



**WICHITA STATE  
UNIVERSITY**

**UNIVERSITY LIBRARIES**

**Development of analytical algorithms  
for the sensor coverage problem**

Item Type	Thesis
Authors	Tholen, Kyle
Publisher	Wichita State University
Rights	Copyright 2015 Kyle Tholen
Download date	2026-05-08 11:12:03
Link to Item	<a href="http://hdl.handle.net/10057/11655">http://hdl.handle.net/10057/11655</a>

# DEVELOPMENT OF ANALYTICAL ALGORITHMS FOR THE SENSOR COVERAGE PROBLEM

A Thesis by

Kyle Tholen

Bachelor of Science, University of Minnesota, 2007

Submitted to the Department of Engineering  
and the faculty of the Graduate School of  
Wichita State University  
in partial fulfillment of  
the requirements of the degree of  
Master of Science

May 2015

© Copyright 2015 by Kyle Tholen

All Rights Reserved

DEVELOPMENT OF ANALYTICAL ALGORITHMS FOR THE SENSOR COVERAGE  
PROBLEM

The following faculty members have examined the final copy of this thesis for form and content, and recommend that it be accepted in partial fulfillment of the requirement for the degree of Master of Science with a major in Aerospace Engineering.

---

Animesh Chakravarthy, Committee Chair

---

John M Watkins, Committee Member

---

James E Steck, Committee Member

## **DEDICATION**

To Katie, Norah, and Graeme

## **ABSTRACT**

The sensor coverage problem requires a group of mobile sensors to autonomously determine trajectories such that the sensors enclose a region of interest. It has many applications such as tracking oil spills underwater, volcanic ash plumes in the air, and so on. This thesis develops analytical control laws so as to distribute autonomous mobile sensors on a plane in a manner so as to ensure that an entire region of interest is covered by at least one sensor. Two scenarios are considered. In the first scenario the shapes of the sensing zones for each sensor are assumed to be circular, and dynamic inversion is used to develop control laws that autonomously position three sensors in such a manner that there is no gap in coverage between them. In the second scenario the shapes of the sensing zones are assumed to be arbitrary (convex or non-convex), and a collision cone approach is combined with dynamic inversion techniques to determine trajectories for the three sensors in such a way that the coverage zones of each sensor touch. Simulations are performed to demonstrate the performance of the designed control laws.

## TABLE OF CONTENTS

Chapter	Page
I. INTRODUCTION .....	1
II. CIRCULAR SENSORS .....	7
Problem formulation .....	7
Control law derivation .....	10
III. CIRCULAR SENSOR SIMULATION RESULTS.....	24
Case 1 .....	25
Case 2.....	28
Case 3.....	30
Case 4.....	32
IV. NON CIRCULAR SENSING RADIUS.....	35
Problem formulation .....	36
Control law derivation .....	37
Example .....	46
V. NON CIRCULAR SENSOR SIMULATION RESULTS .....	54
Case 1 .....	55
Case 2.....	58
Case 3.....	60
Case 4.....	62
VI. CONCLUSIONS.....	65
VII. WORKS CITED .....	67

## CHAPTER 1

### INTRODUCTION

Autonomous mobile sensors can be deployed to monitor a region of interest. Note that a mobile sensor will have a sensing coverage area that may or may not be omnidirectional. There are a number of applications for autonomous mobile sensor networks. These applications can occur in the air, underwater, and on ground. For instance, when volcanic eruptions occur, volcanic ash enters the atmosphere and wind currents can cause the volcanic ash to spread over large areas. In 2010, an eruption of the volcano Eyjafjallajökull in Iceland [1] caused widespread disruption of commercial jet traffic across northern Europe. The smoke and ash in the air caused by volcanic eruption leads to a reduction in visibility, and the ash poses a risk to the engines of aircraft that fly through it. An aerial autonomous mobile sensing network can be deployed after an eruption to monitor the density of ash and smoke to help determine if commercial jets can be safely flown in the area.

Aerial sensor coverage networks could also be used to monitor atmospheric conditions. Aircraft are sometimes flown into tropical storms and hurricanes to gather important meteorological data that cannot be measured by weather satellites, such as accurate wind speed and the barometric pressure throughout the storm. In the United States, the United States Air Force Reserve's 53rd weather Reconnaissance Squadron fly Lockheed WC-130J aircraft directly into a hurricane and penetrate the eye several times in a mission at altitudes ranging from 500 feet to 10,000 feet [2]. The information that is collected can be very important for forecasting the strength and trajectory of hurricanes and accurate forecasts can help determine the need for evacuating areas that are in the path of a hurricane. It is very dangerous for a crew to repeatedly

fly into a hurricane, but an autonomous sensor coverage network can be used to collect the data without the risk to an aircrew.

Aerial sensor networks can be used to perform surveillance. An autonomous aerial sensor coverage network can be used to monitor a region. This can be particularly useful in a region where real time information is very important, such as a battlefield. Over a battlefield, piloted aircraft face the risk of enemy fire, while remotely piloted drones can require the pilot to be focused and alert for shifts that may last for several hours. Furthermore, the remote communication between the drone and pilot may be disrupted, rendering the drone uncontrollable. In such cases, an autonomous aerial sensor coverage network can be deployed to monitor a battlefield or track a target over long periods of time, without risking the safety of a human pilot, and without the possibility of human error or communication disruptions that affect piloted drones [3].

There are also underwater applications for autonomous sensor coverage networks. There has been a history of oil spills in the ocean, whether due to oil tanker transport ships wrecking and spilling oil, or off shore oil drills malfunctioning and spilling oil. Whenever an oil spill occurs, it has a disruptive effect on the entire ecosystem of the effected region. It can be dangerous for birds and marine life that are covered in oil, and an oil spill can also have a negative effect on local economies that are dependent on the ocean. An autonomous underwater sensor network can be deployed into the region to monitor the presence and quantity of oil, how far the spill spreads, and how effective the cleanup efforts are [4].

Since the advent of sailing, ships have been sunk whether from storms, accidents, or naval battles. Many ancient shipwrecks provide clues to archeologist as to how ancient civilizations lived, and many Spanish trade ships sank while transporting gold from the new

world back to Spain. Autonomous underwater sensor networks could be deployed to search the ocean for shipwrecks. Once the shipwrecks are located, it may be impractical for divers to dive to the wreck due to the depth or other safety concerns. If the wreck is old enough, it may be impossible to recover any portion of the wreck, but an autonomous underwater sensor network could document the entire wreck with photographs and film.

Ocean currents and temperatures have a tremendous effect on the worldwide climate. Warming or cooling of the ocean can cause a shift in ocean currents and cause the locations of different species of fish providing challenges for fishing industries. Changing ocean temperatures can also cause some regions to be warmer and drier, and others to be cooler and wetter than normal, providing challenges to regions dependent on agriculture. An autonomous underwater sensor network can be deployed to monitor ocean temperatures and currents to better forecast weather patterns.

When multiple autonomous mobile sensors are deployed in a mobile sensing network, the sensor coverage problem needs to be addressed. Sensor coverage problems may be categorized into the area-coverage type, and the boundary-tracking type. In the boundary tracking problem, it is the perimeter of the region of interest that needs to be enclosed by the sensors, whereas in an area coverage type problem, the entire area of interest needs to be covered. The sensor coverage problem is concerned with autonomously placing the mobile sensors in such a way as to ensure that the entire region of interest can be observed by at least one sensor, without any gaps in coverage. To solve the sensor coverage problem, the geometry of the region of interest, and the geometry of the sensing area of each mobile sensor must also be considered, in order to ensure that there are no coverage gaps in a given region of interest [5] [6].

There are several approaches to solving the sensor coverage problem [7] [8] [9] [10] [11] [12] [13] [14], whether the sensors are assumed to be heterogeneous or not [15] [16], depending on how the sensors are able to communicate with each other [3] [17] [18]. Some of the early methods for coordinating multiple robot sensor systems included dividing the region of interest up equally among the available sensors, and each sensor would be assigned a region to cover [7]. The Gabriel graph approach can be used to get near complete coverage. The method requires a desired distance between sensors to be specified, and then each sensor moves in a coordinated fashion until all the sensors are positioned in the desired configuration. The ability of the Gabriel graph method to achieve complete coverage depends on the choice of the distances between each sensor. If the separation distance is too large, then complete coverage cannot be achieved, and if the separation distance is chosen to be too small, then extra sensors could be required to achieve complete coverage [19].

Another method is the Voronoi based coverage algorithms [20] [17]. The idea is that the region of interest is partitioned into regions known as Voronoi cells. The Voronoi partitions are defined such that each sensor has complete coverage of its Voronoi cell, and when the coverage area of two sensors overlap, the Voronoi partition for each sensor is the area closest to the center of the sensor. As the sensors move in space, the Voronoi partitions can also change. A cost function is defined based on the Voronoi partitions, and then a gradient descent algorithm is used to move the sensors to minimize the cost function. This method requires that each Voronoi cell be computed, and that each sensor have geometric knowledge of the sensors in each adjacent Voronoi cell. Depending on the number of sensors, it can take a long time to compute all of the Voronoi cells. An alternative approach that assumes each sensor is omnidirectional, is to divide the region of interest into points, and rather than a cost function to minimize the area of the

sensing void between sensors, use a cost function to drive the distance between the point and the center of the nearest sensor. Once that distance is less than the sensing radius of the sensor, the point is considered covered by the sensor coverage network. There are infinitely many points in a region to consider. To simplify the calculations, the region can be sampled into a grid of points and once each point in the grid is covered, then complete coverage is assumed. The accuracy of this method is limited by the number of points chosen to be in the grid. [20]

An algorithm based on virtual forces can be used to optimize the positioning of sensors in a region [21]. Like the methods described above, the virtual force method requires a predefined desired distance between sensors. If the distance between the sensors is too large, then an attractive virtual force is calculated and the sensors are moved closer together. If the distance between the sensors is too small, then a repulsive virtual force is calculated and the sensors are moved apart. The virtual forces can also account for obstacles and the boundary of the sensor coverage region of interest through attractive and repulsive virtual forces. This method allows for a more autonomous deployment of sensors than the Gabriel graph method because the sensors can orient themselves around boundaries and obstacles and the sensor coverage areas can overlap as necessary, but it still requires a desired separation distance to be defined ahead of time.

This thesis contributes two control algorithms, both of which are analytical in nature. The first control algorithm guides three sensors with circular sensing zones placed at random initial positions and assembles them such that there is no gap in the coverage area between the three sensors. The second control algorithm guides three sensors that do not necessarily have circular sensor coverage zones, but instead may have any arbitrary convex or non-convex shaped coverage zone, together such that the sensor coverage zone of each sensor makes contact with

the sensor coverage zone of the other two sensors. These control algorithm do not require a desired separation distance to be specified, and it does not require the sensors to have uniform sensor coverage zone sizes. The first control algorithm (circular sensing zone) simply requires that the radius of the circular sensing zone of each sensor be known, while the second control algorithm (arbitrarily shaped sensing zone) requires that the shape of each sensor coverage zone be defined. Toward the development of the second control algorithm, the collision cone framework [22] [23] [24] is used. While the collision cone approach in [22] [23] [24] was originally developed for the purpose of collision avoidance of autonomous vehicles, this thesis demonstrates its applicability to addressing the sensor coverage problem as well.

In chapter 2, a control algorithm is developed to guide three sensors with circular sensing zones together such that there is no gap in the sensor coverage between the sensors. In Chapter 3 the control algorithm is tested with four simulation test cases. In chapter 4 a control algorithm is developed to guide three sensors with arbitrary sensing zones together such that the sensing zones of all three sensors touch. In chapter 5 the non circular sensing zone control algorithm is tested with four simulation test cases.

## CHAPTER 2

### CIRCULAR SENSORS

When multiple robots with sensors attached are used to detect an event, the goal is to assemble the robots from arbitrary initial positions to one such that there is no gap in the coverage area of the sensors. We consider three sensors, of which one sensor is the leader which is free to move on a two dimensional plane in an independent trajectory, while the other two sensors,  $F_1$  and  $F_2$ , are followers that will move towards the lead sensor,  $L$ , and assemble in such a way that there is no gap in the coverage area enclosed by the three sensors. The sensors are assumed to have a circular sensing area, and the sensing radius of each sensor is known a priori. The sensors are heterogeneous in the sense that they can have different sensing radii. The vertices of the three sensors form the triangle, and if the sensors move together in a manner such that the perimeter of that triangle is driven to zero, then the coverage gap will also go to zero. A dynamic control law is used to achieve this objective.

#### Problem Formulation

To define the control law, first the system state variables need to be defined. The velocities of the sensors,  $L$ ,  $F_1$  and  $F_2$ , are represented as,  $v_L(t)$ ,  $v_1(t)$ , and  $v_2(t)$ , applied at angles  $\alpha_L(t)$ ,  $\alpha_1(t)$ , and  $\alpha_2(t)$ , respectively. The two follower sensors,  $F_1$  and  $F_2$  are at distances,  $r_1(t)$  and  $r_2(t)$ , from the leader with line of sight angles  $\theta_1(t)$  and  $\theta_2(t)$ . The relative accelerations of the two followers are  $a_1(t)$  and  $a_2(t)$ , applied at angles  $\delta_1(t)$  and  $\delta_2(t)$ , respectively. The distances, velocities, accelerations, and angles of the three sensor system are given in Figure 1.

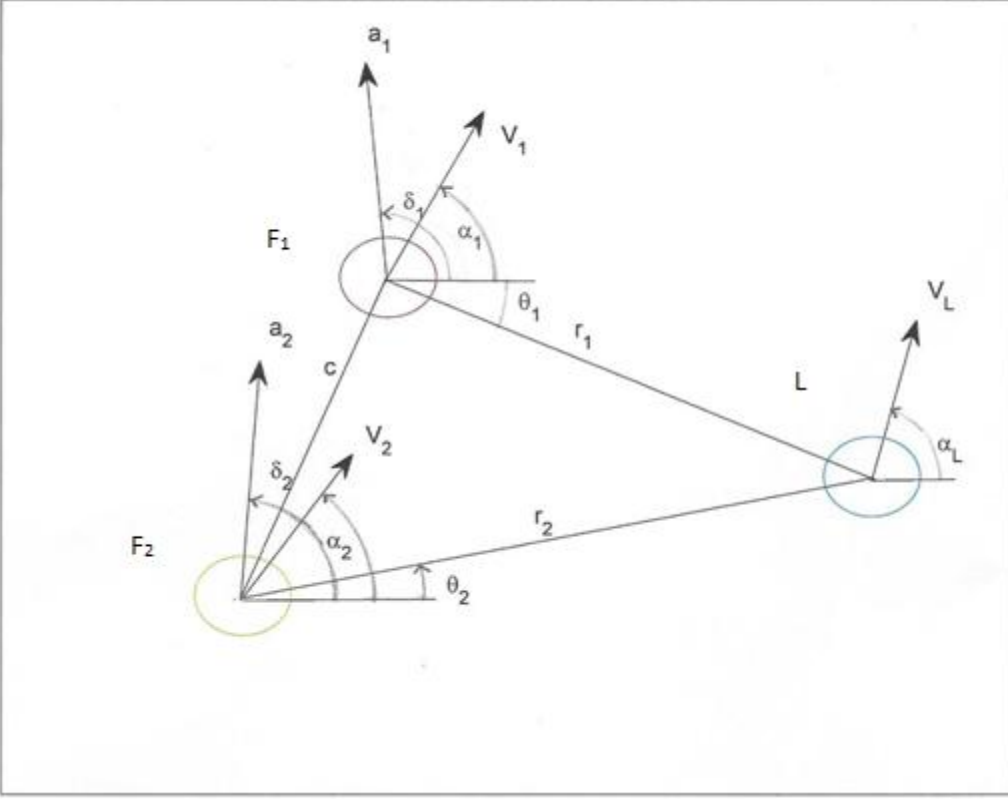


Figure 1 Geometry of the three sensors in the sensor coverage problem

In the interest of saving space, the following abbreviations will be used:

$$\cos() = c() \quad (1)$$

$$\sin() = s() \quad (2)$$

A line of sight is the line joining the two sensors, and the line of sight angle is measured with respect to the horizon. The line of sights are  $F_1L$ ,  $F_2L$ , and  $F_1F_2$ . The relative velocity

components resolved along, and perpendicular to the lines-of-sight are as follows:

$$v_{r1} = r_1' = v_L c(\alpha_L - \theta_1) - v_1 c(\alpha_1 - \theta_1) \quad (3)$$

$$v_{\theta 1} = r_1 \theta_1' = v_L s(\alpha_L - \theta_1) - v_1 s(\alpha_1 - \theta_1) \quad (4)$$

$$v_{r2} = r_2' = v_L c(\alpha_L - \theta_2) - v_2 c(\alpha_2 - \theta_2) \quad (5)$$

$$v_{\theta 2} = r_2 \theta_2' = v_L s(\alpha_L - \theta_2) - v_2 s(\alpha_2 - \theta_2) \quad (6)$$

$$v_{r1}' = \theta_1' v_{\theta 1} - c(\delta_1 - \theta_1) a_1 \quad (7)$$

$$v_{\theta_1}' = -\theta'v_{r_1} - s(\delta_1 - \theta_1)a_1 \quad (8)$$

$$v_{r_2}' = \theta'v_{\theta_2} - c(\delta_2 - \theta_2)a_2 \quad (9)$$

$$v_{\theta_2}' = -\theta'v_{r_2} - s(\delta_2 - \theta_2)a_2 \quad (10)$$

The nonlinear kinematic state equations governing the lines-of-sight can be written in the form  $x' = f(x,u)$  as follows:

$$\begin{bmatrix} r_1 \\ \theta_1 \\ v_{\theta_1} \\ v_{r_1} \\ r_2 \\ \theta_2 \\ v_{\theta_2} \\ v_{r_2} \end{bmatrix} = \begin{bmatrix} v_{r_1} \\ \frac{v_{\theta_1}}{r_1} \\ -\frac{(v_{r_1}v_{\theta_1})}{r_1} - s(\delta_1 - \theta_1)a_1 \\ \frac{v_{\theta_1}^2}{r_1} - c(\delta_1 - \theta_1)a_1 \\ v_{r_2} \\ \frac{v_{\theta_2}}{r_2} \\ -\frac{(v_{r_2}v_{\theta_2})}{r_2} - s(\delta_2 - \theta_2)a_2 \\ \frac{v_{\theta_2}^2}{r_2} - c(\delta_2 - \theta_2)a_2 \end{bmatrix} \quad (11)$$

where  $a_1$  and  $a_2$  are the control inputs which are the accelerations of the two followers. The above equations assume the leader's acceleration to be zero, and they can be suitably modified when the leading acceleration is non-zero. The three lines-of-sight form the perimeter of a triangle as shown in Figure 1, with edges  $r_1$ ,  $r_2$ , and a third variable,  $c$ , which is found as:

$$c = [(r_1c(\theta_1) + r_2c(\theta_2))^2 + (r_1s(\theta_1) + r_2s(\theta_2))^2]^{(1/2)} \quad (12)$$

we define an output function  $y$  as the perimeter of this triangle.

$$y = r_1 + r_2 + c \quad (13)$$

While the leader sensor may or may not be accelerating along its trajectory, it is assumed that the follower sensors do not have knowledge of the leader's acceleration.

## Control Law Derivation

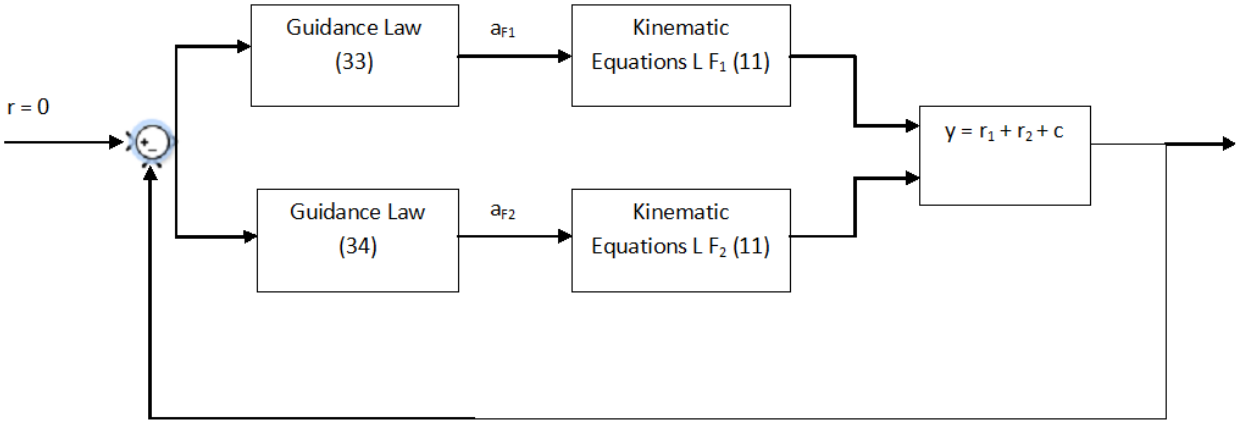


Figure 2 Block diagram of the control law used to control  $F_1$  and  $F_2$  for the circular sensing zone case.

To address this sensor coverage problem, we use nonlinear dynamic inversion to find control inputs  $a_1$  and  $a_2$  such that  $y$  is driven to zero.

Dynamic inversion [25] [26] will be used to find the control inputs  $a_1$  and  $a_2$ . For a system

$$\dot{x}' = f(x,u) \quad (14)$$

that can be described in an input affine form as:

$$\dot{x}' = f(x) + g(x)u \quad (15)$$

with an output

$$y = h(x) \quad (16)$$

define the tracking error

$$e = r_{\text{ref}} - y \quad (17)$$

then

$$e' = r_{\text{ref}}' - y' \quad (18)$$

both  $r_{\text{ref}}$  and  $r_{\text{ref}}'$  are zero for this sensor coverage problem

$$e' = -ky, \text{ where } k > 0 \quad (19)$$

$$y' = -ky \quad (20)$$

Where  $k$  is the feedback control gain and can be used to control how aggressively the controller drives  $e'$  to zero. The output,  $y$ , is differentiated until the control input term,  $u$ , appears. If the control input appears in the first derivative of  $y$ , and has the form:

$$y' = -ky = \nabla h(f(x)+g(x)u) = L_f h(x) + L_g h(x)u \quad (21)$$

where for this sensor coverage problem:

$$\nabla = [\partial h/\partial r_1 \quad \partial h/\partial \theta_1 \quad \partial h/\partial v_{\theta 1} \quad \partial h/\partial v_{r1} \quad \partial h/\partial r_1 \quad \partial h/\partial \theta_1 \quad \partial h/\partial v_{\theta 1} \quad \partial h/\partial v_{r1}] \quad (22)$$

If the number of control inputs is equal to the number of system outputs, then the system is said to be square and the control input,  $u$ , can be solved directly as follows:

$$u = 1/(L_g h)(-L_f h - ky) \quad (23)$$

For this sensor coverage problem the output,  $y$ , is a perimeter formed by the three sensors, and is a scalar, but there are two control inputs,  $a_1$  and  $a_2$ , so the system is not square. For the non square system, the Moore-Penrose Pseudo-Inverse [27] of  $L_g h(x)$  can be used to solve for the control inputs. For this sensor coverage problem where  $y'$  and  $L_f h(x)$  are both scalar, and  $L_g h(x)$  is a  $1 \times 2$  matrix and  $u$  is a  $2 \times 1$  vector, the pseudo inverse is:

$$L_g h^+(x) = L_g h^T(x) [L_g h(x) L_g h^T(x)]^{-1} \quad (24)$$

then the control inputs  $u$  can be determined as:

$$u = L_g h^+(x) [-L_f h(x) - ky] \quad (25)$$

For the sensor coverage problem, the control inputs appear after the second differentiation of  $y$  with respect to time.

