Leader-Follower SLAM Based Navigation and Fleet Management Control

BY

Omar Salem Al Buraiki

A Thesis Presented to the

DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

In

Systems Engineering

December, 2012

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN- 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **Omar Salem Al-Buraiki** under the direction of his thesis advisor and approved by his thesis committee, has been presented and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **Master of Science in Systems Engineering.**

Thesis Committe

17/12/12

Dr.Sami El Ferik
(Advisor)

Dec 17, 12

Dr. Fouad Al-Sunni
Department Chairman

Dec 17, 12

Dr. Fouad Al-Sunni
(Member)

M. Faizan 17/12/12

Dr. Mohammed F. Maysorwalla
(Member)

Dr. Salam A. Zummo
Dean of Graduate Studies

عمادة الدراسات العليا
DEANSHIP OF GRADUATE STUDIES

18/12/12

Date

LEADER FOLLOWER SLAM BASED NAVIGATION AND FLEET MANAGEMENT CONTROL

Omar Salem Al Buraiki

Master Thesis

Submitted for the Partial Fulfillment of the Requirements for the Degree of

Master of Science

SYSTEMS ENGINEERING DEPARTMENT

December, 2012

To my parents

Preface

This thesis is submitted to King Fahd University of Petroleum and Minerals (KFUPM) for partial fulfillment of the requirements for the degree of Master of Science**.**

ACKNOWLEDGMENTS

My unreserved praises and thankfulness are for Allah, the Most Compassionate, and the Most Merciful. He blessed me with his bounties. May his peace and blessing be upon the prophet Muhammad, and his family.

Acknowledgment is also due to King Fahd University of Petroleum and Minerals and Hadhramout Establishment for Human Development (HEHD) for scholarship and supporting me during master program.

It is a pleasure to thank those who made this thesis possible. In the following lines some of them are gratefully acknowledged. However, these words cannot express the gratitude and respect I feel for all of those conserned.

Firstly, I would like to thank my advisor Dr. Sami El Ferik for guiding me in the right directions to make this work possible. He was always kind, understanding and sympathetic twords me.

I also would like to thank Dr. Fouad Al-Sunni and Dr. M. Faizan Mysorewala for serving on my committee. In addition, I would like to thank my friends for their continuous support, advice and encouragement.

Last but not least, I would like to show my gratitude to my family. I am particularly indebted to my parents for their never-ending encouragement and on-going support. Very special thanks go to my beloved wife, beloved sons, brothers and sisters for always being there to support me with words of wholehearted encouragement.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

PD	:	Positive Definite.
SLAM	:	Simultaneous Localization and Mapping.
EKF	:	Extended Kalman Filter.
DOF	:	Degree of Freedom.
PF	:	Potential Fields.
NN	:	Neural Networks.
Da	:	Damping Factor.
P_{ce}	:	Center Potential.
P_{rep}	:	Repulsive Potential.
UAV	:	Unmanned Aerial Vechile.
FBL	:	Feedback Linearization.

ABSTRACT

Full Name : Omar Salem Mubarak Al Buraiki

Thesis Title : Leader Follower SLAM Based Navigation And Fleet Management Control

Major Field : Systems Engineering

Date of Degree : December, 2012

In this thesis, the problem of fleet formation under leader-follower strategy and the whole fleet navigation has been addressed using simultaneous localization and mapping (SLAM) navigation unit and artificial potential field formation approach. The framework is developed to control the navigation and formation of a fleet of nonholonomic robots. The proposed framework is developed by installing the SLAM navigation unit on the group leader and, on the assumption that the leader is located in the group center, the potential fields formation control strategy is used to localize all followers around their leader.

The formation control is developed such that the potential fields control inputs for the followers to control their desired positions around their leader. The nonlinearities of the nonholonomic robot is canceled using feedback linearization, and the case when all robot dynamics are known is considered.

The formation stability of each follower is analyzed using the Lyapunov method and its derivation is provided.

The formation control also designed on the assumption that all system dynamics are unknown. In this case on-line NN weight tuning algorithms used to estimate the robot

dynamics to guarantee tracking a desired following path. The followers will rack its desired trajectory which is generated based on the potential fields.

This system will be able to handle cooperative tasks like exploring unknown environments, delivering goods and so on.

ملخص الرسالة

الاسم الكامل: عمر سالم مبارك البريكي

عنوان الرسالة: تصميم نظام تحكم لإدارة اسطول من الروبوتات الاتوماتيكية تتموضع حول بعضها وفق نظام القائد والتابع وتحدد مسارها بإستخدام وحده ملاحة تزامنية

التخصص: هندسة نظم

تاريخ الدرجة العلمية: ديسمبر 2012 م

في هذه الأطروحة، تم معالجة مشكلة تشكيل أسطول من الربوتات بإتباع استراتيجية القائد والتابع في حين ان هذا الاسطول يستخدم نظام الملاحة التزامني لتحديد المسار. تم تطوير نظام للسيطرة والتحكم في تشكيل هذا الأسطول من الروبوتات. فقد تم تصميم النظام المقترح عن طريق تثبيت وحدة الملاحة SLAM على الروبوت الذي يستخدم قائدا للأسطول المفترض أن يقع في مركز القائد للمجموعة، وقد تم تطوير نظام للتحكم في تموضع الربوبوتات في هذا الاسطول حول بعضها البعض ولمنع تصادمها بناءا على استخدام خاصية مجالات الجذب والتباعد.

في حالة المعرفة التامة لديناميكا الربوتات تم إلغاء المركبات غير الخطية لكل روبوت بإستخدام التغذية الراجعة الخطية ثم تم تطبيق نظام تحكم التموضع والمسار. أما في حالة عدم معرفة ديناميكا الربوبوتات فقد تم استخدام الشبكات العصبية لتعريف ديناميكا الروبوت ومن ثم تم تطبيق نظام تحكم التموضع والمسار المقترح.

استقرار نظام التحكم المقترح تم تحليها واثباتها باستخدام طريقة ليابانوف. النظام الذي تم تصميمة قادرا على التعامل مع مهام قد تكون صعبة او معقدة مثل استكشاف الاماكن او البيئات المجهولة ودخول المناطق غير الامنه بالنسبة للبشر، وايضا عمليات النقل من مكان الى آخر وغيرها.

CHAPTER 1

INTRODUCTION

1.1 Motivation

The mobile robot can be defined as an automatic machine (electro-mechanical design) that is capable of movement in a given environment”. Recently, autonomous mobile robots have attracted the attention of a great number of researchers.

The researchers concentrate their work on robotics based on some agreements common to special features of the robots like robot’s tasks, such as moving around, operating mechanically, sensing, manipulating environments, and exhibiting smart behavior, especially when the robot is expected to behave like humans or animals, either mentally or physically.

Mobile robots are also found in industry, military and security environments. They also appear as consumer products, for entertainment or to perform certain tasks like vacuuming, gardening, delivering goods and some other common household tasks.

During the last decade, many researchers have dedicated their efforts to construct revolutionary machines and to provide them with some kind of artificial intelligence to perform some of the most unpleasant, risky or monotonous tasks historically assigned to humans.

An initial classification of mobile robotics based on their applications has been developed in the literature. There exist two types of robotics: 1) general-purpose autonomous robots; 2) dedicated robots. The general-purpose autonomous robots typically possess the ability to navigate autonomously within the environment, handling some basic tasks and communicating with humans or other robots. Dedicated robots concentrate on a specific service or task, such as car production, packaging and automated guided vehicles.

In recent years the multi-robot system has became more attractive to the interest of researchers because it has more advantages than a single robot system, such as the ability to finish complex tasks more efficiently than doing these tasks by using a single robot. For example, multiple robots can estimate their position faster and more accurately due to their ability of exchanging information related to their positions whenever they sense each other.

Many design strategies for controlling a group of mobile robots have been formulated based on the required application. In many applications the mobile robots have to work cooperatively to integrate certain tasks or actions. For example, when a group of robots is used to explore an unknown environment the returned data from their sensors will be better than using a single robot for this task.

A new framework for multiple cooperative manipulation tasks will be proposed. One of the robots will be able to lead the others in an unknown environment. At the same time, all the robots in the group will keep the desired formation with each other and with their group's leader, and the whole system will be integrated to handle approach, organization and transportation control modes.

This framework will be developed by a combination and improvement of two stages. The first one is mobile robot localization at each time step when the mobile robot follows its trajectory based on the current observation and the last available knowledge about the navigation environment. This process will be installed by running the simultaneous localization and mapping SLAM on the group leader, the problem of localization is addressed by using laser scanning [1] by updating the vehicle position information at high frequency. The objective of this stage is to place a mobile robot in an unknown location in an unknown environment and then the leader robot can incrementally build a map of this unknown environment, and at the same time use this map simultaneously to navigate autonomously.

Figure 1.1: SLAM mapping.

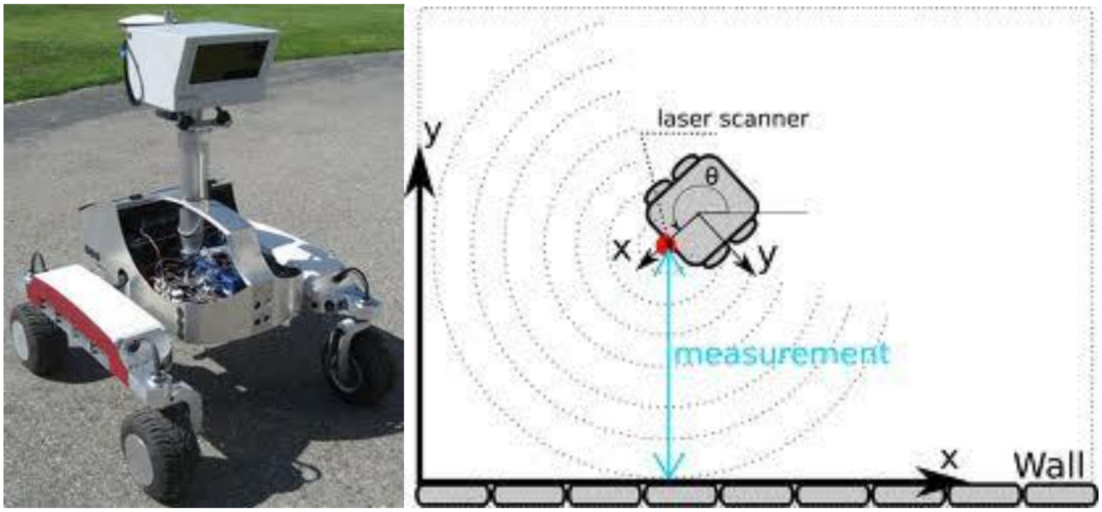


Figure 1.2: Robot equipped with laser scanner and (x, y) measurements.

The second stage is the group formation based on the artificial potential fields. The potential fields are a useful strategy for formation control consideration for a group of robots that allows one to control the behavior of a group of robots.

1.2 Problem formulations and contributions

This thesis will develop a new framework for handling cooperative tasks which will be built by integrating robot group formations based on the artificial potential fields combined with SLAM navigation. This approach is described in the following:

1. The new framework will extend Lorenzo *et al.* [32], from point mass holonomic agents to nonholonomic vehicles.
2. This framework allows one of the robots to lead the others in an unknown environment, assuming that the leader configuration is used as the group field center. At the same time all the agents in the fleet will keep their

formation shape based on the potential fields. The cost function (3.52), will be used for analyzing the formation stability.

3. The whole system will be integrated to handle cooperative tasks. (i.e. it will satisfy Song and Kumar framework [29])

1.3 Thesis Organization

This thesis will address the problem mentioned in the previous section, and it will be constructed as follows:

Chapter 2: Literature Review, summarizes the previous work in:

- ✓ SLAM.
- ✓ Potential fields for multi-robot group formation

Chapter3: Preliminaries, which will contain the general description for:

- ✓ Nonholonomic mobile robot modeling.
- ✓ SLAM overview.
- ✓ SLAM based laser pose tracking for simple robot model.
- ✓ Potential fields for multi-robot group formation.
- ✓ Neural networks control.

Chapter 4: New framework, there will be a design description for our proposed framework that can handle:

- ✓ Leader followers nonholonomic robots formation based on potential fields and SLAM navigation.

- ✓ For known dynamics model the feedback linearization control will be used to cancel the nonlinearities of the nonholonomic vehicles.
- ✓ Group formation for cooperative manipulation tasks, that the whole system will be integrated into:
    - One of the robots will be able to lead the others in an unknown environment.
    - All the robots in the group will keep their formation with each other and with their group leader.
    - The whole system will be able to handle a cooperative tasks.
- ✓ The cost function $\tilde{V}$equation (3.50) will be used for analyzing the formation stability.
- ✓ For unknown dynamics system a neural network control will be used model the system dynamics and tracking the inner loop errors.
- ✓ Simulation results.
- ✓ A comprison study between the implemented controllers.

Chapter 5: Thesis contributions, future work and conclusions.

CHAPTER 2

LITERATURE REVIEW

2.1 Simultaneous Localization and Mapping (SLAM)

A robot generates its trajectory based on Simultaneous Localization and Mapping (SLAM). The mapping problem is addressed by using the current observations and the last available knowledge about the navigation environment. From the literature, the simultaneous localization and mapping is used when a robot is placed at an unknown location in an unknown environment and requested to incrementally build a map of this unknown environment and use it to navigate autonomously.

During the last two decades, many researchers considered mapping and localization problems. Smith and Cheeseman[5] addressed the mapping problem based on Extended Kalman filter (EKF) and established the concept of stochastic map. They considered the state vector as a non-separable entity with spatial correlation between map features and mobile robot. Whyte [6] introduced a statistical basis for describing relationships between landmarks and manipulating geometric uncertainty. Ayache [7] addressed the problem of visual navigation. Crowley [8] considered sonar based navigation of mobile robots using Kalman filter-type algorithms. The initial analysis of the SLAM problem,its solution formulation and many of its results were developed by Csorba [15]. Also, many groups of researchers worked on localization and map building, for example, Leonard *et al*. [16], Castellanos *et al*. [17],and [18]. Whyte et al.[9,10] proposed a study of SLAM in two parts. Part I [9] provided a historical description of SLAM's early developments especially of the probabilistic form of the SLAM problem, essential solution methods and significant implementations. Part II [10] was concerned with recent advances in computational methods and new formulations of the SLAM problem with a focus on three key areas: computational complexity, data association, and environment representation. These two summary papers, [9] and [10], are a comprehensive introduction to the SLAM problem.

In the above cited work, it was assumed that the unknown environment is static. In the case of a dynamic environment, Wang and Thorpe [11] developed a consistency-based moving object detector and provided a framework to solve the SLAM and the detection and tracking of moving objects. They integrated SLAM with moving objects tracking. John *et al*. [12] applied an expectation maximization technique to the SALM algorithm for detecting moving land marks. They optimized the final map by removing the moveable landmarks. Recently, Wu and Sun [13] presented an extended Kalman filter algorithm for SLAM and moving object detection. They found the measurements of moving object and environment landmarks based on the combination of occupy grid map and their algorithm. Throughout the last decade, many SLAM algorithms [19] were proposed for detection and tracking of moving objects in dynamic environments based on different techniques: with reversible data association [14]; by using laser scanners [20]; and with vision sensors [21].

The mobile robot system equipped with SLAM solution to navigate autonomously is one of the most reported topics in robotic navigation studies. The SLAM solutions can be applied to robotics to explore environments if it is dangerous or impossible for humans to visit, such as underground [14] and underwater [28].

A typical SLAM problem has two main solutions based on two approaches, namely the Bayesian theory [24], and scan matching [23]. Feng and Milios, developed two iterative algorithms for matching two range scans. The first one solves the displacement between the two scans based on distance function minimization. The second establishes correspondences between scans based on iterative point correspondence matching algorithm [22].

The SLAM problem solution has been given in many papers. Dissanayake *et. al*. [27] propose a solution for the general SLAM problem by providing its convergence properties and illustrating SLAM algorithm implementation for an outdoor environment. Huang and Dissanayake [33] gave an analysis for the convergence properties of extended Kalman filter (EKF) SLAM. SLAM based extended information filter (EIF) is treated in [34][39][40].

The observability of the SLAM problem, which is the SLAM system's ability for computing the current state from previous observations and sequence control actions, is addressed for different SLAM versions in [41][42][43][44]. The problem of SLAM convergence or the convergence of state estimate uncertainty to a finite value is solved in [27] [33] [45] [46] [47]. The sensor bias is an active SLAM problem. Many solutions to this problem are given in [48] [49] [50].

SLAM solutions for tracking the moving landmarks have not been proposed in literature, but recently Esaka *et. al.* [25] proposed SLAM for leader's tracking by using correlation-based map matching method. Also they presented a method for a leader-follower platooning where the follower builds an occupancy grid map[26]. Many papers addressed the problem of localization by using laser scanning [1] by updating the vehicle position information at high frequency. Furthermore,using laser scanning information to detect landmarks was proposed by J. Guivant [2].

2.2 Leader-Follower Formation Control and Potential Fields

One possible application of a group of robots is in the exploration of unknown environments. A group of mobile robots which moves inside an unknown environment can acquire much more information than single robot exploration.

