\sin(\alpha_X - \theta_X)\right].[\lambda_X \sin\alpha_i - \lambda_Y \cos\alpha_i] \qquad (2.2.2.4)$$

$$+ \frac{V_X V_i}{\lambda_X^2 + \lambda_Y^2}\left[\frac{\partial y_X}{\partial V_{r,X}}\cos(\alpha_X - \theta_X)\right.$$

$$\left.- \frac{\partial y_X}{\partial V_{\theta,X}}\sin(\alpha_X - \theta_X)\right].[\lambda_X \cos\alpha_i - \lambda_Y \sin\alpha_i]$$

$$N_{iv} = \frac{-1}{V_X}\left[\frac{\partial y_X}{\partial V_{r,X}}\cos(\alpha_X - \theta_X) + \frac{\partial y_X}{\partial V_{\theta,X}}\sin(\alpha_X - \theta_X)\right].[\lambda_X \cos\alpha_i + \lambda_Y \sin\alpha_i] \qquad (2.2.2.5)$$

$$+ \frac{V_X}{\lambda_X^2 + \lambda_Y^2}\left[\frac{\partial y_X}{\partial V_{r,X}}\sin(\alpha_X - \theta_X)\right.$$

$$\left.- \frac{\partial y_X}{\partial V_{\theta,X}}\cos(\alpha_X - \theta_X)\right].[\lambda_X \sin\alpha_i - \lambda_Y \cos\alpha_i]$$

We defined here the control inputs as $u_{\alpha,i}$ and $u_{V,i}$, where $u_{\alpha,i} \equiv \dot{\alpha}_i$ and $u_{V,i} \equiv \dot{V}_i$ represent the acceleration components governing the heading change and speed change, respectively, of mobile agents A_i. Here, $u_{V,i}$ represents the longitudinal acceleration and the lateral acceleration is computed from $u_{V,i}V_i$.

From above matrix Equation (2.2.2.1), last row equation helps to meet objective (b). From reference [1], for satisfying both objective (b) and (c), we look at the kinematics of each of the links A_iA_j. The governing kinematics equations are given as follows and its quantities are schematically shown in Fig.5 from ref. [1].

$$\dot{r_{ij}} = V_{rij} \qquad (2.2.2.6)$$

$$\dot{\theta_{ij}} = \frac{V_{\theta ij}}{r_{ij}} \qquad (2.2.2.7)$$

$$\dot{V}_{\theta ij} = -\frac{V_{\theta ij}V_{rij}}{r_{ij}} - a_i \sin(\delta_i - \theta_{ij}) + a_j \sin(\delta_j - \theta_{ij}) \tag{2.2.2.8}$$

$$\dot{V}_{rij} = \frac{V_{\theta ij}^2}{r_{ij}} - a_i \cos(\delta_i - \theta_{ij}) + a_j \cos(\delta_j - \theta_{ij}) \tag{2.2.2.9}$$

where, δ_i and δ_j represent the directions of the acceleration vectors of a_i and a_j, respectively.

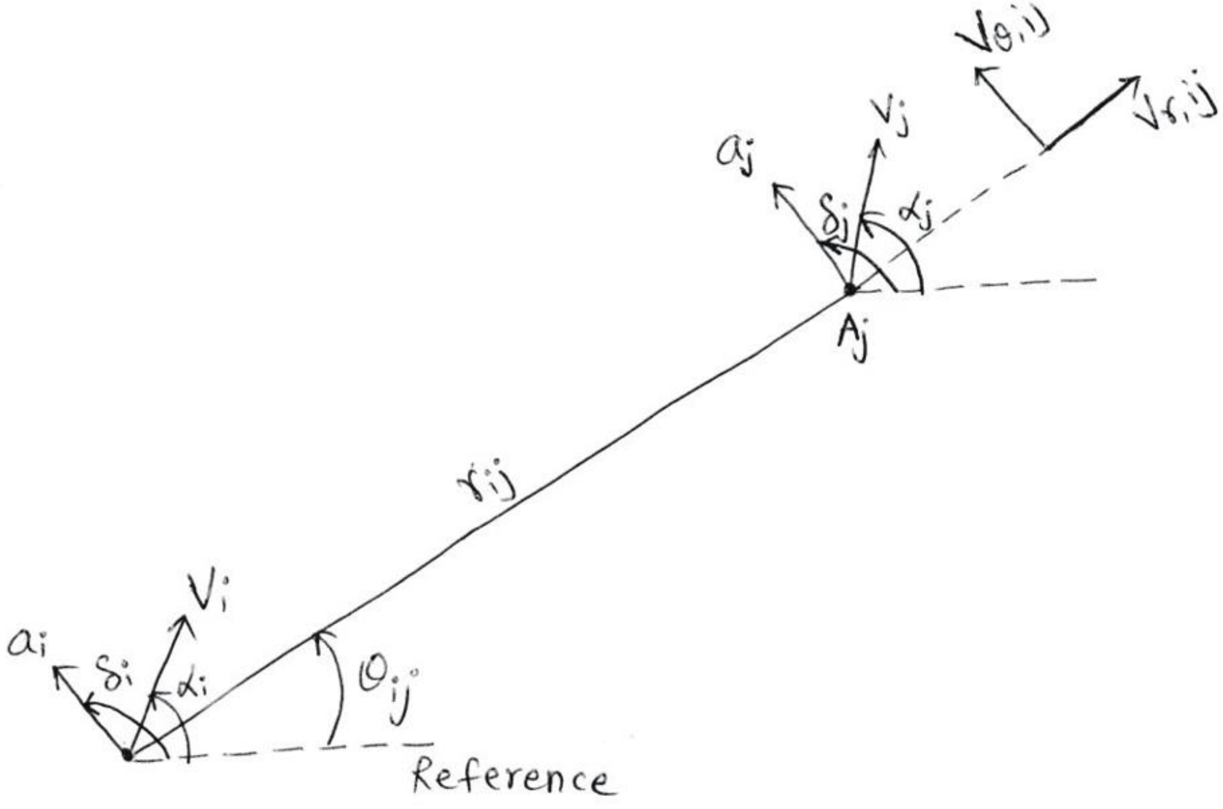

Figure 5. Schematic of rigid link between mobile agent A_i and A_j [1]

From above equations (2.2.2.6) - (2.2.2.9), it is evident that when a_i and a_j are equal then the $V_{r,ij} = 0$, $V_{\theta,ij} = 0$ and r_{ij} is constant. Considering θ_{14} as the angle made by open end A_1A_4 and θ_{14d} as the desired orientation of this line. Here, we have the equation from ref. [1] which is as below for acceleration of A_1 and A_4 such the pursuing mobile agents will be able to surround the target with the links at the appropriate orientation.

$$\begin{aligned} & -u_{\alpha,1}V_1\cos(\alpha_1-\theta_{1n}) \qquad (2.2.2.10) \\ & -u_{V,1}\sin(\alpha_1-\theta_{1n}) \\ & +u_{\alpha,n}V_n\cos(\alpha_n-\theta_{1n}) \\ & +u_{V,n}\sin(\alpha_n-\theta_{1n}) \\ & =V_{\theta,1n}\left(\frac{2V_{r,1n}}{r_{1n}}-K_1\right)-r_{1n}K_2\left(\theta_{1n}-\theta_{1n,d}\right)+r_{1n}K_1\dot{\theta}_{1,nd} \end{aligned}$$

This equation makes sure that pursuing mobile agents cage the target from the open end A_1A_4 with required angle of that open end. This equation gives appropriate accelerations to the open-end agents A_1 and A_4 to follow the expected orientation and formation.

The remaining rows in the matrix equation (2.2.2.1), helps us to meet objective (c) which dictates that the distance between the agents A_1A_2, A_2A_3 and A_3A_4 need to remain constant throughout the engagement. From reference [1], we know that the distance between the agents A_iA_j can be kept constant when $\dot{V}_{rij}=0$ with an assumption that initially $V_{rij}=0$. From Eq. (2.2.2.11), we get following condition which will maintain distance between the agents. This condition formulates the lateral and longitudinal accelerations of every mobile agent which ensures that inter-agent's distances between A_1A_2, A_2A_3 and A_3A_4 are constant.

$$\begin{aligned} & u_{V,i}\cos\left(\alpha_i-\theta_{ij}\right)-u_{\alpha,i}V_i\sin\left(\alpha_i-\theta_{ij}\right)-u_{V,j}\cos\left(\alpha_j-\theta_{ij}\right) \qquad (2.2.2.11) \\ & +u_{\alpha,j}V_j\sin\left(\alpha_j-\theta_{ij}\right)=V_{\theta,ij}^2/r_{ij}\,, \qquad \forall(i,j)\in E \end{aligned}$$

Therefore, equation (2.2.2.1) represents of $2n - 1$ equations with $2n$ unknowns which makes an under-determined system of equations, with there being one more unknown than the number of equations which have nonunique solutions. Solution of this equation (2.2.2.1) results in lateral and longitudinal acceleration of every agent. The general solution of above under-determined equation is obtained by Moore- Penrose pseudoinverse. Here we used 'pinv' command in MATLAB to find pseudoinverse of A as $U = pinv(A) * Y$.

2.3. Gripping

The above caging guidance law ensures that the target is caged in the convex hull of the mobile agents. This research was referred from ref. [1]. This thesis discusses the subsequent gripping of the target along with the above guidance law. Gripping the target is said to have occurred when one of the mobile agents touches the target, that means, r of one of the agents to the target needs to go to zero. Thus, the gripping guidance law is developed and is implemented on the relevant mobile agents during the end of the engagement. In our scenario $A_1 \ and \ A_4$ will be mobile agents which will be used to grip the target as shown in Fig.6. So, the gripping guidance law will be applied only to the A_1 and A_4 mobile agents after specific time.

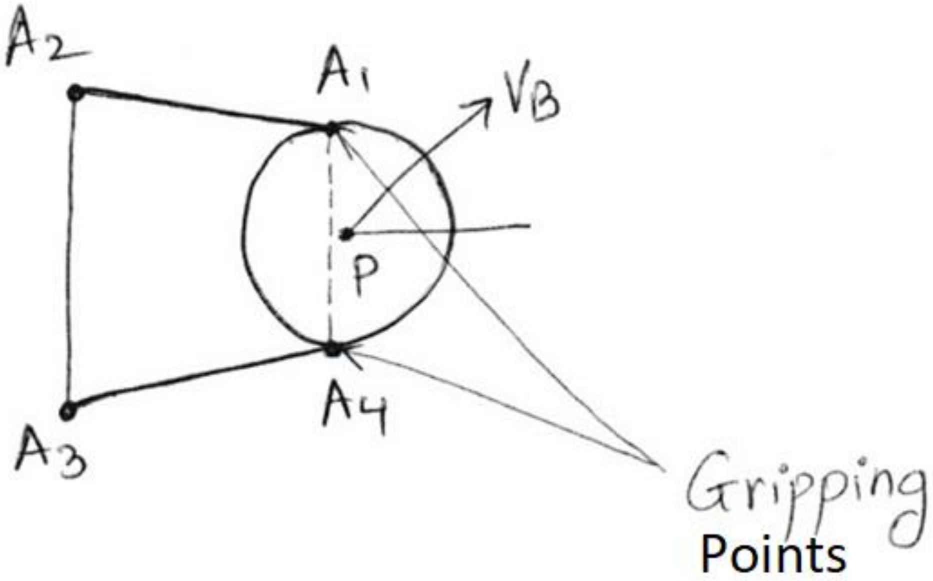

Figure 6. Schematic of 4 mobile agents gripping the target

The as collision cone function between $A_1$to target and A_4 to target with value of w as zero is used, to ensure that the agents $A_1\ and\ A_4$ will approach the target with the requisite acceleration required to graze the circular target's circumference. Applying acceleration to $A_1\ and\ A_4$ will result in altering the formation between the agents. However, the distances A_1A_2 and A_3A_4 should remain constant. Therefore, relative velocity between agents 1 and 2 i.e., V_{r12} and relative velocity between agents 3 and 4 i.e., V_{r34} need to be zero. This means that the distance r_{12} between agents A_1A_2 and distance r_{34} between agents A_3A_4 remains constant. Here control inputs i.e., lateral, and longitudinal accelerations for both mobile agents will be $u_{\alpha,1} = \dot{\alpha}_1$ and $u_{V,1} = \dot{V}_1$, similarly $u_{\alpha,4} = \dot{\alpha}_4$ and $u_{V,4} = \dot{V}_4$

Considering A_1, and its collision function to the target as:

$$y_1 = \frac{r_1^2 V_{\theta 1}^2}{V_{\theta 1}^2 + V_{r1}^2} - R^2 \tag{2.3.1}$$

Since, $V_{r12}\ =\ 0$ i.e., rate of change of distance $\dot{r_{12}} = 0$. This leads to

$$V_1 \cos(\alpha_1 - \theta_{12}) = V_2 \cos(\alpha_2 - \theta_{12}) \tag{2.3.2}$$

Differentiating equation (2.3.2) with respect to $V_1, \alpha_1\ and\ \theta_1$ with $\dot{\theta}_{12} = 0$ and solving it with dynamic inversion of Eq. (2.3.1), we find out the lateral and longitudinal acceleration for A_1 which must be applied to grip the target. Here control inputs will be $u_{\alpha,1} = \dot{\alpha}_1$ and $u_{V,1} = \dot{V}_1$. Upon solving the equation, the result is as follows.

$$\dot{\alpha}_1 = [\ (\ \dot{V}_2 u_1 \cos(\alpha_2 - \theta_{12})) - (\dot{\alpha}_2 V_2 u_1 \sin(\alpha_{2-}\theta_{12})) + \frac{(k_1 y_1 \cos(\alpha_1 - \theta_{12}))}{c_1} + \frac{(V_2 u_1 \sin(\alpha_2 - \theta_{12}) V_{\theta 12})}{r_{12}} - \frac{(V_1 u_1 \sin(\alpha_1 - \theta_{12}) V_{\theta 12})}{r_{12}}] \frac{1}{D_1 V_1} \tag{2.3.3}$$

$$\dot{V}_1 = \frac{\left(\frac{-k_1 y_1}{c_1}\right) + \dot{\alpha}_1 V_1 w_1}{u_1} \tag{2.3.4}$$

$$where, c_1 = \frac{2 V_{\theta 1} V_{r1} r_1^2}{\left(V_{r1}^2 + V_{\theta 1}^2\right)^2} \tag{2.3.5}$$

$$w_1 = V_{r1} \cos(\alpha_1 - \theta_1) + V_{\theta 1} \sin(\alpha_1 - \theta_1) \tag{2.3.6}$$

$$u_1 = V_{\theta 1} \cos(\alpha_1 - \theta_1) - V_{r1} \sin(\alpha_1 - \theta_1) \tag{2.3.7}$$

$$D_1 = w_1 \cos(\alpha_1 - \theta_{12}) - u_1 \sin(\alpha_1 - \theta_{12}) \tag{2.3.8}$$

Similarly, we follow the above procedure for agent A_4, and get the following lateral and longitudinal acceleration equations.

$$\dot{\alpha}_4 = [\ (\ \dot{V}_3 u_4 \cos(\alpha_3 - \theta_{34})) - (\dot{\alpha}_3 V_3 u_4 \sin(\alpha_{3-}\theta_{34})) + \frac{(k_4 y_4 \cos(\alpha_4 - \theta_{34}))}{c_4} + \frac{(V_3 u_4 \sin(\alpha_3 - \theta_{34}) V_{\theta 34})}{r_{34}} - \frac{(V_4 u_4 \sin(\alpha_4 - \theta_{34}) V_{\theta 34})}{r_{34}}] \frac{1}{D_4 V_4} \tag{2.3.9}$$

$$\dot{V}_4 = \frac{\left(\frac{-k_4 y_4}{c_4}\right) + \dot{\alpha}_4 V_4 w_4}{u_4} \tag{2.3.10}$$

$$where, c_4 = \frac{2 V_{\theta 4} V_{r4} r_4^2}{\left(V_{r4}^2 + V_{\theta 4}^2\right)^2} \tag{2.3.11}$$

$$w_4 = V_{r4}\cos(\alpha_4 - \theta_4) + V_{\theta 4}\sin(\alpha_4 - \theta_4) \qquad (2.3.12)$$

$$u_4 = V_{\theta 4}\cos(\alpha_4 - \theta_4) - V_{r4}\sin(\alpha_4 - \theta_4) \qquad (2.3.13)$$

$$D_4 = w_4\cos(\alpha_4 - \theta_{34}) - u_4\sin(\alpha_4 - \theta_{34}) \qquad (2.3.14)$$

Initially, all agents follow the caging guidance law and use the acceleration from (2.2.2.1) required to approach the target. After an interval of time, the gripping guidance law from (2.3.3) – (2.3.14) is applied to the agents 1 and 4. Now, the agents 1 and 4 follow the gripping guidance law required to touch the target and agents 2 and 3 follow the caging guidance law required to cage the target. Thus, the simultaneous application of caging and gripping guidance laws results in agent 1 and 2 successfully touching the target while the formation between the distances A_2A_3, A_1A_2 and A_3A_4 remain unchanged.