$$e'' = r_{ref}'' - y'' \quad (26)$$

For this problem,  $r_{ref}''$  is zero,

$$e'' = -k_1 e' - k_2 e \text{ where } k_1 > 0 \text{ and } k_2 > 0 \quad (27)$$

then

$$y'' = -k_1 y' - k_2 y = L_f^2 h(x) + L_g^2 h(x) u \quad (28)$$

then for this non square problem

$$L_g^2 h^+(x) = L_g^2 h^T(x) [L_g^2 h(x) L_g^2 h^T(x)]^{-1} \quad (29)$$

$$u = L_g^2 h^+(x) (-k_1 y' - k_2 y - L_f^2 h(x)) \quad (30)$$

The first differentiation yields:

$$y' = v_{r1} + v_{r2} + (2(c(\theta_2)r_2 - c(\theta_1)r_1)(c(\theta_2)v_{r2} - c(\theta_1)v_{r1} - s(\theta_2)r_2 v_{\theta 2}/r_2 + s(\theta_1)r_1 v_{\theta 1}/r_1) + 2(s(\theta_2)r_2 s(\theta_1)r_1)(s(\theta_2)v_{r2} - s(\theta_1)v_{r1} + c(\theta_2)r_2 v_{\theta 2}/r_2 - c(\theta_1)r_1 v_{\theta 1}/r_1))/(2((s(\theta_2)r_2 - s(\theta_1)r_1)^2 + (c(\theta_2)r_2 - c(\theta_1)r_1)^2)^{(1/2)})) \quad (31)$$

and the second differentiation yields:

$$\begin{aligned} y'' = & v_{\theta 2}^2/r_2 + v_{\theta 1}^2/r_1 - a_1 c(\delta_1 - \theta_1) - a_2 c(\delta_2 - \theta_2) - ((2v_{r2}s(\theta_2) - 2r_1s(\theta_1))(s(\theta_1)(v_{\theta 1}^2/r_1 - a_1 c(\delta_1 - \theta_1)) - s(\theta_2)(v_{\theta 2}^2/r_2 - a_2 c(\delta_2 - \theta_2))) + a_1 s(\delta_1 - \theta_1) + a_2 s(\delta_2 - \theta_2) + (v_{\theta 2}^2 s(\theta_2))/r_2 - (v_{\theta 1}^2 s(\theta_1))/r_1) + \\ & (2v_{r2}c(\theta_2) - 2r_1c(\theta_1))(c(\theta_1)(v_{\theta 1}^2/r_1 - a_1 c(\delta_1 - \theta_1)) - c(\theta_2)(v_{\theta 2}^2/r_2 - a_2 c(\delta_2 - \theta_2))) + a_1 s(\delta_1 - \theta_1) + a_2 s(\delta_2 - \theta_2) + (v_{\theta 2}^2 c(\theta_2))/r_2 - (v_{\theta 1}^2 c(\theta_1))/r_1 - (v_{\theta 2}c(\theta_2) - v_{\theta 1}c(\theta_1) + v_{r2}s(\theta_2) - v_{r1}s(\theta_1))(2s(\theta_2)(v_{\theta 2}^2/r_2 - a_2 c(\delta_2 - \theta_2)) - 2v_{\theta 1}c(\theta_1) - 2v_{r1}s(\theta_1) + (2v_{\theta 2}v_{r2}c(\theta_2))/r_2) - (v_{r2}c(\theta_2) - v_{r1}c(\theta_1) - v_{\theta 2}s(\theta_2) + v_{\theta 1}s(\theta_1))(2c(\theta_2)(v_{\theta 2}^2/r_2 - a_2 c(\delta_2 - \theta_2)) - 2v_{r1}c(\theta_1) + 2v_{\theta 1}s(\theta_1) - (2v_{\theta 2}v_{r2}s(\theta_2))/r_2))/2((v_{r2}c(\theta_2) - r_1c(\theta_1))^2 + (v_{r2}s(\theta_2) - r_1s(\theta_1))^2)^{(1/2)}) - (((2v_{r2}c(\theta_2) - 2r_1c(\theta_1))(v_{r2}c(\theta_2) - v_{r1}c(\theta_1) - v_{\theta 2}s(\theta_2) + v_{\theta 1}s(\theta_1)) + (2v_{r2}s(\theta_2) - 2r_1s(\theta_1))(v_{\theta 2}c(\theta_2) - v_{\theta 1}c(\theta_1) + v_{r2}s(\theta_2) - v_{r1}s(\theta_1)))(2v_{r2}c(\theta_2) - 2r_1c(\theta_1))(c(\theta_2)(v_{\theta 2}^2/r_2 - a_2 c(\delta_2 - \theta_2)) - v_{r1}c(\theta_1) + v_{\theta 1}s(\theta_1) - (v_{\theta 2}v_{r2}s(\theta_2))/r_2) + (2v_{r2}s(\theta_2) - 2r_1s(\theta_1))(s(\theta_2)(v_{\theta 2}^2/r_2 - a_2 c(\delta_2 - \theta_2)) - v_{\theta 1}c(\theta_1) - v_{r1}s(\theta_1) + (v_{\theta 2}v_{r2}c(\theta_2))/r_2)))/4((v_{r2}c(\theta_2) - r_1c(\theta_1))^2 + (v_{r2}s(\theta_2) - r_1s(\theta_1))^2)^{(3/2)}) \quad (32) \end{aligned}$$

Next we solve for the acceleration control inputs  $a_1$  and  $a_2$  using equation (30)

$$\begin{aligned}
a_1 = & ((c(\theta_1 - \delta_1) + ((s(\theta_1 - \delta_1) + s(\theta_1)c(\theta_1 - \delta_1))(2s(\theta_1)r_1 - 2s(\theta_2)r_2) + (s(\theta_1 - \delta_1) + c(\theta_1)c(\theta_1 - \\
& \delta_1))(2c(\theta_1)r_1 - 2c(\theta_2)r_2)))/(2(r_1^2 - 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(1/2)}))(k_1(v_{r1} + v_{r2} + 2r_1v_{r1} + 2r_2v_{r2} - r_1v_{r2}c(\theta_1 - \\
& \theta_2) - v_{r1}r_2c(\theta_1 - \theta_2) + s(\theta_1 - \theta_2)r_1r_2(v_{\theta_1}/r_1 - v_{\theta_2}/r_2)) - ((c(\theta_1)v_{\theta_1} - c(\theta_2)v_{\theta_2} + s(\theta_1)v_{r1} - \\
& s(\theta_2)v_{r2})(2c(\theta_1)v_{\theta_1} - 2c(\theta_2)v_{\theta_2} + 2s(\theta_1)v_{r1} - 2s(\theta_2)v_{r2}) - (c(\theta_1)v_{r1} - c(\theta_2)v_{r2} - s(\theta_1)v_{\theta_1} + \\
& s(\theta_2)v_{\theta_2})(2c(\theta_1)v_{r1} - 2c(\theta_2)v_{r2} - 2s(\theta_1)v_{\theta_1} + 2s(\theta_2)v_{\theta_2}))/2(r_1^2 - 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(1/2)} + k_2(r_1 + r_2 \\
& + ((s(\theta_1)r_1 + s(\theta_2)r_2)^2 + (c(\theta_1)r_1 + c(\theta_2)r_2)^2)^{(1/2)} + v_{\theta_1}^2/r_1 + v_{\theta_2}^2/r_2 - (r_1v_{r1} + r_2v_{r2} - r_1v_{r2}c(\theta_1 - \theta_2) - \\
& v_{r1}r_2c(\theta_1 - \theta_2) - s(\theta_1 - \theta_2)r_1v_{\theta_2} + s(\theta_1 - \theta_2)v_{\theta_1}r_2)^2/(r_1^2 - 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(3/2)}))/((c(\theta_1 - \delta_1) + ((s(\theta_1 \\
& - \delta_1) + s(\theta_1)c(\theta_1 - \delta_1))(2s(\theta_1)r_1 - 2s(\theta_2)r_2) + (s(\theta_1 - \delta_1) + c(\theta_1)c(\theta_1 - \delta_1))(2c(\theta_1)r_1 - 2c(\theta_2)r_2)))/(2(r_1^2 - \\
& 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(1/2)}))^2 + (c(\theta_2 - \delta_2) + ((s(\theta_2 - \delta_2) + s(\theta_2)c(\theta_2 - \delta_2))(2s(\theta_1)r_1 - 2s(\theta_2)r_2) + (s(\theta_2 - \\
& \delta_2) + c(\theta_2)c(\theta_2 - \delta_2))(2c(\theta_1)r_1 - 2c(\theta_2)r_2)))/(2(r_1^2 - 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(1/2)}))^2) \quad (33)
\end{aligned}$$

$$\begin{aligned}
a_2 = & ((c(\theta_2 - \delta_2) + ((s(\theta_2 - \delta_2) + s(\theta_2)c(\theta_2 - \delta_2))(2s(\theta_1)r_1 - 2s(\theta_2)r_2) + (s(\theta_2 - \delta_2) + c(\theta_2)c(\theta_2 - \\
& \delta_2))(2c(\theta_1)r_1 - 2c(\theta_2)r_2)))/(2(r_1^2 - 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(1/2)}))(k_1(v_{r1} + v_{r2} + 2r_1v_{r1} + 2r_2v_{r2} - r_1v_{r2}c(\theta_1 - \\
& \theta_2) - v_{r1}r_2c(\theta_1 - \theta_2) + s(\theta_1 - \theta_2)r_1r_2(v_{\theta_1}/r_1 - v_{\theta_2}/r_2)) - ((c(\theta_1)v_{\theta_1} - c(\theta_2)v_{\theta_2} + s(\theta_1)v_{r1} - \\
& s(\theta_2)v_{r2})(2c(\theta_1)v_{\theta_1} - 2c(\theta_2)v_{\theta_2} + 2s(\theta_1)v_{r1} - 2s(\theta_2)v_{r2}) - (c(\theta_1)v_{r1} - c(\theta_2)v_{r2} - s(\theta_1)v_{\theta_1} + \\
& s(\theta_2)v_{\theta_2})(2c(\theta_1)v_{r1} - 2c(\theta_2)v_{r2} - 2s(\theta_1)v_{\theta_1} + 2s(\theta_2)v_{\theta_2}))/2(r_1^2 - 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(1/2)} + k_2(r_1 + r_2 + \\
& ((s(\theta_1)r_1 + s(\theta_2)r_2)^2 + (c(\theta_1)r_1 + c(\theta_2)r_2)^2)^{(1/2)} + v_{\theta_1}^2/r_1 + v_{\theta_2}^2/r_2 - (r_1v_{r1} + r_2v_{r2} - r_1v_{r2}c(\theta_1 - \theta_2) - \\
& v_{r1}r_2c(\theta_1 - \theta_2) - s(\theta_1 - \theta_2)r_1v_{\theta_2} + s(\theta_1 - \theta_2)v_{\theta_1}r_2)^2/(r_1^2 - 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(3/2)}))/((c(\theta_1 - \delta_1) + ((s(\theta_1 \\
& - \delta_1) + s(\theta_1)c(\theta_1 - \delta_1))(2s(\theta_1)r_1 - 2s(\theta_2)r_2) + (s(\theta_1 - \delta_1) + c(\theta_1)c(\theta_1 - \delta_1))(2c(\theta_1)r_1 - 2c(\theta_2)r_2)))/(2(r_1^2 - \\
& 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(1/2)}))^2 + (c(\theta_2 - \delta_2) + ((s(\theta_2 - \delta_2) + s(\theta_2)c(\theta_2 - \delta_2))(2s(\theta_1)r_1 - 2s(\theta_2)r_2) + (s(\theta_2 - \\
& \delta_2) + c(\theta_2)c(\theta_2 - \delta_2))(2c(\theta_1)r_1 - 2c(\theta_2)r_2)))/(2(r_1^2 - 2c(\theta_1 - \theta_2)r_1r_2 + r_2^2)^{(1/2)}))^2) \quad (34)
\end{aligned}$$

then when the accelerations  $a_1$  and  $a_2$  are applied to the  $F_1$  and  $F_2$  sensors respectively, the sensors will move in a coordinated fashion. As the perimeter of the triangle formed by the three sensors decreases, the coverage gap decreases. When the three sensors are close enough such that all three sensing circles overlap, then the coverage gap is defined by a Hyperbolic triangle [28], as shown in Figure 3.

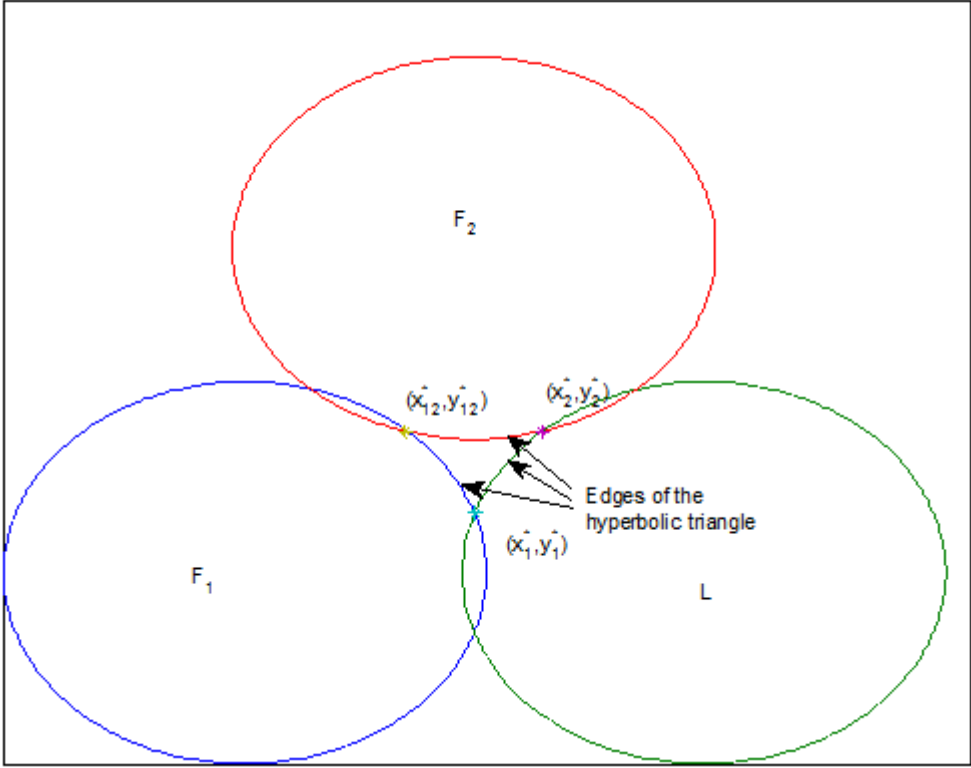
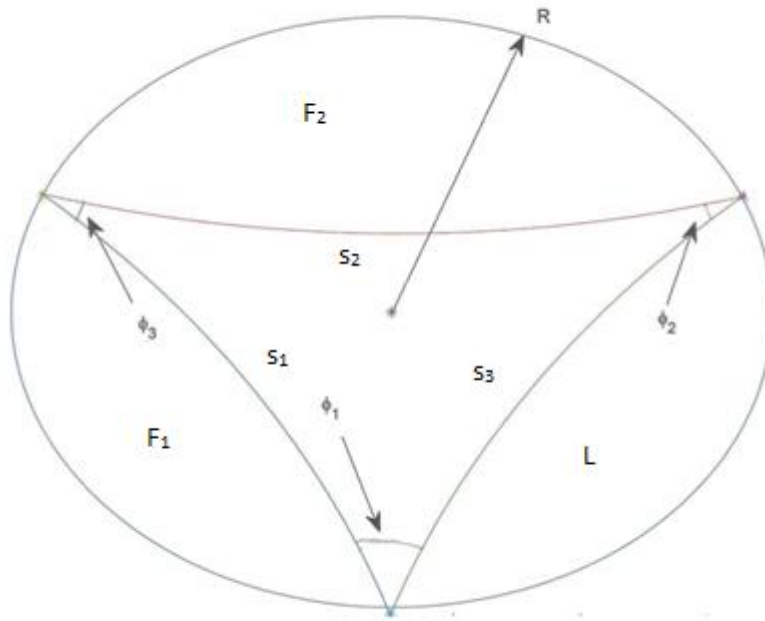


Figure 3 Hyperbolic triangle formed when the coverage area of the three sensors overlap.

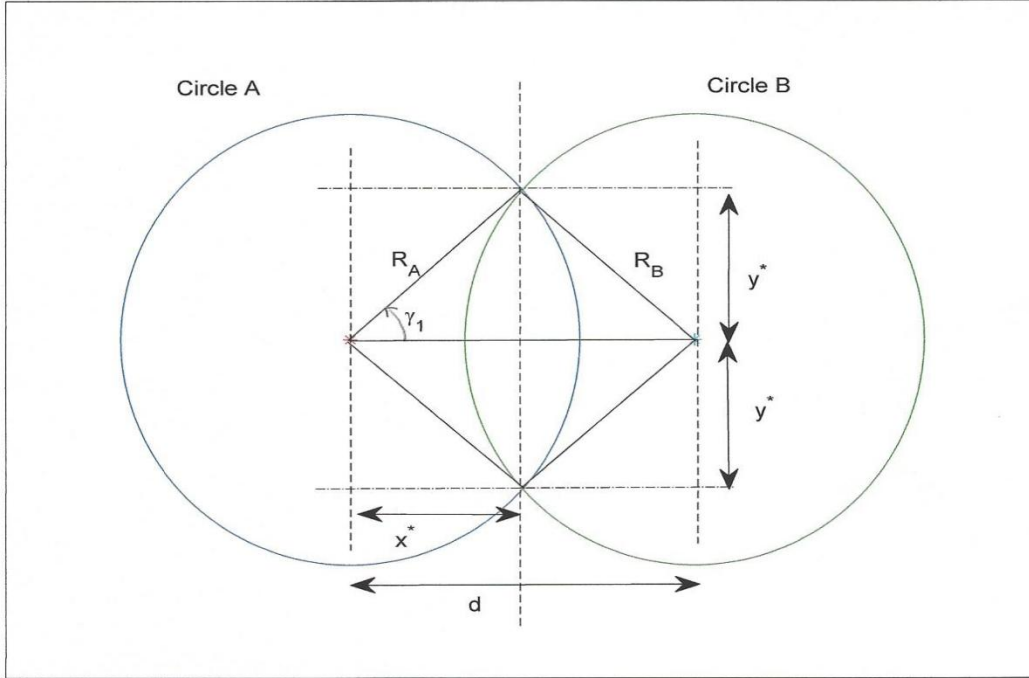


**Figure 4 Interior angles of the hyperbolic triangle, the circle that touches the vertices of the hyperbolic triangle, and radius of the circle.**

A hyperbolic triangle is a non Euclidian triangle where the three edges are convex curves rather than straight lines as seen in Figure 3. The three angles of a hyperbolic triangle always add up to less than 180 degrees [29], and area of a hyperbolic triangle is found from:

$$\text{area} = R^2 (\pi - \varphi_1 - \varphi_2 - \varphi_3) \quad (35)$$

where R is the radius of the circle that passes through all three vertices,  $\varphi_1$ ,  $\varphi_2$ , and  $\varphi_3$  are the internal angles of the hyperbolic triangle as seen in Figure 4. The angles  $\varphi_1$ ,  $\varphi_2$ , and  $\varphi_3$  need to be found as a function of the state variables. As a first step, we need to find the vertices of the hyperbolic triangle. For two circles as shown in Figure 5, there are two points where the circles intersect.



**Figure 5** Geometry of the intersection points between two circles.

The x and y coordinates of the intersections points, relative to the vertex of Circle<sub>A</sub> are:

$$x^* = R_A \cos(\gamma_1) \quad (36)$$

$$y^* = R_A \sin(\gamma_1) \quad (37)$$

from the Law of Cosines:

$$\gamma_1 = \cos^{-1}((-R_B^2 + d^2 + R_A^2)/(2dR_A)) \quad (38)$$

solving for  $x^*$

$$x^* = (-R_B^2 + d^2 + R_A^2)/(2d) \quad (39)$$

from the Pythagorean Theorem:

$$R_A^2 = x^{*2} + y^{*2} \quad (40)$$

solving for  $y^*$

$$y^* = \pm(R_A^2 - ((-R_B^2 + d^2 + R_A^2)/(2d))^2)^{1/2} \quad (41)$$

next we find the coordinates of the three vertices of the hyperbolic triangle  $(x^*_1, y^*_1)$ ,  $(x^*_2, y^*_2)$ , and  $(x^*_{12}, y^*_{12})$  in the inertial reference frame. We also need to define  $\theta_{12}$ , the line of sight angle between  $F_1$  and  $F_2$ .

$$\theta_{12} = \tan^{-1}((y_{F2} - y_{F1})/(x_{F2} - x_{F1})) \quad (42)$$

where  $x_{F1}$ ,  $x_{F2}$ ,  $y_{F1}$ , and  $y_{F2}$  are the x and y coordinates of the vertices of  $F_1$  and  $F_2$ . We solve for the  $x^*$  and  $y^*$  coordinates between the two sensors, and then rotate the coordinate axis about line of sight angle between the two sensors to find the inertial coordinates. There are two points of intersect between each sensor, denoted as  $x^*_{1a}$ ,  $y^*_{1a}$ ,  $x^*_{1b}$ ,  $y^*_{1b}$ ,  $x^*_{2a}$ ,  $y^*_{2a}$ ,  $x^*_{2b}$ ,  $y^*_{2b}$ ,  $x^*_{12a}$ ,  $y^*_{12a}$ ,  $x^*_{12b}$ , and  $y^*_{12b}$ ,

$$x^*_{1a} = (-R_3^2 + r_1^2 + R_1^2)/(2r_1)c(\theta_1) - (R_1^2 + ((-R_3^2 + r_1^2 + R_1^2)/(2r_1))^2)^{1/2}s(\theta_1) + x_{F1} \quad (43)$$

$$y^*_{1a} = (-R_3^2 + r_1^2 + R_1^2)/(2r_1)s(\theta_1) + (R_1^2 + ((-R_3^2 + r_1^2 + R_1^2)/(2r_1))^2)^{1/2}c(\theta_1) + y_{F1} \quad (44)$$

$$x^*_{1b} = (-R_3^2 + r_1^2 + R_1^2)/(2r_1)c(\theta_1) + (R_1^2 + ((-R_3^2 + r_1^2 + R_1^2)/(2r_1))^2)^{1/2}s(\theta_1) + x_{F1} \quad (45)$$

$$y^*_{1b} = (-R_3^2 + r_1^2 + R_1^2)/(2r_1)s(\theta_1) - (R_1^2 + ((-R_3^2 + r_1^2 + R_1^2)/(2r_1))^2)^{1/2}c(\theta_1) + y_{F1} \quad (46)$$

$$x^*_{2a} = (-R_3^2 + r_2^2 + R_2^2)/(2r_2)c(\theta_2) - (R_2^2 + ((-R_3^2 + r_2^2 + R_2^2)/(2r_2))^2)^{1/2}s(\theta_2) + x_{F2} \quad (47)$$

$$y^*_{2a} = (-R_3^2 + r_2^2 + R_2^2)/(2r_2)s(\theta_2) + (R_2^2 + ((-R_3^2 + r_2^2 + R_2^2)/(2r_2))^2)^{1/2}c(\theta_2) + y_{F2} \quad (48)$$

$$x^*_{2b} = (-R_3^2 + r_2^2 + R_2^2)/(2r_2)c(\theta_2) + (R_2^2 + ((-R_3^2 + r_2^2 + R_2^2)/(2r_2))^2)^{1/2}s(\theta_2) + x_{F2} \quad (49)$$

$$y^*_{2b} = (-R_3^2 + r_2^2 + R_2^2)/(2r_2)s(\theta_2) - (R_2^2 + ((-R_3^2 + r_2^2 + R_2^2)/(2r_2))^2)^{1/2}c(\theta_2) + y_{F2} \quad (50)$$

$$x^*_{12a} = (-R_3^2 + c^2 + R_1^2)/(2r_1)c(\theta_1) - (R_1^2 + ((-R_2^2 + c^2 + R_1^2)/(2c))^2)^{1/2}s(\theta_1) + x_{F12} \quad (51)$$

$$y^*_{12a} = (-R_3^2 + c^2 + R_1^2)/(2r_1)s(\theta_1) + (R_1^2 + ((-R_2^2 + c^2 + R_1^2)/(2c))^2)^{1/2}c(\theta_1) + y_{F12} \quad (52)$$

$$x^*_{12b} = (-R_3^2 + c^2 + R_1^2)/(2r_1)c(\theta_1) + (R_1^2 + ((-R_2^2 + c^2 + R_1^2)/(2c))^2)^{1/2}s(\theta_1) + x_{F12} \quad (53)$$

$$y^*_{12b} = (-R_3^2 + c^2 + R_1^2)/(2r_1)s(\theta_1) - (R_1^2 + ((-R_2^2 + c^2 + R_1^2)/(2c))^2)^{1/2}c(\theta_1) + y_{F12} \quad (54)$$

where  $R_1$  is the sensing radius of the  $F_1$  sensor,  $R_2$  is the sensing radius of the  $F_2$  sensor, and  $R_3$  is the sensing radius of the Leader sensor. The relative positioning of the sensors determines which

intersection points comprise the vertices of the hyperbolic triangle. The combination of points that give the smallest perimeter will be the vertices of the hyperbolic triangle,  $(x^*_1, y^*_1)$ ,  $(x^*_2, y^*_2)$ , and  $(x^*_{12}, y^*_{12})$ . To find angles  $\varphi_1$ ,  $\varphi_2$ , and  $\varphi_3$ , we need to calculate the arc lengths which make up the edges of the hyperbolic triangle as seen in Figure 3. Once the arc lengths are known, we can use the Hyperbolic Law of Sines, and the Hyperbolic Law of Cosines [28] to solve for angles  $\varphi_1$ ,  $\varphi_2$ , and  $\varphi_3$ .