Formation control is a useful strategy to keep a group of unmanned robots working together and moving in a desired formation based on the required application. Leader-follower formation is one of these applications. Leader-follower formation control is addressed in the literature. Desai *et. al.* proposed a leader-follower formation scheme and stabilized their relative distance and orientation by using feedback linearization [51]. Dierks and Jagannathan developed a framework for a nonholonomic leader-follower formation using backstepping based control by considering the robots' dynamics [52]. Xiaohai and Jizhong presented a leader-follower approach of nonholonomic mobile robots based on the direct Lyapunov method of kinematic control [53]. A framework for achieving the desired formation for a group of nonholonomic mobile robots based on automatic switching between control modes is developed by Fierro *et. al.*[54].

Potential fields are a useful strategy for formation control of a fleet of robots. In the recent decade many control strategies for controlling the formation shape of a fleet of robots were proposed based on the potential fields approach. Song and Kumar [29] analyzed the force equilibrium to show how potential fields can be used for different shape generation and formation control. Also, they adopted a control framework for controlling a fleet of robots for handling cooperative tasks. Chaimowicz *et al*.[30], addressed an approach for shape formation and control to manipulate a group of robots to arbitrary shapes. Hsieh and Kumar extended Chaimowicz's approach and developed decentralized controllers for group desired shape formation [31]. Sabattini *et al*. [32] presented a novel decentralized control approach for holonomic mobile robots formation. Their strategy is not only to handle robots shape formation, but also to determine the robot's positions. Kowdiki *et. al.* implemented a potential fields formation control strategy but they considered a point mass robot [55].

However, our approach will extend the Lorenzo Sabattini *et al.* approach [32], to make one of the robots lead the others in an unknown environment, and at the same time all the agents in the fleet will keep their formation shape based on the potential fields. In addition, our new framework will extend the work in Lorenzo *et al.* [32] from point mass holonomic agents to nonholonomic vehicles. The whole system will be able to satisfy the Song and Kumar framework [29] to give star, square,pentagon, circule for any number of nonholonomic robots in handling and manipulating cooperative tasks.

CHAPTER 3

PRELIMINARIES

3.1 Introduction

In this chapter we will present theoretical background for the models used. The model of nonholonmic mobile robot that is used here is a Car-Like robot. In addition, simultaneous localization and mapping (SLAM) modeling and potential fields approach for mobile robots group formation will be presented.

3.2 Nonholonmic Mobile Robot Modeling

The robot used in this study is a car-like robot [35]. Generally the mobile robot with 2-dimensonal coordination space system and subjected to m constraints can be described in general coordinates (q) by

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) + \tau_d = B(q)\tau - A^T(q)\lambda, \tag{3.1}$$

Where

$M(q) \in \mathcal{R}^{n\times n}$: Symmetric PD inertia matrix.

$V_m(q, \dot{q}) \in \mathcal{R}^{n \times n}$: Centripetal and corioles matrix.

$F(\dot{q}) \in \mathcal{R}^{n \times 1}$: Surface friction.

$G(q) \in \mathcal{R}^{n \times 1}$: Gravitational vector.

$\tau_d \in \mathcal{R}^{n \times 1}$: Unknown disturbance.

$B(q) \in \mathcal{R}^{n \times r}$: Input transformation matrix.

$\tau \in \mathcal{R}^{n \times 1}$: Input vector.

$A^T(q) \in \mathcal{R}^{m \times n}$: Is matrix associated with constraints.

$\lambda \in \mathcal{R}^{m \times 1}$: Constraint forces vector.

If we consider all kinematic equality constraints are not time-dependent, which follows that [35]

$$A(q)\dot{q} = 0 \tag{3.2}$$

Let $S(q)$ be a full rank matrix of a set of smooth and linearly independent vector fields in null space of $A(q)$,i.e.,

$$S^T(q)A^T(q) = 0 \tag{3.3}$$

If we consider (3.2) and (3.3), it will be possible to find a vector time function $\mathrm{v}(t) \in \mathcal{R}^{n-m}$ such that

$$\dot{q} = S(q)\mathrm{v}(t) \tag{3.4}$$

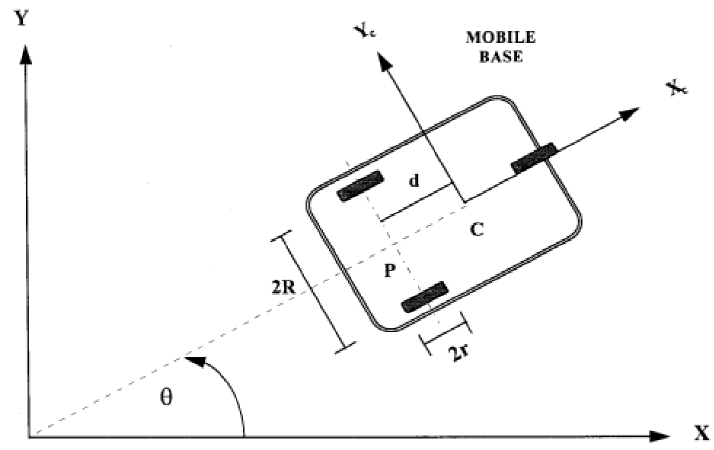


Figure 3.1. A nonholonomic car-like mobile robot.[35]

The position of the robot shown in figure 3.1 is in an inertial Cartesian frame $\{O, X, Y\}$ and it is presented by the vector $q = [x, y, \theta]^T$. The kinematic equations of the motion (3.4) can be presented in terms of linear and angular velocities by

$$S(q) = \begin{bmatrix} \cos(\theta) & -dsin(\theta) \\ sin(\theta) & d\cos(\theta) \\ 0 & 1 \end{bmatrix} \tag{3.5}$$

$$\mathrm{v} = \begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{3.6}$$

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -dsin(\theta) \\ sin(\theta) & d\cos(\theta) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{3.7}$$

The matrices that define the model dynamics based on figure (3.1) are expressed in (3.1) where,

$$M(q) = \begin{bmatrix} m & 0 & md\sin(\theta) \\ 0 & m & -md\cos(\theta) \\ md\sin(\theta) & -md\cos(\theta) & I \end{bmatrix} \tag{3.8}$$

$$V(q,\dot{q}) = \begin{bmatrix} -md\dot{\theta}^2\cos(\theta) \\ -md\dot{\theta}^2\sin(\theta) \\ 0 \end{bmatrix} \quad (3.9)$$

$$G(q) = 0 \quad (3.10)$$

$$B(q) = \frac{1}{r}\begin{bmatrix} \cos(\theta) & \cos(\theta) \\ \sin(\theta) & \sin(\theta) \\ R & -R \end{bmatrix} \quad (3.11)$$

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (3.12)$$

Now the system (3.1) will be transformed to be appropriate for control consideration. To arrive at that, we need to differentiate (3.4) and substitute it into (3.1). We then multiply the resulting equation by S^T. The final motion equations of the nonholonomic mobile robot will be given as[35]

$$\dot{q} = S\mathrm{v} \quad (3.13)$$

$$S^T MS\dot{\mathrm{v}} + S^T(M\dot{S} + V_m S)\mathrm{v} + \bar{F} + \bar{\tau}_d = S^T B\tau \quad (3.14)$$

Where $\mathrm{v}(\mathrm{t}) \in \mathcal{R}^{n-m}$ is a vector of the velocity. Equation (3.14) will be rewritten as

$$\bar{M}(q)\dot{\mathrm{v}} + \bar{V}_m(q,\dot{q})\mathrm{v} + \bar{\mathrm{F}}(\mathrm{v}) + \bar{\tau}_d = \bar{B}\tau \quad (3.15)$$

$$\bar{\tau} = \bar{B}\tau \quad (3.16)$$

where,

$\bar{M}(q) \in \mathcal{R}^{r\times r}$: Symmetric, PD inertia matrix.

$\bar{V}_m(q,\dot{q}) \in \mathcal{R}^{r\times r}$: The centripetal and coriolis matrix.

$\bar{\mathrm{F}}(\mathrm{v}) \in \mathcal{R}^{r\times 1}$: The surface friction.

$\bar{\tau}_d$: Unknown disturbance.

$\bar{\tau} \in \mathcal{R}^{r\times 1}$: Input vector.

Equation (3.15) describes the system behavior in the vehicle coordinates. This means $S(q)$ is the transformation matrix which transforms the velocities of the vehicle coordinates ‘v’ to Cartesian coordinates velocities $\dot{\mathrm{q}}$.

The parameter	Value	Unit
m	10	Kg
I	5	Kg-m^2
R	0.5	m
r	0.05	m

Table 3.1: The vehicle parameters.

3.3 Simultaneous Localization and Mapping (SLAM)

This section will introduce the mathematical modeling of SLAM problems. This modeling is based on [4], [3] and [9]. In simple words, the Simultaneous Localization and

Mapping is a coupled problem of localization and mapping. This has a solution only if the two problems are processed together.

3.3.1 Process Model

The SLAM problem can be formulated by assuming a vehicle (mobile robot) which has a known kinematic model moving through an environment which is populated with a number of (features) landmarks. The vehicle has a sensor to measure the relative location between the vehicle and any one of the individual landmarks as shown in figure 3.2, [3].

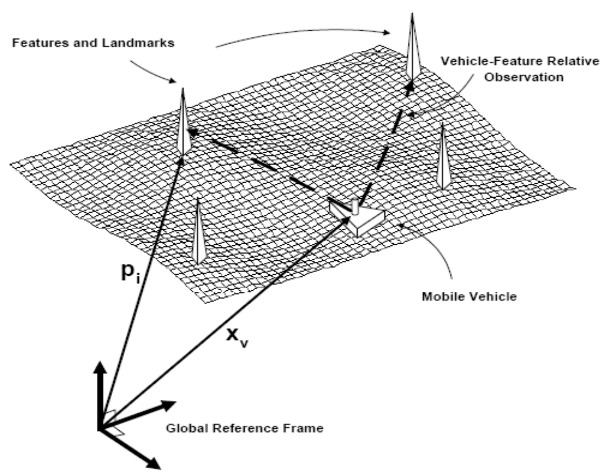


Figure (3.2). The relative location measurements between the vehicle and landmarks.[3]

The state vector of the system contains the states of the vehicle model and all landmark positions (states). If we assume a linear discrete time model of the vehicles and its state is given as $x_v(k)$, then the vehicle process model is given as:

$$x_v(k+1) = F_v(k) + u_v(k+1) + w_v(k+1) \tag{3.17}$$

where

$F_v(k)$ is the state transition matrix.

$u_v(k)$ is control inputs vector.

$w_v(k)$ is a vector of uncorrelated process noise errors with zero mean and covariance $Q_v(k)$.

Based on the assumption that the landmarks are static, its state transition equation will be

$$P_i(k+1) = P_i(k) = P_i \tag{3.18}$$

If we have N landmarks the state vector of all the landmarks is given as

$$P = [P_1{}^T \dots \dots P_N{}^T]^T, \tag{3.19}$$

and the overall state vector for both the vehicle and landmarks is

$$x(k) = [\, x_v{}^T(k) \quad P_1{}^T \dots \dots P_N{}^T]^T \tag{3.20}$$

The complete state transition model can be written as

$$\begin{bmatrix} x_v(k+1) \\ P_1 \\ \vdots \\ P_N \end{bmatrix} = \begin{bmatrix} F_v(k) & 0 & \dots & 0 \\ 0 & I_{P_1} & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & I_{P_N} \end{bmatrix} \begin{bmatrix} x_v(k) \\ P_1 \\ \vdots \\ P_N \end{bmatrix} + \begin{bmatrix} u_v(k+1) \\ 0_{P_1} \\ \vdots \\ 0_{P_N} \end{bmatrix} + \begin{bmatrix} w_v(k+1) \\ 0_{P_1} \\ \vdots \\ 0_{P_N} \end{bmatrix} \tag{3.21}$$

I_{P_i} is the $\dim(P_i) \,\mathrm{x}\, \dim(P_i)$ identity matrix and $0_{\mathrm{P_i}}$ is the $\dim(P_i)$ null vector.

3.3.2 Observation Model

The vehicle has a sensor to measure the relative location between the vehicle and any one of the individual landmarks. Moreover based on the assumption that the observations are linear, the observation model of any i^{th} landmark will be written as

$$z_i(k) = H_i x(k) + v_i(k) \tag{3.22}$$

$$z_i(k) = H_{pi}p - H_v x_v(k) + v_i(k) \tag{3.23}$$

Where $v_i(k)$ is a vector of uncorrelated observation errors with zero mean and variance $R_i(k)$. H_i is the observation matrix that relates the sensor output $z_i(k)$ to the state vector $x(k)$ when observing the i^{th} landmark and is written in the form

$$H_i = [-H_v, 0 \dots 0, H_{pi}, 0 \dots 0] \tag{3.24}$$

Which says that we have relative observations between the landmark and the vehicle itself.

3.3.3 Estimation Process

In the estimation process we need to estimate the state $x(k)$ based on our observations z. This will be easily achieved using the Kalman filter which will handle the function of estimation of landmarks and vehicle locations. The Kalman filter recursively calculates the estimation of the state $x(k)$ in accordance with the process and observations. The Kalman Filter proceeds through three stages:

I. Prediction

At time k if the initial values of the estimate $\hat{x}(k|k)$ of the state $x(k)$ and the covariance estimate of $P(k|k)$ is initialized, the prediction can be calculated at time $k+1$ for state estimate as follow

$$\hat{x}(k+1|k) = F(k)\hat{x}(k|k) + u(k) \tag{3.25}$$

and the observation (relative to the i^{th} landmark) as

$$\hat{z}_i(k+1|k) = H_i(k)\,\hat{x}(k+1|k) \tag{3.26}$$

and also the state covariance

$$P(k+1|k) = F(k)P(k|k)F^T(k) + Q(k) \tag{3.27}$$

II. Observation

After the prediction, the ith landmark observation is calculated based on the observation model (3.22). The innovation (the discrepancy between the actual measurement z_k and the predicted measurement $\hat{z}_k$) is given as

$$v_i(k+1) = z_i(k+1) - \hat{z}_i(k+1|k) \tag{3.28}$$

And its covariance matrix is

$$S_i(k+1) = H_i(k)\, P(k+1|k){H_i}^T(k) + R_i(k+1) \tag{3.29}$$

III. Update

The update of state estimate and its corresponding covariance are calculated as follows:

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + W_i(k+1)w_i(k+1) \tag{3.30}$$

$$P(k+1|k+1) = P(k+1|k) - W_i(k+1)S(k+1){W_i}^T(k+1) \tag{3.31}$$

where

$W_i(k+1)$ is the gain matrix which is given by

$$W_i(k+1)w_i = P(k+1|k){H_i}^T(k){S^{-1}}_i(k+1) \tag{3.32}$$

3.3.4 Kalman Filter

The state of our process, $x(k)$ needs to be estimated based on our measurement $z(k)$, This is the exact definition of the Kalman filter. Kalman filter recursively computes the estimates of state x(k) which is evolving according to the process and observation models. The filter proceeds in three stages prediction, observation and update as discussed above [3].

Define a state $x(k)$ at time k evolves from state $(k-1)$ and its measurement $z(k)$ given by

$$x(k) = Ax(k-1) + Bu(k-1) + w(k) \quad \text{(Process Model)} \tag{3.33}$$

$$z(k) = Hx(k) + v(k) \qquad \text{(Observation Model)} \tag{3.34}$$

where

A: Is the state transition model.

B : Is the control input matrix.

H : Is the observation matrix, which relates the measurement $z(k)$ to the state vector $x(k)$.

w: Is the process noise, which assumed to have a normal probability distribution with zero mean and covariance Q:

$$\mathrm{p(w)} \sim \mathrm{N(0, Q)} \tag{3.35}$$

v is the observation noise, which assumed to have a normal probability distribution with zero mean and covariance R:

$$p(v) \sim N(0, R) \tag{3.36}$$

Table (3.2) presented the Kalman filter algorithm.

If there is a nonlinearity in the system, the linear Kalman filter cannot deal with it. In situations like this, the extended Kalman filter can be used for estimation, because it uses the linearized process model and the observation equations to predict the states.

Kalman Filter Algorithm		
Prediction	*Predicted state*	$\hat{x}(k\|k-1) = A(k)\hat{x}(k-1\|k-1) + B(k)u(k-1)$
	Predicted covariance	$P(k\|k-1) = A(k)P(k-1\|k-1)A(k)^T + Q(k)$
Observation	*Innovation*	$\tilde{y}(k) = z(k) - H(k)\hat{x}(k\|k-1)$
	Innovation covariance	$S(k) = H(k)P(k\|k-1)H(k)^T + R(k)$
Update	*Kalman gain*	$K(k) = P(k\|k-1)H(k)^T S(k)^{-1}$
	Updated state	$\hat{x}(\, k \mid k) = \hat{x}(\, k \mid k-1) + K(k)\tilde{y}(k)$
	Updated covariance	$P(k\|k) = (I - K(k)H(k))P(k\|k-1)$

Table (3.2). Kalman filter algorithm

3.3.5 Extended Kalman Filter

If the nonlinearity is associated in the system the Extended Kalman Filter (EKF) is used for estimation. In this case the process is given as [3]

$$x(k) = f(x(k-1), u(k-1), w(k-1)) \tag{3.37}$$

and the opservation model will be

$$z(k) = h(x(k), v(k)) \tag{3.38}$$

where f , h are non-linear functions.

the process and opservation vectors can be approximated without noise values $w(k)$ and $v(k)$ because in the practice the noise cannot be measured at each time step. The approximated process model

$$\tilde{x}(k) = f(\hat{x}(k-1), u(k), 0) \tag{3.39}$$

and approximated observation model

$$\tilde{z}(k) = h(\tilde{x}(k), 0) \tag{3.40}$$

$\hat{x}(k)$: is some a posteriori estimate of the process state

For nonlinear process estimation, the system is needed to be linearized at the current state

$$x(k) = \tilde{x}(k) + A(x(k-1) - \hat{x}(k-1) + Ww(k-1)) \tag{3.41}$$

$$z(k) = \tilde{z}(k) + J_h\big(x(k) - \tilde{x}(k)\big) + Vv(k) \tag{3.42}$$

where

$x(k)$: The actual state vector.