Chapter 3 Lidar Measurements & Extended Kalman Filter

3.1. Lidar Measurements

3.1.1 Creating a Scenario

The guidance Laws of the previous chapter need to know the relative distances and velocities of the agents to the target. So, to know the target location we introduce a Lidar sensor on every mobile agent. A LIDAR is Light Detection and Ranging sensing method used for measuring the exact distance to an object. It throws light on the object and uses the point clouds to obtain information of location of that object.

Creation of a real-life scenario in MATLAB, is done using ‘uavScenario’ function where one can introduce a platform, surroundings, and sensors. The ‘UAVplatform’ is used to create each mobile agent and a target. One can define their trajectories from the above guidance laws. Lidar is placed below each moving mobile agent using ‘uavSensor’ function in MATLAB. Here one can define lidar specifications, the platform on which the sensor is to be placed and the mounting location of that sensor. In Lidar specification, one can adjust the azimuth and elevation resolution, azimuth and elevation limits and maximum range of Lidar up to which it can observe the target. Scatter plot gives Lidar information in form of point clouds with noise or without noise. To introduce noise in point clouds, use ‘randn’ function. Since the above guidance laws are based on virtual centroid of a quadrilateral so the aim in this thesis is to track the relative location of the target with respect to the virtual centroid using a Lidar on every mobile agent. For this, use ‘pctransform’ function in MATLAB which helps to transform the 3D point clouds which we receive from the Lidar placed below each of mobile agents.

There is a translational transformation of point clouds from actual position to virtual centroid position. The ‘pcmerge’ command is used to merge all the point clouds from each lidar on each mobile agent. By doing this, one can see the target and its point clouds from the virtual centroid of all mobile agents.

3.1.2 Calculating range and bearing of the target

After translating and merging the point clouds to the virtual centroid, each individual point of the Lidar point cloud is represented in polar coordinates $S = (d_i, \alpha_i)$, where d_i is the distance of the data point from the origin of the virtual centroid coordinate system and α_i is the relative angle as shown in figure below. Fig_ shows the point clouds created by lidar on the detecting object and helps to explain the polar coordinates S as explained above.

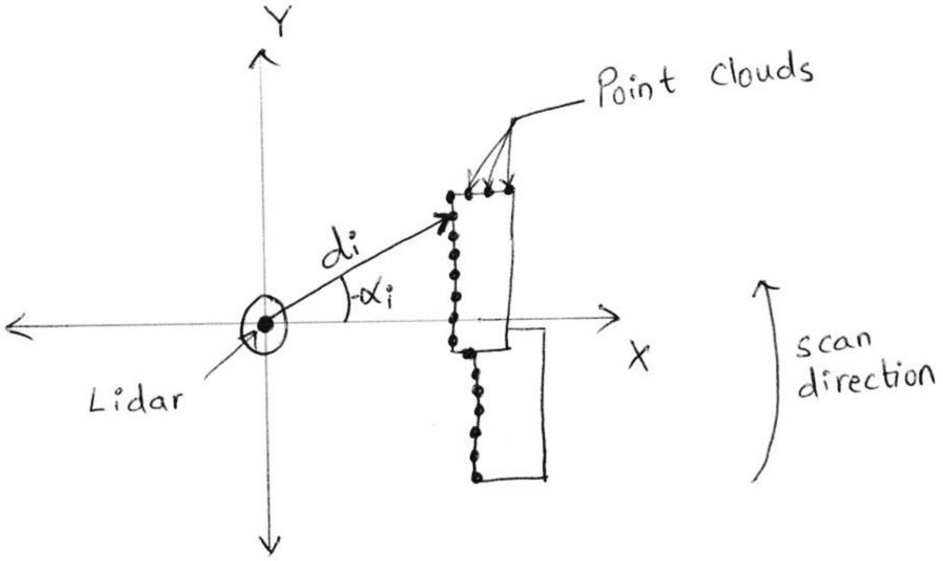

Figure 7. Illustration of the Lidar measurements

In this section, we refer to reference [5] to estimate the position of the target from the generated point clouds. To locate the target, minimization of an objective function F over a space of two parameters, namely the coordinates x_c , y_c of the center of the circle, expressed as a relative polar coordinate in respect to the virtual center helps

$$F(d,\alpha) = \sum_{i=1}^{N^k} \left(\sqrt{(\varepsilon_x d_i \cos(\alpha_i) - x_C)^2 + \left(\varepsilon_y d_i \sin(\alpha_i) - y_C\right)^2} - R \right)^2 \qquad (3.1.2.1)$$

Where N^k is the number of data points present in current cluster K. R is the radius of circular target. $\varepsilon_x \ and\ \varepsilon_y$ are the signs reflecting the scan angle convention of the frontal or the rear lidar.

The above algorithm is the circle fitting algorithm where the clustered point clouds, having approximately the size of the target are supplied into a least square optimization algorithm extended with a modification of the Levenberg-Marquardt algorithm [5]. We assume that the lidar doesn't see the rest of its neighborhood and the point clouds are generated only based on the moving target. The result of the circle fitting algorithm shown in Fig.7 illustrates the situation where the lidar detects the target present in the experimental area.

The above circle fitting algorithm gives a set of coordinates $(x_c\ , y_c)$ of the target in the local coordinate frame of reference. To find the precise location of the target on the map we used the following equation from [5]:

$$\begin{bmatrix} x_C^g \\ y_C^g \end{bmatrix} = \begin{bmatrix} \cos(\alpha_0) & -\sin(\alpha_0) \\ \sin(\alpha_0) & \cos(\alpha_0) \end{bmatrix} \begin{bmatrix} x_c \\ y_c \end{bmatrix} + \begin{bmatrix} x_0^g \\ y_0^g \end{bmatrix} \tag{3.1.2.2}$$

Where the superscript g refers to the global coordinate frame and $[x_0, y_0, \alpha_0]$ is the pose of the lidar (virtual center) in the global frame as shown in Fig.8. Therefore, using Eq. (3.1.2.1) and Eq. (3.1.2.2), range (r) is x_c and bearing angle (θ) is also found. In this way, two dynamic states out of four are achieved at each instant and these will be used for finding the remaining two states i.e., V_r and V_θ in the next section.

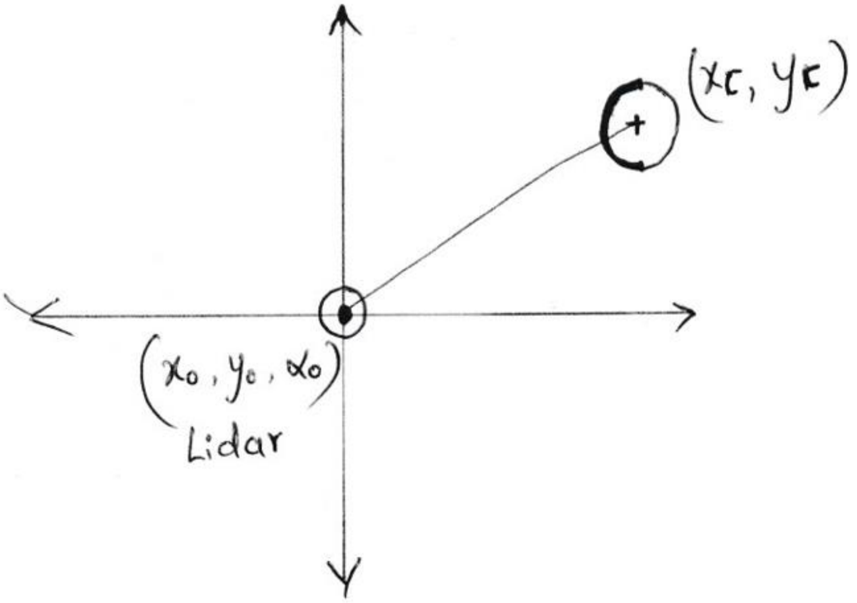

Figure 8. Result of circle fitting algorithm

3.2. Introduction of Extended Kalman Filter

From the above lidar measurements it was possible to determine the range (r) and bearing angle (θ) of the target with respect to the virtual center (X) of four mobile agents. We next need to find V_r and V_θ which will complete states of the agents which will be then used for feedback into the guidance laws. Thus, V_r and V_θ will be the estimated states, while $r\ and\ \theta$ will be the measured states from the Lidars. As the state equations are nonlinear, we introduce an Extended Kalman Filter (EKF). EKF is the nonlinear version of the Kalman filter which linearizes about an estimate of the current mean state trajectory.

In the Extended Kalman filter, the state transition and observation models don't need to be linear functions of the state. The physical system is represented as a Continuous-time model, while discrete-time measurements are taken for state estimation. Therefore, the system model is given by [7]

$$\dot{x}(t) = f(x(t), u(t), t) + G(t)w(t), w(t) \sim N\big(0, Q(t)\big) \qquad (3.2.1)$$

$$\tilde{y}_k = \mathrm{h}(x_k) + v_k, v_k \sim N(0, R_k) \qquad (3.2.2)$$

Where, $x(t), u(t), \tilde{y}_k$ are the state vector, control inputs and the measured values $(r\ and\ \theta)$ from lidar respectively. $w(t)$ and $v(t)$ are the process and observation noise which are both assumed to be zero mean multivariate Gaussian noise with covariance $Q(t)$ and R_k respectively. In the given problem statement, the equation $h(x_k)$ is linear and Eq. (3.2.2) can be rewritten as

$$y_k = C \begin{bmatrix} r \\ \theta \\ V_r \\ V_\theta \end{bmatrix} + v_k \tag{3.2.3}$$

$$where, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{3.2.4}$$

The function f can be used to compute the predicted state from the previous estimate and similarly the function h can be used to compute the predicted measurement from the predicted state. However, $f\ and\ h$ cannot be applied to the covariance directly. Instead, a matrix of partial derivatives (the Jacobian) is computed. At each time step, the Jacobian is evaluated with the current predicted states. These matrices are then used in the Kalman filter equations. This process essentially linearizes the non-linear function around the current estimate. The Continuous Discrete Extended Kalman Filter (CDEKF) is implemented as follows from ref. [7]

Initialize $$\hat{x}(t_0) = \hat{x}_0 \tag{3.2.5}$$

$$P_0 = E\{\tilde{x}(t_0)\tilde{x}^T(t_0)\} \tag{3.2.6}$$

Here $\hat{x}_0\ and\ P_0$ are initial guesses where $\hat{x}$ is a column matrix with initial guesses of all four states whereas P_0 is taken as an identity matrix. We consider $\hat{x}_0 = \hat{x}_k^-$ and $P_0 = P_k^-$ for the first step. In the next step we calculate the Kalman gain K_k as follows

Gain $$K_k = P_k^- C^T (C P_k^- C^T + R_k)^{-1} \text{ where } K_k \text{ is Kalman gain} \tag{3.1.2.2}$$

The initial guesses for the state and the covariance are updated using the K_k and the measured value y_k

Update $$\hat{x}_k^+ = \hat{x}_k^- + K_k[\tilde{y}_k - \hat{x}_k^-] \qquad (3.1.2.3)$$

$$P_k^+ = [I - K_k C]P_k^- \qquad (3.1.2.4)$$

Finally, the updated state values are propagated in time using the state dynamical equations and updated covariance matrix is propagated using the Jacobian.

Propagation $$\dot{\hat{x}}(t) = f(\hat{x}(t), u(t), t) \qquad (3.1.2.5)$$

$$\dot{P_k}^- = F(t)P_k^- + P_k^- F^T(t) + GQG^T \; where \; F(t) = \left.\frac{\partial f}{\partial x}\right|_{\hat{x}(t),u(t)} \qquad (3.1.2.6)$$

where $F(t)$ is a linearized matrix form of state dynamics which is represented as follows

$$F(t) = \begin{bmatrix} 0 & 0 & 0 & 1 \\ -\frac{V_\vartheta}{r^2} & 0 & \frac{1}{r} & 0 \\ \frac{V_\theta V_r}{r^2} & \dot{V}_X \cos(\alpha - \theta) + \dot{\alpha}_{\mathrm{x}} \sin(\alpha - \theta) + ab \sin(\beta - \theta) & -\frac{V_r}{r} & -\frac{V_\theta}{r} \\ \frac{-V_\theta^2}{r^2} & -\dot{V}_X \sin(\alpha - \theta) + \dot{\alpha}_{\mathrm{x}} \cos(\alpha - \theta) + ab \cos(\beta - \theta) & \frac{2V_\theta}{r} & 0 \end{bmatrix} \qquad (3.1.2.7)$$

Here ab is magnitude of acceleration of target. The estimated states $\hat{x}_k^+$ are as used as feedback to compute the control inputs for the current time t and the steps of the CDEKF are repeated in time to generate the estimated states and the control inputs till the simulation is complete.

Chapter 4 Simulation Results

The simulation is divided into three stages. In the first stage, the mobile agents follow the developed guidance laws in an ideal environment where an assumption is made that the system has full state measurements, which are perfect. The trajectory of the mobile agents collected in the first stage is considered as the ideal trajectory the agents are required to follow. In the next stage, a lidar sensor is added to each agent which introduces real time sensor noise in the system. The sensor generates the target location which is used to calculate the radial distance and angle between the agent and target. The distance and angle are compared with the true values and will be used in the Extended Kalman Filter for generating estimated states. The estimated states are further compared with the true states obtained by perfect model in the first stage. In the third stage of simulations, the outputs from the Extended Kalman Filter will be fed back into the guidance laws of the agents.