The Hyperbolic Law of Sines is:

$$s(\varphi_1)/\sinh(s_1) = s(\varphi_2)/\sinh(s_2) = s(\varphi_3)/\sinh(s_3) \quad (55)$$

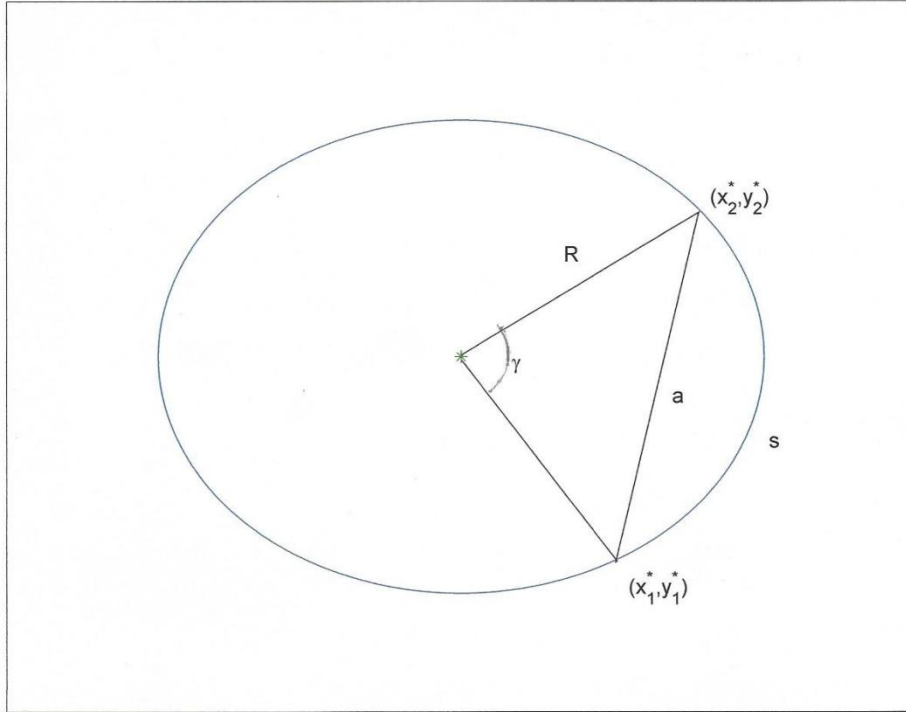
and the Hyperbolic Law of Cosines, which has two parts:

$$\cosh(s_3) = \cosh(s_1)\cosh(s_2) - \sinh(s_1)\sinh(s_2)\c(\varphi_3) \quad (56)$$

and

$$\cosh(s_3) = (\c(\varphi_1)\c(\varphi_2) + \c(\varphi_3))/(s(\varphi_1)s(\varphi_2)) \quad (57)$$

where  $s_1$  is the arc length opposite  $\varphi_1$ ,  $s_2$  is the arc length opposite  $\varphi_2$ , and  $s_3$  is the arc length opposite  $\varphi_3$ . To calculate the arc length, [30] we look at two points on the perimeter of a circle as in Figure 6.



**Figure 6 Geometry of a circle with two points on the perimeter, the chord length and the arc length between the two points.**

The straight line distance between two points on a circle, the chord length  $a$  is a function of the angle  $\gamma$  and  $R$  in Figure 6

$$a = 2R\sin(\gamma/2) = ((x_2-x_1)^2 + (y_2-y_1)^2)^{1/2} \quad (58)$$

then

$$\gamma = 2\sin^{-1}(((x_2-x_1)^2 + (y_2-y_1)^2)^{1/2}/(2R)) \quad (59)$$

and the arc length

$$s = R\gamma \quad (60)$$

$$s = 2R\sin^{-1}(((x_2-x_1)^2 + (y_2-y_1)^2)^{1/2}/(2R)) \quad (61)$$

Next we use equation (61) to find  $s_1$ ,  $s_2$ , and  $s_3$

$$s_1 = 2R_1\sin^{-1}(((x_{12}^*-x_1^*)^2 + (y_{12}^*-y_1^*)^2)^{1/2}/(2R_1)) \quad (62)$$

$$s_2 = 2R_2\sin^{-1}(((x_{12}^*-x_2^*)^2 + (y_{12}^*-y_2^*)^2)^{1/2}/(2R_2)) \quad (63)$$

$$s_3 = 2R_3\sin^{-1}(((x_{21}^*-x_1^*)^2 + (y_{21}^*-y_1^*)^2)^{1/2}/(2R_3)) \quad (64)$$

then using equation (56) to solve for  $\varphi_3$

$$\varphi_3 = \cos^{-1}(-(\cosh(s_3) - \cosh(s_1)\cosh(s_2))/(\sinh(s_1)\sinh(s_2))) \quad (65)$$

then using equation (55)

$$\varphi_1 = \sin^{-1}(\sinh(s_1)s(\varphi_3)/\sinh(s_3)) \quad (66)$$

$$\varphi_2 = \sin^{-1}(\sinh(s_2)s(\varphi_3)/\sinh(s_3)) \quad (67)$$

the last piece of the hyperbolic area equation is the radius R of the circle that passes through each vertex of the hyperbolic triangle. For a triangle with sides a, b, and c, the diameter of the circle that touches each vertex of the triangle is: [31]

$$\text{diameter} = abc/(2 * \text{area of the triangle}) \quad (68)$$

Heron's Formula can be used to calculate the area of a triangle using only the edge lengths: [32]

$$A = 1/4*(4a^2b^2-(a^2+b^2-c^2)^2) \quad (69)$$

then the radius of the circle that touches all three vertices of the hyperbolic triangle is:

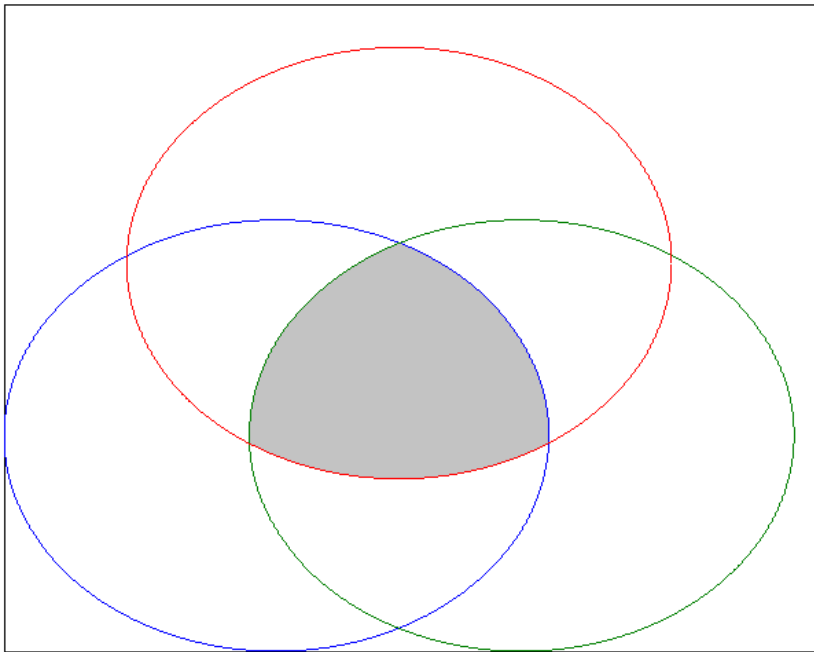
$$R = (((x_2^* - x_1^*)^2 + (y_2^* - y_1^*)^2)((x_2^* - x_3^*)^2 + (y_2^* - y_3^*)^2)((x_3^* - x_1^*)^2 + (y_3^* - y_1^*)^2))^{1/2} / (2|x_1^*y_2^* + x_2^*y_3^* + x_3^*y_1^* - x_1^*y_3^* - x_2^*y_1^* - x_3^*y_2^*|) \quad (70)$$

Using the state variables and calculating the vertices of the hyperbolic triangle formed by the three sensors, the hyperbolic triangle area is:

$$\begin{aligned} \text{area} = & (((x_2^* - x_1^*)^2 + (y_2^* - y_1^*)^2)((x_2^* - x_3^*)^2 + (y_2^* - y_3^*)^2)((x_3^* - x_1^*)^2 + (y_3^* - y_1^*)^2))^{1/2} / (2|x_1^*y_2^* + \\ & x_2^*y_3^* + x_3^*y_1^* - x_1^*y_3^* - x_2^*y_1^* - x_3^*y_2^*|)^2 (\pi - \sin^{-1}(\sinh(2R_1 \sin^{-1}(((x_{12}^* - x_1^*)^2 + (y_{12}^* - \\ & y_1^*)^2)^{1/2} / (2R_1))))s(\cos^{-1}(-(\cosh(2R_3 \sin^{-1}(((x_{23}^* - x_1^*)^2 + (y_{23}^* - y_1^*)^2)^{1/2} / (2R_3)))) - \cosh(2R_1 \sin^{-1}(((x_{12}^* - \\ & x_1^*)^2 + (y_{12}^* - y_1^*)^2)^{1/2} / (2R_1))))\cosh(2R_2 \sin^{-1}(((x_{12}^* - x_2^*)^2 + (y_{12}^* - y_2^*)^2)^{1/2} / (2R_2)))) / (\sinh(2R_1 \sin^{-1} \\ & 1(((x_{12}^* - x_1^*)^2 + (y_{12}^* - y_1^*)^2)^{1/2} / (2R_1))))\sinh(2R_2 \sin^{-1}(((x_{12}^* - x_2^*)^2 + (y_{12}^* - \\ & y_2^*)^2)^{1/2} / (2R_2)))))) / \sinh(2R_3 \sin^{-1}(((x_{23}^* - x_1^*)^2 + (y_{23}^* - y_1^*)^2)^{1/2} / (2R_3)))) - \sin^{-1}(\sinh(2R_2 \sin^{-1}(((x_{12}^* - x_2^*)^2 \\ & + (y_{12}^* - y_2^*)^2)^{1/2} / (2R_2))))s(\cos^{-1}(-(\cosh(2R_3 \sin^{-1}(((x_{23}^* - x_1^*)^2 + (y_{23}^* - y_1^*)^2)^{1/2} / (2R_3)))) - \cosh(2R_1 \sin^{-1} \end{aligned}$$

$$\begin{aligned}
& \sqrt{((x_{12}^* - x_1^*)^2 + (y_{12}^* - y_1^*)^2) / (2R_1)} \cosh(2R_2 \sin^{-1}(\sqrt{((x_{12}^* - x_2^*)^2 + (y_{12}^* - y_2^*)^2) / (2R_2)})) / (\sinh(2R_1 \sin^{-1}(\sqrt{((x_{12}^* - x_1^*)^2 + (y_{12}^* - y_1^*)^2) / (2R_1)})) \sinh(2R_2 \sin^{-1}(\sqrt{((x_{12}^* - x_2^*)^2 + (y_{12}^* - y_2^*)^2) / (2R_2)})))) / \sinh(2R_3 \sin^{-1}(\sqrt{((x_{12}^* - x_1^*)^2 + (y_{12}^* - y_1^*)^2) / (2R_3)})) - \cos^{-1}(-(\cosh(2R_3 \sin^{-1}(\sqrt{((x_{12}^* - x_1^*)^2 + (y_{12}^* - y_1^*)^2) / (2R_3)})) - \cosh(2R_1 \sin^{-1}(\sqrt{((x_{12}^* - x_1^*)^2 + (y_{12}^* - y_1^*)^2) / (2R_1)})) \cosh(2R_2 \sin^{-1}(\sqrt{((x_{12}^* - x_2^*)^2 + (y_{12}^* - y_2^*)^2) / (2R_2)})) / (\sinh(2R_1 \sin^{-1}(\sqrt{((x_{12}^* - x_1^*)^2 + (y_{12}^* - y_1^*)^2) / (2R_1)})) \sinh(2R_2 \sin^{-1}(\sqrt{((x_{12}^* - x_2^*)^2 + (y_{12}^* - y_2^*)^2) / (2R_2)})))))) \quad (71)
\end{aligned}$$

When  $r_1 < R_1 + R_3$ ,  $r_2 < R_2 + R_3$ , and  $c < R_1 + R_2$ , then coverage area of each sensor overlaps with the coverage area of the other two, then the sensors are orientated in a fashion similar to Figure 3, and the sensing radii are said to overlap. Before the three sensor sensing radii overlap, the hyperbolic area equation yields an imaginary result, when the area of the hyperbolic triangle area reaches zero the coverage gap has been closed. As the sensors continue to move together, the hyperbolic triangle changes to an elliptical triangle, as seen in Figure 7, and the hyperbolic triangle area equation above also yields an imaginary result.



**Figure 7** The shaded region is an elliptical triangle, with three convex curved edges.

There are two sets of stopping conditions, the first stopping condition is applied to the first follower sensor when it reaches the leader and their sensing radii overlap, the second stopping condition is reached when the sensing radii of each sensor overlaps and the hyperbolic area equation has an imaginary component meaning that there is no coverage hole as shown in Figure 8. A pseudo code for the stopping condition is:

```
if  $r_1 < R_1 + R_3$  and  $r_2 > R_2 + R_3$ 
```

```
    stop relative motion of  $F_1$ 
```

```
endif
```

```
if  $r_2 < R_2 + R_3$  and  $r_1 > R_1 + R_3$ 
```

```
    stop relative motion of  $F_2$ 
```

```
endif
```

```
if  $r_1 < R_1 + R_3$  and  $r_2 < R_2 + R_3$  and  $c < R_1 + R_2$  and the area of the hyperbolic triangle from equation (71) is imaginary
```

```
    stop relative motion of  $F_1$  and  $F_2$ 
```

```
else
```

```
     $a_1 =$  equation (33)
```

```
     $a_2 =$  equation (34)
```

```
endif
```

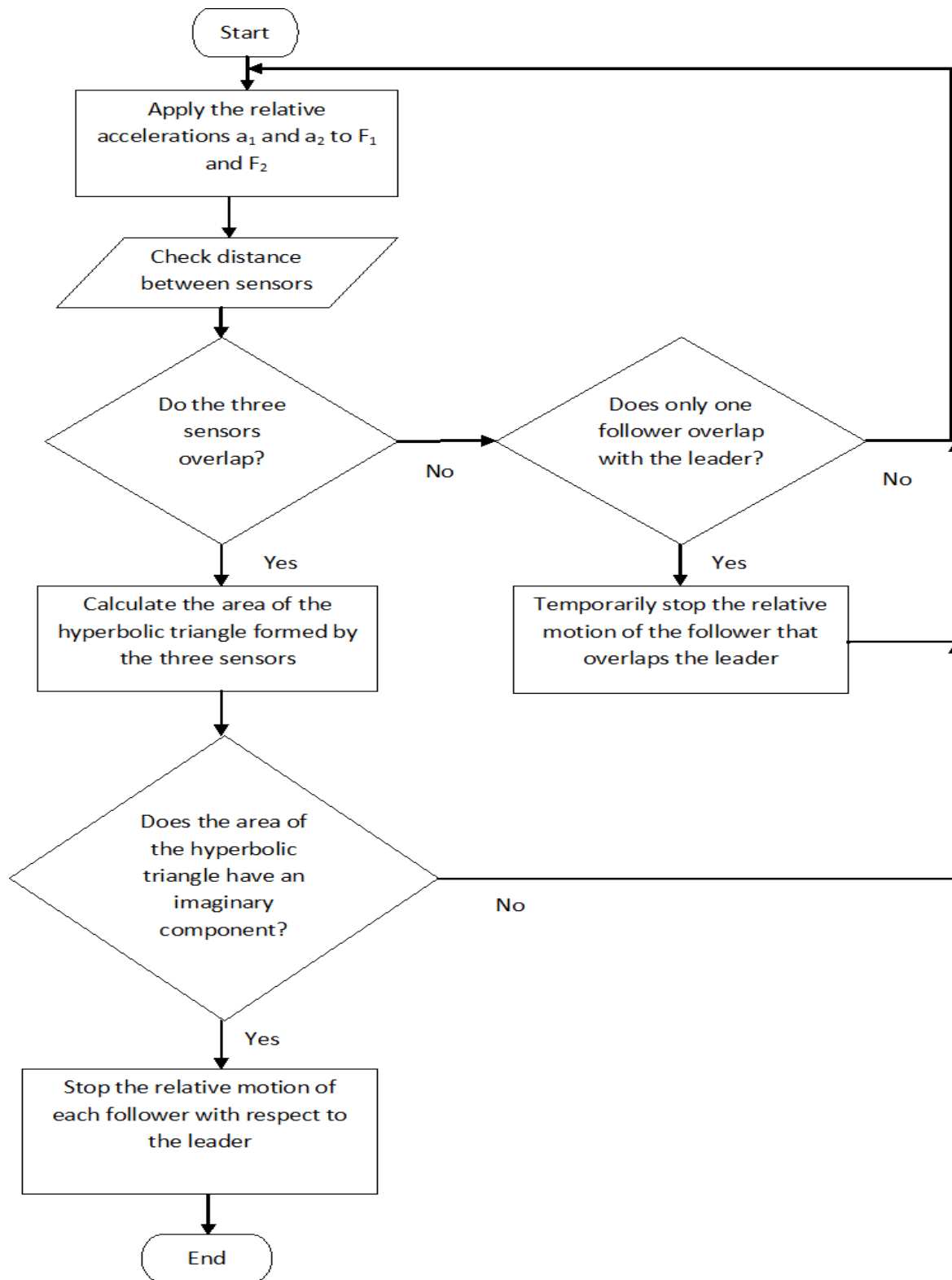


Figure 8 Control algorithm used for the circular sensor problem.

## CHAPTER 3

### CIRCULAR SENSOR SIMULATION RESULTS

To test the effectiveness of the control laws equations (33) and (34), simulations are performed. Simulations are run from several different initial positions with different initial velocities, and the results are that the coverage hole is closed in each case. In each case, the initial and final position and path of each sensor is plotted, and the commanded accelerations that are applied to both follower sensors, are shown

For the circular sensing radius problem, four test cases with various initial conditions are shown in Table 1. The leader is tracing an independent trajectory on the plane, and  $F_1$  and  $F_2$  are trying to close the coverage gap using equations (33) and (34).

Table 1  
Circular sensing radius test conditions

	Case 1	Case 2	Case 3	Case 4
Leader initial x position	15 m	15 m	15 m	15 m
Leader initial y position	18 m	18 m	18 m	18 m
Leader final x position	20.99 m	15	15 m	55.43 m
Leader final y position	25.95 m	18	18 m	-23.423 m
$r_1(0)$	18.3848 m	19.7231 m	19.7231 m	18.868 m
$r_2(0)$	18.868 m	13.1529 m	13.1529 m	18.3848 m
$\theta_1(0)$	.7854 rad	1.0391 rad	2.1025 rad	2.1295 rad
$\theta_2(0)$	2.1295 rad	-.1526 rad	-.1526 rad	.7854 rad
$v_{\theta_1}(0)$	0 m/s	.1 m/s	-.125 m/s	0 m/s
$v_{\theta_2}(0)$	0 m/s	.25 m/s	.11 m /s	0 m/s

Table 1 (continued)

$v_{r1}(0)$	-1 m/s	-1 m/s	-2 m/s	-1 m/s
$v_{r2}(0)$	-1 m/s	-1 m/s	-1 m/s	-1 m/s
$k_1$	1	1	1	1
$\delta_1$	0 rad	0 rad	0 rad	0 rad
$\delta_2$	0 rad	0 rad	0 rad	0 rad

The dynamic inversion control law is used to drive  $y$ , equation (13), towards zero, however in each case the stopping condition is reached before  $y$  equals zero. The only way  $y$  can equal zero is if the vertices of each sensor sit on top of each other, which is not desirable because the three sensors would all be sensing the same region and would have no benefit over using just one sensor. The stopping condition is reached when the coverage gap is closed, which always occurs before  $y$  reaches zero.

### Case 1

The leader moves on a path from an initial position. The leader does not follow a straight line on the path from its initial position to its final position.  $F_1$  begins below and to the left of the leader, and  $F_2$  begins below and to the right of the leader. Both followers move toward the leader, and as the trajectory of the leader changes the trajectories of the two followers also change in response. The control law is able to drive both followers towards the leader and close the coverage gap.

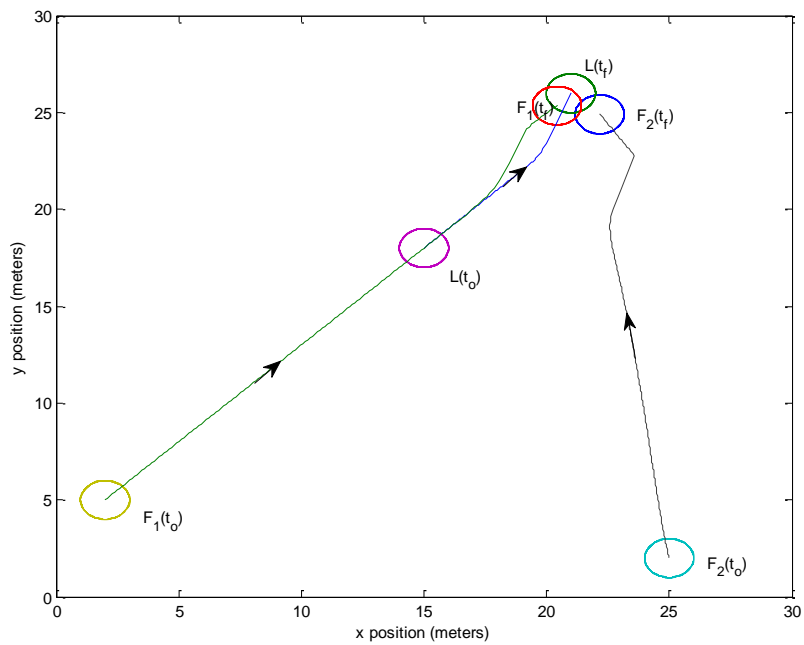


Figure 9 Case 1: Initial and final positions of each sensor. Solid lines indicate the trajectory of the sensors.

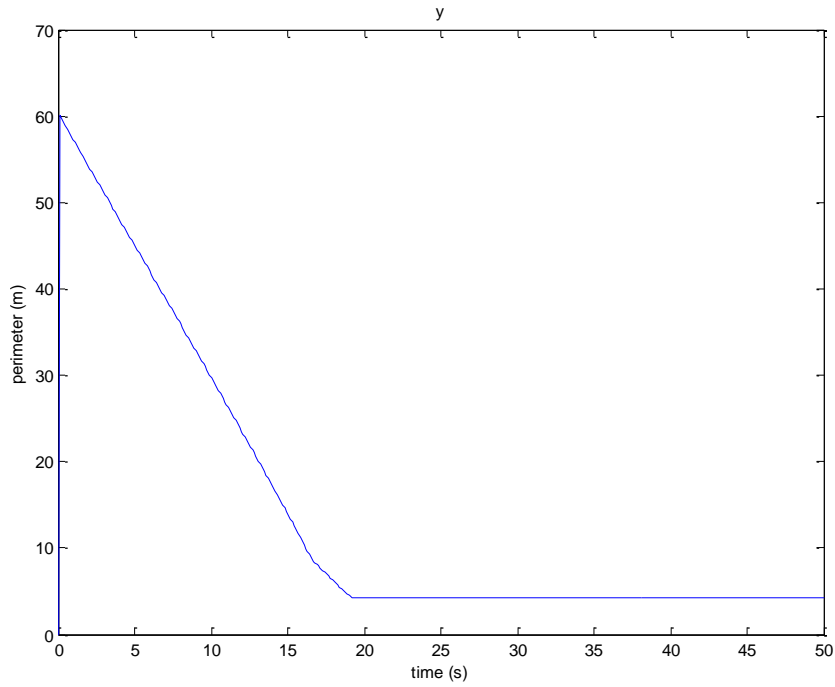


Figure 10 Case 1: Value of the output  $y$  that is used to develop the dynamic inversion control law.

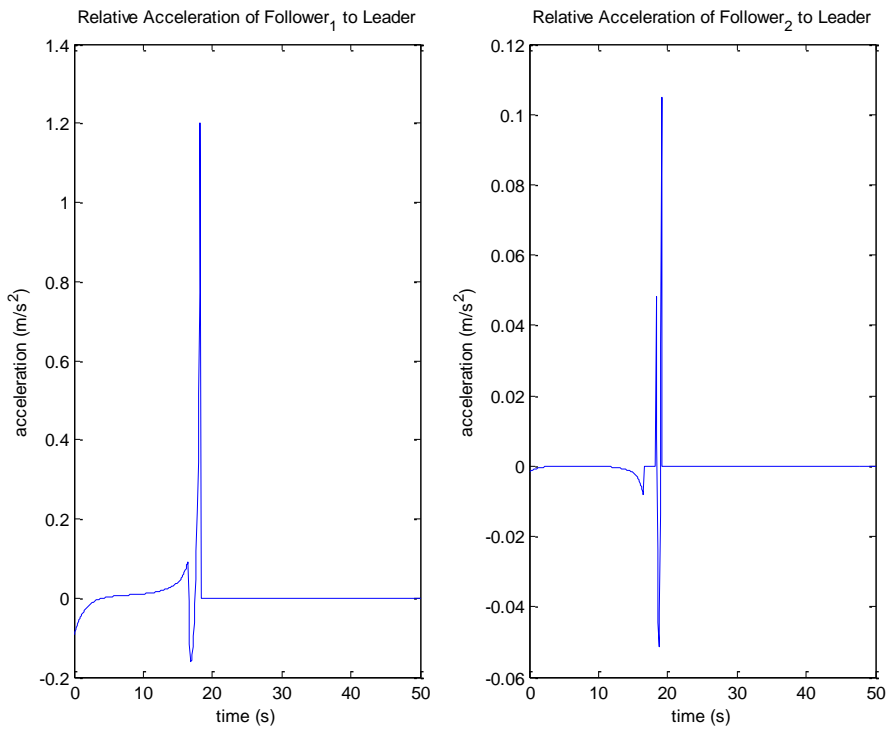


Figure 11 Case 1: Acceleration of Follower<sub>1</sub> and Follower<sub>2</sub>.