$z(k)$: The measurement vector.

$\tilde{x}(k)$: The approximate state vector.

$\tilde{z}(k)$: The approximate measurement vector.

$\hat{x}(k)$: Posteriori estimate of the state at step k.

$w(k)$: Process noise.

$v(k)$: Measurement noise

A: : Jacobian matrix of partial derivatives of f with respect to x.

W: Jacobian matrix of partial derivatives of f with respect to w.

J_h: Jacobian matrix of partial derivatives of h with respect to x.

V: Jacobian matrix of partial derivatives of h with respect to v.

The extended Kalman filter algorithm is presented in table (3.3) bellow.

Extended Kalman Filter Algorithm		
Prediction	*Predicted state* *Predicted covariance*	$\hat{x}(k)^{-} = f(\hat{x}(k-1), u(k-1), 0)$ $P(k)^{-} = A(k)P(k-1)A(k)^{T} + W(k)Q(k-1)W(k)^{T}$
Observation	*Innovation* *Innovation covariance*	$\tilde{y}(k) = z(k) - H(k)$ $S(k) = J_h(k)P(k)^{-}J_h(k)^{T} + V(k)R(k)V(k)^{T}$
Update	*Kalman gain* *Updated state* *Updated covariance*	$K(k) = P(k)^{-}J_h(k)^{T}S(k)^{-1}$ $\hat{x}(k) = \hat{x}(k)^{-} + K(k)\tilde{y}(k)$ $P(k) = (I - K(k)J_h(k))P(k)^{-}$

Table (3.3). Extended Kalman filter algorithm

3.3.6 SLAM Based Laser Pose Tracking for Simple Robot Model

The problem of localization using laser scanning is addressed in [1] by updating the vehicle position information at high frequency. Using laser scanning information to detect landmarks was proposed by Guivant [2].

A simple robot model is used for the modeling purpose. The robot position is defined in global coordinates and the direction control is represented in the robot coordinates. The robot is equipped with a laser sensor for range and bearing calculations between the autonomous robot and the landmarks. The landmarks are defined as B_i , $i = 1 \ldots n$, and the distance and bearing measurements are calculated based on the robot coordinates (x_l, y_l) where

$$z(k) = (r, \beta) \tag{3.43}$$

r: is the distance between LM and laser,

β: is the sensor bearing w. r. t the robot coordinate frame.

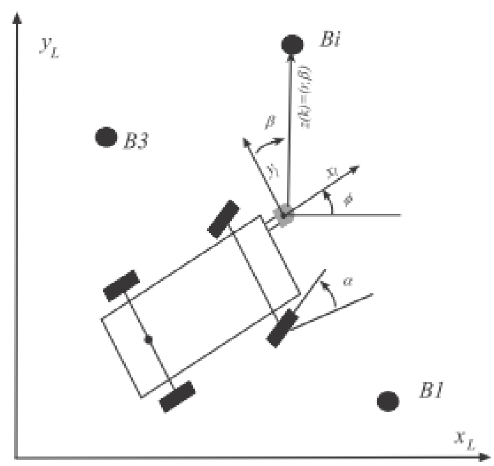


Figure (3.3) vehicle coordinate system

If the vehicle is controlled by its velocity V and the steering angle α with respect to the trajectory prediction of the vehicle back axle center, the process model will be

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} V\cos(\phi) \\ V\sin(\phi) \\ \frac{V}{L}\tan(\propto) \end{bmatrix} \tag{3.44}$$

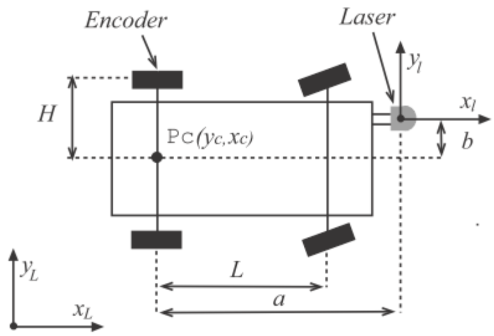


Figure (3.4). Car-like robot model parameters.

For updating, the center of the back axle should be translated to represent the kinematic vehicle based on the trajectory of the laser center, from figures (3.3) and (3.4), [2] the full state representation is written as follow

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} V\, cos(\phi) - \frac{V}{L}(a \cdot sin(\phi) + b \cdot cos(\phi)) \cdot tan(\alpha) \\ V\, sin(\phi) + \frac{V}{L}(a \cdot cos(\phi) - b \cdot sin(\phi)) \cdot tan(\alpha) \\ \frac{V}{L} tan(\propto) \end{bmatrix} \tag{3.45}$$

The final discrete model in global coordinates is represented in the form

$$\begin{bmatrix} x(k) \\ y(k) \\ \phi(k) \end{bmatrix} =$$

$$\begin{bmatrix} x(k-1) + T \cdot V(k-1)\, cos(\phi(k-1)) - \frac{V(k-1)}{L}(a \cdot sin\big(\phi(k-1)\big) + b \cdot cos(\phi(k-1))) \cdot tan(\alpha(k-1)) \\ y(k-1) + T \cdot V(k-1)\, sin\big(\phi(k-1)\big) + \frac{V(k-1)}{L}\big(a \cdot cos\big(\phi(k-1)\big) - b \cdot sin\big(\phi(k-1)\big)\big) \cdot tan\big(\alpha(k-1)\big) \\ \frac{V(k-1)}{L} tan\big(\propto (k-1)\big) \end{bmatrix} \tag{3.46}$$

where

T: Sampling time

The observation model is given by

$$\begin{bmatrix} z_r^i \\ z_\beta^i \end{bmatrix} = \begin{bmatrix} \sqrt{(x_L - x_i)^2 + (y_L - y_i)^2} \\ atan\left(\frac{(y_L - y_i)}{(x_L - x_i)}\right) - \phi + \frac{\pi}{2} \end{bmatrix} \tag{3.47}$$

where

$(x_i,\ y_i)$ are the i^{th} landmark position.

The estimated position of the landmark becomes part of the process state vector. If the robot begins its navigation from an unknown place and obtains the environment measurements, this procedure will be used to incrementally build a map and localize the vehicle based on this map[2].

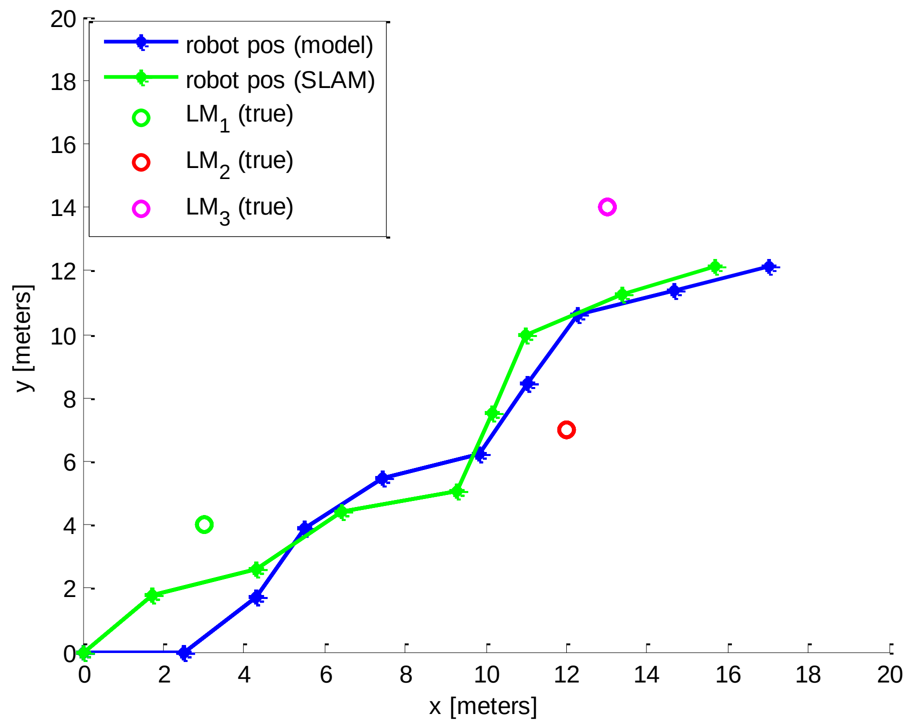


Figure (3.5). Full SLAM implementation in MATLAB

3.4 Mobile robots group formation based on potential fields approach

Potential fields are a useful strategy for formation control of a fleet of robots. One possible application of a group of robots is in the exploration of unknown environments. A group of mobile robots which moves inside an unknown environment can acquire much more information than single robot exploration.

The potential fields approach can be used as a formation control strategy, that can be applied to a group of robots moving in a two dimensional space. The main idea is to make a group of robots move in a formation of a desired shape. The formation control strategy considers a group of 'n' point mass holonomic agents described by the following dynamics:[32]

$$\ddot{x}_i = u_i \qquad i = 1, \dots \dots . . , n \tag{3.48}$$

$x_i \in \mathcal{R}^2$, is the i^{th} agent's position. If the agents are needed to localize themselves on the circumcircle of a regular polygon, the distance between each two neighboring agents is equal to L, and the radius of the circumcircle of the polygon is $\boldsymbol{r}$.

The desired formation can be obtained by implementing the following control law:

$$u_i = f_{ci} + \sum_{j=1, j\neq i}^{n} f_{aij} - b\dot{x}_i \tag{3.49}$$

$$f_{ci} = -\nabla_{x_i} V_{ci}(x_i) \tag{3.50}$$

f_{ci}: is the force between the center (leader) and and follower i. This term is to keep each agent at distance $\boldsymbol{r}$ from the center of the formation.

$$f_{aij} = -\nabla_{x_i} V_{aij}(x_i, x_j) \tag{3.51}$$

This term is used to regulate the distances between the agents.

Thus, the summation of attractive and repulsive potentials can control the formation behavior. When all agents are at a distance r from the group center, the control action at (3.50) is null. When the distances between each two neighboring agents is equal to L, the control action in (3.51) is null also.

In the desired group formation, the composition of these potential fields is null control action[32]. In other words, the regular polygon formation means that the system is asymptotic stable.

To get the desired formation, we consider our stable configuration shape is a regular polygon. Choose the all system energy as a Lyapunov candidate function '$\widetilde{\mathrm{V}}$'[32] :

$$\tilde{V} = \sum_{i=1}^{n} \left[E_{ci}(P_i) + \sum_{j=1; j\neq i}^{n} E_{aij}(P_i, P_j) + \frac{1}{2} \mathrm{v}_i^{\mathrm{T}} \overline{\mathrm{M}} \mathrm{v}_i \right] \tag{3.52}$$

E_{ci} is the energy of the center potential and E_{aij} is the energy of interagent potential.

$$\tilde{V} = \sum_{i=1}^{n} [Pot_i + Kin_i] \tag{3.53}$$

$$Pot_i = E_{ci}(P_i) + \sum_{j=1; j\neq i}^{n} E_{aij}(P_i, P_j) \tag{3.54}$$

$$Kin_i = \sum_{\mathrm{i}=1}^{\mathrm{n}} \frac{1}{2} \mathrm{v_i}^{\mathrm{T}} \overline{\mathrm{M}}\, \mathrm{v_i} \tag{3.55}$$

where $P_i = \begin{bmatrix} x_i \\ y_i \\ \theta_i \end{bmatrix}$; $P_j = \begin{bmatrix} x_j \\ y_j \\ \theta_j \end{bmatrix}$. For the desired shape formation the system will be stable if and only if the $\tilde{V}(P_i) \geq 0$.

The time derivative of a Lyapunov candidate function is '$\dot{\tilde{V}}$':

$$\dot{\tilde{V}}(P_i) = \sum_{i=1}^{n} \{ \ [\nabla_{p_i} Pot_i] \, \dot{P}_i + [Kin_i] \dot{\mathrm{v}} \} \tag{3.56}$$

$$\dot{\tilde{V}} = \sum_{i=1}^{n} \left[\nabla_{P_i} E_{ci}(P_i) + \sum_{j=1; j \neq i}^{n} \nabla_{P_i} E_{aij}(P_i, P_j) \right] \dot{P}_i + \sum_{=1}^{\mathrm{n}} \frac{\partial}{\partial \mathrm{v_i}} \left[\frac{1}{2} \mathrm{v_i}^{\mathrm{T}} \bar{M} \mathrm{v_i} \right] \dot{\mathrm{v}}_{\mathrm{i}} \tag{3.57}$$

The cost function (3.52) will be used in chapter 4 for analyzing the formation stability.

3.5 Neural Networks

A neural network function is used as a powerful estimater to estimate the unmodeled dynamics in a nonholonomic mobile robot [56]. Here the on-line neural network tuning algorithms are used to guarantee the vehicle's desired path tracking.

3.5.1 Feedforward Neural Networks

The feedforward neural network is given in figure (3.6). It has two layers. Its output is the vector y, and x is its input vector.

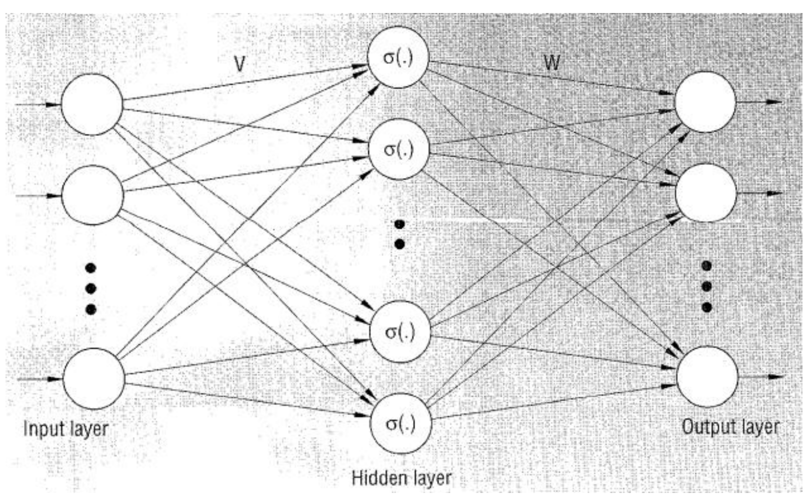


Figure (3.6). Multilayer feedforward neural network[57]

The two layer neural network input output relation is given by the following formula[57]:

$$y_i = \sum_{j=1}^{N_h}[W_{ij}\sigma\left(\sum_{k=1}^{n} V_{jk}x_k + \theta_{Vj}\right) + \theta_{Wi}] \tag{3.58}$$

where

$y \in \mathcal{R}^m, x \in \mathcal{R}^n, i = 1, \dots \dots, m.$

$\sigma(\cdot)$: The activation function.

N_h: Hidden-layer neurons.

V_{jk}: Weights of first interconnections.

W_{ij}: Weights of second interconnections.

θ_{Vj}, θ_{Wi}: Threshold offsets or 'bias'.

Some common choices of activation function are given in figure(3.7).

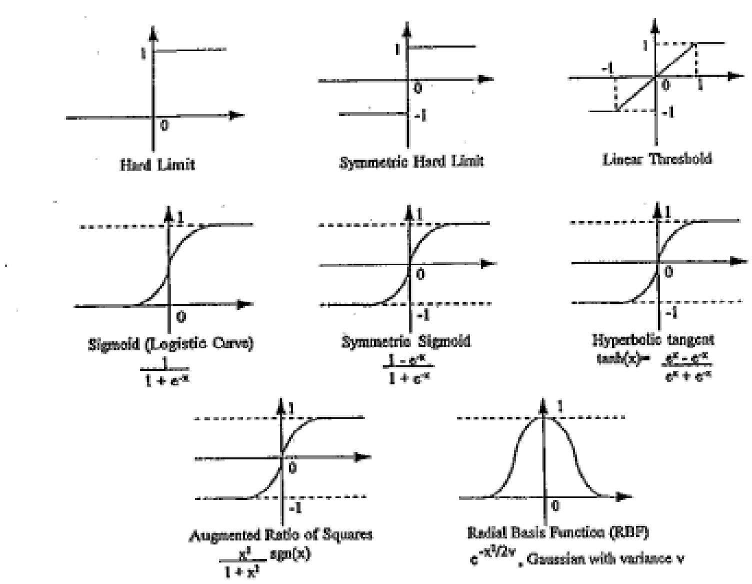


Figure (3.7): Some common neural network activation functions[56].

The expresion of neural network output formula can be written in the vector case as:

$$y = W^T\sigma(V^T x) \tag{3. 59}$$

The thresholds are included in the weight matrix as its first column.

3.5.2 Function approximation property

One of the main neural network properties is the function approximation property, which is important for control purposes. If $f(x)$ is assumed to be a smooth function from $\mathcal{R}^n$ to $\mathcal{R}^m$,we can see that, when x is belongs to compact set $\mathcal{R}^s \in \mathcal{R}^n$ for a number of hidden layer neurons N_h, the weights and thresholds will be given as

$$f(x) = W^T\sigma(V^T x) + \varepsilon \tag{3.60}$$

W and V are the ideal weights that will give a perfect $f(x)$ approximation.