The trajectory of the mobile agents and target in the first stage of the simulation is depicted in Fig. 9. The pursuing mobile agents $A_1, A_2, A_3\ and\ A_4$ are moving with initial speeds of $V_1 = V_2 = V_3 = V_4 = 10m/s$. The target circle B is moving with constant speed of $V_B = 5m/s$. The initial positions of the pursuing agents are $(13.6, 13.6)\ m, (5, 8.6)\ m, (5, -8.6)\ m\ and\ (13.6, -13.6)\ m$. All mobile agents have an initial heading angle of 65° $(\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 65°)$. The target circle B has an initial heading angle of $\beta = 180°$. The point X is taken as the centroid of the quadrilateral formed by pursuing agents, that is, $\lambda 1 = \lambda 2 = \lambda 3 = \lambda 4 = ¼ = 0.25$.

The initial position of the center of the target circle is taken as $(150, 150)\ m$. For these initial conditions, we obtain $r_X\ (0)\ =\ 205.66\ m, V_{r,X}\ (0) =\ -12.922\frac{m}{s}, V_{\theta,X}\ (0)\ =\ 30.2980\ m/s$. It is seen that $V_{r,X}\ (0)\ <\ 0$, which means that the initial engagement geometry is such that the pursuing agents with grippers are moving into the target circle. The radius of the target circle is taken as $R\ =\ 5\ m$. Here the target maneuvers with a lateral acceleration of $0.5\ m/s^2$.

In simulation cases, it is assumed that the pursuing mobile agents are carrying a rope line as an open-chain, and the open end of this chain is carried by the grippers A$_1$ and A$_4$. The open end of the chain is depicted with dashed lines in Fig. 1, while the other edges are depicted with solid lines. Given all above initial conditions, it is evident that the pursuing agents, while pursuing the target, also need to simultaneously rotate their formation, to approach and grip the target from the open end of the chain formed by grippers A$_1$ and A$_4$. Here, desired angle of open end of the chain is $\theta_{14}\ =\ -10°$. We choose the reference $w\ =\ -24$.

4.1 Simulation with Perfect State information

It is evident from Fig. 9, the pursuing mobile agents can successfully surround and grip the target while simultaneously rotating their formation. They eventually grip the target, which is said to occur when one of the agents comes in direct contact with the target. It is also clearly seen from same the figure that pursuing agents catch and grip the target from the direction of the dashed line, which forms the open end of the chain. The pursuing mobile agents use the guidance laws obtained by solving Equation (2.2.2.1) and then apply gripping guidance laws obtained by solving Equations (2.3.3) and (2.3.9) law after 6 seconds.

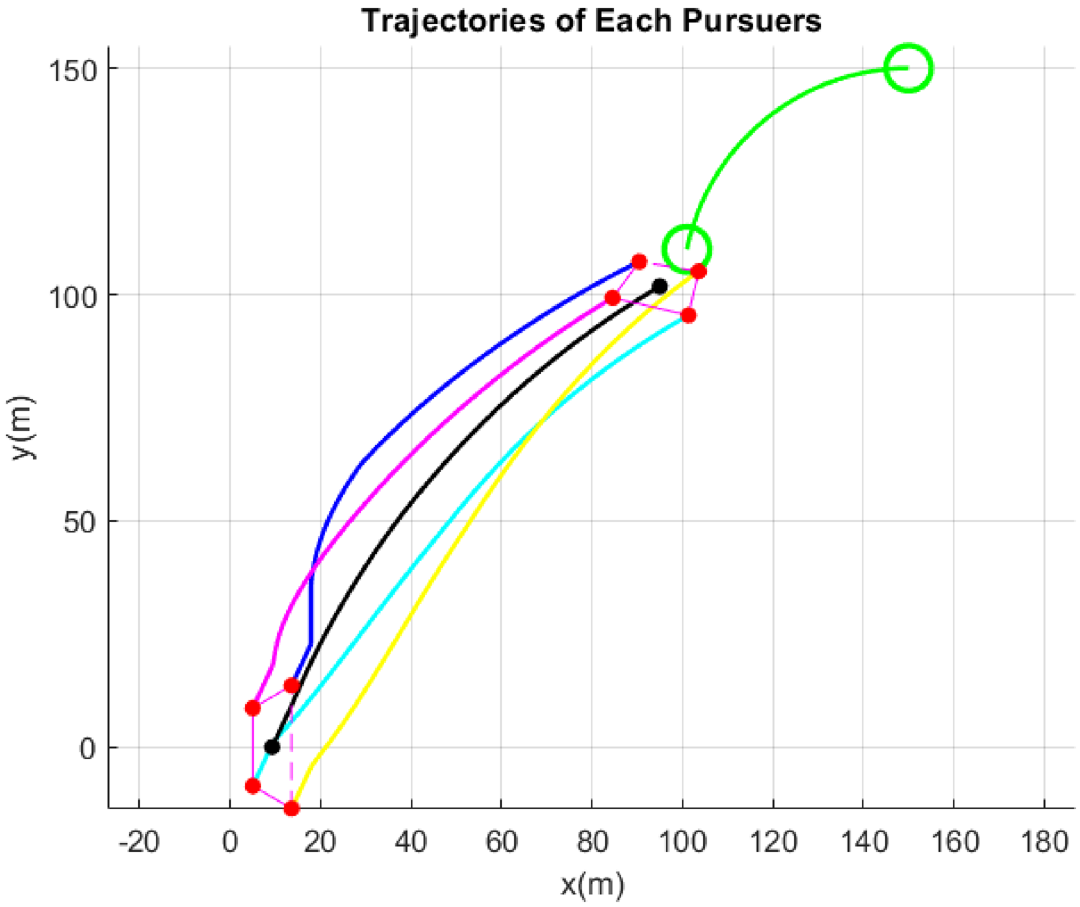

Figure 9. Trajectories of 4 mobile agents and a target with Perfect State information

Detailed plots, showing time histories are given in below figures. Fig. 10 shows the plot of the collision cone function y_X. The initial value of y_X is positive, indicating that the initial engagement is such that the relative velocity vector $(V_{\theta X}, V_{rX})$ lies outside the collision cone. Using the developed guidance laws, the pursuing mobile agents successfully drive y_X to a negative value i.e., to $w = -24$, (which in conjunction with $V_{rX} < 0$), is equivalent to driving the relative velocity vector into the collision cone.

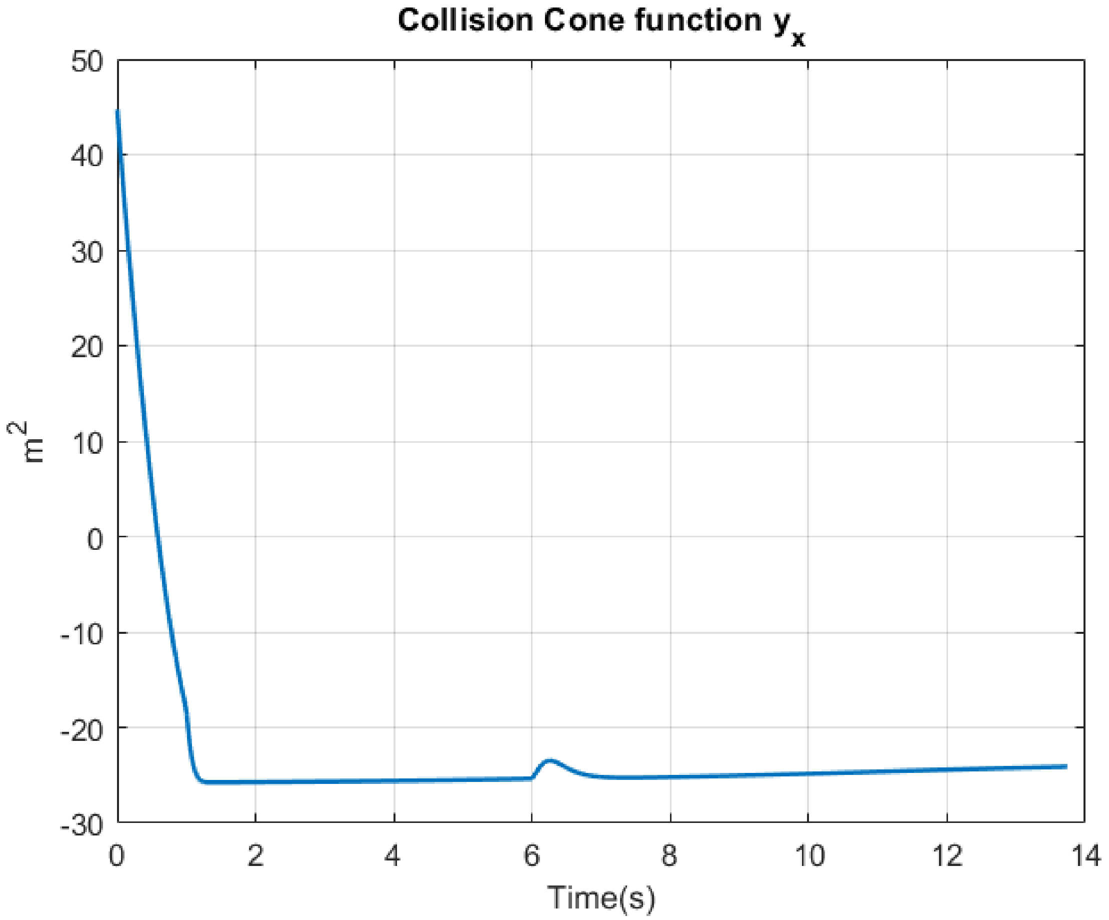

Figure 10. Detailed time history of collision cone function y_X

The time histories of the heading angles of the pursuing agents are shown in Fig. 11, and those of their speeds are in Fig. 12.

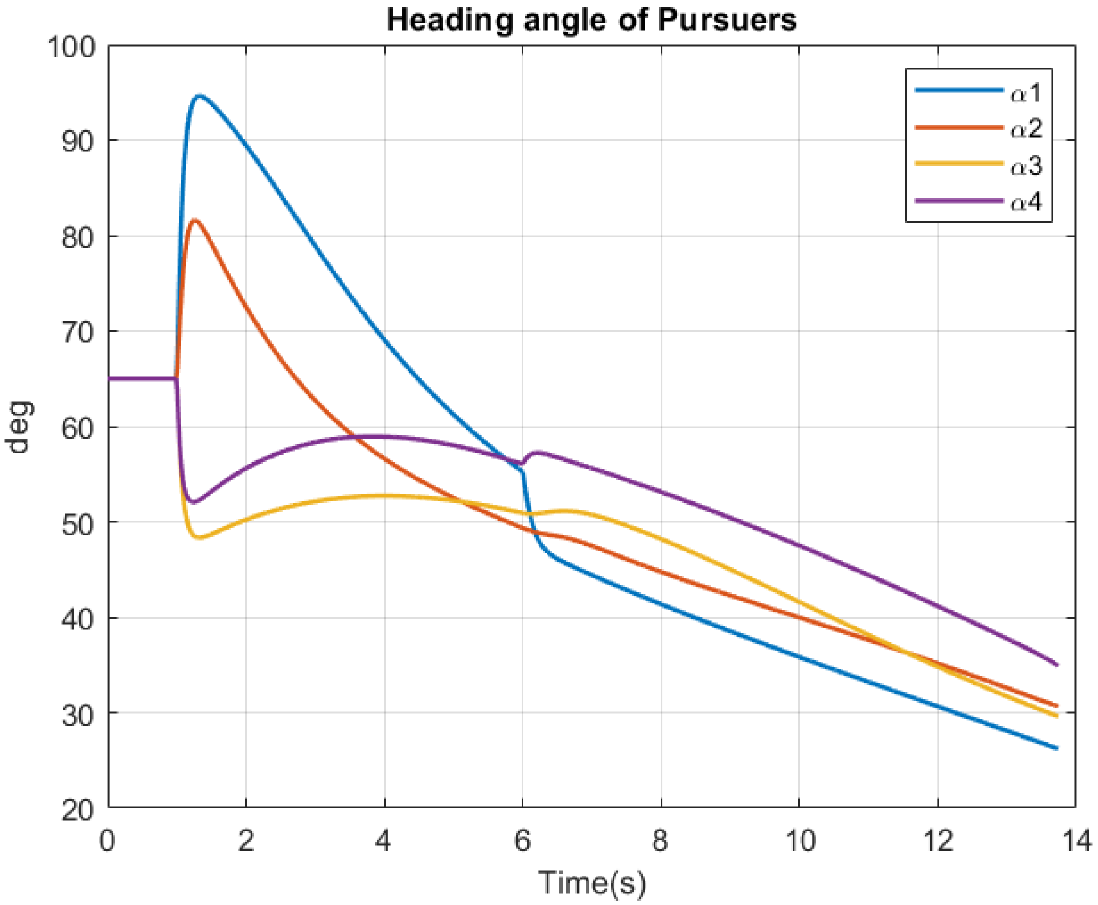

Figure 11. Detailed time history of heading angle of pursuing agents

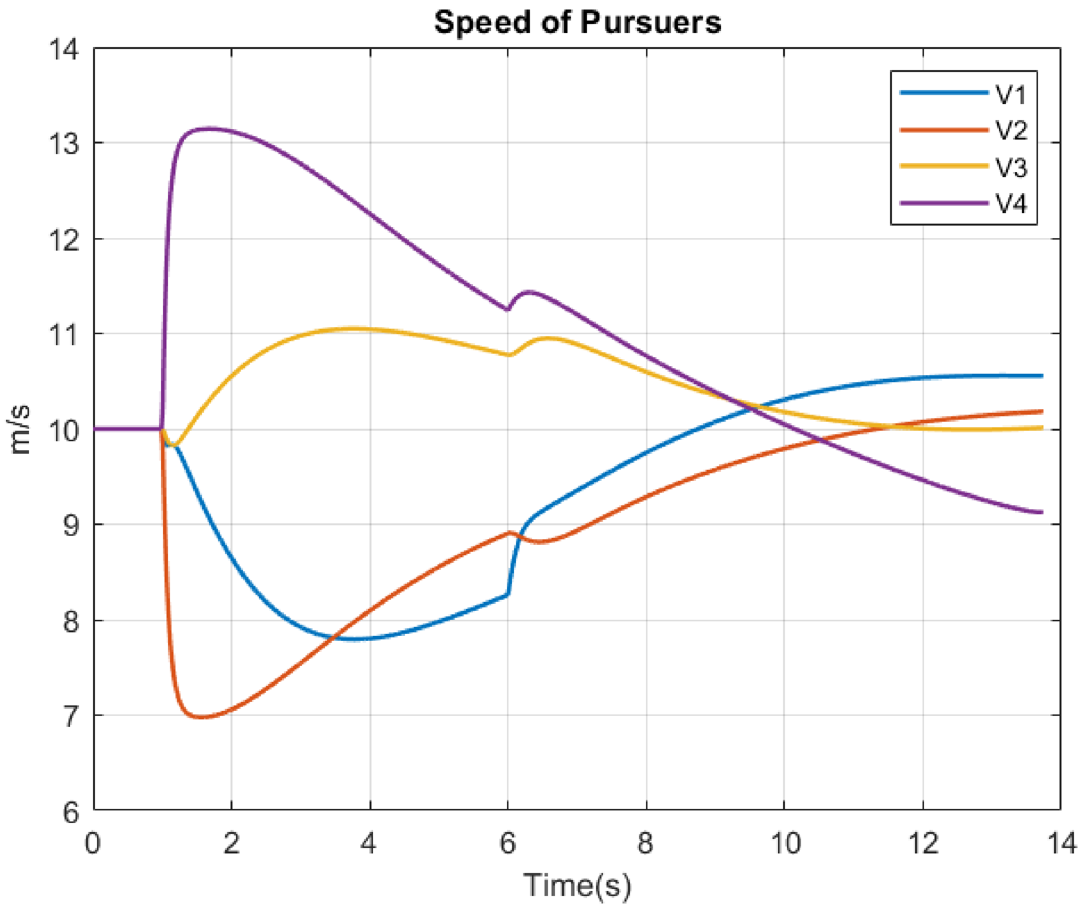

Figure 12. Detailed time history of speed of pursuing agents

The longitudinal and lateral accelerations of the pursuing agents are shown in Fig. 13 and Fig. 14.