## Case 2

$F_1$  begins below and to the left of the leader, while  $F_2$  begins above and to the left of the leader. The leader is assumed to be stationary in this case. This is not limiting in any way since the follower accelerations, equations (33) and (34) have been determined in a relative velocity framework. Both followers begin with an initial velocity that does not point directly at the leader, but the control laws equations (33) and (34) are able to redirect the followers towards the leader and close the coverage gap.

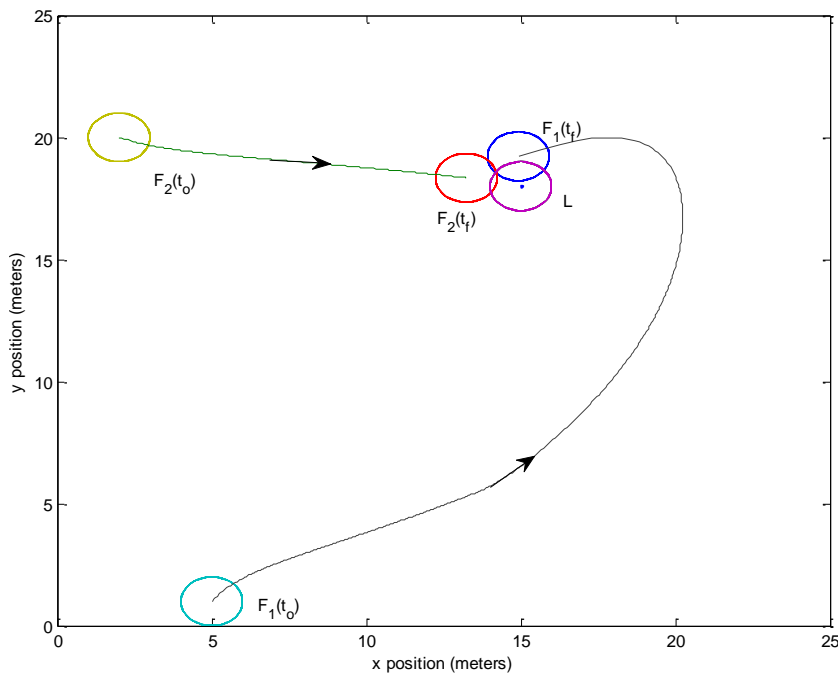


Figure 12 Case 2: Initial and final positions of each sensor. Solid lines indicate the trajectory of the sensors.

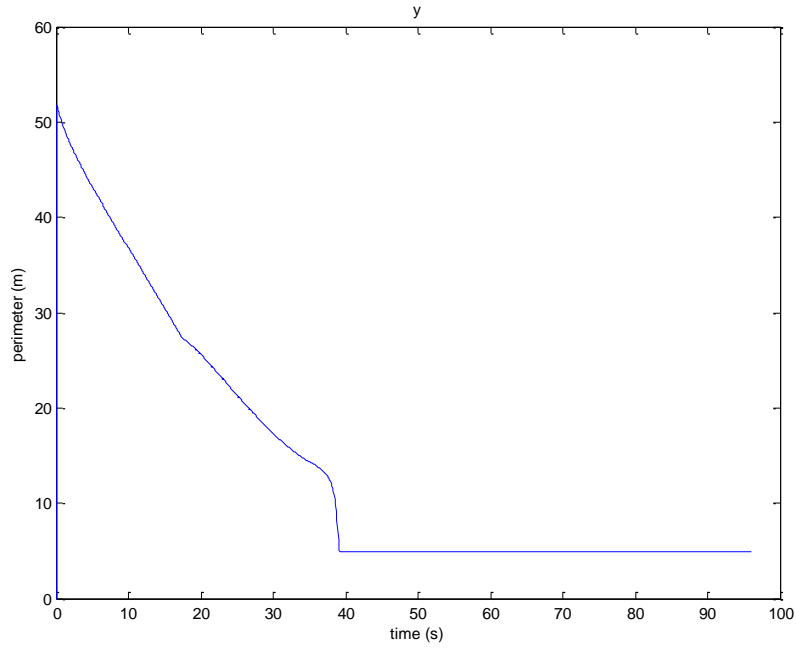


Figure 13 Case 2: Value of the output  $y$  that is used to develop the dynamic inversion control law.

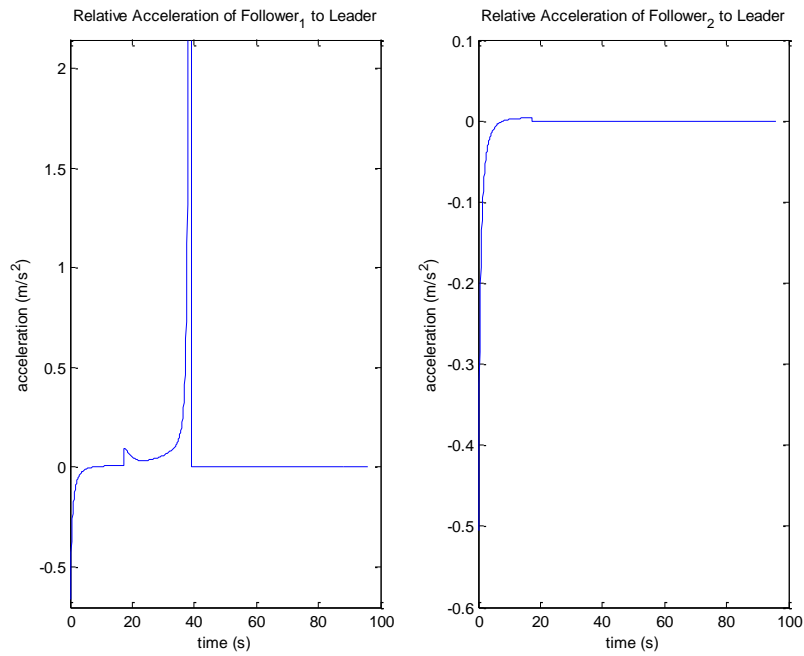


Figure 14 Case 2: Acceleration of Follower<sub>1</sub> and Follower<sub>2</sub>

### Case 3

The leader is again assumed to be stationary.  $F_1$  begins below and to the right of the leader, while  $F_2$  begins above and to the right of the leader. Both followers have initial velocities that do not point directly at the leader. Equation (34) applies a small acceleration to redirect  $F_2$  to the bottom left side of the leader.  $F_1$  has an initial trajectory that points to the right of the leader, equation (33) applies an acceleration to  $F_1$  to redirect  $F_1$ , but the magnitude of accelerations due to equation (33) causes a few overshoots in the change in trajectory of  $F_1$ , before  $F_1$  is put on the path to close the coverage gap with  $F_2$  and the leader.

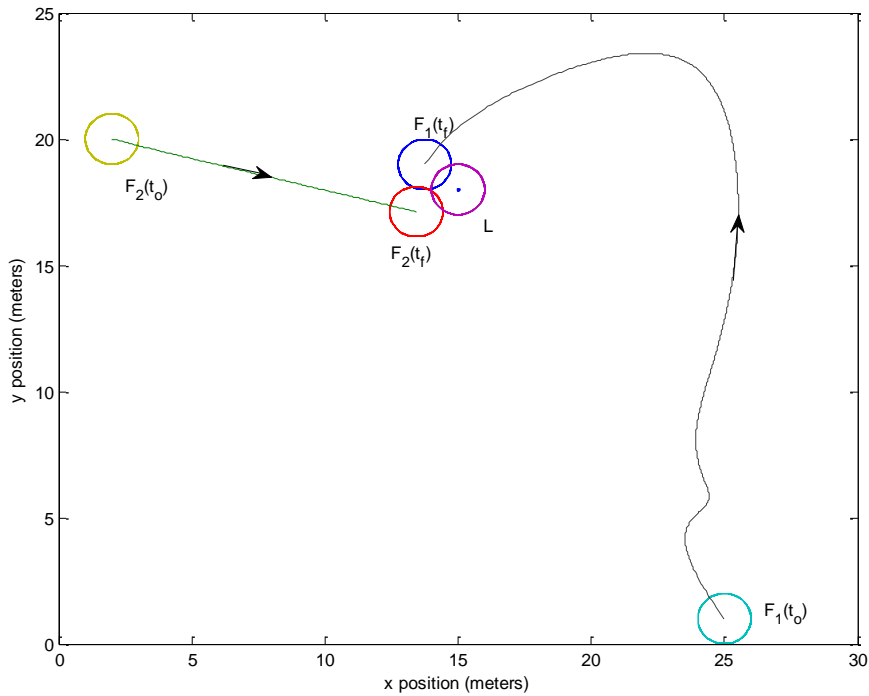


Figure 15 Case 3: Initial and final positions of each sensor. Solid lines indicate the trajectory of the sensors.

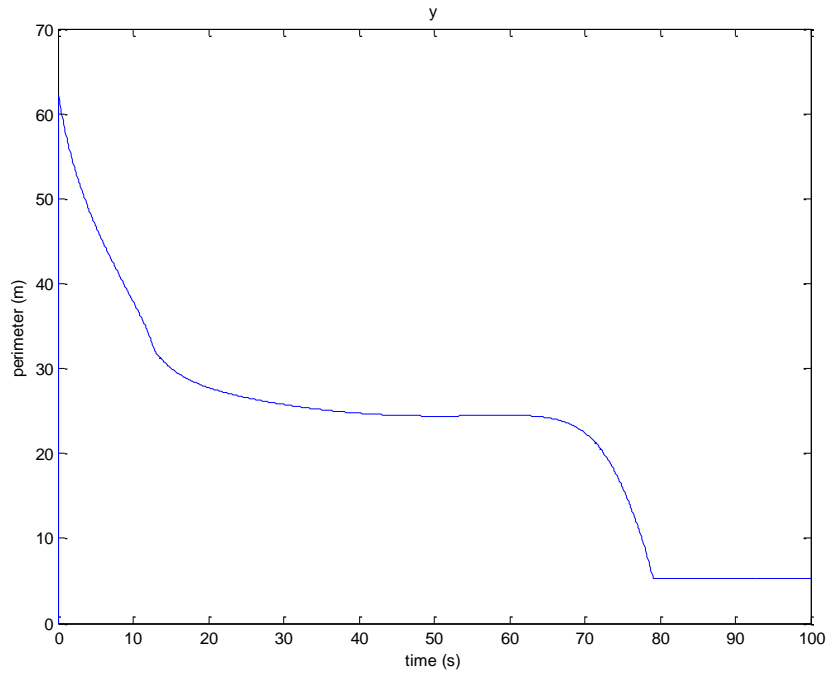


Figure 16 Case 3: Value of the output  $y$  that is used to develop the dynamic inversion control law.

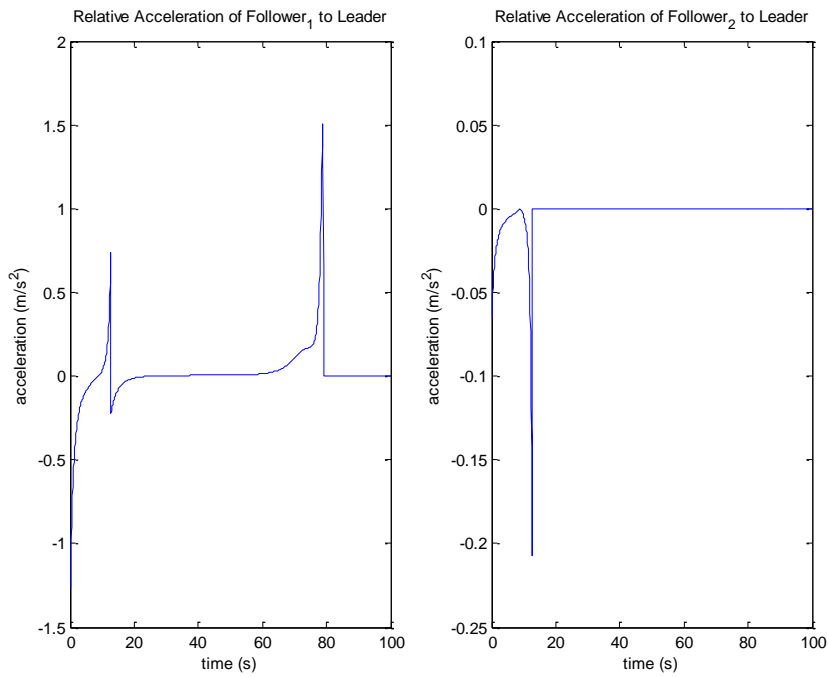


Figure 17 Case 3: Acceleration of Follower<sub>1</sub> and Follower<sub>2</sub>.

### Case 4

In test case 4,  $F_1$  begins below and to the left of the leader, and  $F_2$  begins below and to the right. The leader begins moving up and to the right, before turning to move down and to the right. Both followers begin moving towards the leader, and are able to change directions when the leader changes directions, close the coverage gap, and continue to move in a coordinated fashion with the leader.

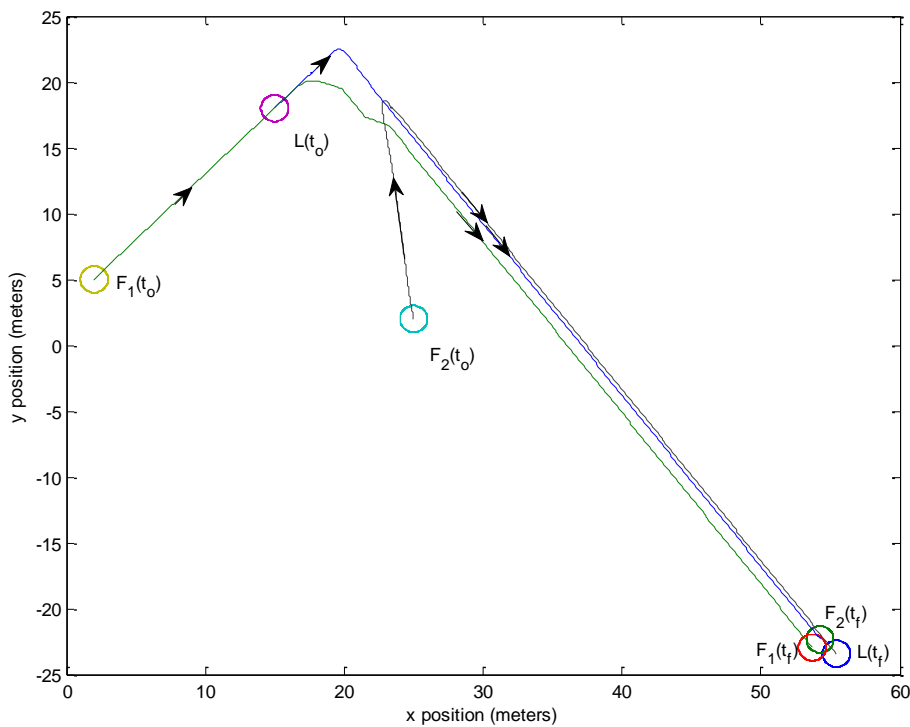
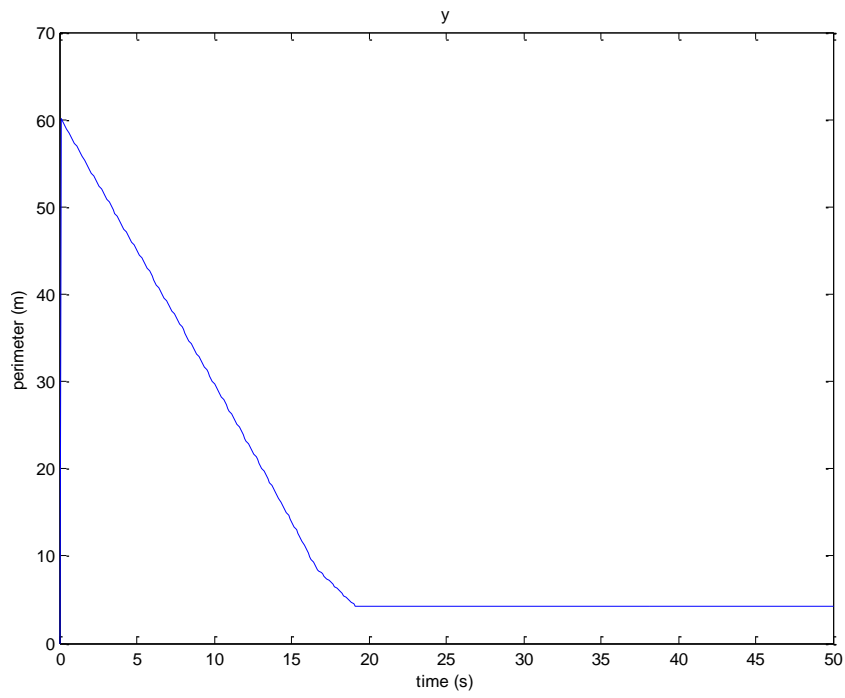


Figure 18 Case 4: Initial and final positions of each sensor. Solid lines indicate the trajectory of the sensors.



**Figure 19 Case 4: Value of the output  $y$  that is used to develop the dynamic inversion control law.**

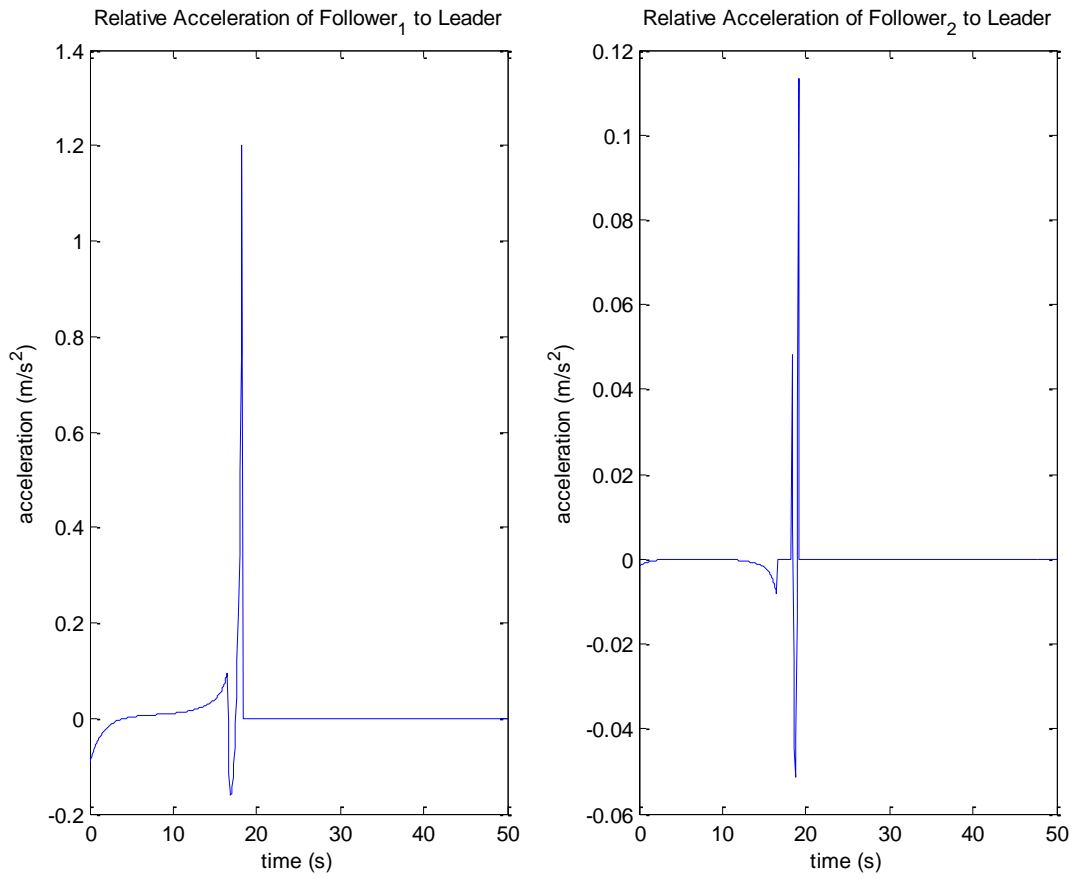


Figure 20 Case 4: Acceleration of Follower<sub>1</sub> and Follower<sub>2</sub>.

## CHAPTER 4

### NON CIRCULAR SENSING ZONE

For the case where the sensors are not assumed to have a uniform circular sensing radius, a different approach is required. Non circular sensing zone can occur for instance, when the sensors use a beam forming antenna. Beam forming is a method to create a desired radiation pattern of an antenna by adding constructively the phase of signal in a desired direction, and nulling the phase of signals in undesired directions. Unlike the circular sensing radius case, a coverage hole area cannot be defined for a group of three sensors with a non uniform sensing zone. Instead, the goal is to develop a control law that will guide the sensors to move together such that the sensing zone of each sensor overlaps with the other two sensors. This can be achieved using a Collision Cone Approach.

The collision cone method [22] [23] [24] is used to predict if a collision is imminent between two objects with unknown trajectories based on the geometry of the two objects and their relative positioning in space. So far the collision cone method has been used as a collision avoidance method. If a collision is determined to be imminent, then evasive maneuvers can be performed. For the sensor coverage problem, it is desired that the sensing area of each sensor overlap, and the collision cone method is used to ensure that each sensor is on the appropriate course to ensure such an overlap. From the collision cone approach, it is shown that if two arbitrary objects (convex or non convex) moving on a plane satisfy the conditions

$$v_r < 0 \tag{72}$$

and

$$v_\theta^2 \leq s^2(\psi/2)(v_\theta^2 + v_r) \tag{73}$$

then the two objects are on a collision course.

### Problem Formulation

The system state variables for the non circular sensing zone problem are defined similarly to the state variables for the circular sensor problem. Each sensor, L, F<sub>1</sub> and F<sub>2</sub>, has a velocity,  $v_L$ ,  $v_1$ , and  $v_2$ , applied at angles  $\alpha_L$ ,  $\alpha_1$ , and  $\alpha_2$ , respectively. The two follower sensors, F<sub>1</sub> and F<sub>2</sub> are at distances  $r_1$  and  $r_2$ , from the leader with line of sight angles  $\theta_1$  and  $\theta_2$ . The line of sight between each sensor pair is defined as the angular bisector of the smallest cone such that a pair of sensors lie on the opposite sides of the vertex of the cone. The two followers have relative accelerations,  $a_1$  and  $a_2$ , applied at angles  $\delta_1$  and  $\delta_2$ , and  $a_1$  and  $a_2$  are the control inputs. The distances, velocities, accelerations, and angles of the three sensor system are given in Figure 21. The line of sight velocities and accelerations match equations (7), (8), (9), and (10), and the nonlinear kinematic state equations match equation (11).