ε : Is a bounded value of *neural network approximation error*.

From equation (3.60) we can conclude that the neural network can give an approximation for any smooth function in a compact set. Then the estimation of $f(x)$ is given by

$$\hat{f}(x) = \widehat{W}^T\sigma\left(\hat{V}^T x\right) \tag{3.61}$$

$\widehat{W}, \hat{V}$: The ideal neural network weight’s estimation. These weights can be estimated by applying the *on-line tuning algorithms*[57].

3.5.3 Weight Tunning Algorithms

The weight tuning algorithms used in literature were the gradient algorithms which built based on the backpropagation alghorithms. This weight tuning algorithms provided by

$$\dot{\widehat{W}} = F\sigma\left(\widehat{V}^T x\right)r^T \tag{3.62}$$

$$\dot{\hat{V}} = Gx(\hat{\sigma}'^{T}\hat{W}r)^{T} \tag{3.63}$$

where F and G are PD design matrices. In the case of sigmoid activation the hidden layer gradient is

$$\hat{\sigma}' = diag\{\sigma\hat{V}^{T}x\}\left[I - diag\{\hat{V}^{T}x\}\right] \tag{3.64}$$

CHAPTER 4

LEADER-FOLLOWER FORMATION CONTROL BASED POTENTIAL FIELD

4.1 Introduction

The aim of this work is to arrange a group of robots in a desired formation around their leader. In the ideal formation case the leader is assumed to be located in the center of the group. The formation proposed strategy is an extension of the potential field control strategy given in [32] with the difference that the model used in this study is a nonholonomic mobile robot instead of a holonomic point mass robot.

Let us assume that we have a fleet of n nonholonomic mobile robots with dynamics similar to those given in equations (3.1). The group leader coordinates are generated by running the SLAM algorithm.

4.2 Leader Navigation Unit

The leader's map is generated by running the SLAM algorithm. The group's leader is equipped with simultaneous localization and mapping system (SLAM navigation unit) the implementation of SLAM is discussed in details in chapter 3. For full SLAM Matlab implementation for real sensor data [2] see figure (4.4).

4.2 The Nonholonomic Robot Model

The whole system consists of two cascaded stages. The first one is the robot system which includes the system dynamics. The second stage is the potential fields formation controller. These two stages are combined together in cascaded form. The general structure of the integrated system is presented below in figure (4.1).

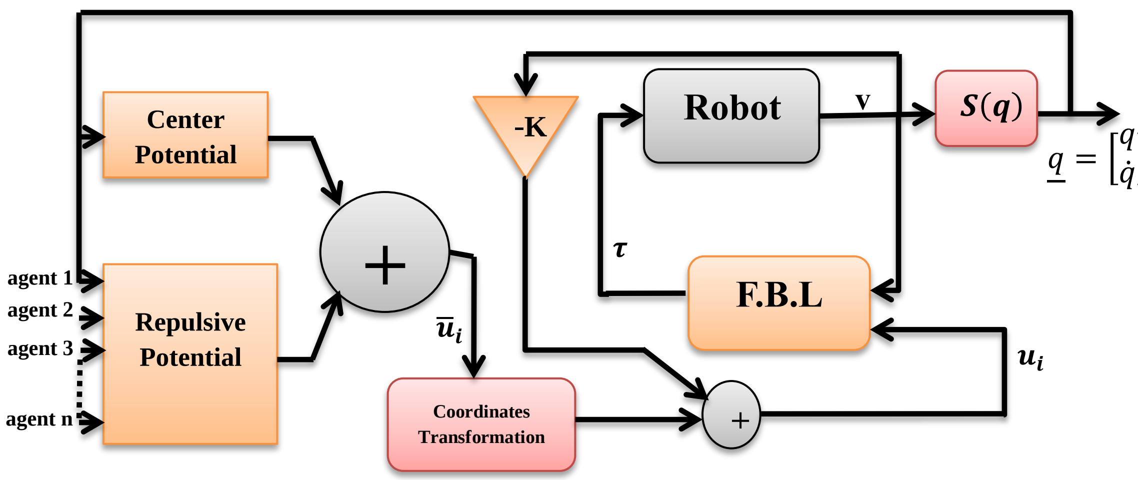


Figure (4.1) The general structure of the proposed system.

4.3 Feedback linearization

A method for controlling the navigation of nonholonomic mobile robots could be based on feedback linearization to cancel the nonlinearities of the nonholonomic vehicles. The linearized system will be valid for both navigation and group formation.

From chapter 3 the nonholonomic mobile robot dynamics is:

$$\bar{M}(q)\dot{\mathrm{v}} + \bar{V}_m(q,\dot{q})\mathrm{v} + \bar{\mathrm{F}}(\mathrm{v}) + \bar{\tau}_d = \bar{B}\tau \tag{4.1}$$

$$\bar{\tau} = \bar{B}\tau \tag{4.2}$$

where

$\bar{M}(q) \in \mathcal{R}^{r\times r}$: Symmetric, PD inertia matrix.

$\bar{V}_m(q,\dot{q}) \in \mathcal{R}^{r\times r}$: The centripetal and coriolis matrix.

$\bar{\mathrm{F}}(\mathrm{v}) \in \mathcal{R}^{r\times 1}$: The surface friction.

$\bar{\tau}_d$: Unknown disturbance.

$\bar{\tau} \in \mathcal{R}^{r\times 1}$: Input vector.

The i^{th} robot system should satisfy the following assumptions [35]

1. Boundedness: $\bar{M}_i(q)$, the norm of $\bar{V}_{m_i}(q,\dot{q})$, and $\bar{\tau}_d$ are bounded.

2. Skew Symmetric: The matrix $\dot{\bar{M}}_i(q) - 2\bar{V}_{m_i}(q,\dot{q})$ is skew symmetric such that

$$x^T\left(\dot{\bar{M}}_i(q) - 2\bar{V}_{m_i}(q,\dot{q})\right)x = 0$$

In the case that, the surface friction $\bar{\mathrm{F}}(\mathrm{v}) = 0$, and unknown disturbance $\bar{\tau}_d = 0$.

$$\dot{\mathrm{v}} = \bar{M}^{-1}(-\bar{V}_m(q,\dot{q})\mathrm{v} + \bar{B}\tau) \tag{4.3}$$

$$\dot{\mathrm{v}} = -\bar{M}^{-1}\bar{V}_m(q,\dot{q})\mathrm{v} + \bar{M}^{-1}\bar{B}\tau \tag{4.4}$$

Then for feedback linearization, choose the input of the robot τ to be

$$\tau = (\bar{M}^{-1}\bar{B})^{-1}(u_i + \bar{M}^{-1}\bar{V}_m(q,\dot{q})\mathrm{v}) \tag{4.5}$$

Where u_i is the vehicle frame proposed control signal which controls the whole system behavior figure (4.1), then by substituting (4.5) in (4.4) the robot system will be linearized to

$$\dot{\mathrm{v}} = u_i \tag{4.6}$$

This means that the linear and the angular velocity of the robot can be controlled directly by the proposed control signal u_i, but all $\dot{\mathrm{v}}$ dynamics in (4.6) is in the robot body frame representation and the proposed control signal $\bar{u}_i$ figure (4.1) is in general inertia frame. Now we need to design a transformation filter to transform the control signal $\bar{u}_i$ to become more appropriate for controlling the robot body frame dynamics.

COORDINATES TRANSFORMATION FILTER

We will start from the inertia frame coordinates.

Assume that

$$\Lambda\ddot{q} = \bar{u} \tag{4.7}$$

Because we are interested in x and y only, define

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

For coordinates transformation we have

$$\dot{q} = S\mathrm{v} \tag{4.8}$$

The linearized robot system is $\dot{\mathrm{v}} = u_i$. By differentiating (4.8) and substituting it for $\ddot{q}$ we will get

$$\Lambda\ddot{q} = \Lambda(\dot{S}\mathrm{v} + \dot{\mathrm{v}}S) \tag{4.9}$$

From (4.7) and (4.9) $\bar{u}$ will be given as,

$$\bar{u} = \Lambda(\dot{S}\mathrm{v} + \dot{\mathrm{v}}S) \tag{4.10}$$

$$\Rightarrow u_i = (\Lambda\mathrm{S})^{-1}(\bar{u} - \Lambda\dot{\mathrm{S}}\mathrm{v}) \tag{4.11}$$

4.4 Shape formation

We assume that each robot has a sensing range equal to $\mathrm{S_r}$ figure (4.2), and each robot can know its neighbors' positions which are located inside its sensing area. All nonholonomic robots in the fleet need to navigate in a desired polygon shape. These robots localize themselves as followers around a moving leader. The distance between each two neighboring followers should be equal to $\mathrm{L} \leq \mathrm{S_r}$. The radius of the circumcircle of the desired polygon is equal to $\mathscr{r}$. The coordinates of the moving leader is $x_c \in \mathcal{R}^2$ which can be generated by SLAM approach on the group leader. From the basics of the geometry, $\mathscr{r}$ will be: [32].

$$\mathscr{r} = \frac{L}{2sin(\pi/n)} \tag{4.12}$$

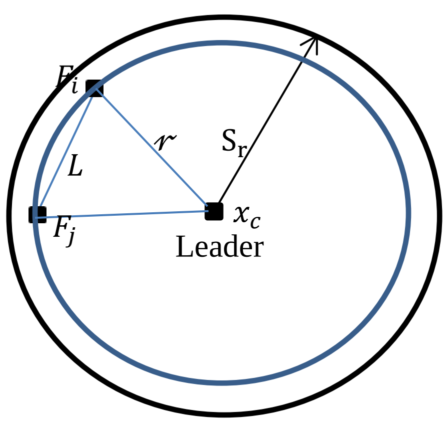


Figure (4.2). The leader sensing range F_i, F_j followers.

4.5 Control Design

The control law will be implemented based on the attractive and repulsive potentials as well as the vehicle damping action. The desired behavior of the system will be obtained by using this control law:

$$u_i = P_{ce} + P_{ij} + Da \tag{4.13}$$

where

P_{ce}: The center potential.

P_{ij}: The interagent potential (agent's potential).

Da: Damping action.

4.5.1 Holonomic model

In the case of the holonomic robot model behaving like a point mass, the complete potential field formation and analysis can be expressed as follows.[32]

Center Potential

The center potential in $\mathcal{R}^2$ is defined as follows:

$$P_{ce} = -\nabla_{x_i} V_{c_i}(x_{f_i}) \tag{4.14}$$

and, $$V_{c_i} = \frac{1}{2} K_c (d_{c_i} - r)^2 \tag{4.15}$$

where $d_{c_i}(t) = \|x_{f_i}(t) - x_c\|$: is the Euclidian distance in the current time between the following i^{th} robot (x_{f_i}) and the moving leader (x_c).

K_c: is a positive constant.

The control action in (4.14) is used to control the distance r between the leader and each i^{th} follower.

$$x_{f_i} = \begin{bmatrix} x_f \\ y_f \end{bmatrix}, x_c = \begin{bmatrix} x_l \\ y_l \end{bmatrix}.$$

$$d_{c_i} = \sqrt{\left(x_f - x_l\right)^2 + (y_f - y_l)^2} \tag{4.16}$$

$$\Rightarrow V_{c_i}(x_{f_i}) = \frac{1}{2} K_c (d_{c_i} - r)^2 \tag{4.17}$$

To calculate the current center potential between the leader and each i^{th} follower we differentiate V_{c_i} with respect to x_{f_i}. This differentiation will be as follows:

$P_{att} = -\nabla_{x_{f_i}} V_{c_i}(x_{f_i}) = -\frac{\partial V_{c_i}(x_{f_i})}{\partial x_{f_i}}$ using chain rule, we obtain

$$-\frac{\partial V_{c_i}(x_{f_i})}{\partial x_{f_i}} = \frac{\partial V_{c_i}}{\partial d_{c_i}} \frac{\partial d_{c_i}}{\partial x_{f_i}} \tag{4.18}$$

From equation (4.17)

$$\frac{\partial V_{c_i}(x_{f_i})}{\partial d_{c_i}} = K_c (d_{c_i} - r) \tag{4.19}$$

define

$$d_{c_i} = D^{1/2} \tag{4.20}$$

$$\Rightarrow D = \left(x_f - x_l\right)^2 + (y_f - y_l)^2 \tag{4.21}$$

differentiate d_{c_i} with respect to x_{f_i} and using the chain rule, we get:

$$\frac{\partial d_{c_i}}{\partial x_{f_i}} = \frac{\partial d_{c_i}}{\partial D} \frac{\partial D}{\partial x_{f_i}} \tag{4.22}$$

$$\frac{\partial d_{c_i}}{\partial D} = \frac{1}{2}(D)^{-1/2} \tag{4.23}$$

and

$$\frac{\partial \mathrm{D}}{\partial x_{f_i}} = \begin{bmatrix} \frac{\partial D}{\partial x_f} & \frac{\partial D}{\partial y_f} \end{bmatrix} = \begin{bmatrix} 2(x_f - x_l) & 2(y_f - y_l) \end{bmatrix}$$

$$= 2\left[x_{f_i} - x_c\right]^T \tag{4.24}$$

Then substitute (4.23) and (4.24) in (4.22) we will get:

$$\frac{\partial d_{c_i}}{\partial x_{f_i}} = \frac{1}{2}(D)^{-\frac{1}{2}} \cdot 2\left[x_{f_i} - x_c\right]^T$$

$$= \frac{1}{d_{c_i}}\left[x_{f_i} - x_c\right]^T \tag{4.25}$$

Finally substitute (4.19) and (4.25) in (4.18). The attractive potential between the leader and each follower will be given as:

$$P_{ce} = \frac{-K_c}{d_{c_i}}(d_{c_i} - r)(x_{f_i} - x_c)^T \tag{4.26}$$

Interagents Potential

The interagents potential between each two followers in $\mathcal{R}^2$ is defined as:[32]

$$P_{rep} = -\nabla_{x_{f_i}} \mathrm{V}_{a_{ij}}(x_{f_i}, x_{f_j}) \tag{4.27}$$

$$\mathrm{V}_{a_{ij}} = \begin{cases} \frac{1}{2} K_a (d_{ij} - L)^2 & d_{ij} < L \\ 0 & otherwise \end{cases} \tag{4.28}$$

where

$d_{ij}(t) = \| x_{f_i}(t) - x_{f_j}(t) \|$ is the distance in the current time between the i^{th} and the j^{th} followers.

$$x_{f_i} = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \qquad x_{f_j} = \begin{bmatrix} x_j \\ y_j \end{bmatrix}; \quad d_{ij} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \tag{4.29}$$

Then by following the same procedure, the repulsive potential between each two neighboring followers is given by:

$$P_{rep} = -\frac{K_a}{d_{ij}} (d_{ij} - L) \left[\left(x_{f_i} - x_{f_j} \right)^T + \left(y_{f_i} - y_{f_j} \right)^T \right] \tag{4.30}$$

K_a: Is a positive constant.

4.5.2 Nonholonomic model

A nonholonomic robot model such as car-like robot is used here . This section will give complete attractive and repulsive potential field analysis for such a system The potential analysis here will include (x, y, θ) instead of (x, y) as expressed in section (4.5.1), where (x, y) are the coordinates of the robot's center of the mass in the inertial Cartesian frame (X, Y), and θ is the orientation of the robot with respect to the inertial frame, see figure (3.1)

Center Potential

In the case of a nonholonomic mobile robot, if the interest is (x, y, θ), the center potential is:

$$P_{att} = -\nabla_{P_i} V_{c_i}(P_i) \tag{4.31}$$

$$V_{c_i} = \frac{1}{2} K_c \left(R_{c_i} - r\right)^2 \tag{4.32}$$

where

$$R_{ci}(t) = \|P_i(t) - P_c\|, \qquad P_i(t) = \begin{bmatrix} x_i(t) \\ y_i(t) \\ \theta_i(t) \end{bmatrix},\ P_c = \begin{bmatrix} x_c \\ y_c \\ \theta_c \end{bmatrix}$$

$$\Rightarrow R_{ci}(t) = \sqrt{(x_i(t) - x_c)^2 + (y_i(t) - y_c)^2 + (\theta_i(t) - \theta_c)^2} \tag{4.33}$$

We have

$$P_{att} = -\nabla_{p_i} V_{ci}(P_i) = -\frac{\partial V_{ci}}{\partial P_i} = -\left(\frac{\partial V_{ci}}{\partial R_{ci}} \cdot \frac{\partial R_{ci}}{\partial P_i}\right) \tag{4.34}$$

Differentiate (4.32) for R_{ci}, we will get

$$\frac{\partial V_{c_i}}{\partial R_{c_i}} = K_c(R_{c_i} - r) \tag{4.35}$$

If $\quad D = (x_i(t) - x_c)^2 + (y_i(t) - y_c)^2 + (\theta_i(t) - \theta_c)^2 \qquad (4.36)$

And $\quad R_{c_i} = D^{1/2} \qquad (4.37)$

Now differentiate R_{c_i} with respect to P_i and use the chain rule, the result will be

$$\frac{\partial R_{ci}}{\partial P_i} = \frac{\partial R_{ci}}{\partial D}\frac{\partial D}{\partial P_i} \tag{4.38}$$

From (4.37) we have

$$\frac{\partial R_{ci}}{\partial D} = \frac{1}{2}D^{-\frac{1}{2}} \tag{4.39}$$

and

$$\frac{\partial D}{\partial P_i} = \begin{bmatrix}\frac{\partial D}{\partial x_i} & \frac{\partial D}{\partial y_i} & \frac{\partial D}{\partial \theta_i}\end{bmatrix}$$

$$= [2(x_i(t) - x_c) \;\; 2(y_i(t) - y_c) \;\; 2(\theta_i(t) - \theta_c)] = 2[(P_i(t) - P_c)]^T \tag{4.40}$$

By substituting (4.39) and (4.40) in (4.38) we will get:

$$\frac{\partial R_{ci}}{\partial P_i} = \frac{1}{R_{ci}}[(P_i(t) - P_c)]^T \tag{4.41}$$

Using (4.35) and (4.41) in (4.34) the attractive potential is

$$P_{ce} = -\frac{1}{R_{ci}}\left[K_c(R_{c_i} - r)\right][(P_i(t) - P_c)]^T \tag{4.42}$$

Interagent Potential

The interagent potential between each two neighboring followers can be expressed as:

$$P_{ij} = -\nabla_{P_i}V_{ij}(P_i, P_j) \tag{4.43}$$

where

$$R_{f_i}(t) = \|P_i(t) - P_j(t)\|, \quad P_i(t) = \begin{bmatrix}x_i(t)\\ y_i(t)\\ \theta_i(t)\end{bmatrix}, \; P_j = \begin{bmatrix}x_j(t)\\ y_j(t)\\ \theta_j(t)\end{bmatrix}$$

$$V_{ij}(P_i, P_j) = \begin{cases} \frac{1}{2} K_a (R_{f_i} - L)^2, & R_{f_i} < L \\ 0\,, & Otherwise \end{cases}$$

(4.44)

where

R_{f_i}: Is the current distance between each two neighboring agents.