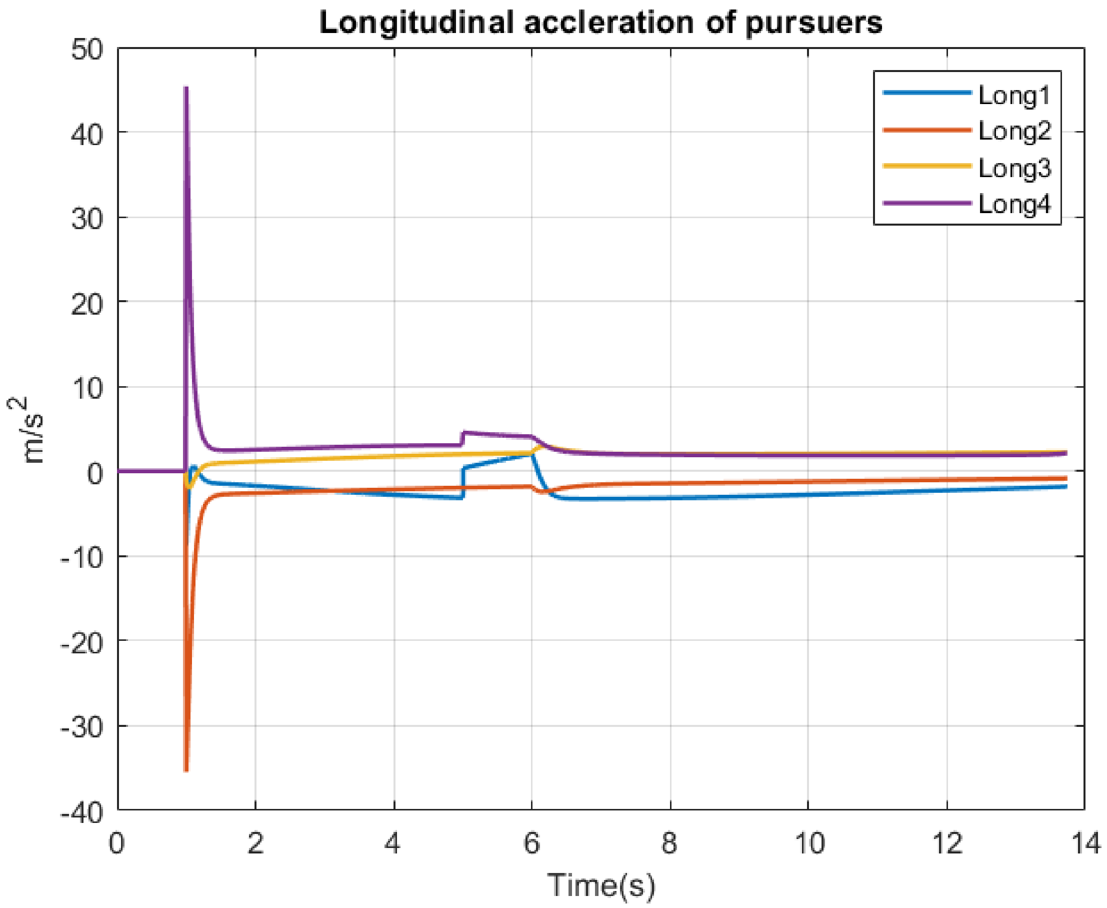

Figure 13. Detailed time history of longitudinal acceleration of pursuing agents

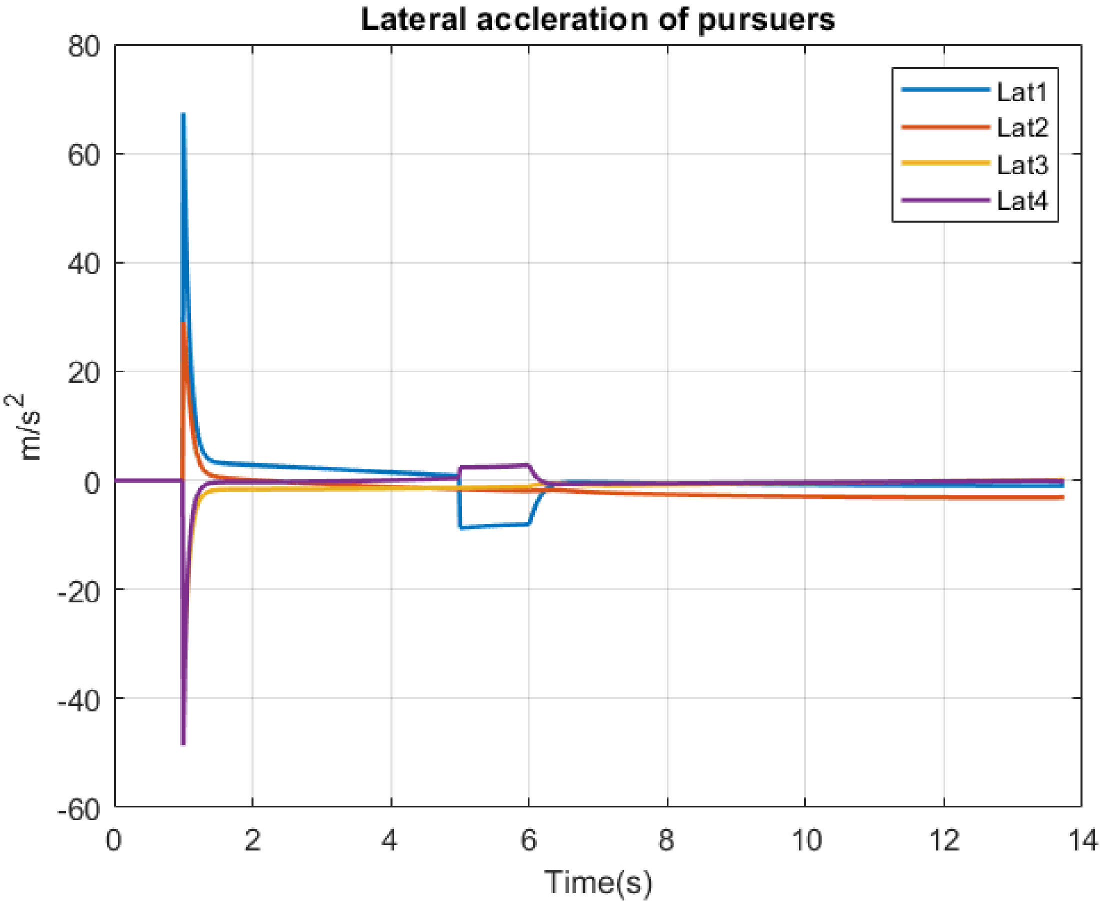

Figure 14. Detailed time history of lateral acceleration of pursuing agents

The rotation of the formation and gripping is evident from Fig. 9, and the fact that this rotation is achieved while keeping the formation otherwise rigid, is seen in Fig. 6, which shows the lengths $r_{12}, r_{23}\ and\ r_{34}$. All these lengths are seen to remain constant during the engagement, and this indicates that the pursuing mobile agents do not collide with one another. Whereas length r_{34} is the open and gripping end whose length decreases when the gripping guidance law starts acting.

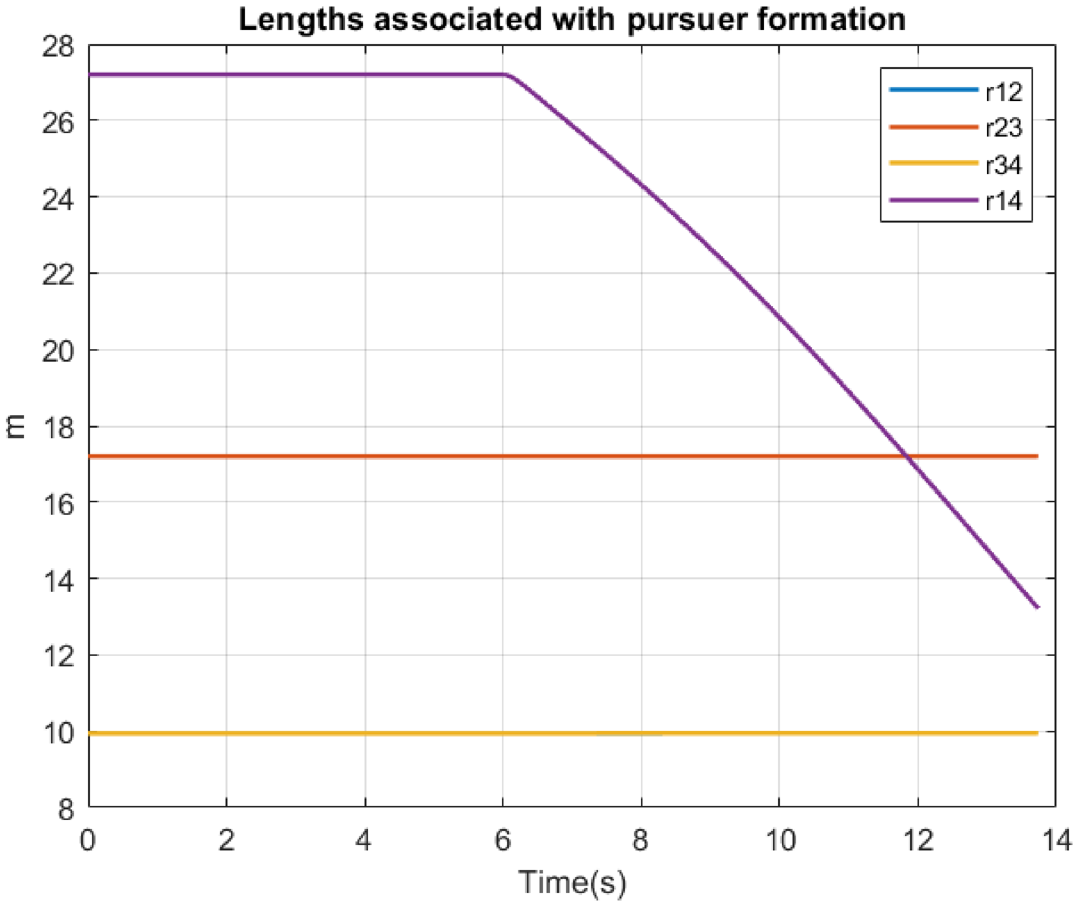

Figure 15. Detailed time history of length associated with pursuing agents

Fig. 16, shows the desired angle of open end of chain approaching towards -10°. Total simulation time is 13.83 sec.

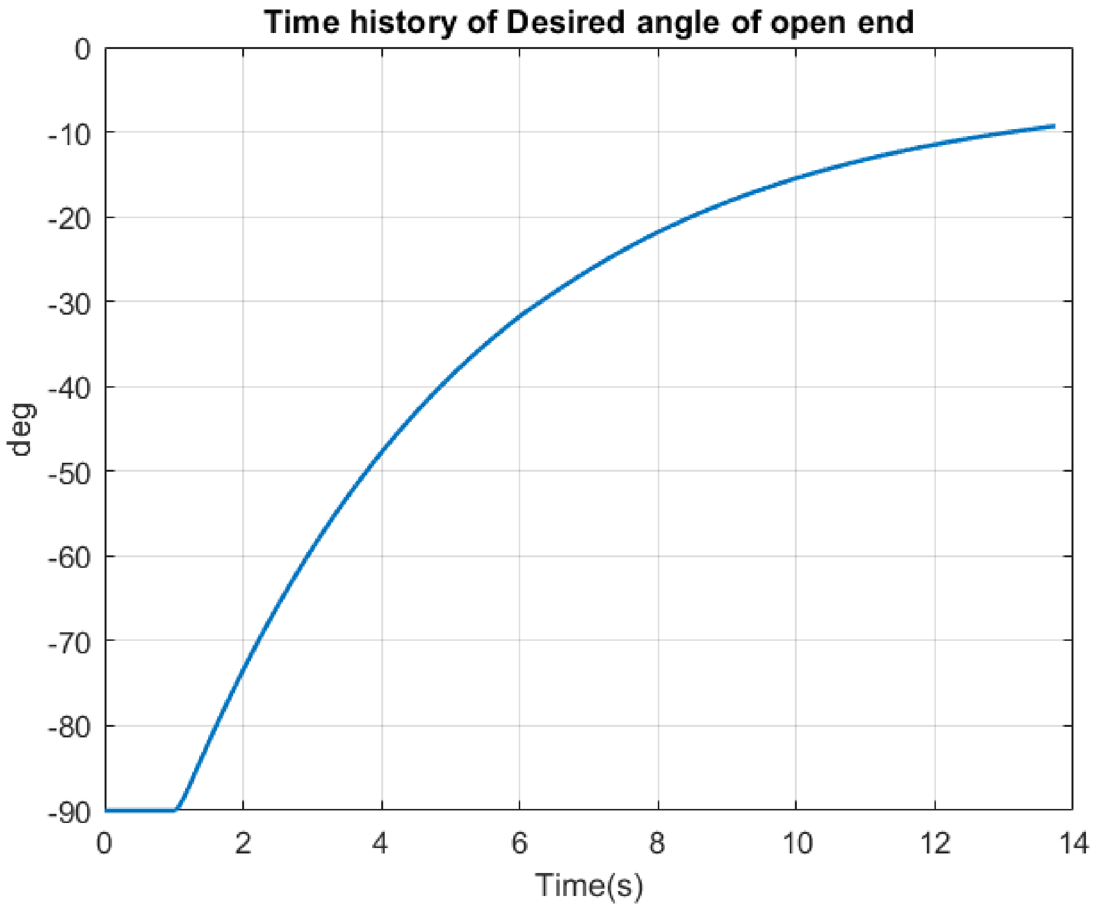

Figure 16. Detailed time history of desired angle of open end

Fig. 17, shows the time histories of the relative velocity components $(V_{\theta X}, V_{rX})$ $and\ r_X$, θ_X .