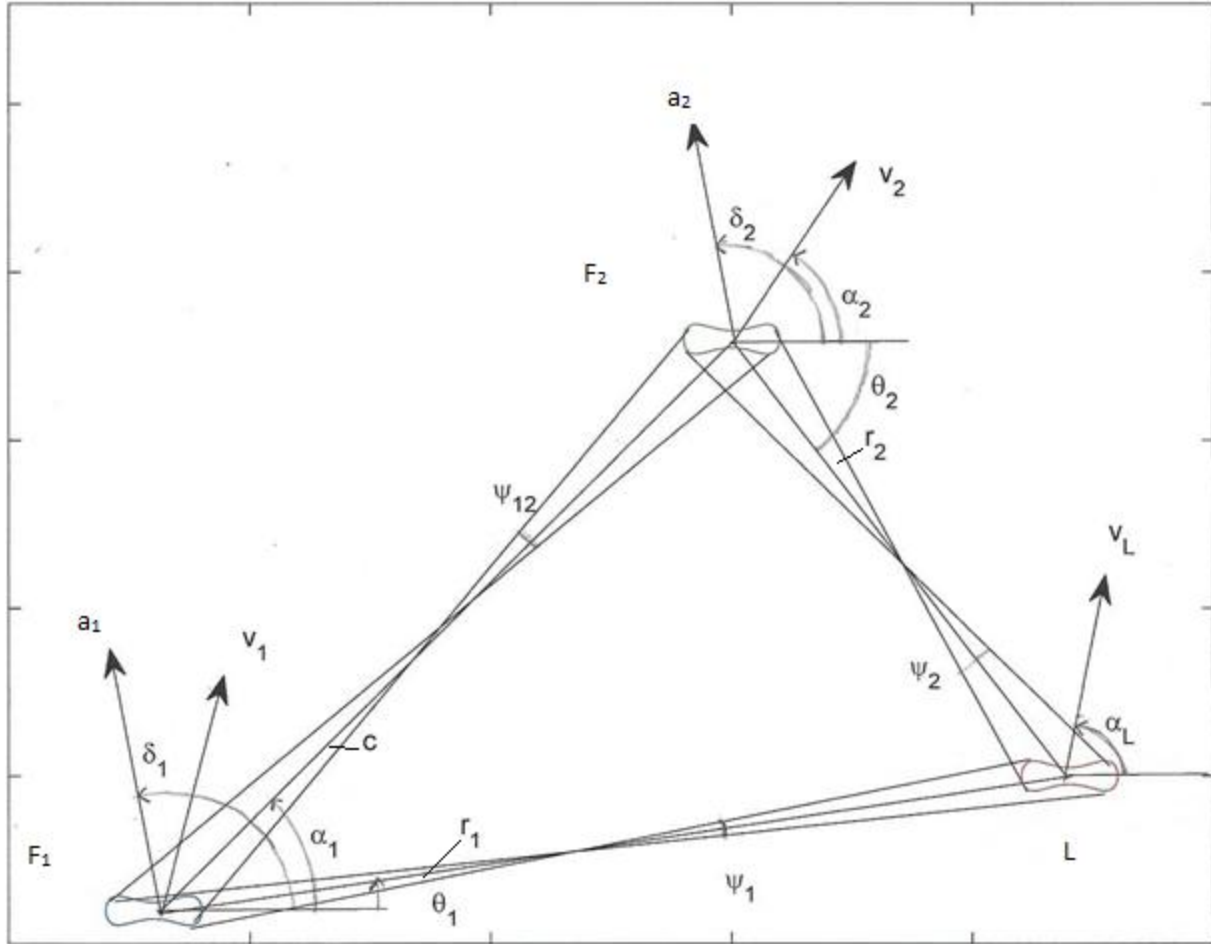


Figure 21 Geometry for the non circular sensing zone problem.

The sensors are labeled the leader,  $F_1$  and  $F_2$ , where the sensor designated as the leader can be stationary, or may move to a desired location, or along a desired trajectory. A control law is developed such that  $F_1$  will move towards both the leader and  $F_2$ . A second control law is developed such that  $F_2$  will move towards the leader.

### Control Law Derivation

In both cases, the control law is developed using dynamic inversion. For  $F_1$ ,  $y_1$  is defined as:

$$y_1 = (v_{\theta 1}^2 - s^2(\psi_1/2)(v_{r1}^2 + v_{\theta 1}^2))^2 + (v_{\theta 12}^2 - s^2(\psi_{12}/2)(v_{r12}^2 + v_{\theta 12}^2))^2 \quad (74)$$

which can be split into two parts,  $y_{1a}$  and  $y_{1b}$ .

$$y_{1a} = (v_{\theta 1}^2 - s^2(\psi_1/2) (v_{r1}^2 + v_{\theta 1}^2))^2 \quad (75)$$

and

$$y_{1b} = (v_{\theta 12}^2 - s^2 (\psi_{12}/2) (v_{r12}^2 + v_{\theta 12}^2))^2 \quad (76)$$

where  $v_{r12}$  and  $v_{\theta 12}$  are the components of the relative velocities between  $F_1$  and  $F_2$ .  $y_{1a} = 0$  implies that  $F_1$  follows a trajectory such that the relative velocity of  $F_1$  with respect to  $L$  is driven to lie on the boundary of the collision cone between  $F_1$  and  $L$ . Similarly,  $y_{1b} = 0$  implies that  $F_1$  follows a trajectory such that the relative velocity of  $F_1$  with respect to  $F_2$  is driven to lie on the boundary of the collision cone between  $F_1$  and  $F_2$ . Both  $v_{r12}$  and  $v_{\theta 12}$  are functions of the state variables, they are defined as:

$$v_{\theta 12} = (v_{r1}(c(\theta_1)s(\theta_2)-s(\theta_1)c(\theta_2))r_2-v_{r2}(c(\theta_1)s(\theta_2)-s(\theta_1)c(\theta_2))r_1 - (c(\theta_1)c(\theta_2)r_2 + s(\theta_1)s(\theta_2)r_2 - r_1) v_{\theta 1} - (c(\theta_1)c(\theta_2)r_1 + s(\theta_1)s(\theta_2)r_1 - r_2) v_{\theta 2}) / (-2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2+r_1^2+r_2^2)^{(1/2)} \quad (77)$$

$$v_{r12} = -(v_{r1}(c(\theta_1)c(\theta_2)r_2+s(\theta_1)s(\theta_2)r_2 - r_1) + v_{r2}(c(\theta_1)c(\theta_2)r_1+s(\theta_1)s(\theta_2)r_1 - r_2) + (c(\theta_1)s(\theta_2) - s(\theta_1)c(\theta_2))(v_{\theta 1}/r_1 - v_{\theta 2}/r_2)r_1r_2) / (-2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2+r_1^2+r_2^2)^{(1/2)} \quad (78)$$

then  $y_{1b}$  becomes:

$$y_{1b} = (s(\psi_{12}/2))^2((v_{r1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) + v_{r2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta 1}/r_1 - v_{\theta 2}/r_2))^2/(r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2) + (v_{r2}(v_{\theta 1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + v_{\theta 2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2)) - (v_{r2}(v_{\theta 1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + v_{\theta 2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2))^2 \quad (79)$$

so that  $y_1 = y_{1a} + y_{1b}$ ,  $y_1$  can only equal zero if both  $y_{1a} = 0$  and  $y_{1b} = 0$ . Note that while in the circular sensing case,  $y$  is defined as the perimeter of the triangle formed by the three lines of sight. In this case  $y_1$  is defined as the sum of squares of the differences between the angles of the relative velocity vector and the boundary between the F and L collision cone. The control input,  $a_1$ , appears in the first derivative of  $y_1$ . The components of  $a_1$  due to  $y_{1a}$  and  $y_{1b}$ ,  $a_{1a}$  and  $a_{1b}$  respectively are split into two parts so the components of  $y_1'$  are found separately. The acceleration is split into two parts so that when  $F_1$  rendezvous with the leader, the stopping condition can be triggered for  $a_{1a}$  and when  $F_1$  rendezvous with  $F_2$  the stopping condition for  $a_{1b}$  is triggered. Once the stopping condition is triggered, it remains in effect as long as the sensing zones of the sensors are in contact with one another. If the sensing zone of  $F_1$  no longer is in contact with the sensing zone of L, then  $a_{1a}$  becomes active again. Similarly if the sensing zone of  $F_1$  no longer overlaps with  $F_2$  then  $a_{1b}$  becomes active again. The stopping condition is defined below.

$$y_{1a}' = -(2v_{\theta 1}^2 - 2s^2(\psi_1)(v_{r1}^2 + v_{\theta 1}^2))(s(\psi_1)^2(2v_{\theta 1}(a_1 s(\theta_1 - \delta_1) - (v_{r1}v_{\theta 1})/r_1) - 2v_{r1}(a_1 c(\theta_1 - \delta_1) - v_{\theta 1}^2/r_1)) - 2v_{\theta 1}(a_1 s(\theta_1 - \delta_1) - (v_{r1}v_{\theta 1})/r_1) + \psi_1' c(\psi_1) s(\psi_1)(v_{r1}^2 + v_{\theta 1}^2)) \quad (80)$$

$$y_{1b}' = (2s^2(\psi_{12}/2) ((v_{r1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) + v_{r2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))(v_{\theta 1}/r_1 - v_{\theta 2}/r_2))^2/(r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2) + (v_{r2}(v_{\theta 1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + v_{\theta 2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))))^2/(r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2)) - (2(v_{r2}(v_{\theta 1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + v_{\theta 2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))))^2/(r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2)))(s^2(\psi_{12}/2) (((v_{r1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) + v_{r2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + r_1r_2(c(\theta_1)s(\theta_2) -$$

$$\begin{aligned}
& c(\theta_2)s(\theta_1))(v_{\theta_1}/r_1 - v_{\theta_2}/r_2))^2(2c(\theta_1)c(\theta_2)r_1v_{r_2} - 2r_2v_{r_2} - 2r_1v_{r_1} + 2c(\theta_1)c(\theta_2)r_2v_{r_1} - 2c(\theta_1)s(\theta_2)r_1v_{\theta_2} + \\
& 2c(\theta_1)s(\theta_2)r_2v_{\theta_1} + 2c(\theta_2)s(\theta_1)r_1v_{\theta_2} - 2c(\theta_2)s(\theta_1)r_2v_{\theta_1} + 2s(\theta_1)s(\theta_2)r_1v_{r_2} + 2s(\theta_1)s(\theta_2)r_2v_{r_1}))/r_1^2 + r_2^2 \\
& - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2)^2 + ((v_{r_2}(v_{\theta_1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + \\
& v_{\theta_2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) + r_2v_{r_1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))))^2(2c(\theta_1)c(\theta_2)r_1v_{r_2} - \\
& 2r_2v_{r_2} - 2r_1v_{r_1} + 2c(\theta_1)c(\theta_2)r_2v_{r_1} - 2c(\theta_1)s(\theta_2)r_1v_{\theta_2} + 2c(\theta_1)s(\theta_2)r_2v_{\theta_1} + 2c(\theta_2)s(\theta_1)r_1v_{\theta_2} - \\
& 2c(\theta_2)s(\theta_1)r_2v_{\theta_1} + 2s(\theta_1)s(\theta_2)r_1v_{r_2} + 2s(\theta_1)s(\theta_2)r_2v_{r_1}))/r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2)^2 - \\
& ((2v_{r_2}(v_{\theta_1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + v_{\theta_2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + \\
& c(\theta_2)s(\theta_1)r_1) + 2r_2v_{r_1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))))((a_2c(\theta_2 - \delta_2) - v_{\theta_2}^2/r_2)(v_{\theta_1}(c(\theta_1)c(\theta_2)r_2 - r_1 + \\
& s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + v_{\theta_2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) - v_{r_2}(v_{\theta_1}(s(\theta_1)s(\theta_2)v_{r_2} - \\
& v_{r_1} + c(\theta_1)c(\theta_2)v_{r_2} - c(\theta_1)s(\theta_2)v_{\theta_2} + c(\theta_2)s(\theta_1)v_{\theta_2} + (c(\theta_1)s(\theta_2)r_2v_{\theta_1})/r_1 - (c(\theta_2)s(\theta_1)r_2v_{\theta_1})/r_1) - ((s(\theta_2) \\
& - \delta_2)((2v_{r_2}^2v_{\theta_2}^2)/r_2 - \psi_2'c(\psi_2/2)(v_{r_2}^2 + v_{\theta_2}^2)))/(2s(\theta_2 - \delta_2)v_{\theta_2} + 2c(\theta_2 - \delta_2)s(\psi_2/2)v_{r_2} - 2s(\theta_2 - \\
& \delta_2)s(\psi_2/2)v_{\theta_2}) + (v_{r_2}v_{\theta_2})/r_2)(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + v_{\theta_2}(s(\theta_1)s(\theta_2)v_{r_1} - v_{r_2} + c(\theta_1)c(\theta_2)v_{r_1} + \\
& c(\theta_1)s(\theta_2)v_{\theta_1} - c(\theta_2)s(\theta_1)v_{\theta_1} - (c(\theta_1)s(\theta_2)r_1v_{\theta_2})/r_2 + (c(\theta_2)s(\theta_1)r_1v_{\theta_2})/r_2) + (a_1s(\theta_1 - \delta_1) - \\
& (v_{r_1}v_{\theta_1})/r_1)(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) + c(\theta_1)c(\theta_2)v_{\theta_1} + c(\theta_2)s(\theta_1)v_{r_1} - (c(\theta_1)c(\theta_2)v_{\theta_2})/r_2 + \\
& (s(\theta_1)s(\theta_2)v_{\theta_1})/r_1 - (s(\theta_1)s(\theta_2)r_1v_{\theta_2})/r_2) + r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(a_1c(\theta_1 - \delta_1) - v_{\theta_1}^2/r_1) + \\
& r_2v_{r_1}((c(\theta_1)c(\theta_2)v_{\theta_1})/r_1 - (c(\theta_1)c(\theta_2)v_{\theta_2})/r_2 + (s(\theta_1)s(\theta_2)v_{\theta_1})/r_1 - (s(\theta_1)s(\theta_2)v_{\theta_2})/r_2) - v_{r_1}v_{r_2}(c(\theta_1)s(\theta_2) - \\
& c(\theta_2)s(\theta_1)))/r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2) + ((2v_{r_1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) + \\
& 2v_{r_2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + 2r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta_1}/r_1 - v_{\theta_2}/r_2))(v_{r_1}(s(\theta_1)s(\theta_2)v_{r_2} - \\
& v_{r_1} + c(\theta_1)c(\theta_2)v_{r_2} - c(\theta_1)s(\theta_2)v_{\theta_2} + c(\theta_2)s(\theta_1)v_{\theta_2} + (c(\theta_1)s(\theta_2)r_2v_{\theta_1})/r_1 - (c(\theta_2)s(\theta_1)r_2v_{\theta_1})/r_1) - \\
& (a_2c(\theta_2 - \delta_2) - v_{\theta_2}^2/r_2)(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) - (a_1c(\theta_1 - \delta_1) - v_{\theta_1}^2/r_1)(c(\theta_1)c(\theta_2)r_2 - r_1 + \\
& s(\theta_1)s(\theta_2)r_2) + v_{r_2}(s(\theta_1)s(\theta_2)v_{r_1} - v_{r_2} + c(\theta_1)c(\theta_2)v_{r_1} + c(\theta_1)s(\theta_2)v_{\theta_1} - c(\theta_2)s(\theta_1)v_{\theta_1} - \\
& (c(\theta_1)s(\theta_2)r_1v_{\theta_2})/r_2 + (c(\theta_2)s(\theta_1)r_1v_{\theta_2})/r_2) + r_1v_{r_2}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta_1}/r_1 - v_{\theta_2}/r_2) + \\
& r_2v_{r_1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta_1}/r_1 - v_{\theta_2}/r_2) + r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(((s(\theta_2 - \delta_2)((2v_{r_2}^2v_{\theta_2}^2)/r_2 -
\end{aligned}$$

$$\begin{aligned}
& \psi_2'c(\psi_2/2)(v_{r2}^2 + v_{\theta 2}^2))/(2s(\theta_2 - \delta_2)v_{\theta 2} + 2c(\theta_2 - \delta_2)s(\psi_2/2)v_{r2} - 2s(\theta_2 - \delta_2)s(\psi_2/2)v_{\theta 2}) + \\
& (v_{r2}v_{\theta 2})/r_2/r_2 + (a_1s(\theta_1 - \delta_1) - (v_{r1}v_{\theta 1})/r_1)/r_1 - (v_{r1}v_{\theta 1})/r_1^2 + (v_{r2}v_{\theta 2})/r_2^2 - r_1r_2(v_{\theta 1}/r_1 - \\
& v_{\theta 2}/r_2)((c(\theta_1)c(\theta_2)v_{\theta 1})/r_1 - (c(\theta_1)c(\theta_2)v_{\theta 2})/r_2 + (s(\theta_1)s(\theta_2)v_{\theta 1})/r_1 - (s(\theta_1)s(\theta_2)v_{\theta 2})/r_2))/ (r_1^2 + r_2^2 - \\
& 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2)) - ((v_{r2}(v_{\theta 1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + \\
& v_{\theta 2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2(2c(\theta_1)c(\theta_2)r_1v_{r2} - \\
& 2r_2v_{r2} - 2r_1v_{r1} + 2c(\theta_1)c(\theta_2)r_2v_{r1} - 2c(\theta_1)s(\theta_2)r_1v_{\theta 2} + 2c(\theta_1)s(\theta_2)r_2v_{\theta 1} + 2c(\theta_2)s(\theta_1)r_1v_{\theta 2} - \\
& 2c(\theta_2)s(\theta_1)r_2v_{\theta 1} + 2s(\theta_1)s(\theta_2)r_1v_{r2} + 2s(\theta_1)s(\theta_2)r_2v_{r1}))/ (r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2))^2 + \\
& ((2v_{r2}(v_{\theta 1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + v_{\theta 2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + \\
& c(\theta_2)s(\theta_1)r_1) + 2r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))(a_2c(\theta_2 - \delta_2) - v_{\theta 2}^2/r_2)(v_{\theta 1}(c(\theta_1)c(\theta_2)r_2 - r_1 + \\
& s(\theta_1)s(\theta_2)r_2) - c(\theta_1)s(\theta_2) + v_{\theta 2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) - v_{r2}(v_{\theta 1}(s(\theta_1)s(\theta_2)v_{r2} - \\
& v_{r1} + c(\theta_1)c(\theta_2)v_{r2} - c(\theta_1)s(\theta_2)v_{\theta 2} + c(\theta_2)s(\theta_1)v_{\theta 2} + (c(\theta_1)s(\theta_2)r_2v_{\theta 1})/r_1 - (c(\theta_2)s(\theta_1)r_2v_{\theta 1})/r_1) - ((s(\theta_2 \\
& - \delta_2)((2v_{r2}^2v_{\theta 2}^2)/r_2 - \psi_2'c(\psi_2/2)(v_{r2}^2 + v_{\theta 2}^2)))/(2s(\theta_2 - \delta_2)v_{\theta 2} + 2c(\theta_2 - \delta_2)s(\psi_2/2)v_{r2} - 2s(\theta_2 - \\
& \delta_2)s(\psi_2/2)v_{\theta 2}) + (v_{r2}v_{\theta 2})/r_2)(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + v_{\theta 2}(s(\theta_1)s(\theta_2)v_{r1} - v_{r2} + c(\theta_1)c(\theta_2)v_{r1} + \\
& c(\theta_1)s(\theta_2)v_{\theta 1} - c(\theta_2)s(\theta_1)v_{\theta 1} - (c(\theta_1)s(\theta_2)r_1v_{\theta 2})/r_2 + (c(\theta_2)s(\theta_1)r_1v_{\theta 2})/r_2) + (a_1s(\theta_1 - \delta_1) - \\
& (v_{r1}v_{\theta 1})/r_1)(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) + c(\theta_1)c(\theta_2)v_{\theta 1} + c(\theta_2)s(\theta_1)v_{r1} - (c(\theta_1)c(\theta_2)v_{\theta 2})/r_2 + \\
& (s(\theta_1)s(\theta_2)v_{\theta 1})/r_1 - (s(\theta_1)s(\theta_2)r_1v_{\theta 2})/r_2) + r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(a_1c(\theta_1 - \delta_1) - v_{\theta 1}^2/r_1) + \\
& r_2v_{r1}((c(\theta_1)c(\theta_2)v_{\theta 1})/r_1 - (c(\theta_1)c(\theta_2)v_{\theta 2})/r_2 + (s(\theta_1)s(\theta_2)v_{\theta 1})/r_1 - (s(\theta_1)s(\theta_2)v_{\theta 2})/r_2) - v_{r1}v_{r2}(c(\theta_1)s(\theta_2) - \\
& c(\theta_2)s(\theta_1)))/ (r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2) + \psi_{12}'c(\psi_{12}/2)s(\psi_{12}/2)((v_{r1}(c(\theta_1)c(\theta_2)r_2 - r_1 \\
& + s(\theta_1)s(\theta_2)r_2) + v_{r2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))(v_{\theta 1}/r_1 - \\
& v_{\theta 2}/r_2))^2/(r_1^2 + r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2) + (v_{r2}(v_{\theta 1}(c(\theta_1)c(\theta_2)r_2 - r_1 + s(\theta_1)s(\theta_2)r_2) - \\
& c(\theta_1)s(\theta_2) + v_{\theta 2}(c(\theta_1)c(\theta_2)r_1 - r_2 + s(\theta_1)s(\theta_2)r_1) + c(\theta_2)s(\theta_1)r_1) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + \\
& r_2^2 - 2c(\theta_1)c(\theta_2)r_1r_2 - 2s(\theta_1)s(\theta_2)r_1r_2))) \tag{81}
\end{aligned}$$

$a_{1a}$  is from the first derivative of  $y_{1a}$ , and  $a_{1b}$  is from the first derivative of  $y_{1b}$ , such that:

$$a_1 = a_{1a} + a_{1b} \quad (82)$$

solving for  $a_{1a}$  and  $a_{1b}$  yields:

$$a_{1a} = (((2v_{r1}v_{\theta1}^2)/r_1 + \psi_1'c(\psi_1/2)s(\psi_1/2)(v_{r1}^2 + v_{\theta1}^2))(2s(\psi_1/2)^2(v_{r1}^2 + v_{\theta1}^2) - 2v_{\theta1}^2) + k_1(s(\psi_1/2)^2(v_{r1}^2 + v_{\theta1}^2) - v_{\theta1}^2)^2)/((s(\psi_1/2)^2(2v_{r1}c(\delta_1 - \theta_1) + 2v_{\theta1}s(\delta_1 - \theta_1)) - 2v_{\theta1}s(\delta_1 - \theta_1))(2s(\psi_1/2)^2(v_{r1}^2 + v_{\theta1}^2) - 2v_{\theta1}^2)) \quad (83)$$

$$\begin{aligned} a_{1b} = & (k_1(s^2(\psi_{12}/2) ((v_{r1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) + v_{r2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\ & r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta1}/r_1 - v_{\theta2}/r_2))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2)) + \\ & (v_{r2}(v_{\theta1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\ & r_1c(\theta_2)s(\theta_1)) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2))) - \\ & (v_{r2}(v_{\theta1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\ & r_1c(\theta_2)s(\theta_1)) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2)))^2 + \\ & (2s^2(\psi_{12}/2) ((v_{r1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) + v_{r2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\ & r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta1}/r_1 - v_{\theta2}/r_2))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2)) + \\ & (v_{r2}(v_{\theta1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\ & r_1c(\theta_2)s(\theta_1)) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2))) - \\ & (2(v_{r2}(v_{\theta1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\ & r_1c(\theta_2)s(\theta_1)) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - \\ & 2r_1r_2c(\theta_1)c(\theta_2)))(s(\psi_{12}/2)^2(((2v_{r1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) + 2v_{r2}(r_1c(\theta_1)c(\theta_2) - r_2 + \\ & r_1s(\theta_1)s(\theta_2)) + 2r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta1}/r_1 - v_{\theta2}/r_2))(v_{r1}(v_{r2}c(\theta_1)c(\theta_2) - v_{r1} - v_{\theta2}c(\theta_1)s(\theta_2) + \\ & v_{\theta2}c(\theta_2)s(\theta_1) + v_{r2}s(\theta_1)s(\theta_2) + (r_2v_{\theta1}c(\theta_1)s(\theta_2))/r_1 - (r_2v_{\theta1}c(\theta_2)s(\theta_1))/r_1) + v_{r2}(v_{r1}c(\theta_1)c(\theta_2) - v_{r2} + \\ & v_{\theta1}c(\theta_1)s(\theta_2) - v_{\theta1}c(\theta_2)s(\theta_1) + v_{r1}s(\theta_1)s(\theta_2) - (r_1v_{\theta2}c(\theta_1)s(\theta_2))/r_2 + (r_1v_{\theta2}c(\theta_2)s(\theta_1))/r_2) + (v_{\theta2}^2/r_2 - \end{aligned}$$