K_a: Is a positive constant.

V_{ij}: is continuously differentiable. If $R_{f_i} < L$ this potential produces a repulsive force, and it will produce a null force if $R_{f_i} \geq L$.

Suppose

$$R_{f_i} = \sqrt{\left[P_i(t) - P_j(t)\right]^T \left[P_i(t) - P_j(t)\right]}\ , R_{f_j} = \sqrt{\left[P_j(t) - P_i(t)\right]^T \left[P_j(t) - P_i(t)\right]} \quad (4.45)$$

$$D_{f_i} = \left[P_i(t) - P_j(t)\right]^T \left[P_i(t) - P_j(t)\right] \quad (4.46)$$

$$= \left(x_i(t) - x_j(t)\right)^2 + \left(y_i(t) - y_j(t)\right)^2 + \left(\theta_i(t) - \theta_j(t)\right)^2) \quad (4.47)$$

Whereas the interagent potential between any two neighboring mobile robots is the force obtained by adding together the repulsive forces of all of them, for such an operation we need to diferentiate D_{f_i} with respect to P_i, and P_j.

$$\frac{\partial D_{f_i}}{\partial P_i} = \begin{bmatrix} \frac{\partial D_{f_i}}{\partial x_i} & \frac{\partial D_{f_i}}{\partial y_i} & \frac{\partial D_{f_i}}{\partial \theta_i} \end{bmatrix} \quad (4.48)$$

$$\frac{\partial D_{f_i}}{\partial P_i} = 2\left[\left(x_i(t) - x_j(t)\right) + \left(y_i(t) - y_j(t)\right) + \left(\theta_i(t) - \theta_j(t)\right)\right] \quad (4.49)$$

and,

$$\frac{\partial D_{f_j}}{\partial P_j} = 2\left[\left(x_j(t) - x_i(t)\right) + \left(y_j(t) - y_i(t)\right) + \left(\theta_j(t) - \theta_i(t)\right)\right] \tag{4.50}$$

Taking the derivative of R_{f_i} with respect to P_i and P_j, then use the chain rule and insert D_{f_i}

$$\frac{\partial R_{f_i}}{\partial(P_i)} = \left(\frac{\partial R_{f_i}}{\partial D_{f_i}} \frac{\partial D_{f_i}}{\partial P_i}\right) \tag{4.51}$$

$$\frac{\partial R_{f_i}}{\partial D_{f_i}} = \frac{1}{2} {D_{f_i}}^{-1/2} \tag{4.52}$$

$$\frac{\partial R_{f_i}}{\partial(P_j)} = \left(\frac{\partial R_{f_i}}{\partial D_{f_j}} \cdot \frac{\partial D_{f_j}}{\partial P_j}\right) \tag{4.53}$$

$$\frac{\partial R_{f_i}}{\partial D_{f_j}} = \frac{1}{2} {D_{f_j}}^{-1/2} \tag{4.54}$$

Using (4.52), (4.49) in (4.51) gives

$$\frac{\partial R_{f_i}}{\partial(P_i)} = \frac{1}{2} {D_{f_i}}^{-1/2}\, 2\left[\left(x_i(t) - x_j(t)\right) \quad \left(y_i(t) - y_j(t)\right) \quad \left(\theta_i(t) - \theta_j(t)\right)\right] \tag{4.55}$$

$$= \frac{1}{2} {D_{f_i}}^{-1/2} \left[\left(P_i(t) - P_j(t)\right)^T\right] \tag{4.56}$$

and using (4.54), (4.50) in (4.53), gives

$$\frac{\partial R_{f_i}}{\partial(P_j)} = \frac{1}{2} {D_{f_j}}^{-1/2}\ 2\left[\left(x_j(t) - x_i(t)\right) + \left(y_j(t) - y_i(t)\right) + \left(\theta_j(t) - \theta_i(t)\right)\right] \tag{5.57}$$

$$= \frac{1}{2} \left(D_{f_j}\right)^{-1/2} 2\left[\left(\mathrm{P}_j(\mathrm{t}) - \mathrm{P}_i(\mathrm{t})\right)^{\mathrm{T}}\right] \tag{5.58}$$

The sum of (4.56) and (4.58) results gives

$$\frac{\partial R_{f_i}}{\partial(P_i,P_j)} = \left(D_{f_i}\right)^{-1/2}\Big(\left[\left(x_i(t)-x_j(t)\right) \quad \left(y_i(t)-y_j(t)\right) \quad \left(\theta_i(t)-\theta_j(t)\right)\right] +$$

$$\left(D_{f_j}\right)^{-1/2}\left[\left(x_j(t)-x_i(t)\right) \quad \left(y_j(t)-y_i(t)\right) \quad \left(\theta_j(t)-\theta_i(t)\right)\right]\Big)$$

$$\frac{\partial R_{f_i}}{\partial(P_i,P_j)} = \left[\left(D_{f_i}\right)^{-1/2}\left(P_i(t)-P_j(t)\right)^T + \left(D_{f_j}\right)^{-1/2}\left(P_j(t)-P_i(t)\right)^T\right] \tag{4.59}$$

On the other hand, from (4.44) the derivative of V_{ij} with respect to d_{ij}

$$\frac{\partial \mathrm{V}_{ij}}{\partial(d_{ij})} = K_a(R_{f_i}-L) \text{ , when } R_{f_i} < L \tag{4.60}$$

and using the chain rule to differentiate V_{ij} w.r.t P_i, P_j ,

$$\frac{\partial \mathrm{V}_{ij}}{\partial(P_i,P_j)} = \frac{\partial \mathrm{V}_{ij}}{\partial R_{f_i}}\frac{\partial R_{f_i}}{\partial(P_i,P_j)} \tag{4.61}$$

Substituting (4.59),(4.60) in (4.61) leads to

$$\frac{\partial \mathrm{V}_{ij}}{\partial(P_i,P_j)} = K_a(R_{f_i}-L)\left[\left(D_{f_i}\right)^{-1/2}\left(P_i(t)-P_j(t)\right)^T + \left(D_{f_j}\right)^{-1/2}\left(P_j(t)-P_i(t)\right)^T\right]$$

$$= K_a(R_{f_i}-L)\left[\left(\frac{1}{R_{f_i}}\right)\left(P_i(t)-P_j(t)\right)^T + \left(\frac{1}{R_{f_j}}\right)\left(P_j(t)-P_i(t)\right)^T\right] \tag{4.62}$$

According to (4.45) we have

$$R_{f_i} = -R_{f_j} \tag{4.63}$$

Substituting (4.63) in (4.62) gives

$$\frac{\partial \mathrm{V}_{ij}}{\partial(P_i,P_j)} = K_a(R_{f_i} - L)\left[\left(\frac{1}{R_{f_i}}\right)\left(P_i(t) - P_j(t)\right)^T - \left(\frac{1}{R_{f_i}}\right)\left(P_j(t) - P_i(t)\right)^T\right] \tag{4.64}$$

The interagent potential is given by

$$\therefore\ P_{ij} = -K_a\left(R_{f_i} - L\right)\frac{1}{R_{f_i}}\left[\left(P_i(t) - P_j(t)\right)^T - \left(P_j(t) - P_i(t)\right)^T\right] \tag{4.65}$$

4.5.3 Stability analysis

To get the desired formation we consider our stable configuration to be a regular polygon shape. Choose the total energy of the closed loop system as a Lyapunov candidate function '$\tilde{V}$' :

$$\tilde{V} = \sum_{i=1}^{n}\left[\mathrm{V}_{c_i}(P_i) + \sum_{j=1;j\neq i}^{n}\mathrm{V}_{ij}(P_i,P_j) + \frac{1}{2}\mathrm{v}_i{}^T\bar{M}\mathrm{v}_i\right] \tag{4.66}$$

$$\tilde{V} = \sum_{i=1}^{n}[Pot_i + Kin_i] \tag{4.67}$$

$$Pot_i = \mathrm{V}_{c_i}(P_i) + \sum_{j=1;j\neq i}^{n}\mathrm{V}_{ij}(P_i,P_j) \tag{4.68}$$

$$Kin_i = \sum_{i=1}^{n}\frac{1}{2}\mathrm{v}_i{}^T\bar{M}\ \mathrm{v}_i \tag{4.69}$$

For the desired shape formation, the system will be stable if and only if $\tilde{V}(P_i) \geq 0$, and the time derivative of a Lyapunov candidate function '$\dot{\tilde{\mathrm{V}}} \leq 0$':

$$\dot{\tilde{V}}(P_i) = \sum_{i=1}^{n}\{\ [\nabla_{p_i}Pot_i]\,\dot{P}_i + [Kin_i]\dot{\mathrm{v}}_i\} \tag{4.70}$$

$$\dot{\tilde{V}}(P_i) = \sum_{i=1}^{n}\left[\nabla_{P_i}\mathrm{V}_{c_i}(P_i) + \sum_{j=1;j\neq i}^{n}\nabla_{P_i}\mathrm{V}_{ij}(P_i,P_j)\right]\dot{P}_i + \sum_{=1}^{n}\frac{\partial}{\partial V_i}\left[\frac{1}{2}\mathrm{v}_i{}^T\bar{M}\mathrm{v}_i\right]\dot{\mathrm{v}}_i \tag{4.71}$$

We have

$$\frac{\partial}{\partial V_i}\left[\frac{1}{2}\mathrm{v}_i{}^T\bar{M}\ \mathrm{v}_i\right]\dot{\mathrm{v}}_i = \mathrm{v}_i{}^T\bar{M}\ \dot{\mathrm{v}}_i \tag{4.72}$$

From (4.8) we have $\dot{q} = S\mathrm{v}_i$,

Also we know that

$$\dot{P}_i = \dot{q} = S\mathrm{v}_i \tag{4.73}$$

$$\dot{\mathrm{v}}_i = u_i \tag{4.74}$$

Substitute (4.74) in (4.72) we get:

$$\frac{\partial}{\partial \mathrm{V}}\left[\frac{1}{2}\mathrm{v}_i{}^{\mathrm{T}}\overline{\mathrm{M}}\ \mathrm{v}_i\right]\dot{\mathrm{v}}_i = \mathrm{v}_i{}^{\mathrm{T}}\overline{\mathrm{M}}u_i \tag{4.75}$$

Now Substitute (4.73) and (4.75) in (4.71)

$$\dot{\tilde{\mathrm{V}}} = \sum_{\mathrm{i=1}}^{\mathrm{n}}\left[\nabla_{P_i}\mathrm{V}_{c_i}(P_i) + \sum_{j=1;j\neq i}^{n}\nabla_{P_i}\mathrm{V}_{ij}(P_i,P_j)\right]S\mathrm{v}_i + \sum_{=1}^{\mathrm{n}}\mathrm{v}_i{}^{\mathrm{T}}\overline{\mathrm{M}}\,u_i \tag{4.76}$$

For simplicity we will use $\nabla_{p_i}Pot_i$ instead of $\left[\nabla_{P_i}\mathrm{V}_{c_i}(P_i) + \sum_{j=1;j\neq i}^{n}\nabla_{P_i}\mathrm{V}_{ij}(P_i,P_j)\right]$, so (4.76) will become:

$$\dot{\tilde{\mathrm{V}}} = \sum_{\mathrm{i=1}}^{\mathrm{n}}\left[\nabla_{p_i}Pot_i\ \right]\mathrm{S}\mathrm{v}_i + \sum_{=1}^{\mathrm{n}}(\mathrm{v}_i{}^{\mathrm{T}}\overline{\mathrm{M}}u_i) \tag{4.77}$$

$$\Rightarrow \dot{\tilde{\mathrm{V}}} = \sum_{\mathrm{i=1}}^{\mathrm{n}}\left(\mathrm{v}_i{}^T\mathrm{S}^T\left[\nabla_{p_i}Pot_i\ \right]^T\right) + \sum_{i=1}^{n}(\sum_{=1}^{\mathrm{n}}\mathrm{v}_i{}^{\mathrm{T}}\overline{\mathrm{M}}\,u_i) \tag{4.78}$$

For the desired configuration the system must be stable. For that we need to choose the control u_i as:

$$u_i = \bar{M}^{-1}\left[-S^T\left[\nabla_{p_i}Pot_i\ \right]^T - K_V\mathrm{v}_i\right] \tag{4.79}$$

Substitute (4.79) in (4.78) and compute $\dot{\tilde{V}}$ we get:

$$\dot{\tilde{V}} = \sum_{i=1}^{n}\left(v_i^T S^T \left[\nabla_{p_i} Pot_i\right]^T\right) + \sum_{i=1}^{n}\left(v_i^T \bar{M}\left(\bar{M}^{-1}\left[-S^T\left[\nabla_{p_i} Pot_i\right]^T - K_V v_i\right]\right)\right) \quad (4.80)$$

$$\Rightarrow \dot{\tilde{V}} = \sum_{i=1}^{n}\left(v_i^T S^T \left[\nabla_{p_i} Pot_i\right]^T\right) + \sum_{i=1}^{n}\left(v_i^T\left(\left[-S^T\left[\nabla_{p_i} Pot_i\right]^T - K_V v_i\right]\right)\right)$$

$$\Rightarrow \dot{\tilde{V}} = \sum_{i=1}^{n}\left(v_i^T S^T \left[\nabla_{p_i} Pot_i\right]^T\right) - \sum_{i=1}^{n}\left(v_i^T S^T \left[\nabla_{p_i} Pot_i\right]^T\right) - \sum_{i=1}^{n} v_i^T K_V v_i$$

$$\Rightarrow \dot{\tilde{V}} = -\sum_{i=1}^{n} v_i^T K_V v_i \leq 0 \quad (4.81)$$

which proves stability.

4.6 Unknown Robot Dynamics

In the previous discussion we considered all robot dynamics are known and the designed formation control strategy works very well when all the dynamical terms $M(q), V_m(q,\dot{q})$ are known. In practice, the robot dynamics or parameters in many applications may be unknown or changing with time. Therefore, in many cases the controller that was discussed in the previous section is not appropriate for such cases.

In this section, we will discuss the formation control on the assumption that all system dynamics are unknown. On-line NN weight tuning algorithms guarantee tracking a desired following path and error dynamics. The follower will localize itself on its desired trajectory based on the potential fields effect.

For unknown robot dynamics the proposed framework is also a combination of two stages. The first one is the potential fields or the formation controller. This stage is discussed in detail in the previous sections. The second one is the NN robot controller to guarantee tracking the desired path and modeling robot dynamics. See figure (4.3).

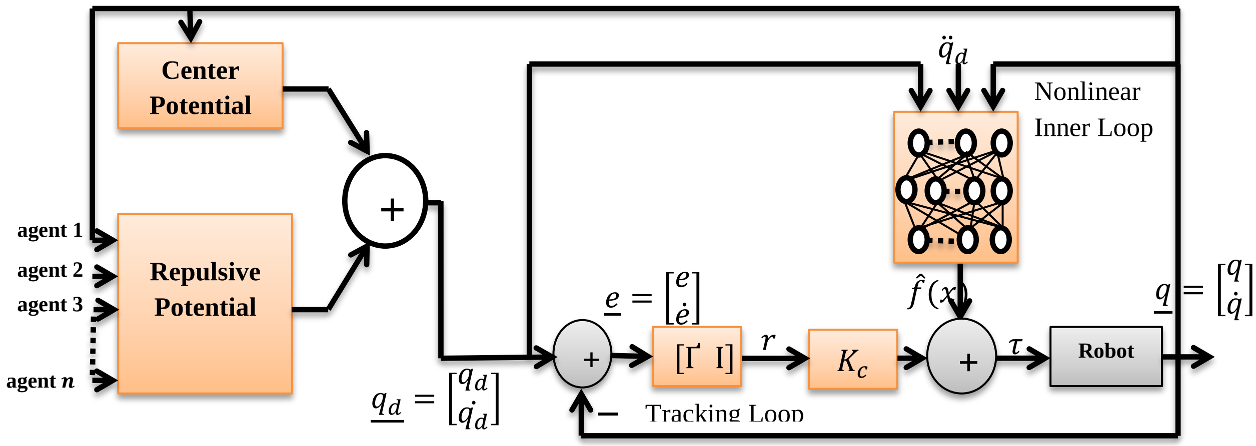


Figure (4.3) The general structure of the system in the case of unknown robot dynamics

4.6.1 Tracking a desired path and the error dynamics

The robot has to follow its desired trajectory q_d. The tracking error is [57] the difference between the desired path trajectory q_d and the estimated trajectory q.

$$E(t) = q_d(t) - q(t) \tag{4.82}$$

Where

$$q_d = \begin{bmatrix} x_f \\ y_f \end{bmatrix} \text{ and } q = \begin{bmatrix} x \\ y \end{bmatrix} \tag{4.83}$$

The filtered tracking error is given by

$$r(t) = \dot{e} - \Gamma e \tag{4.84}$$

$\Gamma > 0$: is a PD design parameter matrix.