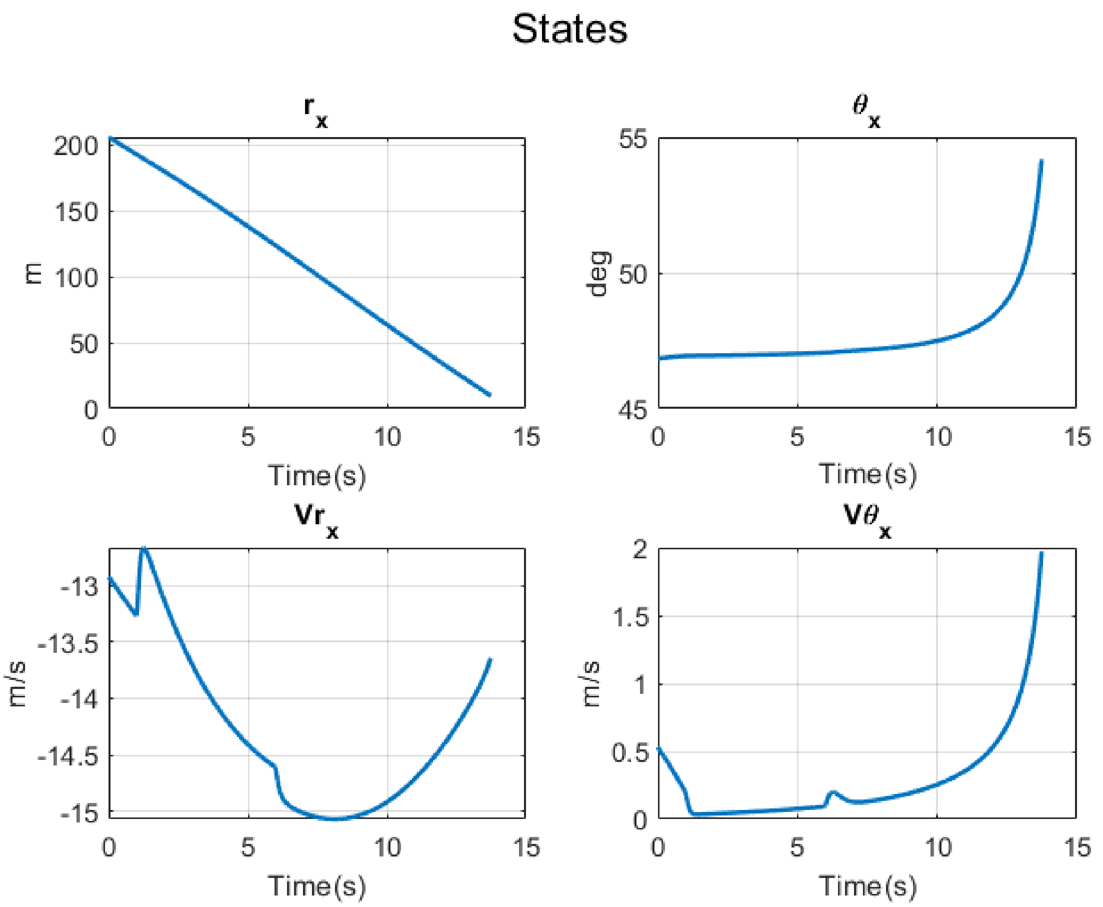

Figure 17. Detailed time history of all the states

4.2 Use of Lidar and EKF (without feedback)

This section shows the simulation results when the mobile agents are each equipped with a lidar sensor. In Fig. 18 red objects are the mobile agents which will be carrying a gripper. Green colored object is a moving target. A cuboid with 5mm length and width has been introduced as a moving target. As our example is in 2D, the ground has been introduced to in the same figure in grey.

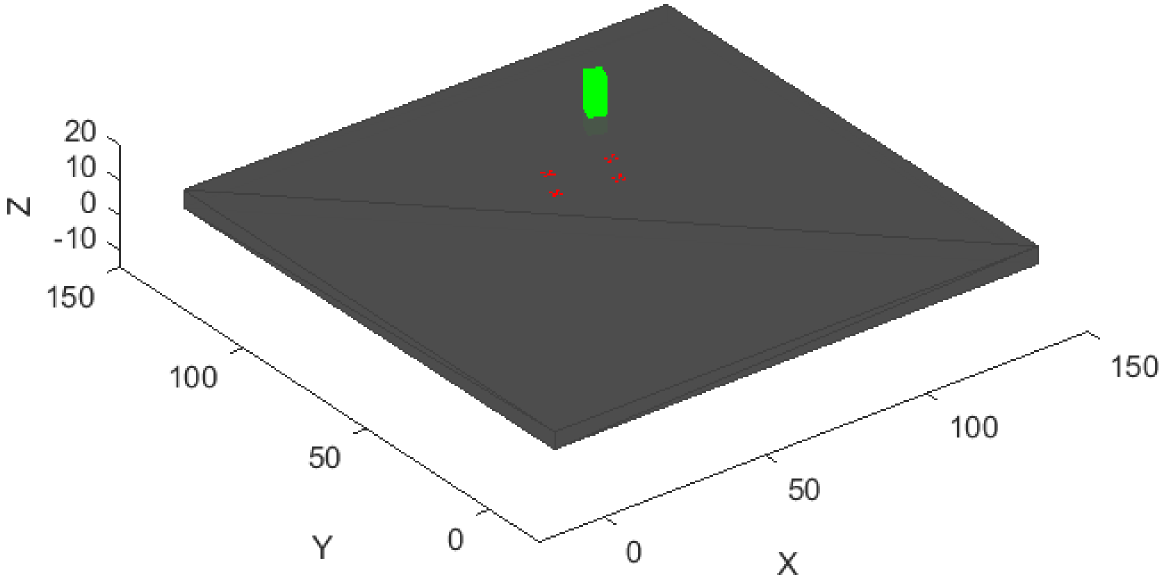

Figure 18. Representation of UAVscenario from MATLAB

Figure 19 shows the actual shape of target seen by the lidar on virtual centroid agent at a given instant.

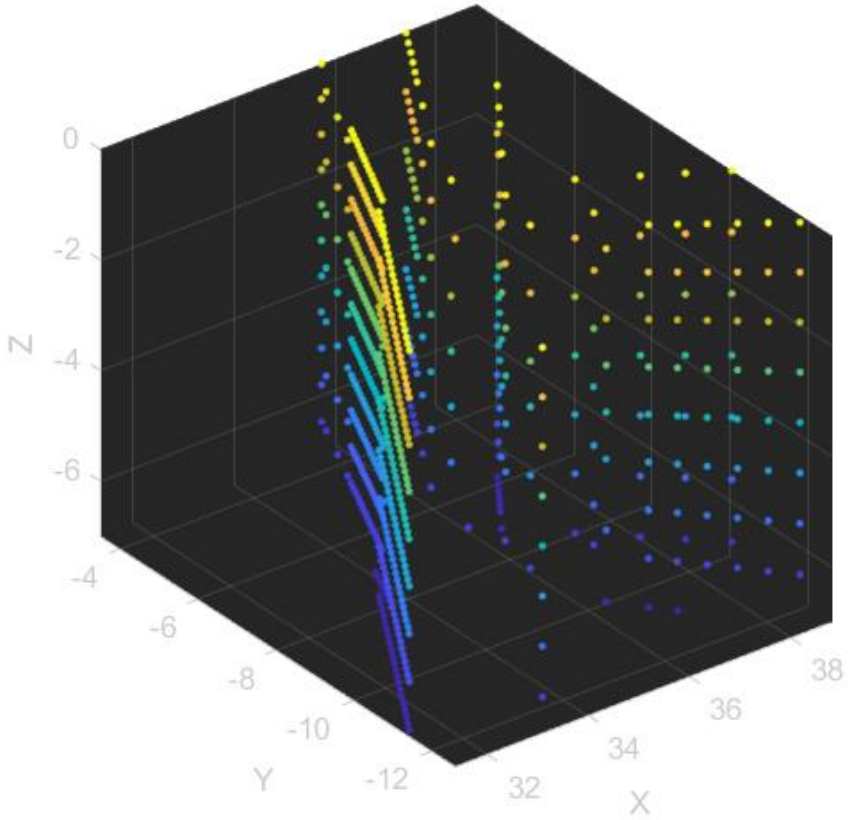

Figure 19. Point cloud representing the target shape seen from Lidar

We set up the Lidar in MATLAB just below each mobile agent with the specification as: Max Range = 210m; Elevation Limits = [-30, 0]; Azimuth Limits = [-180, 180]; Elevation Resolution = 1.25; Azimuth Resolution = 0.3324099. Fig. 14 contains the point clouds which corresponds to the target, and the origin is assumed to be the location of the virtual central lidar obtained by merging cloud points from the individual lidar sensors.

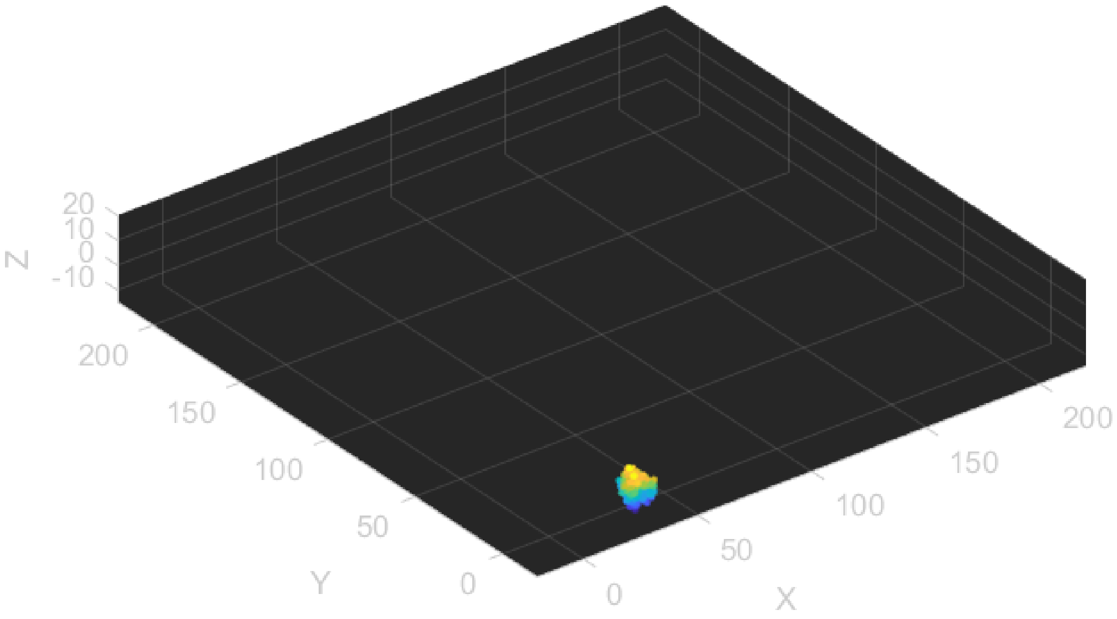

Figure 20. Point Clouds with noise representing the target position

Below simulation graph shows the comparison between true values and measured value from Lidar. Fig. 21 and Fig. 22 shows r_X i.e., range to the target and θ_X i.e., bearing to the target respectively.

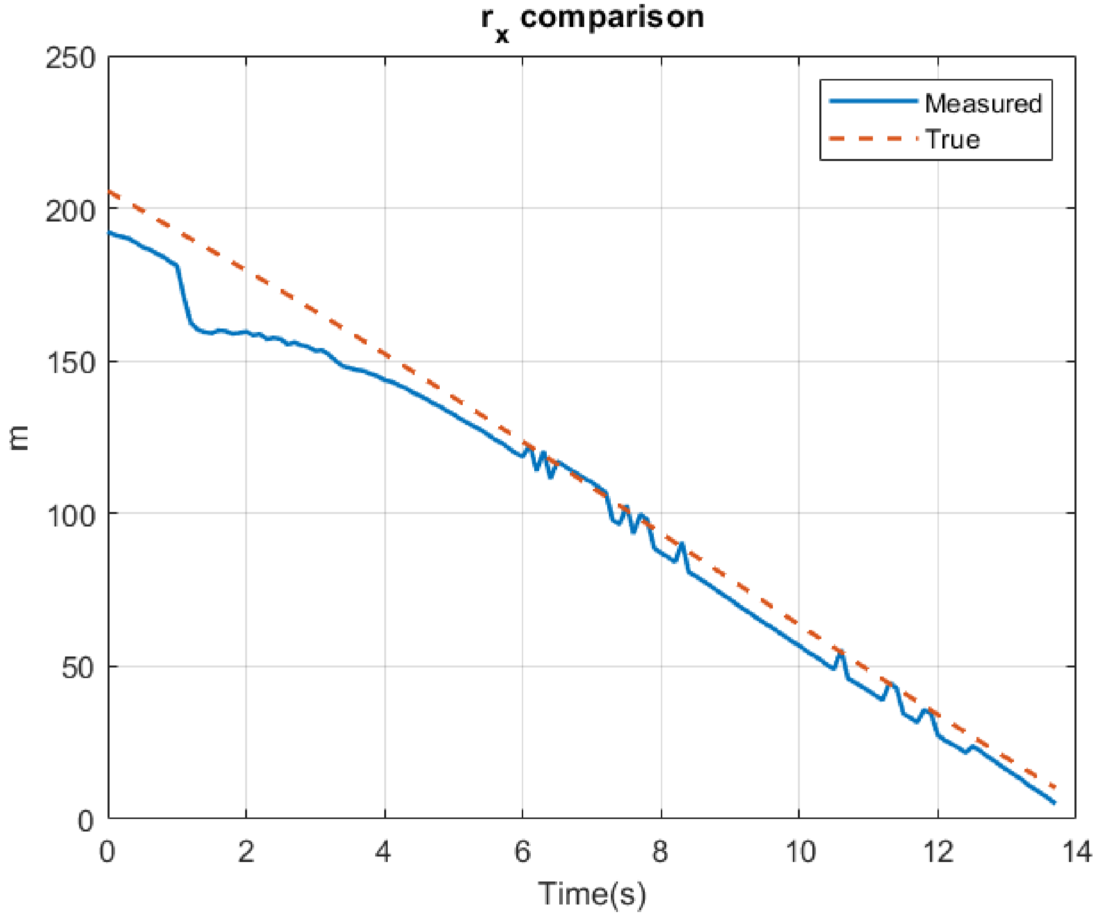

Figure 21. Comparison of r_X from lidar and from perfect model

It takes first 6 seconds to transform and merge all the point clouds from different Lidars to form a one cluster point cloud. That is why, in Fig. 21 we see that the r_X measured value matches true value after 6 seconds. Similarly in Fig. 22, we see a big difference in angle between true and measured value of θ_X for initial 6 seconds. After 6 seconds we can see the difference of a maximum 5° which is acceptable for further calculation.