$$\begin{aligned}
& (c(\delta_2 - \theta_2)(k^2(s^2(\psi_2/2)(v_{r2}^2 + v_{\theta 2}^2) - v_{\theta 2}^2) + (2v_{r2}v_{\theta 2}^2)/r_2 + \psi_2'c(\psi_2/2)s(\psi_2/2)(v_{r2}^2 + \\
& v_{\theta 2}^2)))/(s(\psi_2/2)^2(2v_{r2}c(\delta_2 - \theta_2) + 2v_{\theta 2}s(\delta_2 - \theta_2)) - 2v_{\theta 2}s(\delta_2 - \theta_2))(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\
& (v_{\theta 1}^2(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)))/r_1 + r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(((v_{r2}v_{\theta 2})/r_2 - (s(\delta_2 - \\
& \theta_2)((2v_{r2}^2v_{\theta 2}^2)/r_2 - \psi_2'c(\psi_2/2)(v_{r2}^2 + v_{\theta 2}^2)))/(2v_{r2}s(\psi_2/2)c(\delta_2 - \theta_2) - 2v_{\theta 2}s(\delta_2 - \theta_2) + 2v_{\theta 2}s(\delta_2 - \\
& \theta_2)s(\psi_2/2)))/r_2 - (2v_{r1}v_{\theta 1})/r_1^2 + (v_{r2}v_{\theta 2})/r_2^2) + r_1v_{r2}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta 1}/r_1 - v_{\theta 2}/r_2) + \\
& r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta 1}/r_1 - v_{\theta 2}/r_2) - r_1r_2(v_{\theta 1}/r_1 - v_{\theta 2}/r_2)((v_{\theta 1}c(\theta_1)c(\theta_2))/r_1 - (v_{\theta 2}c(\theta_1)c(\theta_2))/r_2 \\
& + (v_{\theta 1}s(\theta_1)s(\theta_2))/r_1 - (v_{\theta 2}s(\theta_1)s(\theta_2))/r_2))/r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2)) + \\
& ((v_{r1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) + v_{r2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + r_1r_2(c(\theta_1)s(\theta_2) - \\
& c(\theta_2)s(\theta_1))(v_{\theta 1}/r_1 - v_{\theta 2}/r_2))^2(2r_1v_{\theta 2}c(\theta_2)s(\theta_1) - 2r_2v_{r2} - 2r_1v_{\theta 2}c(\theta_1)s(\theta_2) - 2r_1v_{r1} + 2r_2v_{\theta 1}c(\theta_1)s(\theta_2) - \\
& 2r_2v_{\theta 1}c(\theta_2)s(\theta_1) + 2r_1v_{r2}s(\theta_1)s(\theta_2) + 2r_2v_{r1}s(\theta_1)s(\theta_2) + 2r_1v_{r2}c(\theta_1)c(\theta_2) + 2r_2v_{r1}c(\theta_1)c(\theta_2)))/r_1^2 + r_2^2 \\
& - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2))^2 + ((2v_{r2}(v_{\theta 1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + \\
& v_{\theta 2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + r_1c(\theta_2)s(\theta_1)) + 2r_2v_{r1}(c(\theta_1)s(\theta_2) - \\
& c(\theta_2)s(\theta_1)))(v_{r2}(v_{\theta 1}(v_{r2}c(\theta_1)c(\theta_2) - v_{r1} - v_{\theta 2}c(\theta_1)s(\theta_2) + v_{\theta 2}c(\theta_2)s(\theta_1) + v_{r2}s(\theta_1)s(\theta_2) + \\
& (r_2v_{\theta 1}c(\theta_1)s(\theta_2))/r_1 - (r_2v_{\theta 1}c(\theta_2)s(\theta_1))/r_1) + v_{\theta 2}(v_{r1}c(\theta_1)c(\theta_2) - v_{r2} + v_{\theta 1}c(\theta_1)s(\theta_2) - v_{\theta 1}c(\theta_2)s(\theta_1) + \\
& v_{r1}s(\theta_1)s(\theta_2) - (r_1v_{\theta 2}c(\theta_1)s(\theta_2))/r_2 + (r_1v_{\theta 2}c(\theta_2)s(\theta_1))/r_2) - ((v_{r2}v_{\theta 2})/r_2 - (s(\delta_2 - \theta_2)((2v_{r2}^2v_{\theta 2}^2)/r_2 - \\
& \psi_2'c(\psi_2/2)(v_{r2}^2 + v_{\theta 2}^2)))/(2v_{r2}s(\psi_2/2)c(\delta_2 - \theta_2) - 2v_{\theta 2}s(\delta_2 - \theta_2) + 2v_{\theta 2}s(\delta_2 - \theta_2)s(\psi_2/2)))(r_1c(\theta_1)c(\theta_2) - \\
& r_2 + r_1s(\theta_1)s(\theta_2)) + v_{\theta 1}c(\theta_1)c(\theta_2) + v_{r1}c(\theta_2)s(\theta_1) - (v_{\theta 2}c(\theta_1)c(\theta_2))/r_2 - (v_{r1}v_{\theta 1}(r_2c(\theta_1)c(\theta_2) - r_1 + \\
& r_2s(\theta_1)s(\theta_2)))/r_1 + (v_{\theta 1}s(\theta_1)s(\theta_2))/r_1 - (r_1v_{\theta 2}s(\theta_1)s(\theta_2))/r_2) + (v_{\theta 2}^2/r_2 - (c(\delta_2 - \theta_2)(k^2(s^2(\psi_2/2)(v_{r2}^2 + \\
& v_{\theta 2}^2) - v_{\theta 2}^2) + (2v_{r2}v_{\theta 2}^2)/r_2 + \psi_2'c(\psi_2/2)s(\psi_2/2)(v_{r2}^2 + v_{\theta 2}^2)))/(s^2(\psi_2/2)(2v_{r2}c(\delta_2 - \theta_2) + 2v_{\theta 2}s(\delta_2 - \\
& \theta_2)) - 2v_{\theta 2}s(\delta_2 - \theta_2))(v_{\theta 1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta 2}(r_1c(\theta_1)c(\theta_2) - r_2 + \\
& r_1s(\theta_1)s(\theta_2)) + r_1c(\theta_2)s(\theta_1)) - r_2v_{r1}((v_{\theta 1}c(\theta_1)c(\theta_2))/r_1 - (v_{\theta 2}c(\theta_1)c(\theta_2))/r_2 + (v_{\theta 1}s(\theta_1)s(\theta_2))/r_1 - \\
& (v_{\theta 2}s(\theta_1)s(\theta_2))/r_2) + v_{r1}v_{r2}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)) + (r_2v_{\theta 1}^2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))/r_1))/r_1^2 + r_2^2 - \\
& 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2)) + ((v_{r2}(v_{\theta 1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) +
\end{aligned}$$

$$\begin{aligned}
& v_{\theta_2}(r_1 c(\theta_1) c(\theta_2) - r_2 + r_1 s(\theta_1) s(\theta_2)) + r_1 c(\theta_2) s(\theta_1) + r_2 v_{r_1}(c(\theta_1) s(\theta_2) - c(\theta_2) s(\theta_1)))^2 (2r_1 v_{\theta_2} c(\theta_2) s(\theta_1) - \\
& 2r_2 v_{r_2} - 2r_1 v_{\theta_2} c(\theta_1) s(\theta_2) - 2r_1 v_{r_1} + 2r_2 v_{\theta_1} c(\theta_1) s(\theta_2) - 2r_2 v_{\theta_1} c(\theta_2) s(\theta_1) + 2r_1 v_{r_2} s(\theta_1) s(\theta_2) + \\
& 2r_2 v_{r_1} s(\theta_1) s(\theta_2) + 2r_1 v_{r_2} c(\theta_1) c(\theta_2) + 2r_2 v_{r_1} c(\theta_1) c(\theta_2)) / (r_1^2 + r_2^2 - 2r_1 r_2 s(\theta_1) s(\theta_2) - 2r_1 r_2 c(\theta_1) c(\theta_2))^2) \\
& - ((2v_{r_2}(v_{\theta_1}(r_2 c(\theta_1) c(\theta_2) - r_1 + r_2 s(\theta_1) s(\theta_2)) - c(\theta_1) s(\theta_2) + v_{\theta_2}(r_1 c(\theta_1) c(\theta_2) - r_2 + r_1 s(\theta_1) s(\theta_2)) + \\
& r_1 c(\theta_2) s(\theta_1)) + 2r_2 v_{r_1}(c(\theta_1) s(\theta_2) - c(\theta_2) s(\theta_1)))(v_{r_2}(v_{\theta_1}(v_{r_2} c(\theta_1) c(\theta_2) - v_{r_1} - v_{\theta_2} c(\theta_1) s(\theta_2) + \\
& v_{\theta_2} c(\theta_2) s(\theta_1) + v_{r_2} s(\theta_1) s(\theta_2) + (r_2 v_{\theta_1} c(\theta_1) s(\theta_2)) / r_1 - (r_2 v_{\theta_1} c(\theta_2) s(\theta_1)) / r_1) + v_{\theta_2}(v_{r_1} c(\theta_1) c(\theta_2) - v_{r_2} + \\
& v_{\theta_1} c(\theta_1) s(\theta_2) - v_{\theta_1} c(\theta_2) s(\theta_1) + v_{r_1} s(\theta_1) s(\theta_2) - (r_1 v_{\theta_2} c(\theta_1) s(\theta_2)) / r_2 + (r_1 v_{\theta_2} c(\theta_2) s(\theta_1)) / r_2) - ((v_{r_2} v_{\theta_2}) / r_2 - \\
& (s(\delta_2 - \theta_2)((2v_{r_2}^2 v_{\theta_2}^2) / r_2 - \psi_2' c(\psi_2/2)(v_{r_2}^2 + v_{\theta_2}^2))) / (2v_{r_2} s(\psi_2/2) c(\delta_2 - \theta_2) - 2v_{\theta_2} s(\delta_2 - \theta_2) + 2v_{\theta_2} s(\delta_2 - \\
& \theta_2) s(\psi_2/2))) (r_1 c(\theta_1) c(\theta_2) - r_2 + r_1 s(\theta_1) s(\theta_2)) + v_{\theta_1} c(\theta_1) c(\theta_2) + v_{r_1} c(\theta_2) s(\theta_1) - (v_{\theta_2} c(\theta_1) c(\theta_2)) / r_2 - \\
& (v_{r_1} v_{\theta_1}(r_2 c(\theta_1) c(\theta_2) - r_1 + r_2 s(\theta_1) s(\theta_2))) / r_1 + (v_{\theta_1} s(\theta_1) s(\theta_2)) / r_1 - (r_1 v_{\theta_2} s(\theta_1) s(\theta_2)) / r_2) + (v_{\theta_2}^2 / r_2 - (c(\delta_2 \\
& - \theta_2)(k_2(s^2(\psi_2/2)(v_{r_2}^2 + v_{\theta_2}^2) - v_{\theta_2}^2) + (2v_{r_2} v_{\theta_2}^2) / r_2 + \psi_2' c(\psi_2/2) s(\psi_2/2)(v_{r_2}^2 + \\
& v_{\theta_2}^2))) / (s(\psi_2/2)^2 (2v_{r_2} c(\delta_2 - \theta_2) + 2v_{\theta_2} s(\delta_2 - \theta_2)) - 2v_{\theta_2} s(\delta_2 - \theta_2))) (v_{\theta_1}(r_2 c(\theta_1) c(\theta_2) - r_1 + r_2 s(\theta_1) s(\theta_2)) \\
& - c(\theta_1) s(\theta_2) + v_{\theta_2}(r_1 c(\theta_1) c(\theta_2) - r_2 + r_1 s(\theta_1) s(\theta_2)) + r_1 c(\theta_2) s(\theta_1)) - r_2 v_{r_1}((v_{\theta_1} c(\theta_1) c(\theta_2)) / r_1 - \\
& (v_{\theta_2} c(\theta_1) c(\theta_2)) / r_2 + (v_{\theta_1} s(\theta_1) s(\theta_2)) / r_1 - (v_{\theta_2} s(\theta_1) s(\theta_2)) / r_2) + v_{r_1} v_{r_2}(c(\theta_1) s(\theta_2) - c(\theta_2) s(\theta_1)) + \\
& (r_2 v_{\theta_1}^2 (c(\theta_1) s(\theta_2) - c(\theta_2) s(\theta_1))) / r_1) / (r_1^2 + r_2^2 - 2r_1 r_2 s(\theta_1) s(\theta_2) - 2r_1 r_2 c(\theta_1) c(\theta_2)) - \\
& ((v_{r_2}(v_{\theta_1}(r_2 c(\theta_1) c(\theta_2) - r_1 + r_2 s(\theta_1) s(\theta_2)) - c(\theta_1) s(\theta_2) + v_{\theta_2}(r_1 c(\theta_1) c(\theta_2) - r_2 + r_1 s(\theta_1) s(\theta_2)) + \\
& r_1 c(\theta_2) s(\theta_1)) + r_2 v_{r_1}(c(\theta_1) s(\theta_2) - c(\theta_2) s(\theta_1)))^2 (2r_1 v_{\theta_2} c(\theta_2) s(\theta_1) - 2r_2 v_{r_2} - 2r_1 v_{\theta_2} c(\theta_1) s(\theta_2) - 2r_1 v_{r_1} + \\
& 2r_2 v_{\theta_1} c(\theta_1) s(\theta_2) - 2r_2 v_{\theta_1} c(\theta_2) s(\theta_1) + 2r_1 v_{r_2} s(\theta_1) s(\theta_2) + 2r_2 v_{r_1} s(\theta_1) s(\theta_2) + 2r_1 v_{r_2} c(\theta_1) c(\theta_2) + \\
& 2r_2 v_{r_1} c(\theta_1) c(\theta_2)) / (r_1^2 + r_2^2 - 2r_1 r_2 s(\theta_1) s(\theta_2) - 2r_1 r_2 c(\theta_1) c(\theta_2))^2 + \\
& \psi_{12}' c(\psi_{12}/2) s(\psi_{12}/2) ((v_{r_1}(r_2 c(\theta_1) c(\theta_2) - r_1 + r_2 s(\theta_1) s(\theta_2)) + v_{r_2}(r_1 c(\theta_1) c(\theta_2) - r_2 + r_1 s(\theta_1) s(\theta_2)) + \\
& r_1 r_2 (c(\theta_1) s(\theta_2) - c(\theta_2) s(\theta_1)) (v_{\theta_1} / r_1 - v_{\theta_2} / r_2))^2 / (r_1^2 + r_2^2 - 2r_1 r_2 s(\theta_1) s(\theta_2) - 2r_1 r_2 c(\theta_1) c(\theta_2)) + \\
& (v_{r_2}(v_{\theta_1}(r_2 c(\theta_1) c(\theta_2) - r_1 + r_2 s(\theta_1) s(\theta_2)) - c(\theta_1) s(\theta_2) + v_{\theta_2}(r_1 c(\theta_1) c(\theta_2) - r_2 + r_1 s(\theta_1) s(\theta_2)) + \\
& r_1 c(\theta_2) s(\theta_1)) + r_2 v_{r_1}(c(\theta_1) s(\theta_2) - c(\theta_2) s(\theta_1)))^2 / (r_1^2 + r_2^2 - 2r_1 r_2 s(\theta_1) s(\theta_2) -
\end{aligned}$$

$$\begin{aligned}
& 2r_1r_2c(\theta_1)c(\theta_2)))/((2s(\psi_2/2)^2((v_{r1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) + v_{r2}(r_1c(\theta_1)c(\theta_2) - r_2 + \\
& r_1s(\theta_1)s(\theta_2)) + r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta 1}/r_1 - v_{\theta 2}/r_2))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - \\
& 2r_1r_2c(\theta_1)c(\theta_2)) + (v_{r2}(v_{\theta 1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta 2}(r_1c(\theta_1)c(\theta_2) - r_2 + \\
& r_1s(\theta_1)s(\theta_2)) + r_1c(\theta_2)s(\theta_1)) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - \\
& 2r_1r_2c(\theta_1)c(\theta_2)) - (2(v_{r2}(v_{\theta 1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta 2}(r_1c(\theta_1)c(\theta_2) - r_2 + \\
& r_1s(\theta_1)s(\theta_2)) + r_1c(\theta_2)s(\theta_1)) + r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))^2/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - \\
& 2r_1r_2c(\theta_1)c(\theta_2)))(s^2(\psi_2/2) (((2v_{r2}(v_{\theta 1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta 2}(r_1c(\theta_1)c(\theta_2) - \\
& r_2 + r_1s(\theta_1)s(\theta_2)) + r_1c(\theta_2)s(\theta_1)) + 2r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))(v_{r2}s(\delta_1 - \theta_1)(r_2c(\theta_1)c(\theta_2) - r_1 + \\
& r_2s(\theta_1)s(\theta_2)) + r_2c(\delta_1 - \theta_1)(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))))/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2)) \\
& + ((c(\delta_1 - \theta_1)(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) + r_2s(\delta_1 - \theta_1)(c(\theta_1)s(\theta_2) - \\
& c(\theta_2)s(\theta_1)))(2v_{r1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) + 2v_{r2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\
& 2r_1r_2(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))(v_{\theta 1}/r_1 - v_{\theta 2}/r_2))/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2)) - \\
& ((2v_{r2}(v_{\theta 1}(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) - c(\theta_1)s(\theta_2) + v_{\theta 2}(r_1c(\theta_1)c(\theta_2) - r_2 + r_1s(\theta_1)s(\theta_2)) + \\
& r_1c(\theta_2)s(\theta_1)) + 2r_2v_{r1}(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1)))(v_{r2}s(\delta_1 - \theta_1)(r_2c(\theta_1)c(\theta_2) - r_1 + r_2s(\theta_1)s(\theta_2)) + \\
& r_2c(\delta_1 - \theta_1)(c(\theta_1)s(\theta_2) - c(\theta_2)s(\theta_1))))/(r_1^2 + r_2^2 - 2r_1r_2s(\theta_1)s(\theta_2) - 2r_1r_2c(\theta_1)c(\theta_2)))) \quad (84)
\end{aligned}$$

The control law for  $F_2$  is from the dynamic inversion of  $y_2$ , where  $y_2$  is defined as

$$y_2 = v_{\theta 2}^2 - s^2(\psi_2/2) (v_{r2}^2 + v_{\theta 2}^2) \quad (85)$$

and the control input  $a_2$  appears in the first derivative of  $y_2$

$$\begin{aligned}
y_2' = & 2v_{\theta 2}(a_2s(\theta_2 - \delta_2) - (v_{r2}v_{\theta 2})/r_2) - s(\psi_2/2)(2v_{\theta 2}(a_2s(\theta_2 - \delta_2) - (v_{r2}v_{\theta 2})/r_2) - 2v_{r2}(a_2c(\theta_2 - \delta_2) \\
& - v_{\theta 2}^2/r_2)) - (\psi_2'c(\psi_2/2)(v_{r2}^2 + v_{\theta 2}^2))/2 \quad (86)
\end{aligned}$$

solving for  $a_2$  yields

$$a_2 = (k_2(s(\psi_2/2)^2(v_{r2}^2 + v_{\theta 2}^2) - v_{\theta 2}^2) + (2v_{r2}v_{\theta 2}^2)/r_2 + \psi_2'c(\psi_2/2)s(\psi_2/2)(v_{r2}^2 + v_{\theta 2}^2))/(s(\psi_2/2)^2(2v_{r2}c(\delta_2 - \theta_2) + 2v_{\theta 2}s(\delta_2 - \theta_2)) - 2v_{\theta 2}s(\delta_2 - \theta_2)) \quad (87)$$

equations (83), (84) and (87) are the analytical equations for the accelerations of  $F_1$  and  $F_2$  that can be used to cause contact between the three sensors when they have non circular sensing zones.

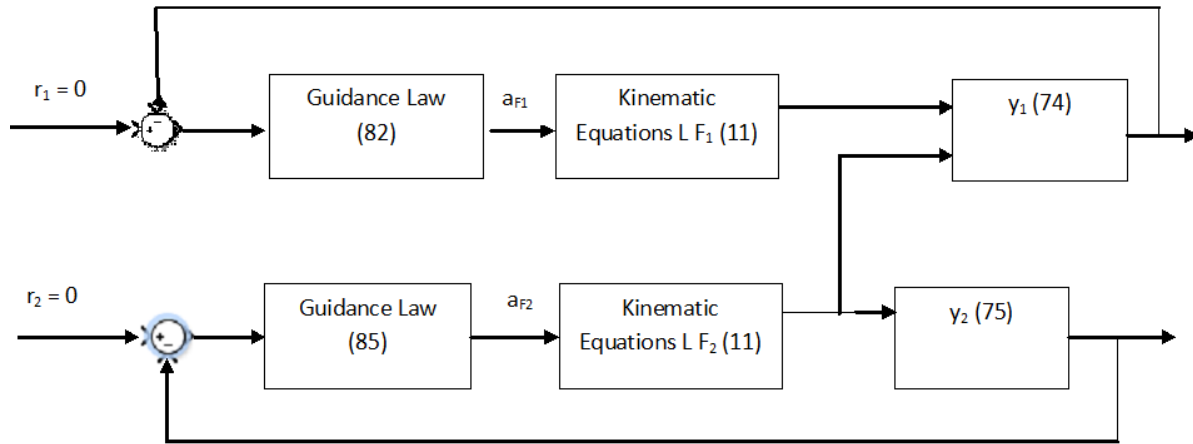


Figure 22 Block diagram of the control law used to control  $F_1$  and  $F_2$  for the non circular sensing zone case.

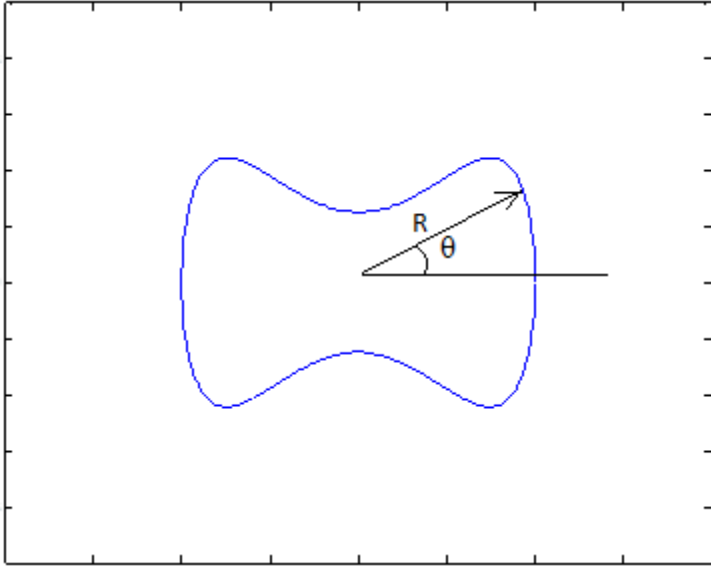
### Example

For the purpose of simulation, a sensor coverage shape must be defined. The shape that is chosen is defined by

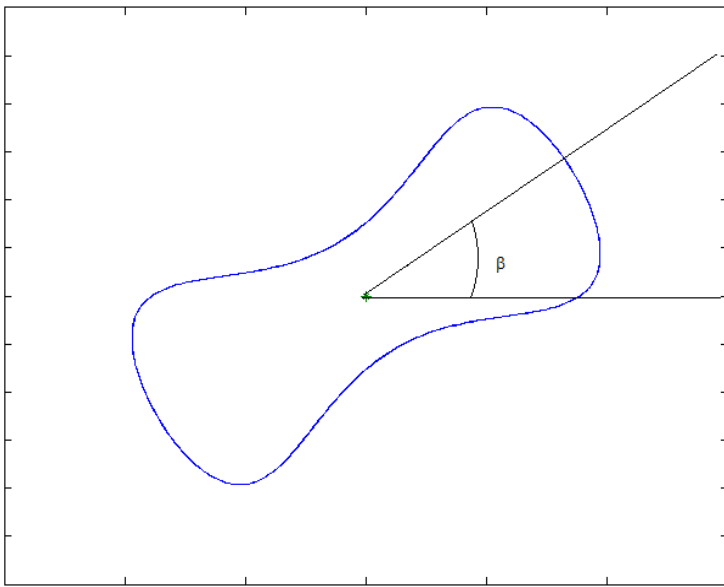
$$x = Rc(\theta) \quad (88)$$

$$y = Rs(\theta) - .75Rs(\theta)^3 \quad (89)$$

where  $R$  is a constant used to adjust the size of the sensor shape. The shape is shown in Figure 23.

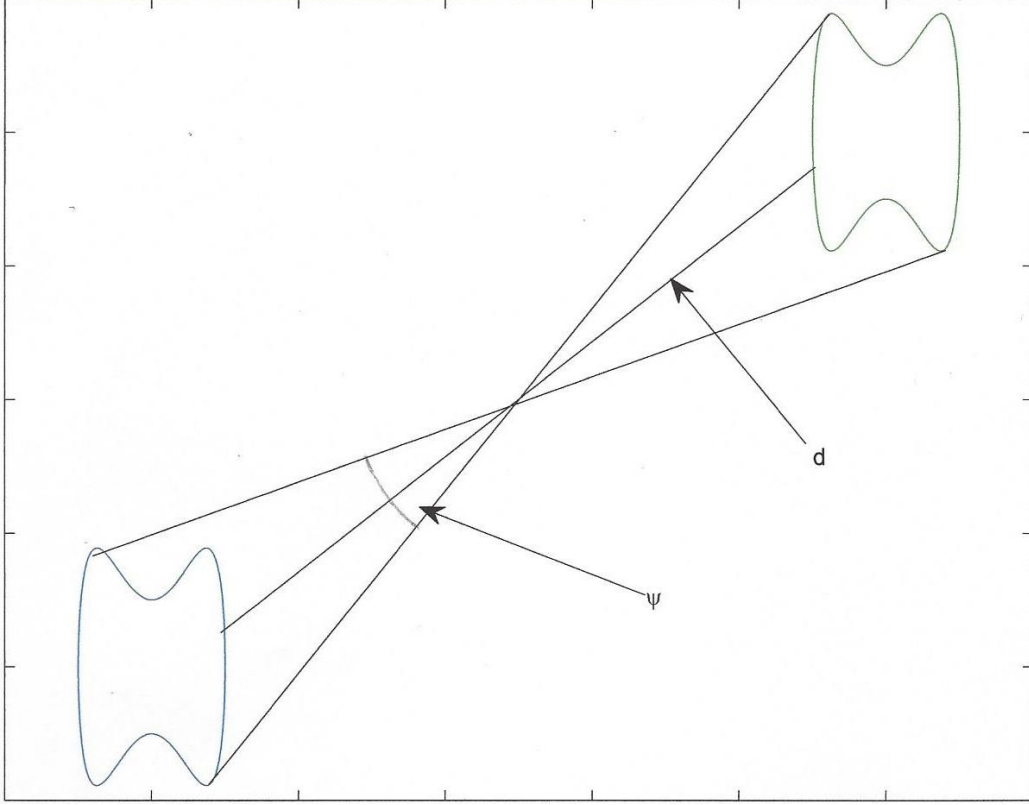


**Figure 23** Shape of the non-convex sensing zone used for simulation.



**Figure 24** Noncircular sensor rotated by an angle  $\beta$

For each sensor pair, an angle  $\psi$  is defined as the angle between two lines tangent to both sensors as shown in Figure 25. The distance between the closest points of each sensor,  $d$ , is also shown in Figure 25. Each sensor can also rotate about its center by an angle  $\beta$ . The calculation of both  $\psi$  and  $d$  depend on the shape of each sensor sensing zone, the orientation of each sensor, and their relative positions.