By using equation (3.1) and deferentiating equation (4.84). The dynamics of the robot can be written in terms of filtered error [56] as

$$M\dot{r} = -V_m r - \tau + f + \tau_d \tag{4.85}$$

where

f: is the nonlinear robot function which equal

$$f(x) = M(q)(\ddot{q}_d - \Gamma\dot{e}) + V_m(q,\dot{q})(\dot{q}_d + \Gamma e) + G(q) + F(\dot{q}) \tag{4.86}$$

To compute $f(x)$ we need to define the vector x

$$x = [e^T \quad \dot{e}^T \quad {q_d}^T \quad {\dot{q}_d}^T \quad {\ddot{q}_d}^T]^T \tag{4.87}$$

where x can be measured.

The apprperiate controller can be used for the path following is derived by putting

$$\tau = \hat{f}(x) + K_c r \tag{4.88}$$

where

$K_c = {K_c}^T$: is the gain matrix, and $K_c r$ is the outer PD tracking loop

$\hat{f}(x)$: is the estimate of $f(x)$.

Using (4.88) controller, the closed loop error dynamics is given by

$$M\dot{r} = -(K_c + V_m)r + \tilde{f} + \tau_d \tag{4.89}$$

The functional estimated error $\tilde{f}$ is:

$$\tilde{f} = f - \hat{f} \tag{4.90}$$

4.6.2 Neural Network Controller

The controller (4.88) will minimize the tracking error based on selecting an appropriate gain value for K_c and estimating $\hat{f}$. The error $r(t)$ and the control signals should be bounded [57]. The structure of this controller is given in figure (4.3). This controller uses PD control in the outer tracking loop and Neural network control in the inner loop to estimate the robot function $f(x)$. The NN controller structure will be selected as

$$\tau = \widehat{W}\sigma\left(\hat{V}^T x\right) + K_c r \tag{4.91}$$

Tunning weights of NN can be selected by using equation (4.89) and selecting K_c which can stabilize the filtered tracking error $r(t)$. [56],[57].

For whole system to be stable the outer formation loop must be slower than the inner robot stabilization loop.

4.7 Simulation Results

In this section we will report the simulation results of the leader motion generated using SLAM [2]. This map, as shown in figure (4.4), is a group leader's path. SLAM algorithm based control is implemented on the leader, which has been used as a navigation generator for all agents in the fleet. The robot system is implemented in MATLAB. The robot parameters used in this study table (3.1), are $m = 10kg, r = 0.05\, m, R = 0.5\, m, I = 5kg\text{-}m^2; d = 0.8m$ .See figure (3.1).

The overall simulation results of the leader-follower group formation and whole fleet navigation are shown in the figures below. From figure (4.5), we can see how the group of three followers can localize themselves and create a star shape around their leader .

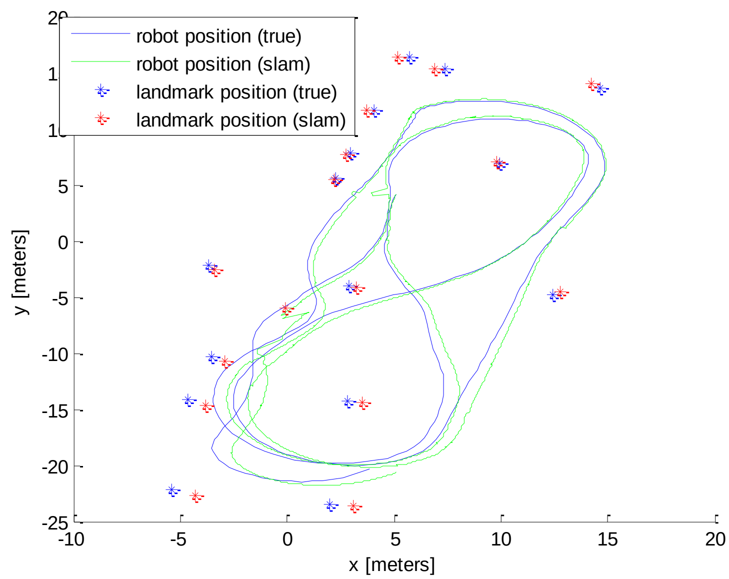
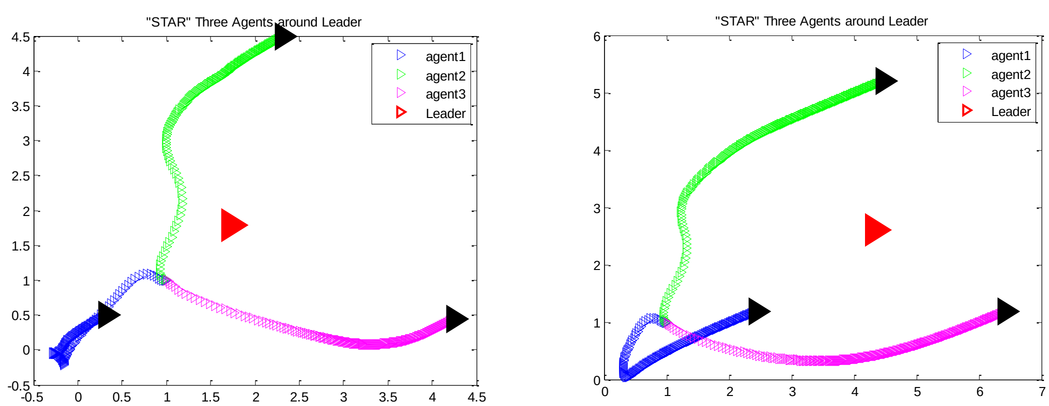
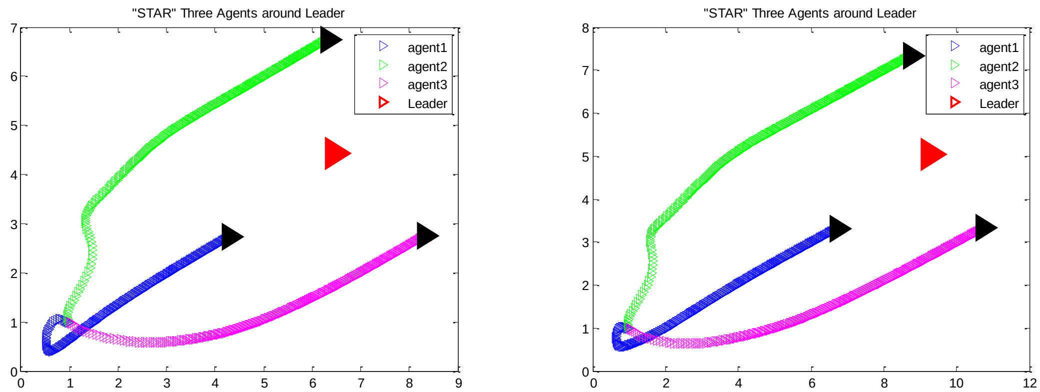


Figure (4.4). Trajectory estimation using SLAM with natural features



a) Leader at (1.6,1.7), followers start from (1,1). b) Leader at (4.2,2.7), followers start from (1,1).



c) Leader at (6.1,4.4), followers start from (1,1). d) Leader at (9.1,4.8), followers start from (1,1)

Figure (4.5). Star shape formation.

Figure (4.6) shows that the fleet of three nonholonomic robots follow their leader based on the potential fields and feedback linearization control assuming all robots dynamics are known. All robots navigate together and follow their leader's SLAM mapping.

Figure (4.7) shows that the fleet of three nonholonomic robots in the case of unknown robot dynamics follow their leader based on the potential fields and NN control. All robots also navigate together and follow their leader's SLAM mapping.

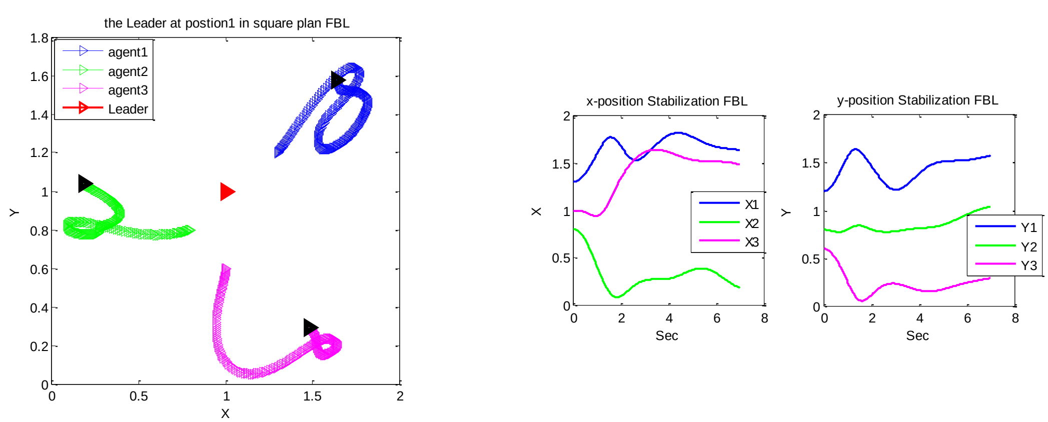


Figure (4.6.a). Three robots formation control response based on potential fields and feedback linearization.

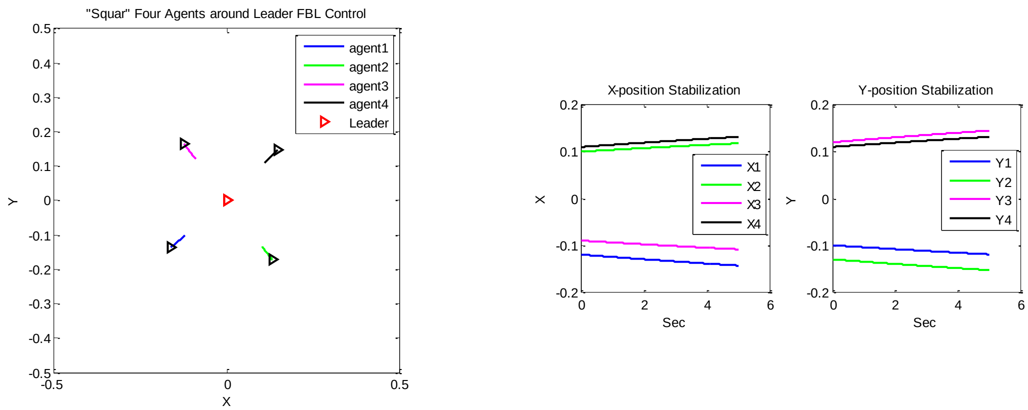


Figure (4.6.b). Four robots formation control response based on potential fields and feedback linearization.

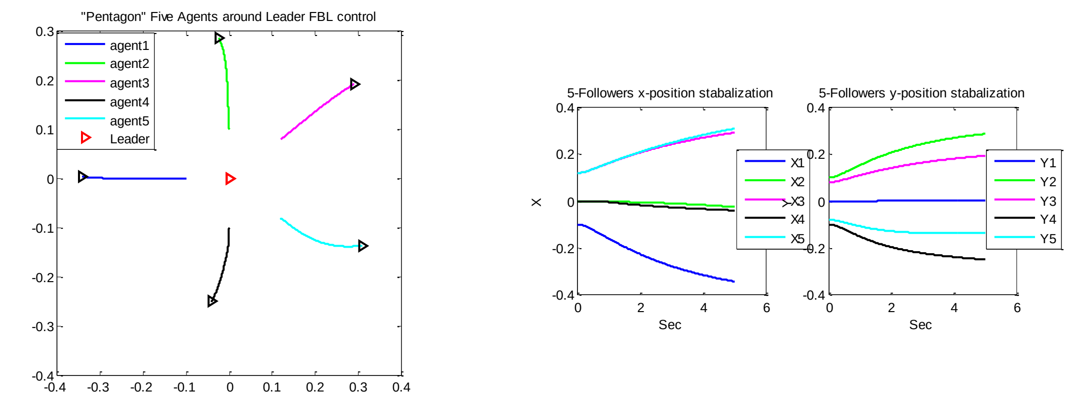


Figure (4.6.c). Five robots formation control response based on potential fields and feedback linearization.

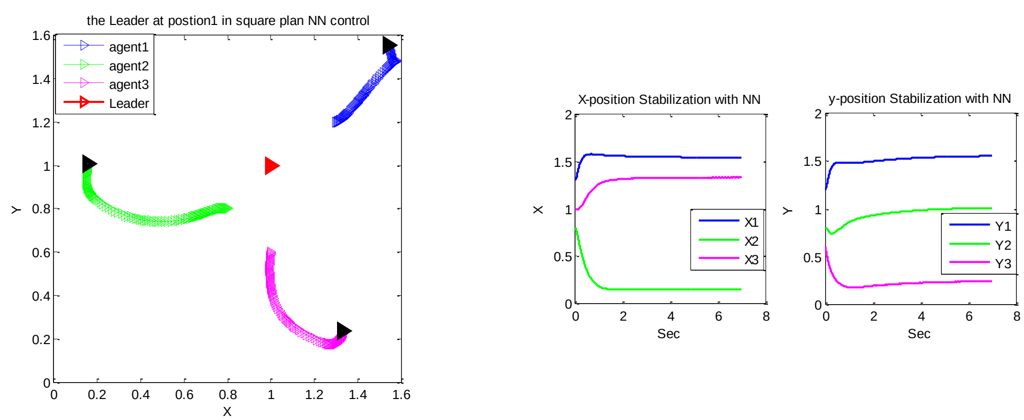


Figure (4.7.a). Three robots formation control response based on potential fields and NN.

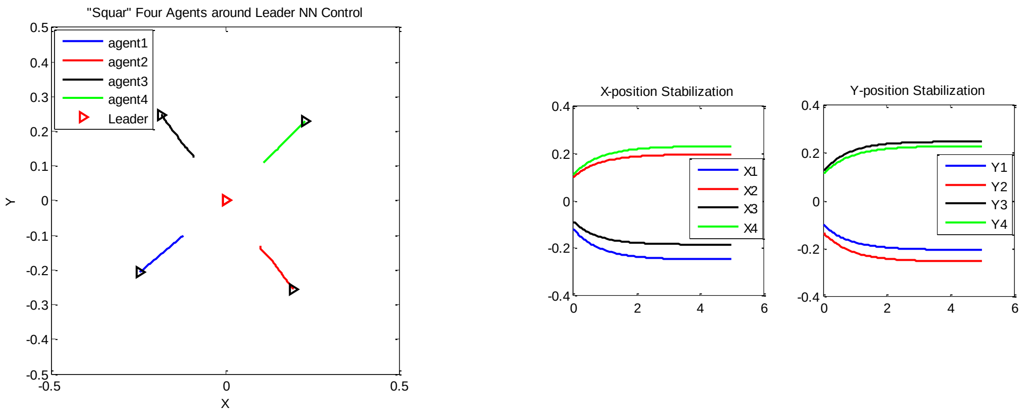


Figure (4.7.b). four robots formation control response based on potential fields and NN.

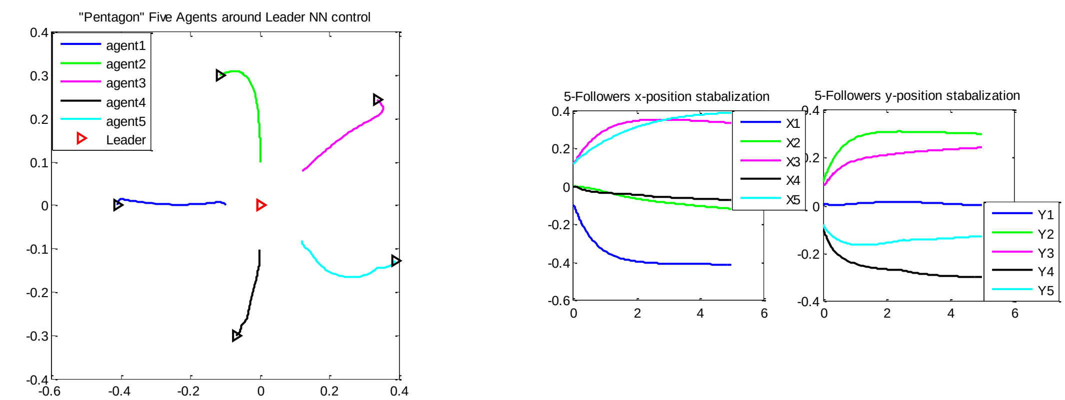


Figure (4.7.c). Five robots formation control response based on potential fields and NN.