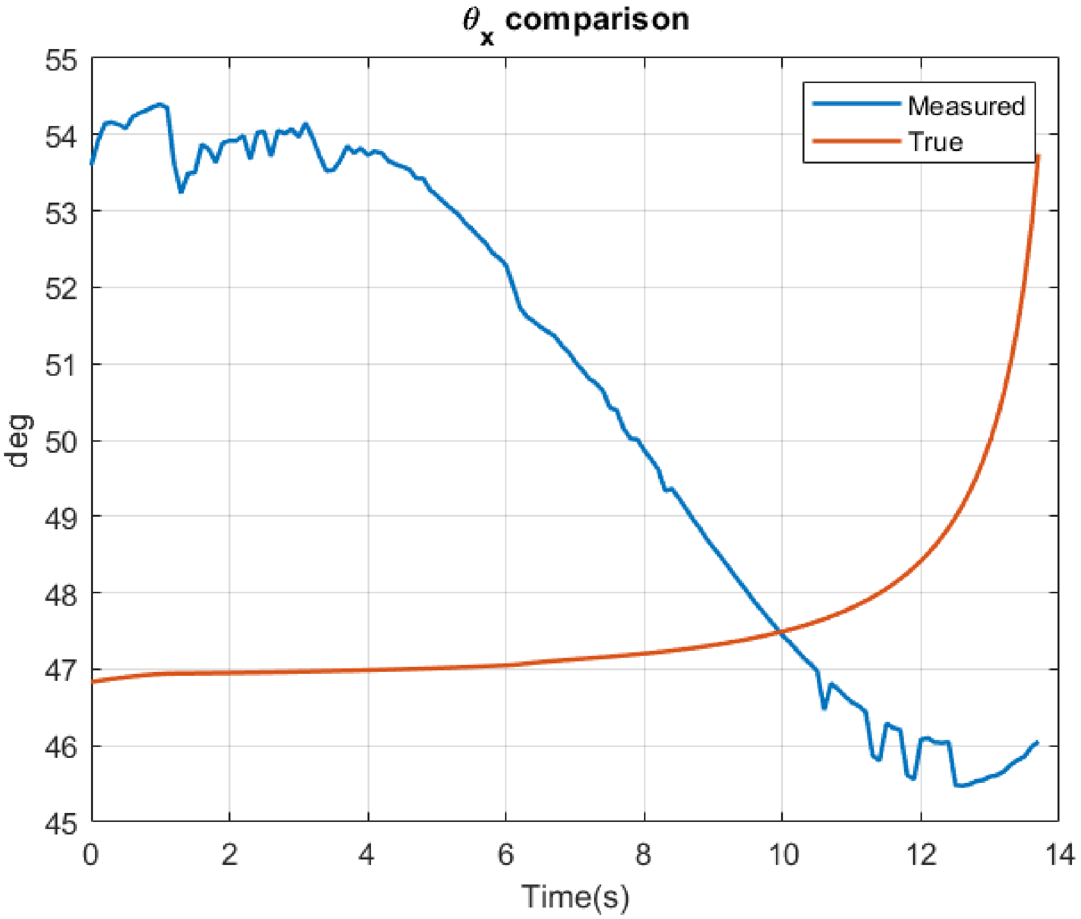

Figure 22. Comparison of θ_X from lidar and from perfect model

Hereafter we used the extended Kalman Filter to estimate the remaining states like $V_{\theta X}$ and V_{rX} $from$ r_X, θ_X which we got from Lidar measurements. All the simulations here are without feedbacking the EKF estimated states to the guidance laws. Fig. 23 will show the comparison between estimated state from EKF (without feedback), true states from perfect model, measured states from Lidar (only for r_X, θ_X). Black dotted lines show the upper and lower bound limits where estimated states should lie between.

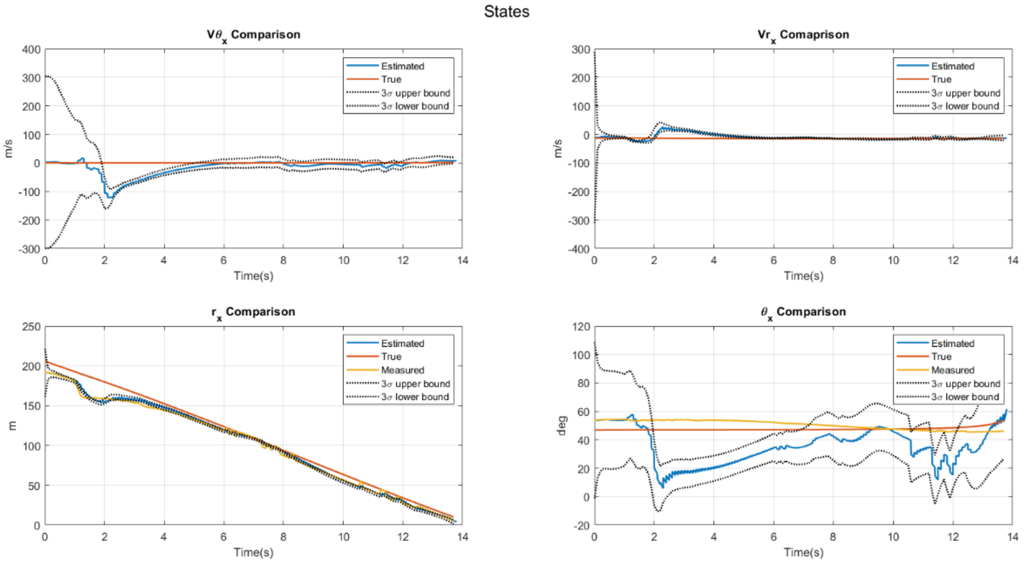

Figure 23. Comparison of all states without feedback

From the above simulation graphs, we can see that estimated states lie between the upper and lower bound without feedback and matched the true values and measured values. This proved that EKF output is now ready for feedbacking the guidance laws.

4.3 Simulation of Guidance Laws with feedback from Lidar and EKF

It is evident from Fig. 24, the simulations from the first stage are achieved from feedbacking the EKF outputs in the guidance laws. It is also clearly seen from same figure, the pursuing grippers catches and grips the target from the direction of the dashed line, which forms the open end of the chain. The pursuing mobile agents use the guidance laws obtained by solving (2.2.2.1) and then apply gripping guidance laws obtained by solving Equations (2.3.3) and (2.3.9) law after 7 seconds.

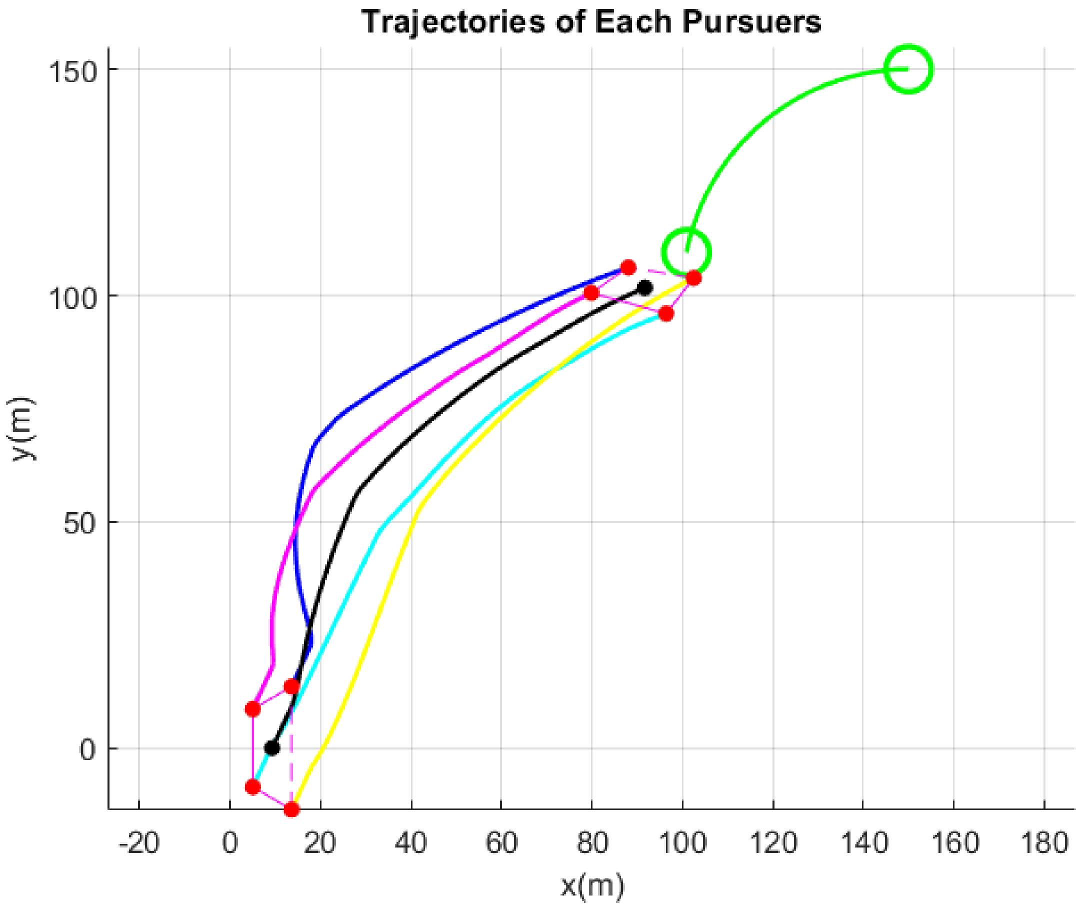

Figure 24. Trajectories of 4 mobile agents and target

Detailed plots, showing time histories are given in below figures. Fig. 25 shows the plot of the collision cone function y_X. Using the developed outputs from Extended Kalman Filter after 6 seconds into guidance laws, the pursuing mobile agents successfully drive y_X to reference value i.e., $w = -24$.

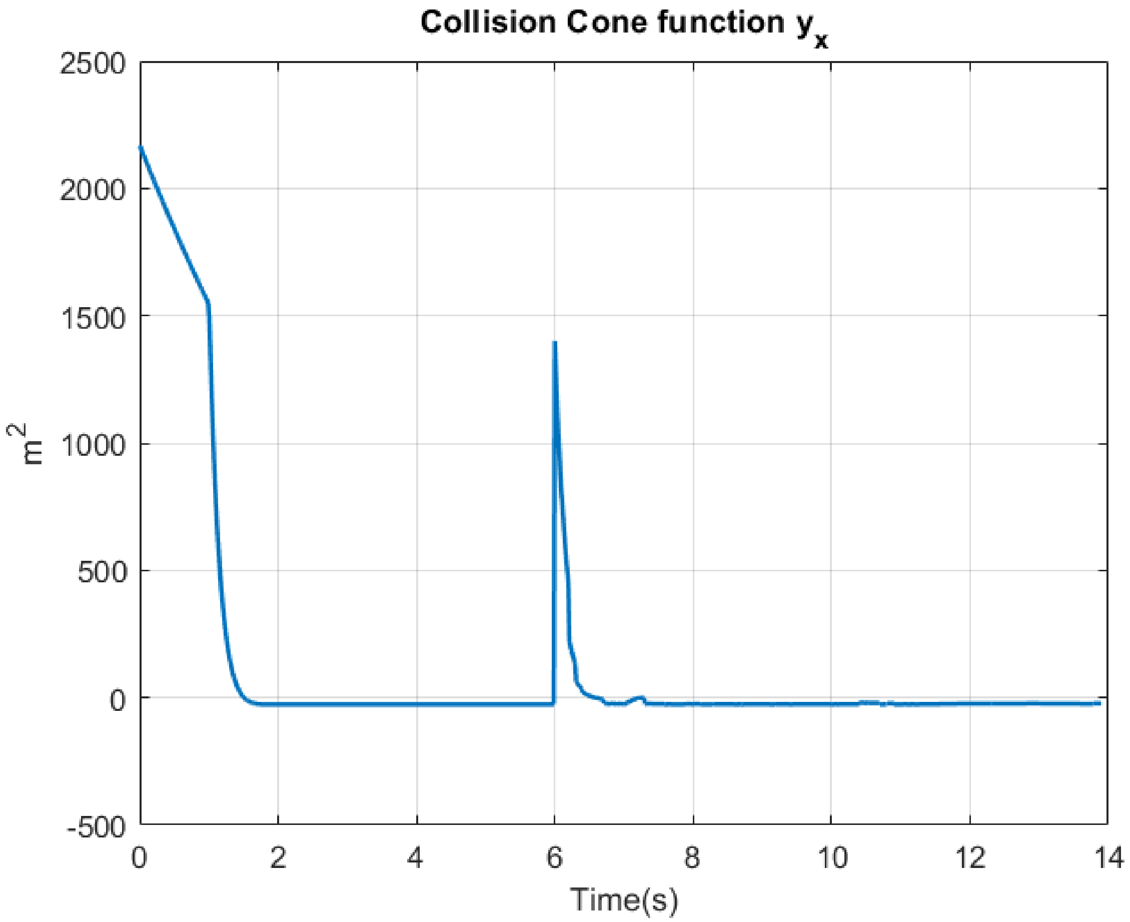

Figure 25. Detailed time history of collision cone function

The time histories of the heading angles of the pursuing agents are shown in Fig. 26, and those of their speeds are in Fig. 27.