**Figure 25 Distance between closest points and angle between tangent lines.**

To determine  $\psi$ , the tangent lines of each sensor need to be calculated. First, the rotation about the center of each sensor,  $\beta$ , is added to the calculation of the x and y components of the perimeter of the sensing zone.

$$x = Rc(\theta)c(\beta) - (Rs(\theta) - .75Rs(\theta)^3)s(\beta) \quad (90)$$

$$y = Rc(\theta)s(\beta) + (Rs(\theta) - .75Rs(\theta)^3)c(\beta) \quad (91)$$

the distance,  $p$ , from the center of the sensor to the edge of its sensing zone is:

$$p = ((Rc(\theta)s(\beta) + (Rs(\theta) - .75Rs(\theta)^3)c(\beta))^2 + (Rc(\theta)c(\beta) - (Rs(\theta) - .75Rs(\theta)^3)s(\beta))^2)^{(1/2)} \quad (92)$$

then the derivative  $dp/d\theta$  is calculated:

$$dp/d\theta = (2(c(\beta)(Rs(\theta) - (3Rs(\theta)^3)/4) + Rc(\theta)s(\beta))(c(\beta)(Rc(\theta) - (9Rc(\theta)s(\theta)^2)/4) - Rs(\theta)s(\beta)) + 2(s(\beta)(Rs(\theta) - (3Rs(\theta)^3)/4) - Rc(\theta)c(\beta))(s(\beta)(Rc(\theta) - (9Rc(\theta)s(\theta)^2)/4) +$$

$$\frac{R_s(\theta)c(\beta))/((c(\beta)(R_s(\theta) - (3R_s(\theta)^3)/4) + R_c(\theta)s(\beta))^2 + (s(\beta)(R_s(\theta) - (3R_s(\theta)^3)/4) - R_c(\theta)c(\beta))^2)^{1/2}}{R_c(\theta)c(\beta))^2)^{1/2}} \quad (93)$$

the slope of the tangent lines are:

$$dy/dx = (dp/d\theta \ s(\theta)+pc(\theta))/(dp/d\theta c(\theta)-ps(\theta)) \quad (94)$$

which for the sensor shape chosen becomes:

$$\begin{aligned} dy/dx = & -(c(\theta)((c(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) + R_c(\theta)s(\beta))^2 + (s(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) - \\ & R_c(\theta)c(\beta))^2)^{1/2} + (s(\theta)(2(c(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) + R_c(\theta)s(\beta))(c(\beta)(R_c(\theta) - (9R_c(\theta)s^2(\theta))/4) - \\ & R_s(\theta)s(\beta)) + 2(s(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) - R_c(\theta)c(\beta))(s(\beta)(R_c(\theta) - (9R_c(\theta)s^2(\theta))/4) + \\ & R_s(\theta)c(\beta))))/(2((c(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) + R_c(\theta)s(\beta))^2 + (s(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) - \\ & R_c(\theta)c(\beta))^2)^{1/2}))/((s(\theta)((c(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) + R_c(\theta)s(\beta))^2 + (s(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) - \\ & R_c(\theta)c(\beta))^2)^{1/2} - (c(\theta)(2(c(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) + R_c(\theta)s(\beta))(c(\beta)(R_c(\theta) - (9R_c(\theta)s^2(\theta))/4) - \\ & R_s(\theta)s(\beta)) + 2(s(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) - R_c(\theta)c(\beta))(s(\beta)(R_c(\theta) - (9R_c(\theta)s^2(\theta))/4) + \\ & R_s(\theta)c(\beta))))/(2((c(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) + R_c(\theta)s(\beta))^2 + (s(\beta)(R_s(\theta) - (3R_s^3(\theta))/4) - \\ & R_c(\theta)c(\beta))^2)^{1/2})) \quad (95) \end{aligned}$$

then  $dy/dx$  for each sensor is equated to the  $\Delta y/\Delta x$  for each sensor sensing perimeter and the tangent points and slopes of the tangent lines are found.

then  $\psi$  is found to be:

$$\psi = (\tan^{-1}(((m_1-m_2)/(1+(m_1m_2)))))) \quad (96)$$

where  $m_1$  and  $m_2$  are the slopes of the tangent lines. During simulation,  $\psi_1'$  and  $\psi_2'$  are calculated numerically.

Next, the minimum distance between sensors needs to be found. If the x and y coordinates of each sensor sensing perimeter is

$$x_1 = R_1c(\theta_1)c(\beta_1)-(R_1s(\theta_1)-.75R_1s(\theta_1)^3)s(\beta_1) + x_{1o} \quad (97)$$

$$y_1 = R_1c(\theta_1)s(\beta_1)-(R_1s(\theta_1)-.75R_1s(\theta_1)^3)c(\beta_1) + y_{1o} \quad (98)$$

$$x_2 = R_2c(\theta_2)c(\beta_2)-(R_2s(\theta_2)-.75R_2s(\theta_2)^3)s(\beta_2) + x_{2o} \quad (99)$$

$$y_2 = R_2c(\theta_2)s(\beta_2)-(R_2s(\theta_2)-.75R_2s(\theta_2)^3)c(\beta_2) + y_{2o} \quad (100)$$

where  $x_{1o}$ ,  $x_{2o}$ ,  $y_{1o}$  and  $y_{2o}$  are the x and y coordinates of the center of each sensor,  $\beta_1$  and  $\beta_2$  are the rotations that each sensor has undergone, and  $\theta_1$  and  $\theta_2$  define where along the perimeter the points  $(x_1, y_1)$  and  $(x_2, y_2)$  are located. Then, the distance between any point on the one sensing perimeter to any point on the other sensing perimeter is:

$$D = ((x_1-x_2)^2 + (y_1-y_2)^2)^{(1/2)} \quad (101)$$

then to find the minimum distance, the critical points at  $\partial D/\partial \theta_1 = 0$  and  $\partial D/\partial \theta_2 = 0$  are calculated.

$$\begin{aligned} \partial D/\partial \theta_1 = & -(2(s(\beta_1)(R_1c(\theta_1) - (9R_1c(\theta_1)s(\theta_1)^2)/4) + R_1c(\beta_1)s(\theta_1))(x_{1o} - x_{2o} - s(\beta_1)(R_1s(\theta_1) - \\ & (3R_1s(\theta_1)^3)/4) + s(\beta_2)(R_2s(\theta_2) - (3R_2s(\theta_2)^3)/4) + R_1c(\beta_1)c(\theta_1) - R_2c(\beta_2)c(\theta_2)) + 2(c(\beta_1)(R_1c(\theta_1) - \\ & (9R_1c(\theta_1)s(\theta_1)^2)/4) + R_1s(\beta_1)s(\theta_1))(y_{1o} - y_{2o} - c(\beta_1)(R_1s(\theta_1) - (3R_1s(\theta_1)^3)/4) + c(\beta_2)(R_2s(\theta_2) - \\ & (3R_2s(\theta_2)^3)/4) + R_1s(\beta_1)c(\theta_1) - R_2s(\beta_2)c(\theta_2)))/(2((x_{1o} - x_{2o} - s(\beta_1)(R_1s(\theta_1) - (3R_1s(\theta_1)^3)/4) + \\ & s(\beta_2)(R_2s(\theta_2) - (3R_2s(\theta_2)^3)/4) + R_1c(\beta_1)c(\theta_1) - R_2c(\beta_2)c(\theta_2))^2 + (y_{1o} - y_{2o} - c(\beta_1)(R_1s(\theta_1) - \\ & (3R_1s(\theta_1)^3)/4) + c(\beta_2)(R_2s(\theta_2) - (3R_2s(\theta_2)^3)/4) + R_1s(\beta_1)c(\theta_1) - R_2s(\beta_2)c(\theta_2))^2)^{(1/2)} \end{aligned} \quad (102)$$

$$\begin{aligned}
dD/d\theta_2 = & (2(s(\beta_2)(R_2c(\theta_2) - (9R_2c(\theta_2)s(\theta_2)^2)/4) + R_2c(\beta_2)s(\theta_2))(x_{1o} - x_{2o} - s(\beta_1)(R_1s(\theta_1) - \\
& (3R_1s(\theta_1)^3)/4) + s(\beta_2)(R_2s(\theta_2) - (3R_2s(\theta_2)^3)/4) + R_1c(\beta_1)c(\theta_1) - R_2c(\beta_2)c(\theta_2)) + 2(c(\beta_2)(R_2c(\theta_2) - \\
& (9R_2c(\theta_2)s(\theta_2)^2)/4) + R_2s(\beta_2)s(\theta_2))(y_{1o} - y_{2o} - c(\beta_1)(R_1s(\theta_1) - (3R_1s(\theta_1)^3)/4) + c(\beta_2)(R_2s(\theta_2) - \\
& (3R_2s(\theta_2)^3)/4) + R_1s(\beta_1)c(\theta_1) - R_2s(\beta_2)c(\theta_2)))/(2((x_{1o} - x_{2o} - s(\beta_1)(R_1s(\theta_1) - (3R_1s(\theta_1)^3)/4) + \\
& s(\beta_2)(R_2s(\theta_2) - (3R_2s(\theta_2)^3)/4) + R_1c(\beta_1)c(\theta_1) - R_2c(\beta_2)c(\theta_2))^2 + (y_{1o} - y_{2o} - c(\beta_1)(R_1s(\theta_1) - \\
& (3R_1s(\theta_1)^3)/4) + c(\beta_2)(R_2s(\theta_2) - (3R_2s(\theta_2)^3)/4) + R_1s(\beta_1)c(\theta_1) - R_2s(\beta_2)c(\theta_2))^2)^{(1/2)}) \quad (103)
\end{aligned}$$

then all critical points that are between the tangent points on each sensor are checked to find the minimum distance,  $d$ . When  $d$  equals zero, the two sensor zones are in contact. During simulation,  $d$  can approach zero but typically never reaches it. When the sensing zone begin to overlap,  $d$  is greater than zero, and increasing, so the stopping condition that is used for simulation is when  $d$  is less than a threshold, which for these simulations was chosen to be .05 meters, and increasing.  $F_2$  needs to only overlap with  $L$ , so when the minimum distance between them is less that the threshold, then the relative motion is stopped.  $F_1$  needs to overlap both  $L$  and  $F_2$ , which may require a larger overlap with one sensor or the other in order to overlap with both. If the minimum distance from  $F_1$  to  $L$  is increasing and  $v_{r1} < 0$ , or if the minimum distance is decreasing and  $v_{r1} > 0$ , then  $F_1$  must overlap  $L$ . If the minimum distance from  $F_1$  to  $F_2$  is increasing and  $v_{r12} < 0$ , or if the minimum distance from  $F_1$  to  $F_2$  is increasing and  $v_{r12} > 0$ , then  $F_1$  must overlap  $F_2$ . The stopping condition for  $F_1$  is reached when either the minimum is less than the threshold, or an overlap is detected for both sensors as in Figure 26.

A pseudo code for the stopping condition is:

if minimum distance from  $F_1$  to  $L$  < threshold and minimum distance from  $F_1$  to  $F_2$  < threshold

```
        stop relative motion of  $F_1$ 
else
     $a_1 = \text{equation (82)}$ 
endif
if minimum distance from  $F_2$  to  $L < \text{threshold}$ 
    stop relative motion of  $F_2$ 
else
     $a_2 = \text{equation (85)}$ 
endif
```

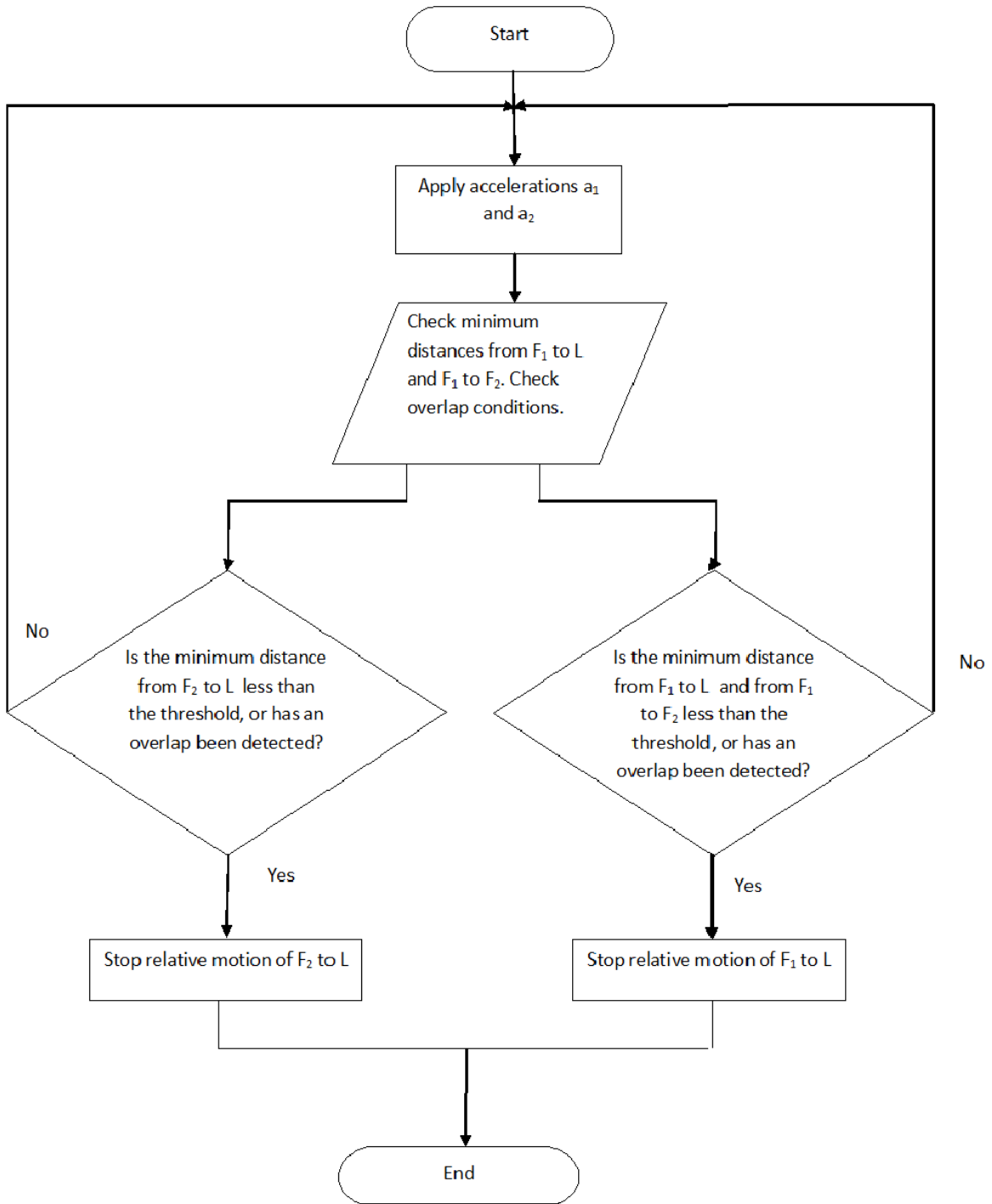


Figure 26 Control algorithm used for the noncircular sensing zone problem.

## CHAPTER 5

### NON CIRCULAR SENSING ZONE SIMULATION RESULTS

The simulation results indicate that from various initial conditions in Table 2. the sensors all come together such that their sensing perimeter either touch or overlap. For each sensor, the rotation angle  $\beta$  is set equal to  $\alpha$ , the angle of the velocity vector relative to the horizon.

Table 2  
Initial conditions used for the non circular sensor simulations

	Case 1	Case 2	Case 3	Case 4
Leader initial x position	15 m	15 m	15 m	15 m
Leader initial y position	18 m	18 m	18 m	18 m
Leader final x position	15 m	15 m	15 m	20 m
Leader final y position	18 m	18 m	18 m	15 m
$r_1(0)$	10.8087 m	15.8087 m	12.0278 m	14.8924 m
$r_2(0)$	7.6301 m	12.7648 m	10.7648 m	10.0333 m
$\theta_1(0)$	.2561 rad	2.3065 rad	-.7296 rad	.281 rad
$\theta_2(0)$	-2.72 rad	.561 rad	-2.647rad	-1.2866 rad
$v_{\theta 1}(0)$	-.5 m/s	-.2 m/s	-.1 m/s	0 m/s
$v_{\theta 2}(0)$	0 m/s	.13 m/s	0 m /s	0 m/s
$v_{r1}(0)$	-1 m/s	-1 m/s	-1 m/s	-1 m/s
$v_{r2}(0)$	-1 m/s	-1 m/s	-1 m/s	-1 m/s
$\delta_1$	0 rad	0 rad	0 rad	0 rad
$\delta_2$	0 rad	0 rad	0 rad	0 rad

Table 2 (continued)

$k_1$	1	1	1	1
$k_2$	1	1	1	1

Case 1:

$F_1$  starts below and to the left of the leader, while  $F_2$  starts above and to the right of the leader.

The lead sensor is stationary, which is not limiting in any way because the control laws equations (83), (84), and (87) are based on the relative positions and velocities of the sensors.

Here  $F_2$  rendezvous with L and stops the relative motion of  $F_2$  to L when the coverage zone of  $F_2$  overlaps with the coverage zone of L.  $F_1$  rendezvous with both L and  $F_2$ , and stops the relative motion of  $F_1$  to L and  $F_2$  when the coverage zone of  $F_1$  overlaps the coverage zones of both L and  $F_2$ .

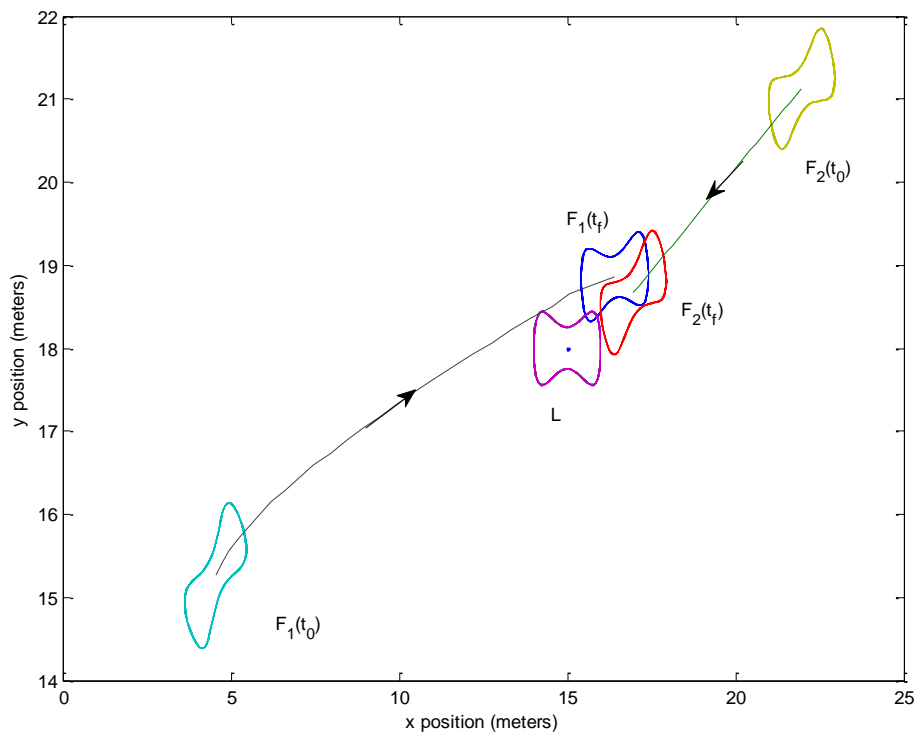


Figure 27 Case 1: Initial and final positions of each sensor.

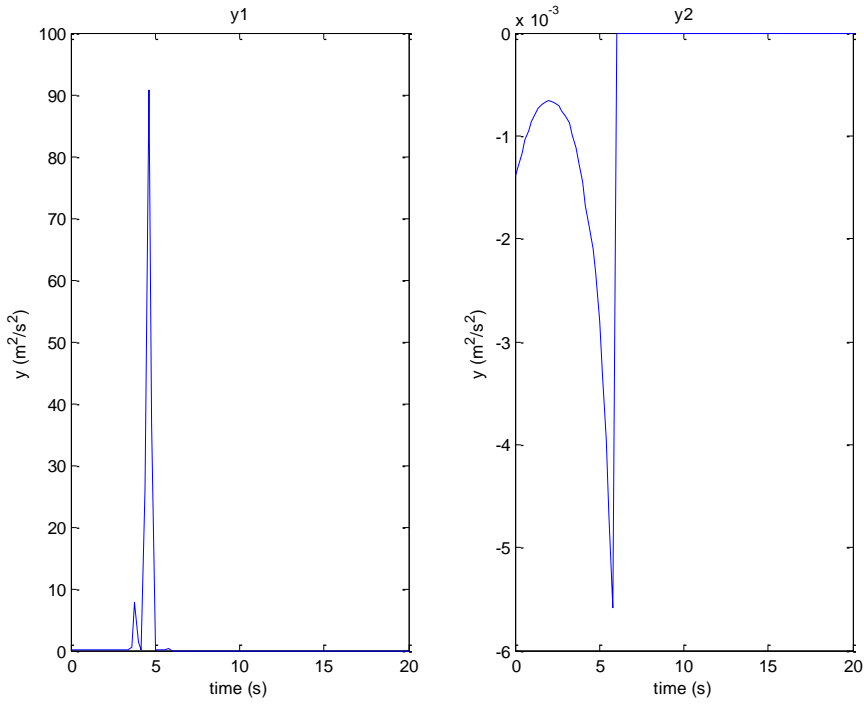


Figure 28 Case 1: Value of the outputs  $y_1$  and  $y_2$  used to develop the dynamic inversion control laws.

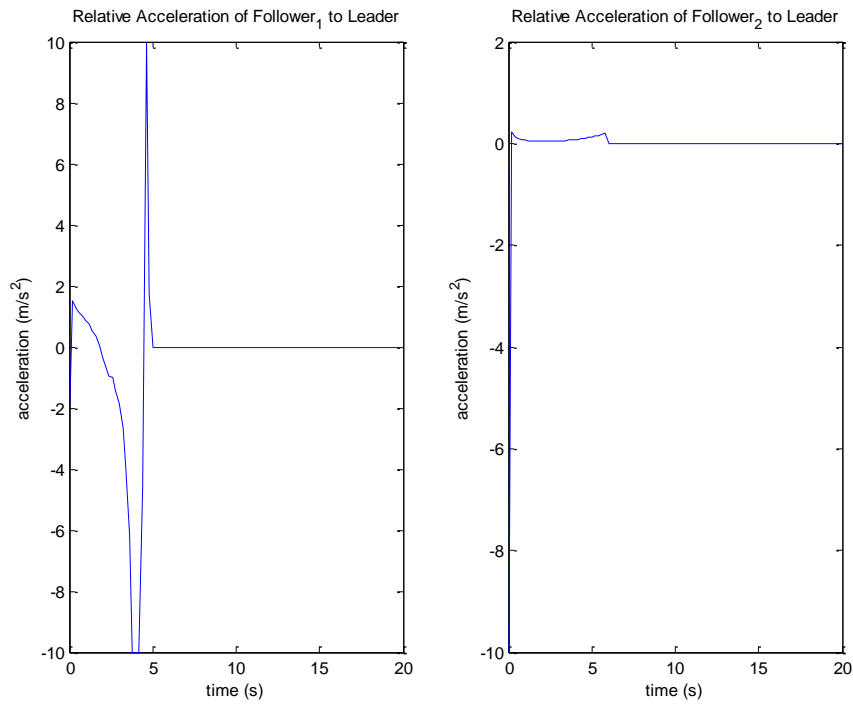
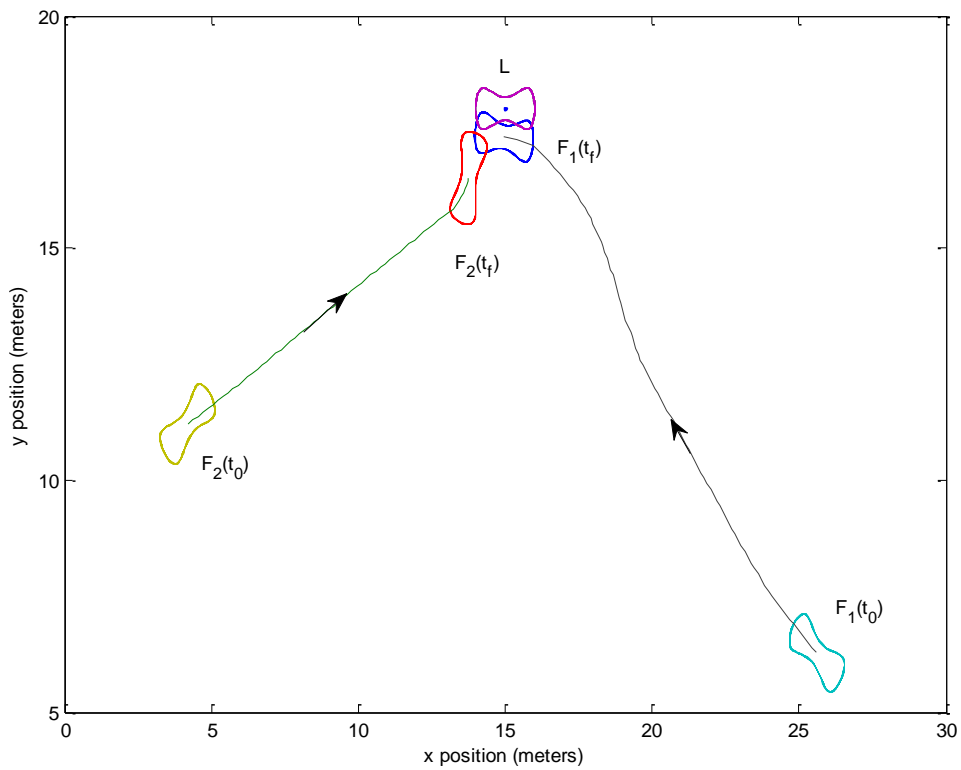


Figure 29 Case 1: Acceleration of Follower<sub>1</sub> and Follower<sub>2</sub>.