The figure (4.8) gives comparison between the two designed controllers' response along the whole navigations path of a group of three robots navigation. The figures on the left hand side is representing the response of the controller of known dynamics system. The figures on the right hand side is representing the response of the controller of unknown dynamics system. Figure (4.9) shows the total groups map in the case of known dynamic system under the potential fields and feedback linearization control. Figure (4.10) shows the total groups map in the case of unknown dynamic system under the potential fields and neural network control.

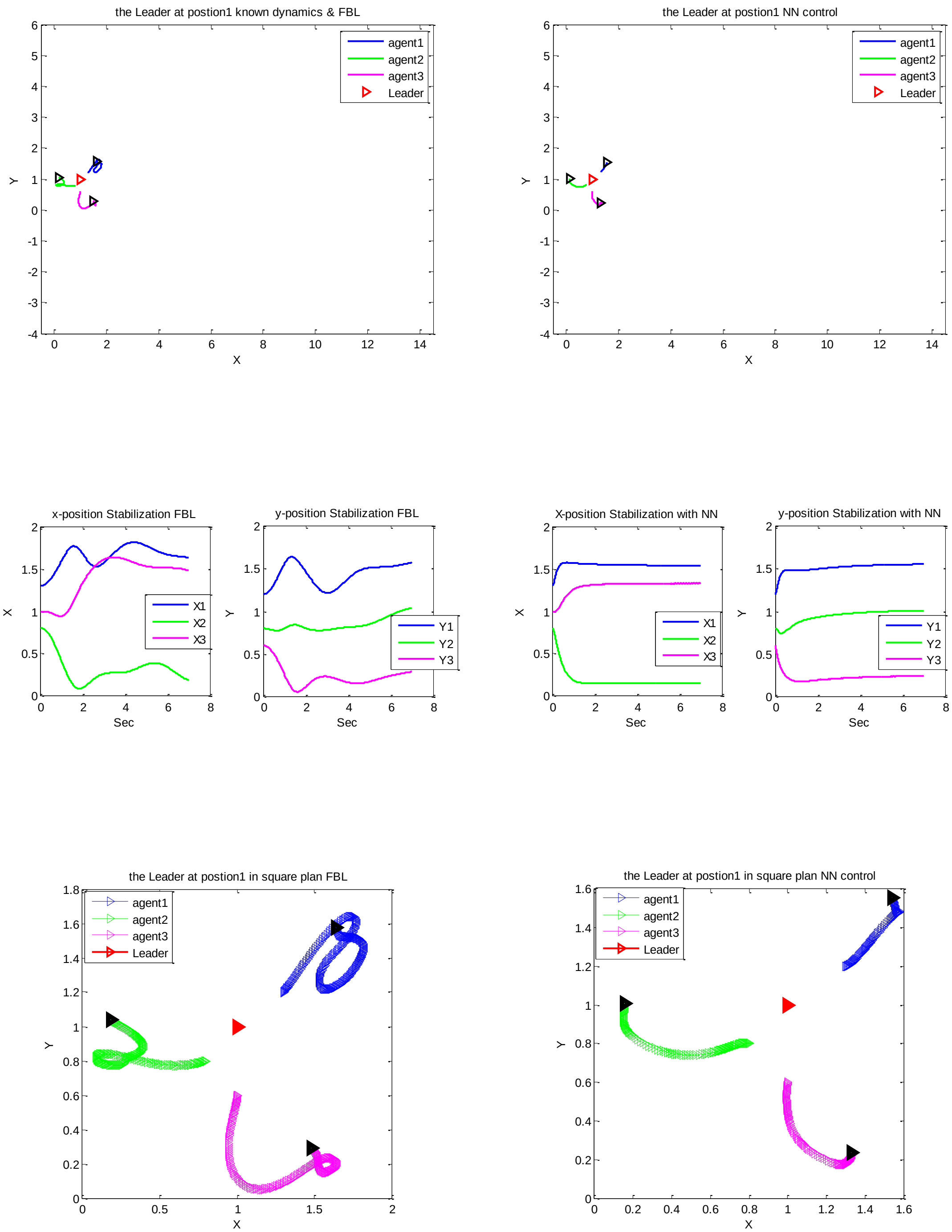


Figure (4.8.1). The leader at the postion $Pc = (1,1)$

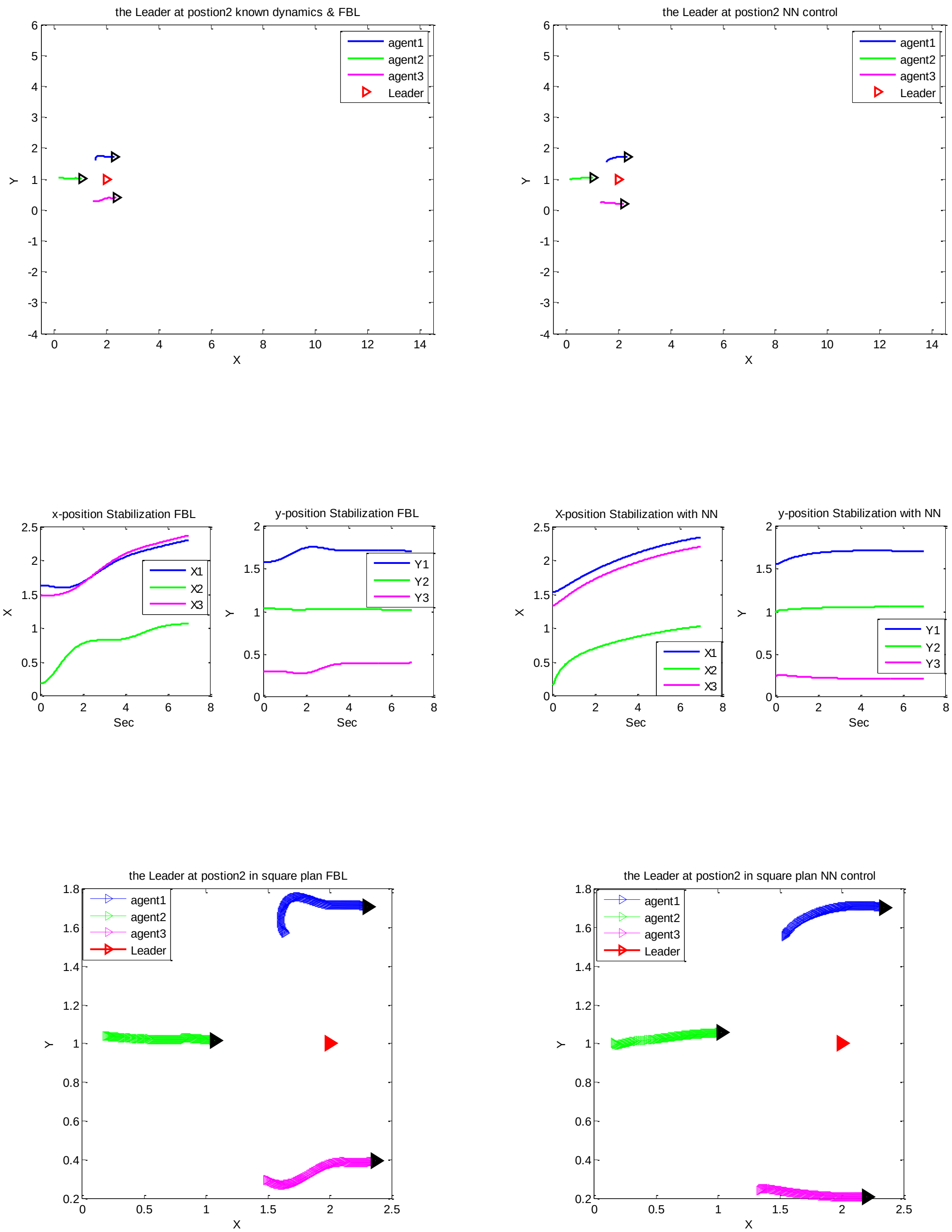