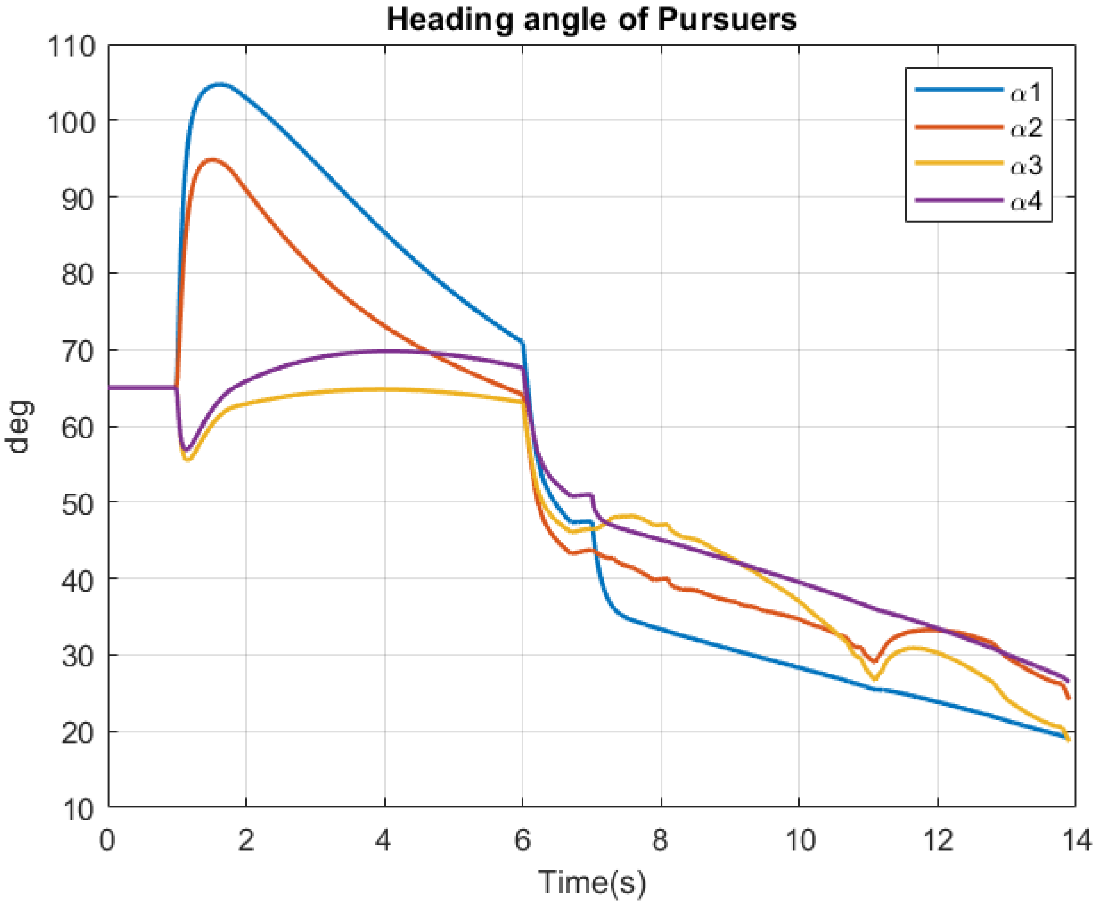

Figure 26. Detailed time history of heading angle of pursuing mobile agents

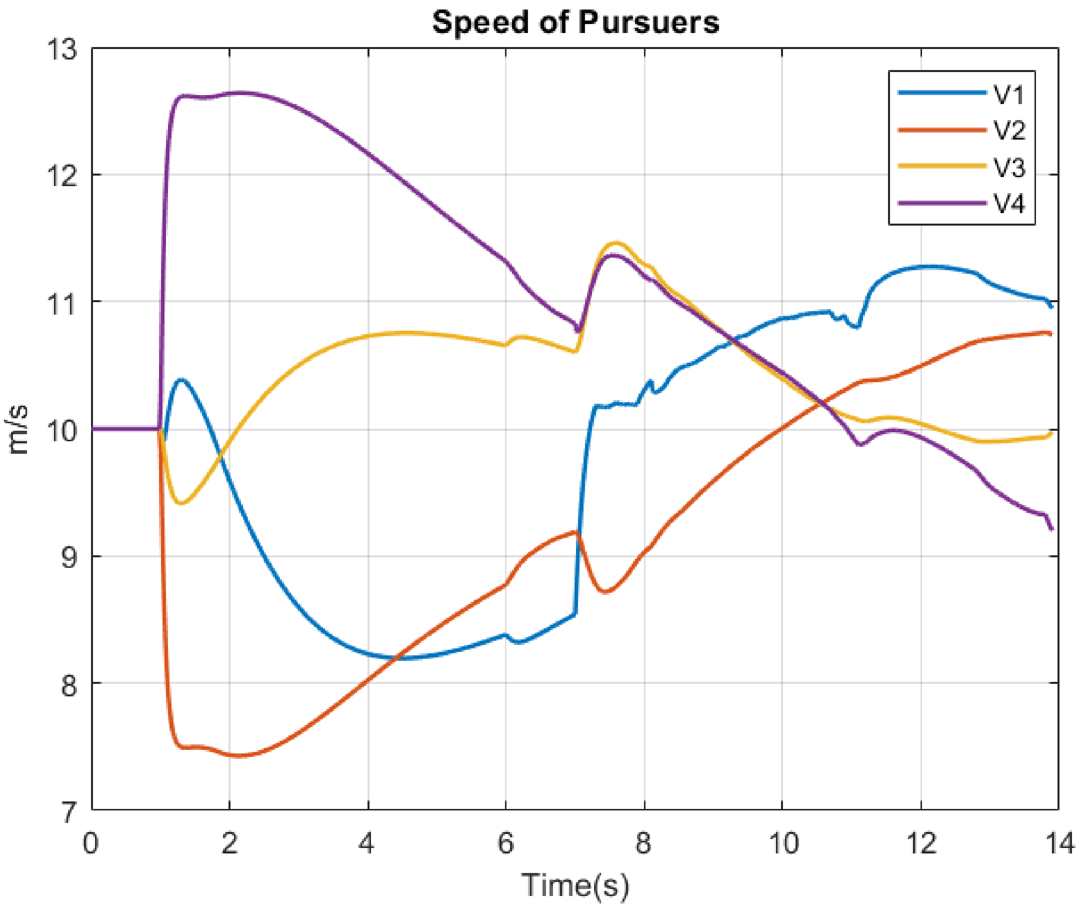

Figure 27. Detailed time history of speed of pursuing mobile agents

The longitudinal and lateral accelerations of the pursuing agents are shown in Fig. 28 and Fig. 29.

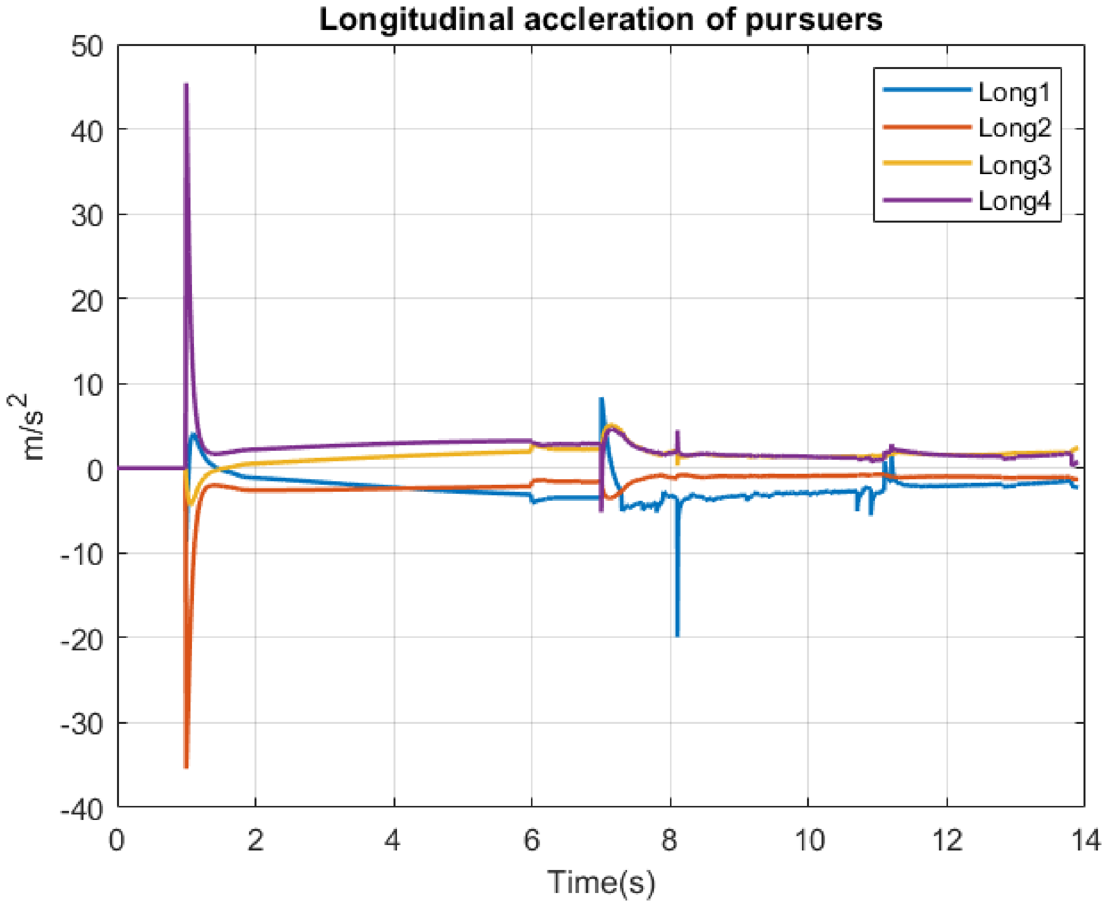

Figure 28. Detailed time history of longitudinal acceleration of pursuing mobile agents

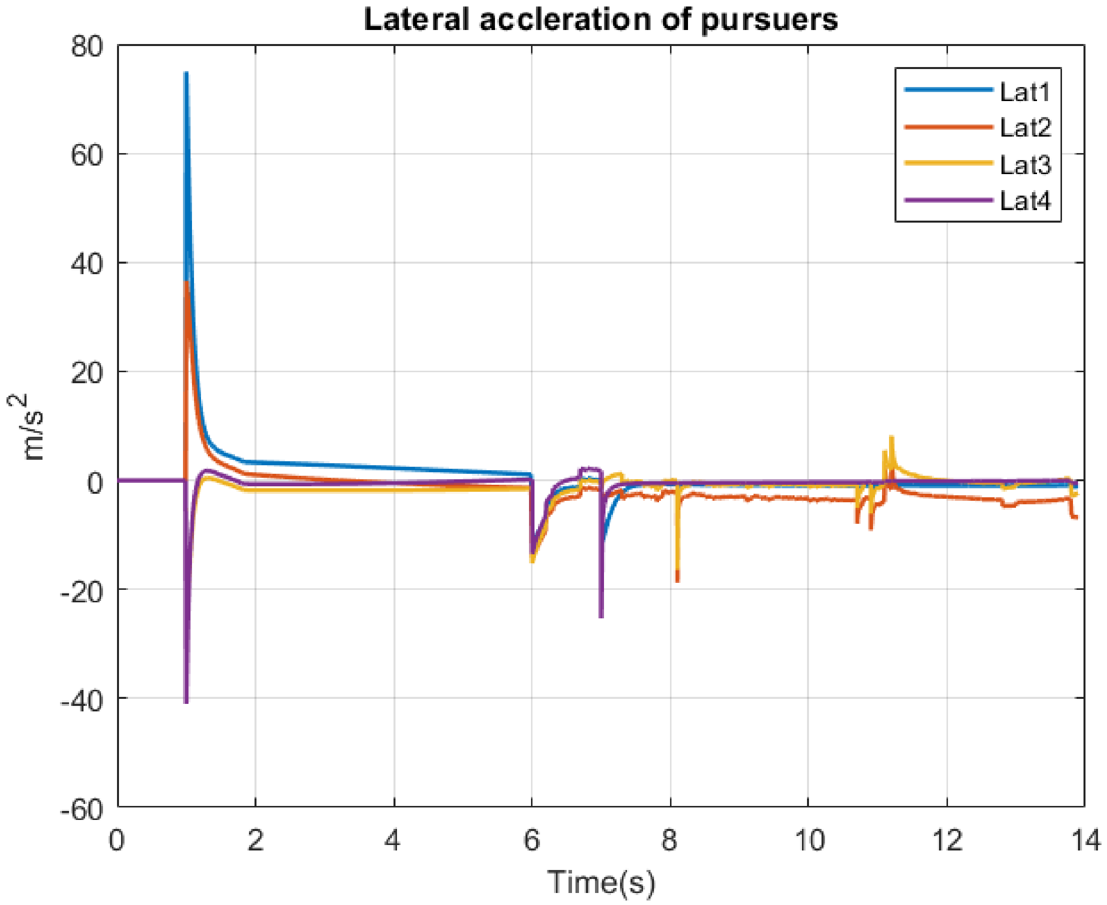

Figure 29. Detailed time history of lateral acceleration of pursuing mobile agents

The rotation of the formation and gripping is evident from Fig. 24, and the fact that this rotation is achieved while keeping the formation otherwise rigid, is seen in same figure, which shows the lengths $r_{12}, r_{23}\ and\ r_{34}$. All these lengths are seen to remain constant during the engagement, and this indicates that the pursuing mobile agents do not collide with one another. Similarly, we can see the decrease in length of open and gripping ends of side r_{34} in below figure which is also seen in Fig.15.

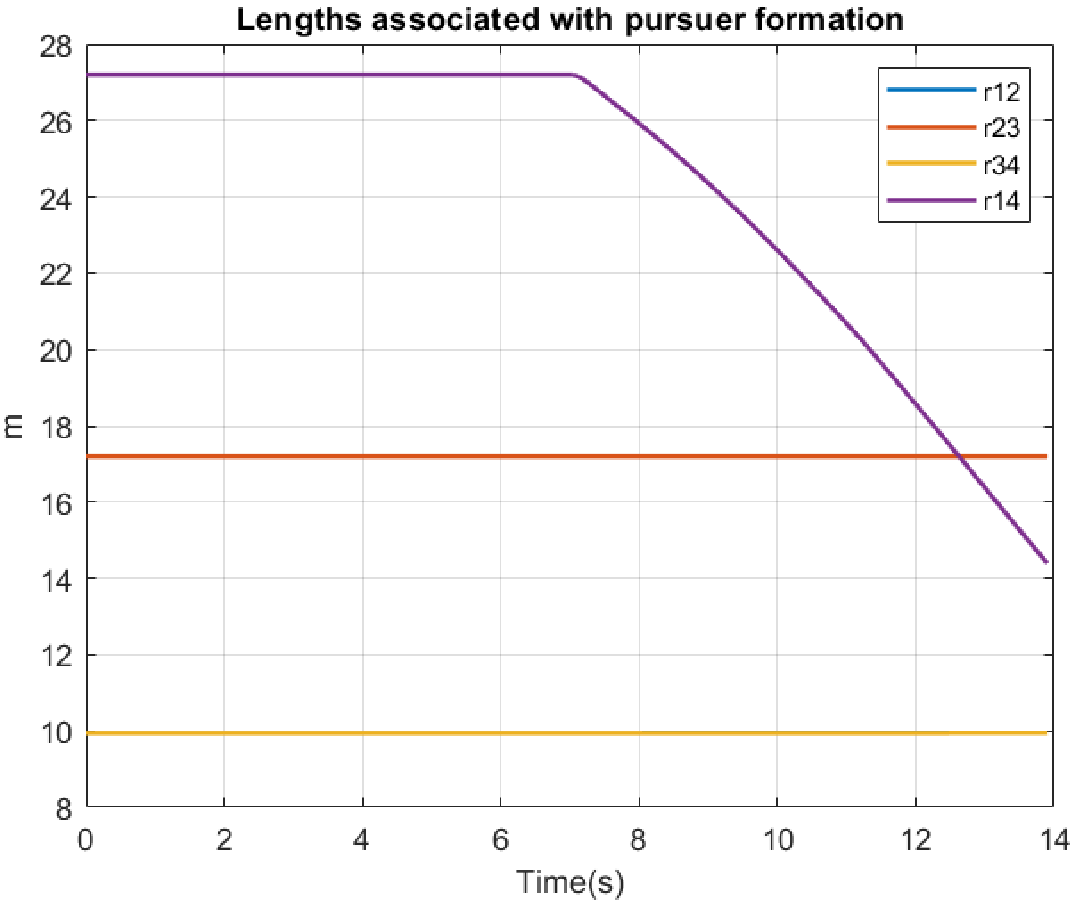

Figure 30. Detailed time history of length associated of pursuing mobile agents

Fig. 31 shows the desired angle of open end of chain approaching towards -10° in 13.95 seconds.