Case 2:

L is stationary,  $F_1$  begins below and to the right of L, and  $F_2$  begins below and to the left of L.  $F_2$  rendezvous with L and stops when the coverage zones touch.  $F_1$  rendezvous first with L, and then continues on to rendezvous with  $F_2$ . The coverage zone of  $F_1$  overlaps the coverage zone of L when the coverage zone of  $F_1$  touches the coverage zone of  $F_2$  and stops.



**Figure 30 Case 2: Initial and final positions of each sensor.**

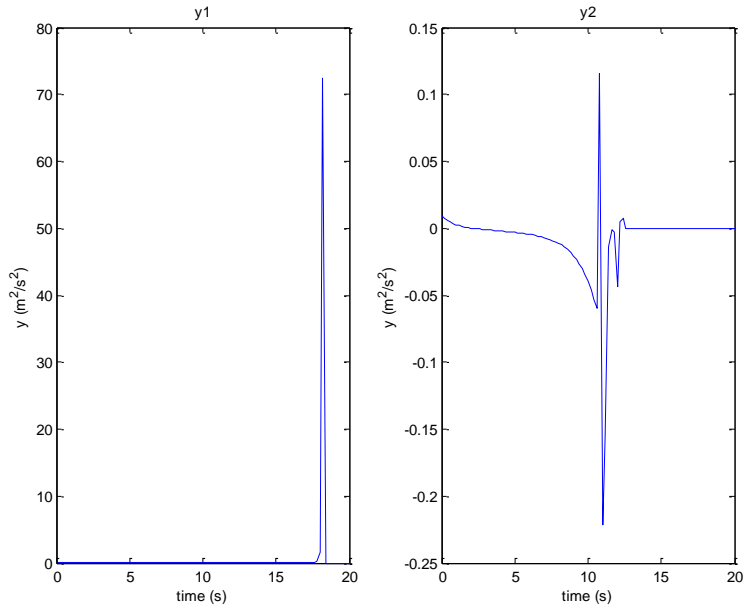


Figure 31 Case 2: Value of the outputs  $y_1$  and  $y_2$  used to develop the dynamic inversion control laws.

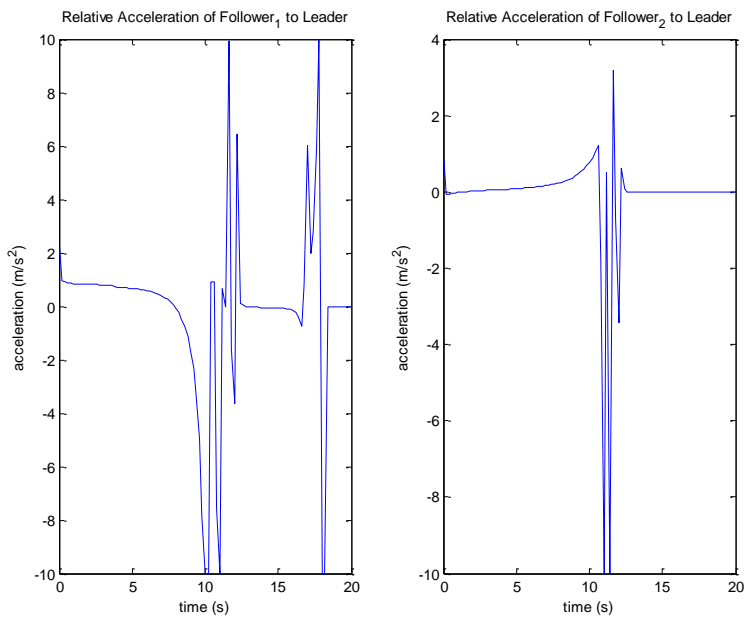
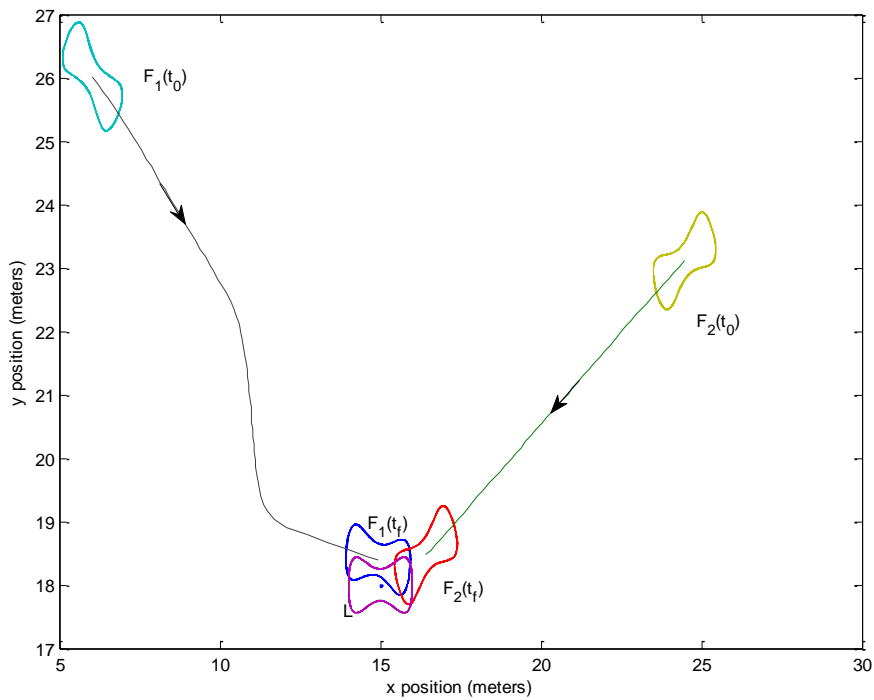


Figure 32 Case 2: Acceleration of Follower<sub>1</sub> and Follower<sub>2</sub>.

Case 3:

L is stationary,  $F_1$  begins above and to the left of L, and  $F_2$  begins above and to the right of L.  $F_2$  rendezvous with L and stops when the sensor coverage zones touch.  $F_1$  rendezvous with both L and  $F_2$  and stops when the sensor coverage zone of  $F_1$  touches both the sensor coverage zones of L and  $F_2$ .



**Figure 33 Case 3: Initial and final positions of each sensor.**

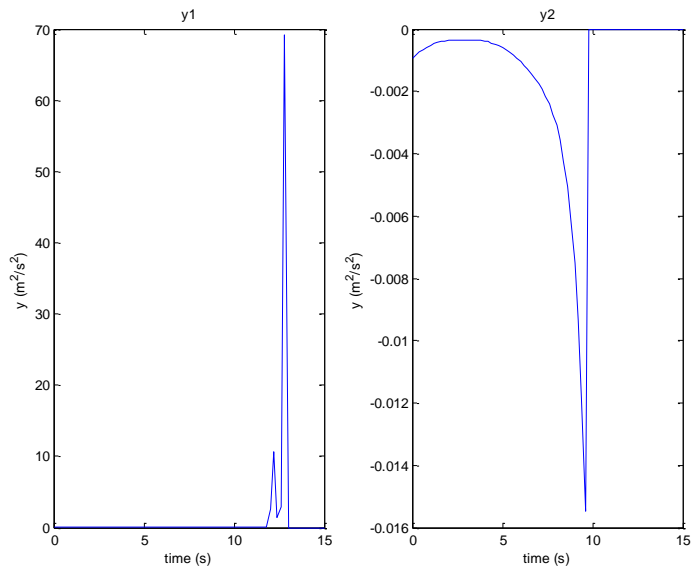


Figure 34 Case 3: Value of the outputs  $y_1$  and  $y_2$  used to develop the dynamic inversion control laws.

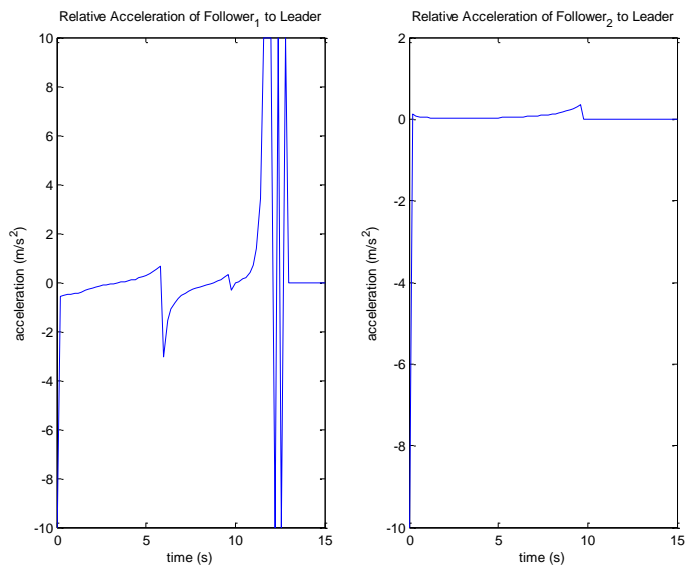
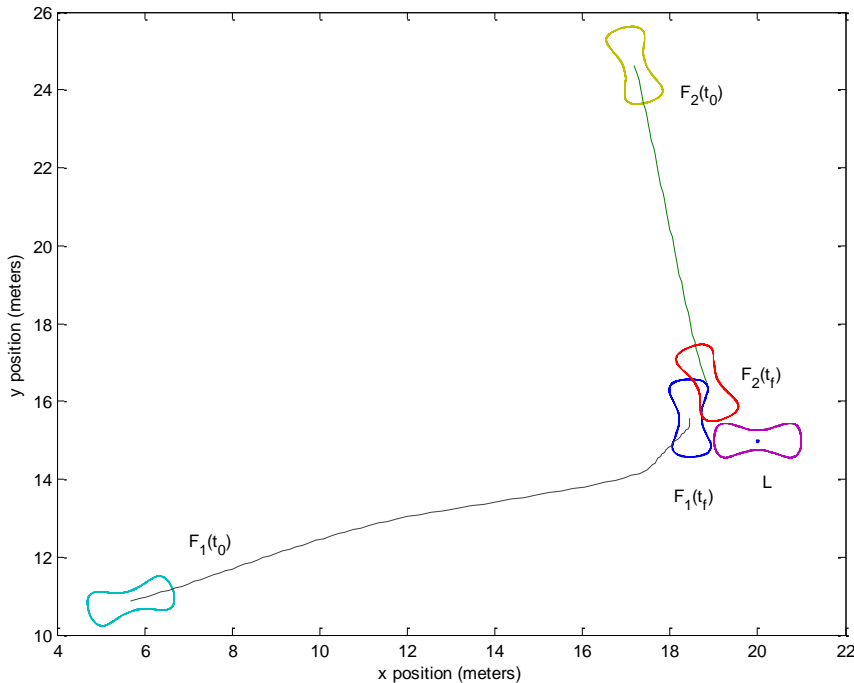


Figure 35 Case 3: Acceleration of Follower<sub>1</sub> and Follower<sub>2</sub>.

Case 4:

$F_1$  begins below and to the left of L, and  $F_2$  begins above and to the left of L.  $F_2$  begins on a trajectory to rendezvous with L stopping when the coverage zone of  $F_2$  touches the coverage zone of L.  $F_1$  begins on a trajectory towards L and  $F_2$ , and as both L and  $F_2$  move, and all three sensors rotate, the trajectory is corrected to complete the rendezvous with both L and  $F_2$ .  $F_1$  rendezvous first with L, and then continues on to rendezvous with  $F_2$ . The coverage zone of  $F_1$  touches the coverage zone of L while overlapping the coverage zone of  $F_2$ , reaching its stopping condition.



**Figure 36 Case 4: Initial and final positions of each sensor.**

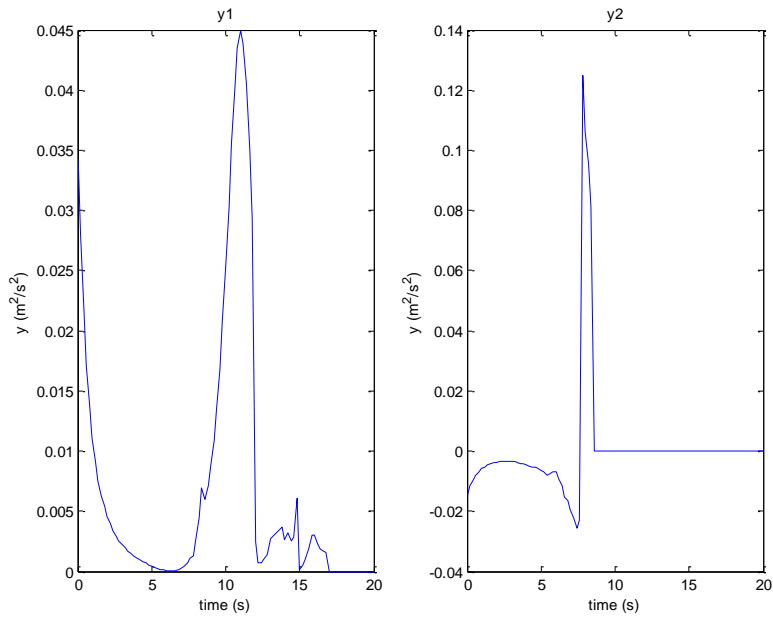


Figure 37 Case 4: Value of the outputs  $y_1$  and  $y_2$  used to develop the dynamic inversion control laws.

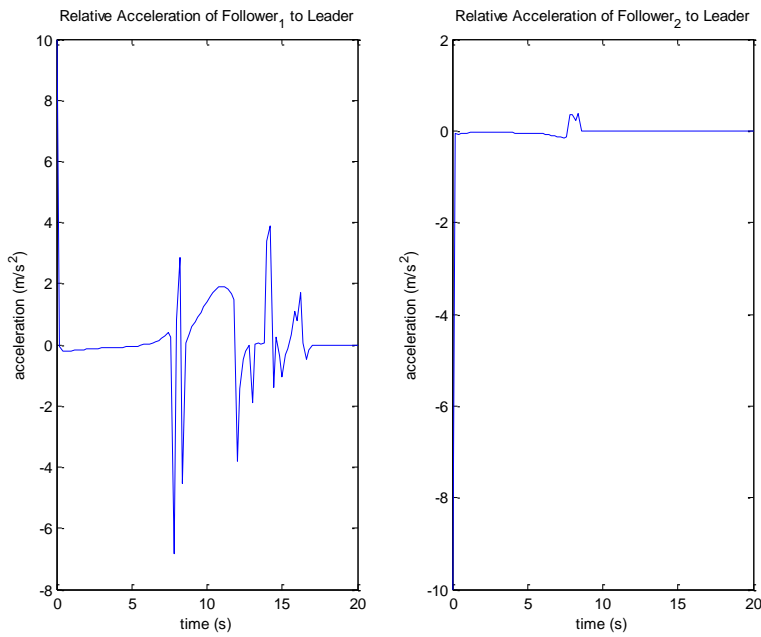


Figure 38 Case 4: Acceleration of Follower<sub>1</sub> and Follower<sub>2</sub>.

For each test case the three sensors assembled such that the sensing zone of each touches.



## **CHAPTER 6**

### **CONCLUSIONS**

In this thesis, two analytical control laws were developed to position three omnidirectional sensors such that there is no gap in coverage between the three sensors. These control laws are developed using the dynamic inversion of the perimeter of the triangle formed by the vertices of each sensor. The result is that accelerations are applied to the two follower sensors so that each follower moves towards the lead sensor and the other follower sensor. The three omnidirectional sensors are able to autonomously assemble in a triangle that has no gaps in coverage between the three sensors.

A second set of control laws are developed for the case when the three sensors are not omnidirectional based on the collision cone approach. This second set of control laws has one control law that applies an acceleration to one of the follower sensors that puts it on a course to rendezvous with the leader sensor and guides it until the sensing area of the follower overlaps the sensing area of the leader. When the sensing area of the follower sensor overlaps the sensing area of the leader, the follower sensor reaches its stopping condition. The other control law is designed to put the follower sensor on a course to rendezvous with both the leader and the other follower. Once the sensing area of this follower overlaps both the sensing area of the leader and the sensing area of the other follower, then it has reached its stopping condition.

There are several possibilities for future work. The control laws were developed assuming all sensors are on a 2-D plane. The problem could be expanded to the third dimension. Another possibility would be to increase the total number of sensors. There are a couple of possible approaches if the number of sensors is increased. One option would be to develop control laws such that if four sensors are used, they would assemble in a quadrilateral with no

coverage gap rather than a triangle, or if five sensors are chosen they would assemble in a pentagon with no coverage gap, etc. Another approach would be to use the same control laws developed in this paper, and designate one sensor as the global leader which would have its follower<sub>1</sub> and follower<sub>2</sub>, and then one of those followers could be designated as the local leader for another set of followers, and this approach could be extended to include any number of sensors.

**WORKS CITED**

## WORKS CITED

- [1] D. S. Loughlin, "Eyjafjallajökull eruption, Iceland | April/May 2010," British Geological Survey, 9 August 2010. [Online]. Available: [http://www.bgs.ac.uk/research/volcanoes/icelandic\\_ash.html](http://www.bgs.ac.uk/research/volcanoes/icelandic_ash.html). [Accessed 7 April 2015].
- [2] P. J. Marson, *The Lockheed Constellation Series*, Tonbridge, Kent: Air-Britain, 1982.
- [3] C. Liu and G. Cao, "Distributed Critical Location Coverage in Wireless Sensor Networks with Lifetime Constraint," in *IEEE Infocom 2012*, Orlando, 2012.
- [4] The Ocean Portal Team, "Ocean Portal," Smithsonian National Museum of Natural History, 2013. [Online]. Available: <http://ocean.si.edu/gulf-oil-spill>. [Accessed 6 April 2015].
- [5] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*, Princeton: Princeton University Press, 2010, pp. 176-188.
- [6] M.-c. Zhao, J. Lei, M.-Y. Wu, Y. Liu and W. Shu, "Surface Coverage in Wireless Sensor Networks," in *IEEE Infocom 2009*, Rio de Janeiro, 2009.
- [7] J. Rogge and D. Aeyels, "Sensor Coverage with a Multi-Robot System," in *IEEE International Symposium on Intelligent Control Part of IEEE Multi-Conference on System and Control*, Singapore, 2007.
- [8] A. H. R. Ko, A.-L. Joussetme and P. Maupin, "A Coverage Dominance Approach for Sensor Deployment Optimization," in *Proceedings of the 14th International Conference on Information Fusion (FUSION)*, Chicago, 2011.
- [9] Y. Liu and W. Liang, "Approximate Coverage in Wireless Sensor Networks," in *The IEEE Conference on Local Computer Networks*, Sydney, 2005.
- [10] S. Rahili and W. Ren, "Game Theory Control Solution for Sensor Coverage Problem in Unknown Environment," in *IEEE 53rd Annual Conference on Decision and Control*, Los Angeles, 2014.

- [11] X. Xu, S. Sahni and N. S. V. Rao, "Minimum-Cost Sensor Coverage of Planar Regions," in *11th International Conference on Information Fusion*, Cologne, 2008.
- [12] V. Gupta, D. E. Jeffcoat and R. M. Murray, "On Sensor Coverage by Mobile Sensors," in *45th IEEE Conference of Decision & Control*, San Diego, 2006.
- [13] L. Paull, C. Thibault, A. Nagaty, M. Seto and H. Li, "Sensor-Driven Area Coverage for an Autonomous Fixed-Wing Unmanned Aerial Vehicle," *IEEE Transaction on Cybernetics*, vol. 44, no. 9, pp. 1605-1618, 2014.
- [14] X. Gong, Z. Junshan and D. Cochran, "When Target Motion Matters: Doppler Coverage in Radar Sensor Networks," in *2013 Proceedings IEEE INFOCOM*, Turin, 2013.
- [15] W. Abbas and M. Egerstedt, "Distribution of Agents in Heterogeneous Multiagent Systems," in *50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, 2011.
- [16] R. Bellazreg, M. Hamdi and N. Boudriga, "On the Impact of Irregular Radio Propagation on Coverage Control and Sleep Scheduling in Wireless Sensor Networks," in *2010 IEEE/ACS International Conference on Computer Systems and Applications*, Hammamet, 2010.
- [17] S. Martinez, J. Cortes and F. Bullo, "Motion Coordination with Distributed Information," *IEEE Control Systems Magazine*, vol. 27, no. 4, pp. 75-88, 2007.
- [18] B. Shucker, T. D. Murphey and J. K. Bennett, "Convergence-Preserving Switching for Topology-Dependent Decentralized Systems," *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1405-1415, 2008.
- [19] B. Shucker, T. Murphey and J. K. Bennett, "A Method of Cooperative Control Using Occasional Non-Local Interactions," in *IEEE Conference on Robotics and Automation*, Orlando, Florida, 2006.

- [20] N. A. B. Ab Aziz, A. W. Mohemmed and B. D. Sagar, "Particle Swarm Optimization and Voronoi Diagram for Wireless Sensor Networks Coverage Optimization," in *International Conference on Intelligent and Advanced Systems*, Kuala Lumpur, 2007.
- [21] A. Dirafzoon, S. M. A. Salahizadeh, S. Emrani and M. B. Menhaj, "Virtual Force Based Individual Particle Optimization for Coverage in Wireless Sensor Networks," in *Electrical and Computer Engineering (CCECE), 2010 23rd Canadian Conference*, Calgary, 2010.
- [22] A. Chakravarthy and D. Ghose, "Collision Cones for Quadric Surfaces," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1159-1166, 2011.
- [23] A. Chakravarthy and D. Ghose, "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, pp. 562-574, 1998.
- [24] A. Chakravarthy and D. Ghose, "Generalization of the Collision Cone Approach for Motion Safety in 3-D Environmets," *Autonomous Robots*, vol. 32, no. 3, pp. 243-266, 2012.
- [25] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*, Upper Saddlle River, New Jersey: Prentice-Hall, Inc., 1991, pp. 213-218.
- [26] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*, 2nd ed., Hoboken, New Jersey: John Wiley & Sons, Inc., 2003, pp. 484-487.
- [27] C. B. Moler, *Numerical Computing with MATLAB*, Philadelphia: Society for Industrial and Applied Mathematics, 2004, pp. 150-151.
- [28] J. W. Anderson, *Hyperbolic Geometry*, 2nd ed., London: Springer, 2005.
- [29] J. Ratcliffe, *Foundations of Hyperbolic Manifolds*, 2nd ed., New York: Springer, 2006, pp. 80-99.

- [30] J. W. Harris and H. Söcker, "Segment of a Circle," in *Handbook of Mathematics and Computational Science*, New York, Springer-Verlag, 1998, pp. 92-93.
- [31] D. Heinrich, *100 Great Problems of Elementary Mathematics*, Dover: Courier Corporation, 1965.
- [32] K. Kendig, "Is a 2000-Year-Old Formula Still Keeping Some Secrets?," *The American Mathematical Monthly*, vol. 107, no. 5, pp. 402-415, 2000.
- [33] M. Ji and M. Egerstedt, "Distributed Coordination Control of Multiagent Systems While Preserving Connectedness," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 693-703, 2007.