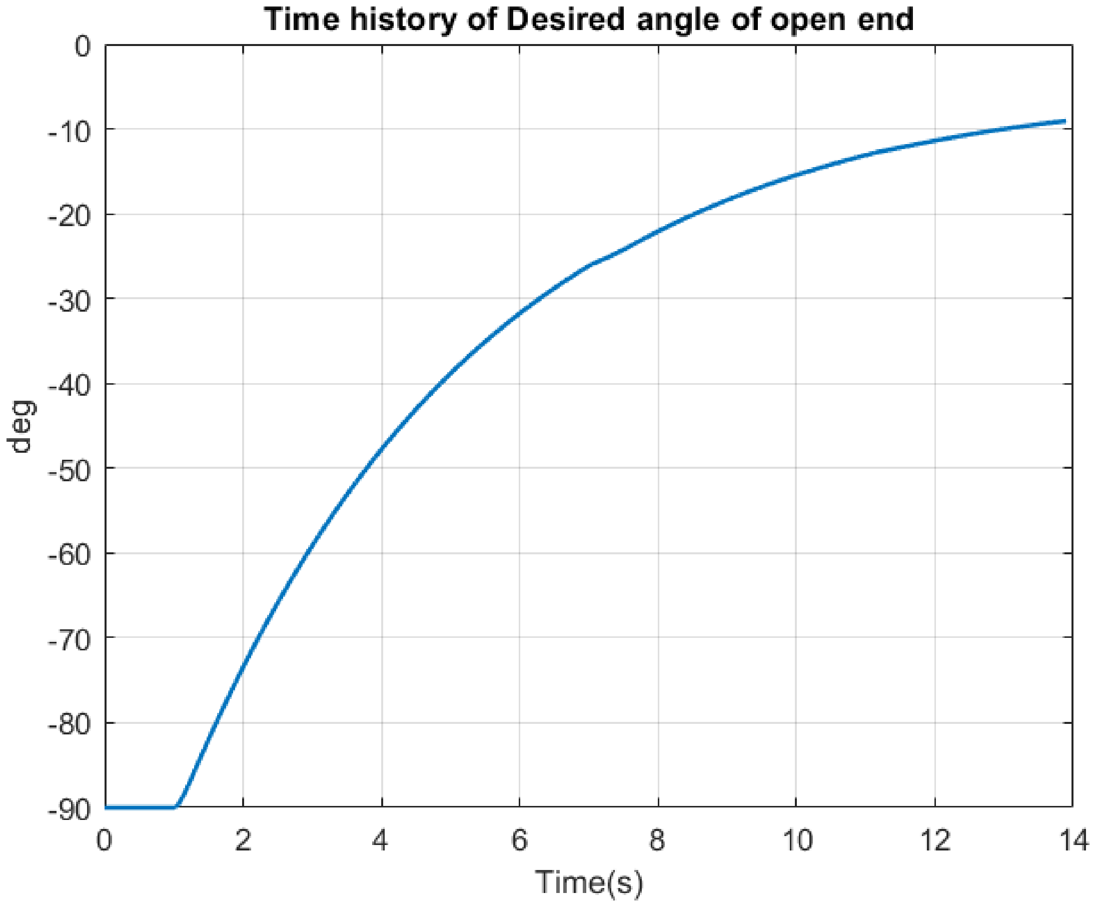

Figure 31. Detailed time history of time history angle of open end

Fig. 32 shows the time histories of the relative velocity components $(V_{\theta X}, V_{rX})\ and\ r_X$, θ_X .

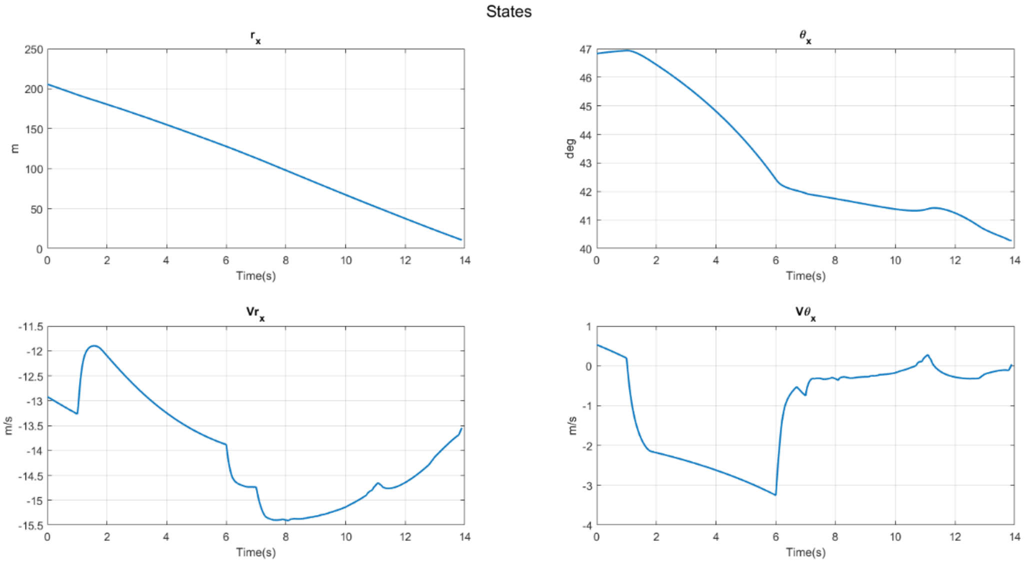

Figure 32. Detailed time history of all the states

Also Fig. 33, shows the time histories of the relative velocity components $(V_{\theta X}, V_{rX})\ and\ r_X\ , \theta_X$ comparing True, Measured and Estimated values showing how closely it matches the perfect model.

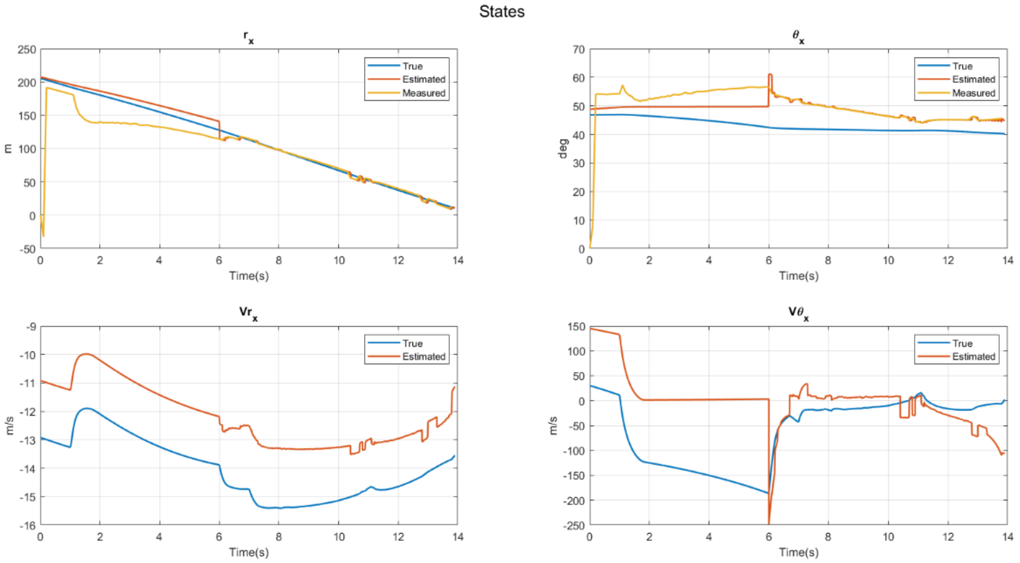

Figure 33. Detailed time history of comparison of all states

Chapter 5 Conclusion

This thesis considers the problem of four mobile agents to pursue, surround and grip a target from inaccessible locations. The development of analytical cooperative pursuit and gripping guidance laws that will meet the motivation of the pursuing mobile agents is addressed in this thesis. These guidance laws are developed based on a collision cone approach. Implementation of lidar sensor on mobile agents was successfully done in MATLAB which enabled in lidar-based relative position estimation and tracking. A distinguishing feature of this work is the fact that each agent only relies on 2D scan provided by a lidar for sensory information. Estimation of the states was achieved using Extended Kalman Filter, and these states were fed back to the guidance laws. This gave the expected simulation results of the trajectories of the mobile agents pursuing, surrounding, and gripping the target. However, by exploiting simple geometric features of the individual mobile agents, we were able to reliably estimate and track the position of the target in the environment.

Future work will include consideration of obstacles which can be present in the vicinity of the target. Thus, the algorithm needs to be enhanced to avoid obstacles along with pursuing and gripping the target while maintaining the formation. Environmental factors like ground, stationary as well as moving objects and other types of noise have not been considered in this thesis. This can be addressed in the future by performing data clustering of the point from lidar which depicts different objects and selection of required clusters using different algorithms to remove unwanted noise. Guidance laws can be generalized to pursue, surround, and grip different shapes of targets.

REFERENCES

[1] Chakravarthy, A., & Ghose, D. (2020). Cooperative Pursuit Guidance to Surround Intruder Swarms Using Collision Cones. Journal Of Aerospace Information Systems, 17(8), 454-469. doi: 10.2514/1. i010790

[2] Baba, Y., Yamaguchi, M., & Howe, R. (1993). Generalized guidance law for collision courses. Journal Of Guidance, Control, And Dynamics, 16(3), 511-516. doi: 10.2514/3.21039

[3] Chakravarthy, Animesh & Ghose, Debasish. (1998). Obstacle avoidance in a dynamic environment: A collision cone approach. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on. 28. 562 - 574. 10.1109/3468.709600.

[4] Coppola, M., McGuire, K.N., Scheper, K.Y.W. et al. On-board communication-based relative localization for collision avoidance in Micro Air Vehicle teams. Auton Robot 42, 1787–1805 (2018). https://doi.org/10.1007/s10514-018-9760-3

[5] Wąsik, A., Ventura, R., Pereira, J., Lima, P., & Martinoli, A. (2015). Lidar-Based Relative Position Estimation and Tracking for Multi-robot Systems. Advances In Intelligent Systems and Computing, 3-16. doi: 10.1007/978-3-319-27146-0_1

[6] Ali, Z., & Li, X. (2019). Modeling and controlling of quadrotor aerial vehicle equipped with a gripper. Measurement And Control, 52(5-6), 577-587. doi: 10.1177/0020294019834040

[7] Optimal Estimation of Dynamic Systems-John L. Crassidis, John L. Junkins 2nd Edition

[8] Chakravarthy, A., and Ghose, D., "Obstacle Avoidance in a Dynamic Environment: A Collision Cone Approach," IEEE Transactions on Systems, Man, and Cybernetics, Part A:

Systems and Humans, Vol. 28, No. 5, Sept. 1998, pp. 562–574. https://doi.org/10.1109/3468.709600

[9] Korpela, C.M., Danko, T.W. & Oh, P.Y. MM-UAV: Mobile Manipulating Unmanned Aerial Vehicle. J Intell Robot Syst 65, 93–101 (2012). https://doi.org/10.1007/s10846-011-9591-3

[10] Chi, W. C., Low, K. H., Hoon, K. H., & Tang, J. (2013). Design of Control Strategy for Autonomous Perching with a Quadrotor. In Applied Mechanics and Materials (Vol. 461, pp. 506–512). Trans Tech Publications, Ltd. https://doi.org/10.4028/www.scientific.net/amm.461.506

[11] Mindy Tieu, Duncan M. Michael, Jeffery B. Pflueger, Manik S. Sethi, Kelli N. Shimazu, Tatiana M. Anthony, and Christopher L. Lee "Demonstrations of bio-inspired perching landing gear for UAVs", Proc. SPIE 9797, Bioinspiration, Biomimetics, and Bioreplication 2016, 97970X (22 April 2016); https://doi.org/10.1117/12.2218167

[12] K. Tadakuma, C. J. Salaan, E. Takane, Y. Okada, K. Ohno and S. Tadokoro, "Design of Aerial Manipulator Suitable for a UAV with Two Passive Rotating Hemispherical Shells," 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), 2018, pp. 1-6, doi: 10.1109/SSRR.2018.8468654.

[13] Maroo, TD, & Wright, AB. "A Novel Gripping System for Delivery of Packages via Unmanned Aerial Vehicles." Proceedings of the ASME 2020 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Volume 10: 44th Mechanisms and Robotics Conference (MR). Virtual, Online. August 17–19, 2020. V010T10A066. ASME. https://doi.org/10.1115/DETC2020-22179

[14] Wang, G., Yu, Y., & Feng, Q. (2016). Design of end-effector for tomato robotic harvesting. IFAC-PapersOnLine, 49(16), 190–193. https://doi.org/10.1016/j.ifacol.2016.10.035

[15] 'Design, Analysis and Fabrication of Quad Copter for Medicine Delivery':: ISSN: 2455-2631, International Journal of Scientific Development and Research, June 2017, Volume 2, Issue 6

[16] 'Design, Analysis, and Fabrication of Quadcopter for Emergency Medical Services using GPS':: ISSN: 2249-5762 (Online) International Journal of Research in Mechanical Engineering & Technology, May-Oct 2017, Volume 7, Issue 2

[17] Bopardikar, S. D., Bullo, F., and Hespanha, J., "Cooperative Pursuit with Sensing Limitations," Proceedings of the American Control Conference, IEEE Publ., Piscataway, NJ, July 2007. https://doi.org/10.1109/ACC.2007.4282474

[18] Zhou, Z., Zhang, W., Ding, J., Huang, H., Stipanovic, D. M., and Tomlin, C., "Cooperative Pursuit with Voronoi Partitions," Automatica, Vol. 72, Oct. 2016, pp. 64–72. https://doi.org/10.1016/j.automatica.2016.05.007

[19] Klein, K., and Suri, S., "Complete Information Pursuit Evasion in Polygonal Environments," Proceedings of the 25th AAAI Conference on Artificial Intelligence, AAAI Publ., Palo Alto, CA, 2011, pp. 1120– 1125. https://doi.org/10.5555/2900423.2900601

[20] Kim, T.-H., and Sugie, T., "Cooperative Control for Target-Capturing Task Based on a Cyclic Pursuit Strategy," Automatica, Vol. 43, No. 8, Aug. 2007, pp. 1426–1431. https://doi.org/10.1016/j.automatica.2007.01.018

[21] Huang, H., Zhou, Z., Zhang, W., Ding, J., Stipanovic, D. M., and Tomlin, C., "Safe-Reachable Area Cooperative Pursuit," IEEE Transactions on Robotics, Vol. 10, No. 5, 2012

[22] Von Moll, A., Casbeer, D. W., Garcia, E., and Milutinović, D., "PursuitEvasion of an Evader by Multiple Pursuers," International Conference on Unmanned Aircraft Systems

(ICUAS), IEEE Publ., Piscataway, NJ, June 2018, pp. 133–142. https://doi.org/10.1109/ICUAS.2018.8453470

[23] Pachter, M., Von Moll, A., Garcia, E., Casbeer, D. W., and Milutinović, D., “Two-on-One Pursuit,” Journal of Guidance, Control, and Dynamics, Vol. 42, No. 7, Feb. 2019, pp. 1638–1644. https://doi.org/10.2514/1.G004068

[24] Jin, S., and Qu, Z., “Pursuit-Evasion Games with Multi-Pursuer vs. One Fast Evader,” 8th World Congress on Intelligent Control and Automation, IEEE Publ., Piscataway, NJ, July 2010, pp. 3184–3189. https://doi.org/10.1109/WCICA.2010.5553770

[25] Jha, B., Tsalik, R., Weiss, M., and Shima, T., “Cooperative Guidance and Collision Avoidance for Multiple Pursuers,” Journal of Guidance, Control, and Dynamics, Vol. 42, No. 7, Feb. 2019, pp. 1506–1518. https://doi.org/10.2514/1.G004139

[26] Wei, M., Chen, G., Cruz, J. B., Jr., Haynes, L. S., Pham, K., and Blasch, E., “Multi-Pursuer Multi-Evader Pursuit-Evasion Games with Jamming Confrontation,” Journal of Aerospace Computing, Information and Communication, Vol. 4, No. 3, March 2007, pp. 693–706. https://doi.org/10.2514/1.25329

[27] Pierson, A., Wang, Z., and Schwager, M., “Intercepting Rogue Robots: An Algorithm for Capturing Multiple Evaders with Multiple Pursuers,” IEEE Robotics and Automation Letters, Vol. 2, No. 2, April 2017, pp. 530–537. https://doi.org/10.1109/LRA.2016.2645516