Figure (4.8.2). The leader at the postion $Pc = (2,1)$

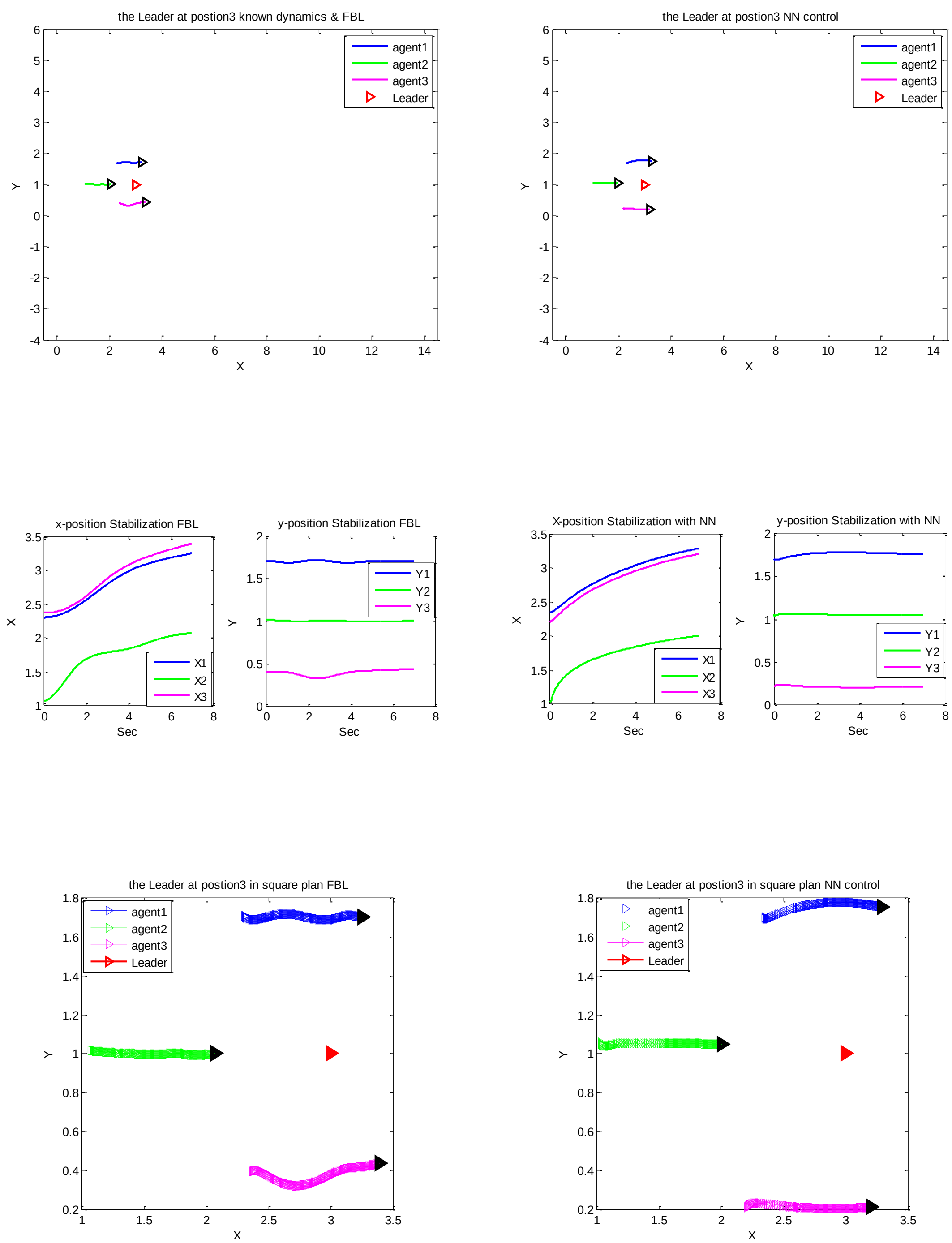


Figure (4.8.3). The leader at the postion $Pc = (3,1)$

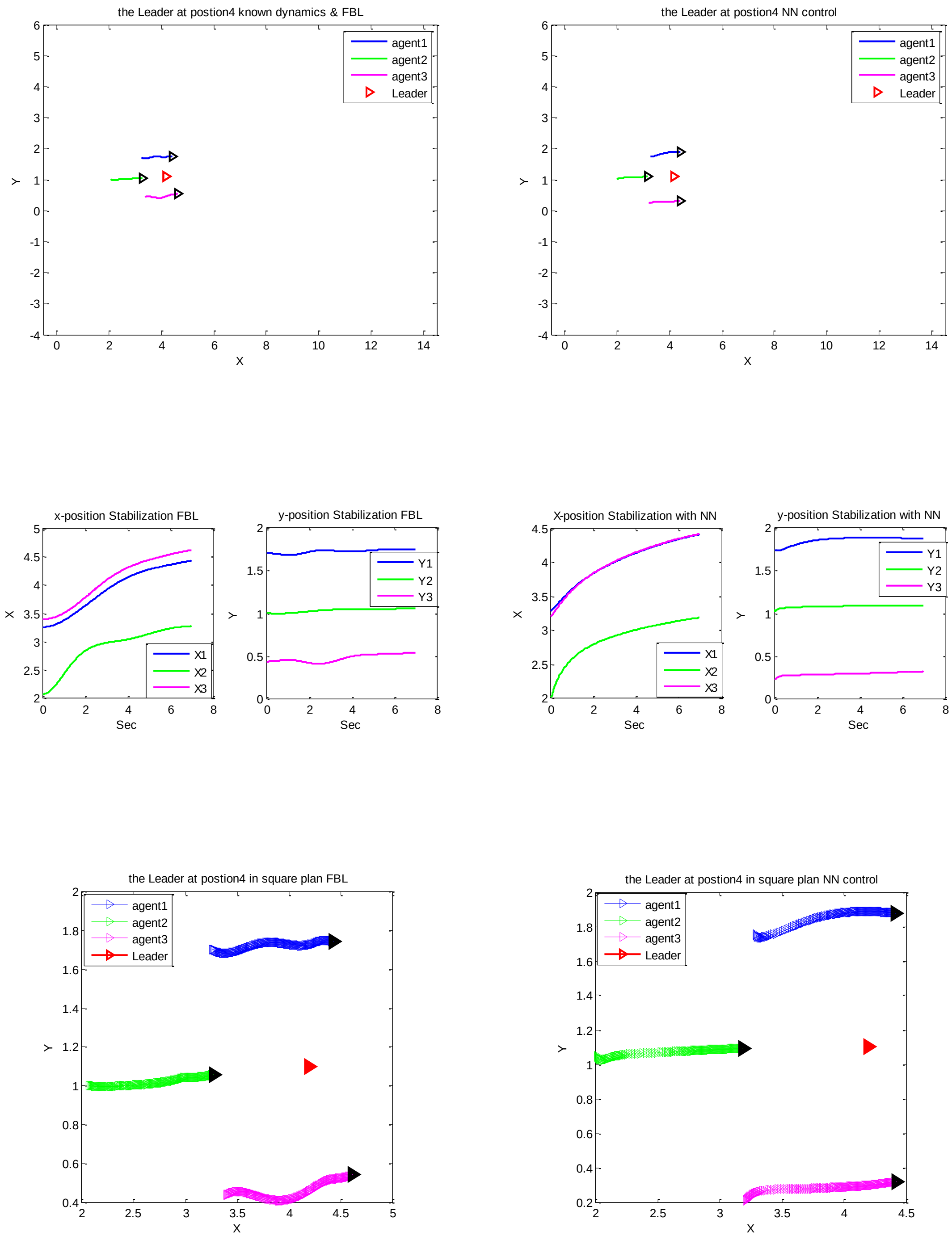


Figure (4.8.4). The leader at the postion $Pc = (4.2,1.1)$

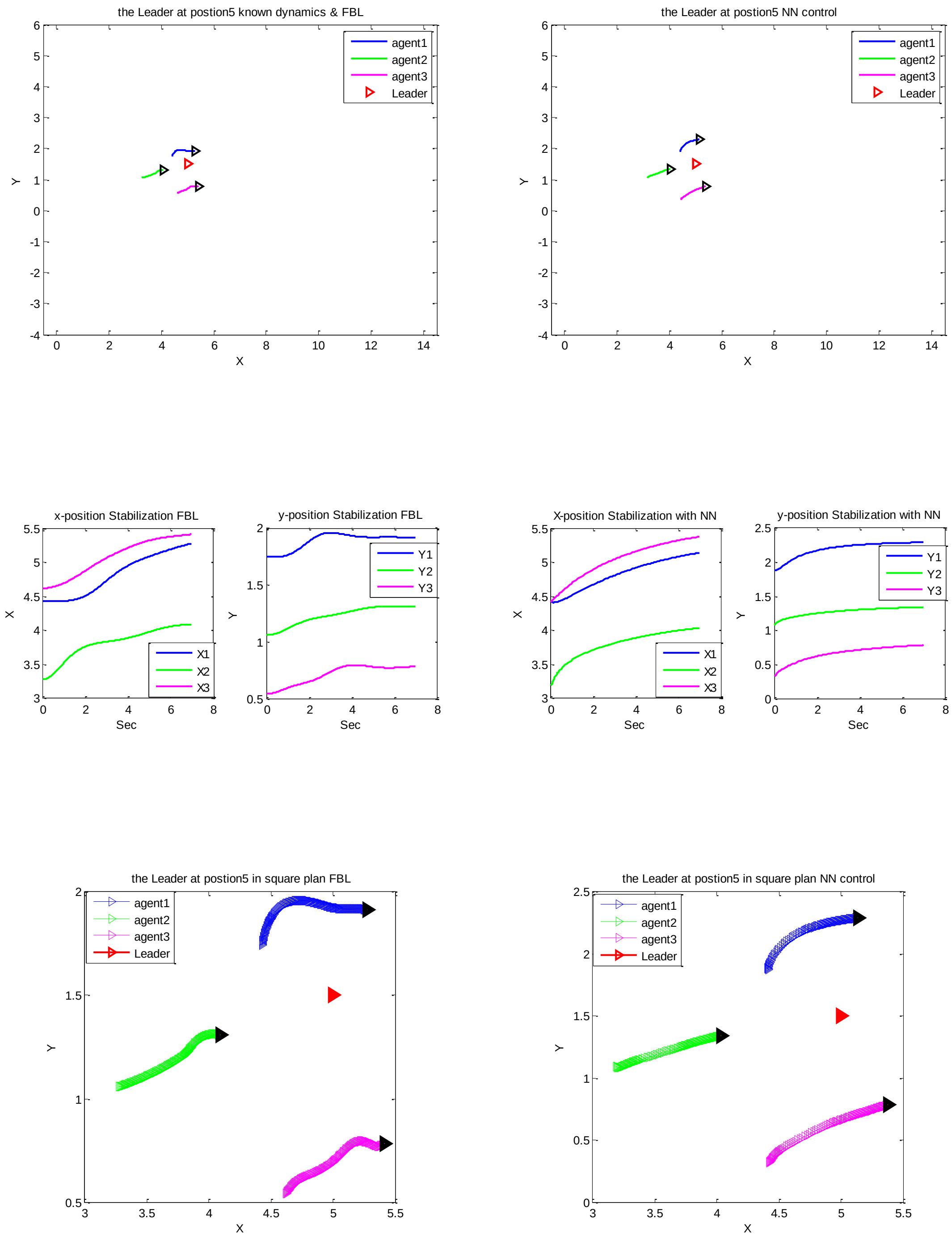


Figure (4.8.5). The leader at the postion $Pc = (5,1.5)$

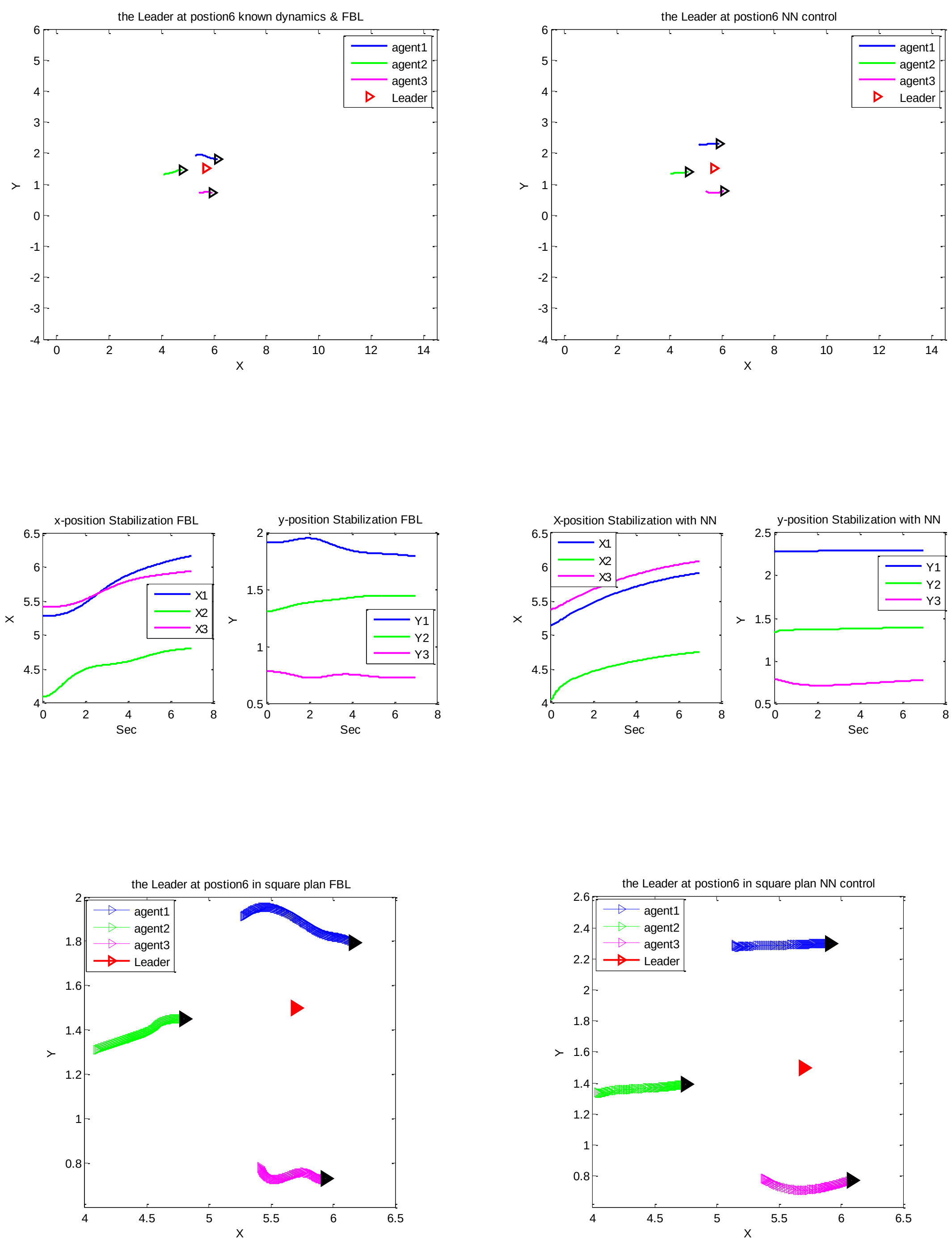


Figure (4.8.6). The leader at the postion Pc = (5.7,1.5)

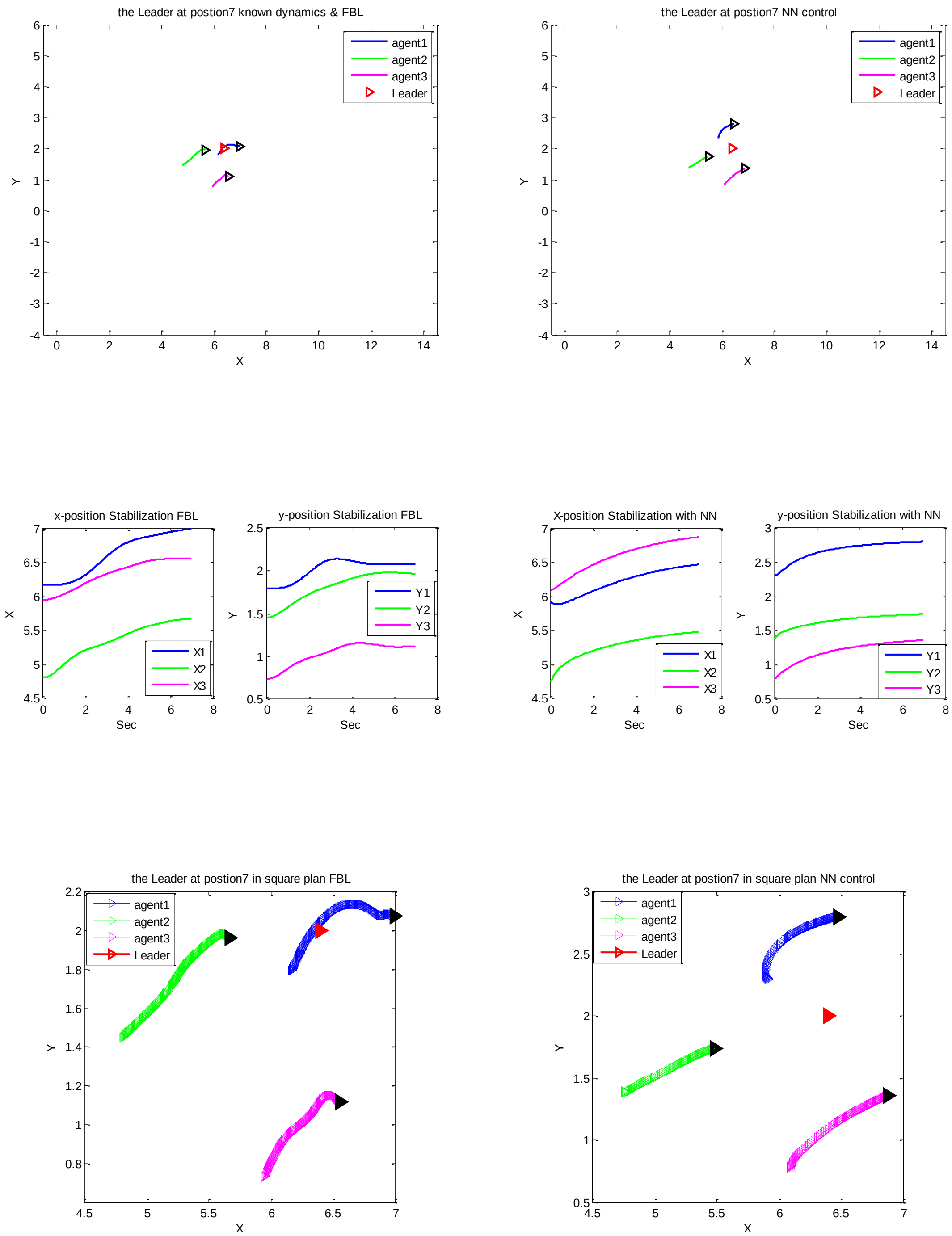


Figure (4.8.7). The leader at the postion Pc = (6.4,2)

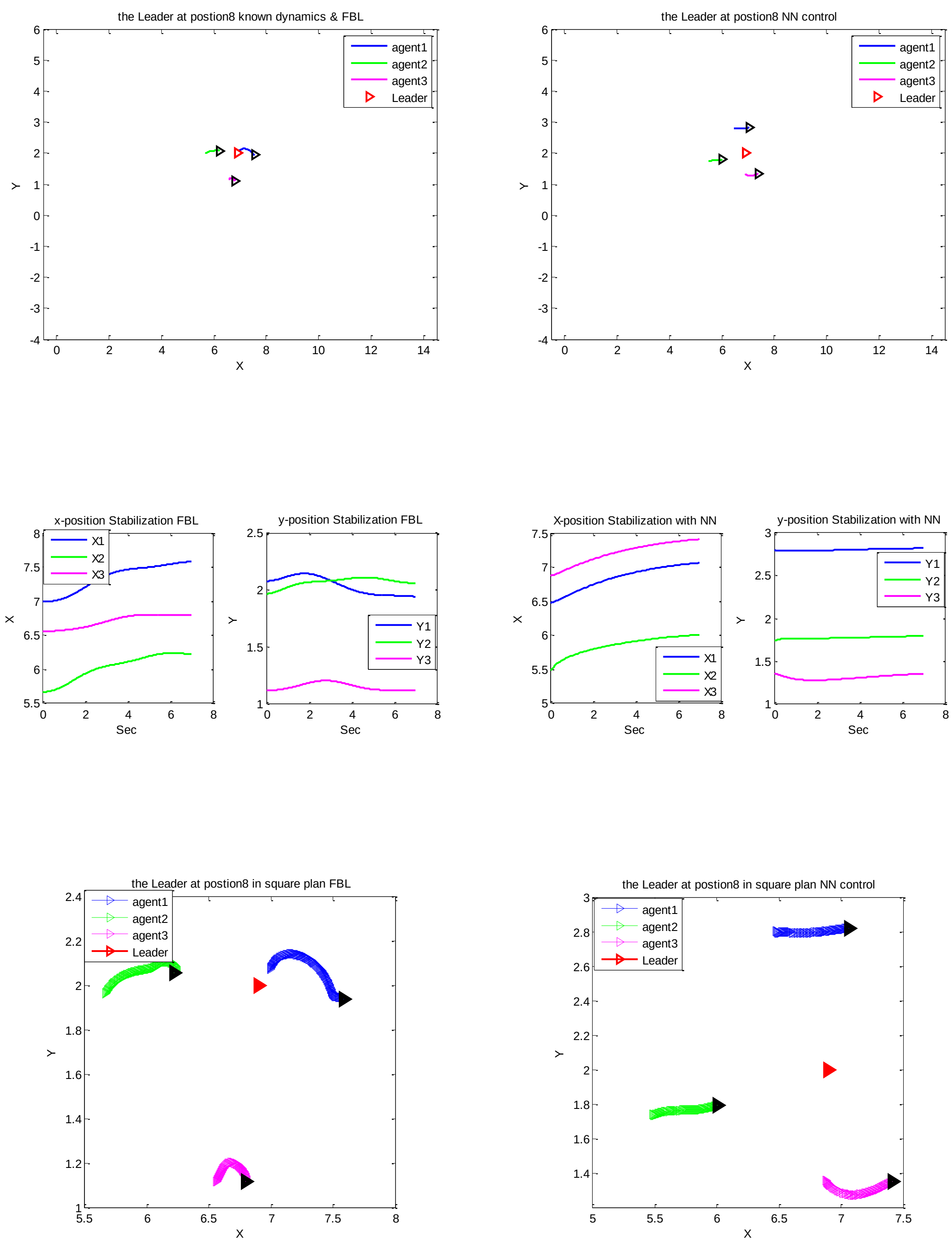


Figure (4.8.8). The leader at the postion Pc = (6.9,2)

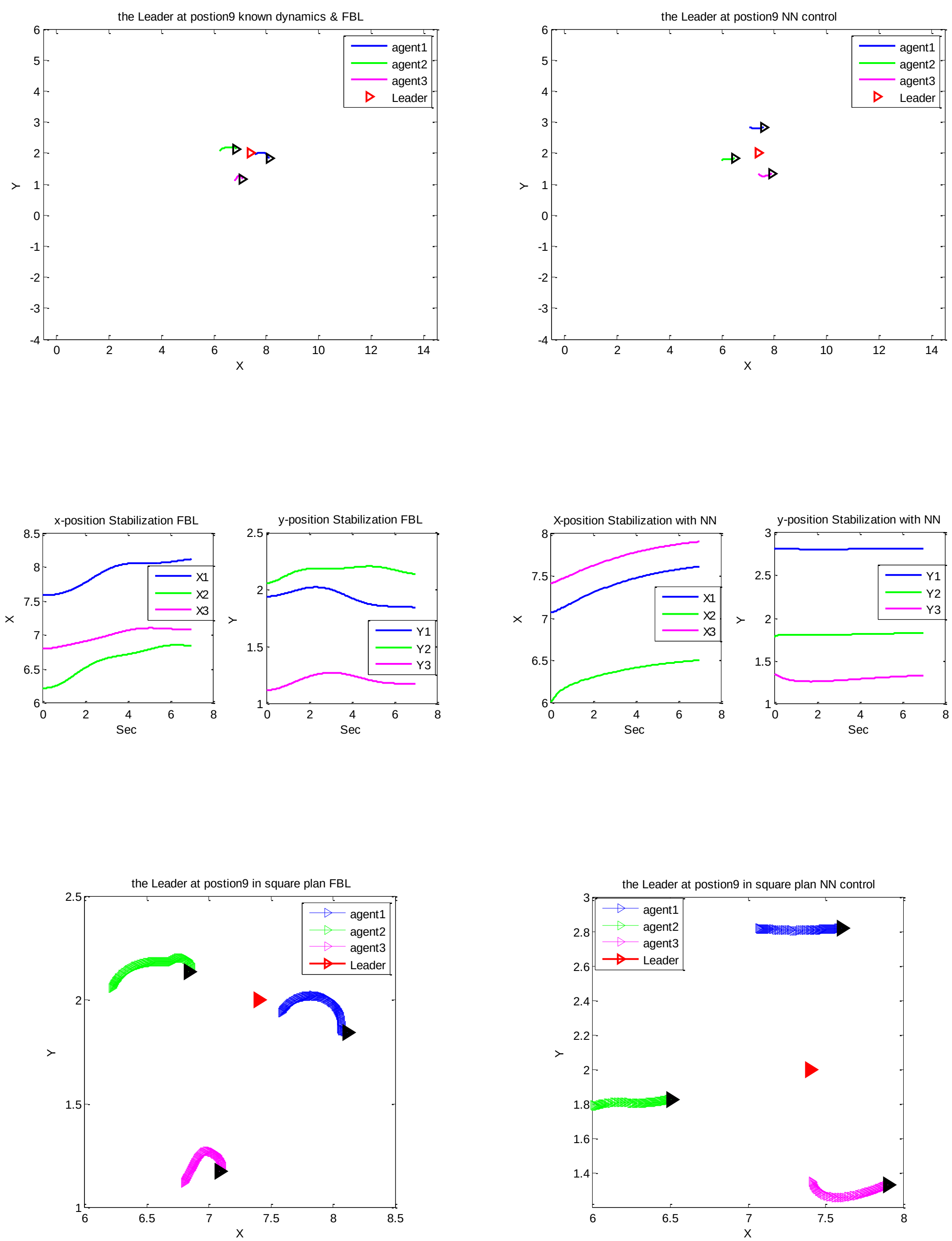


Figure (4.8.9). The leader at the postion Pc = (7.4,2)

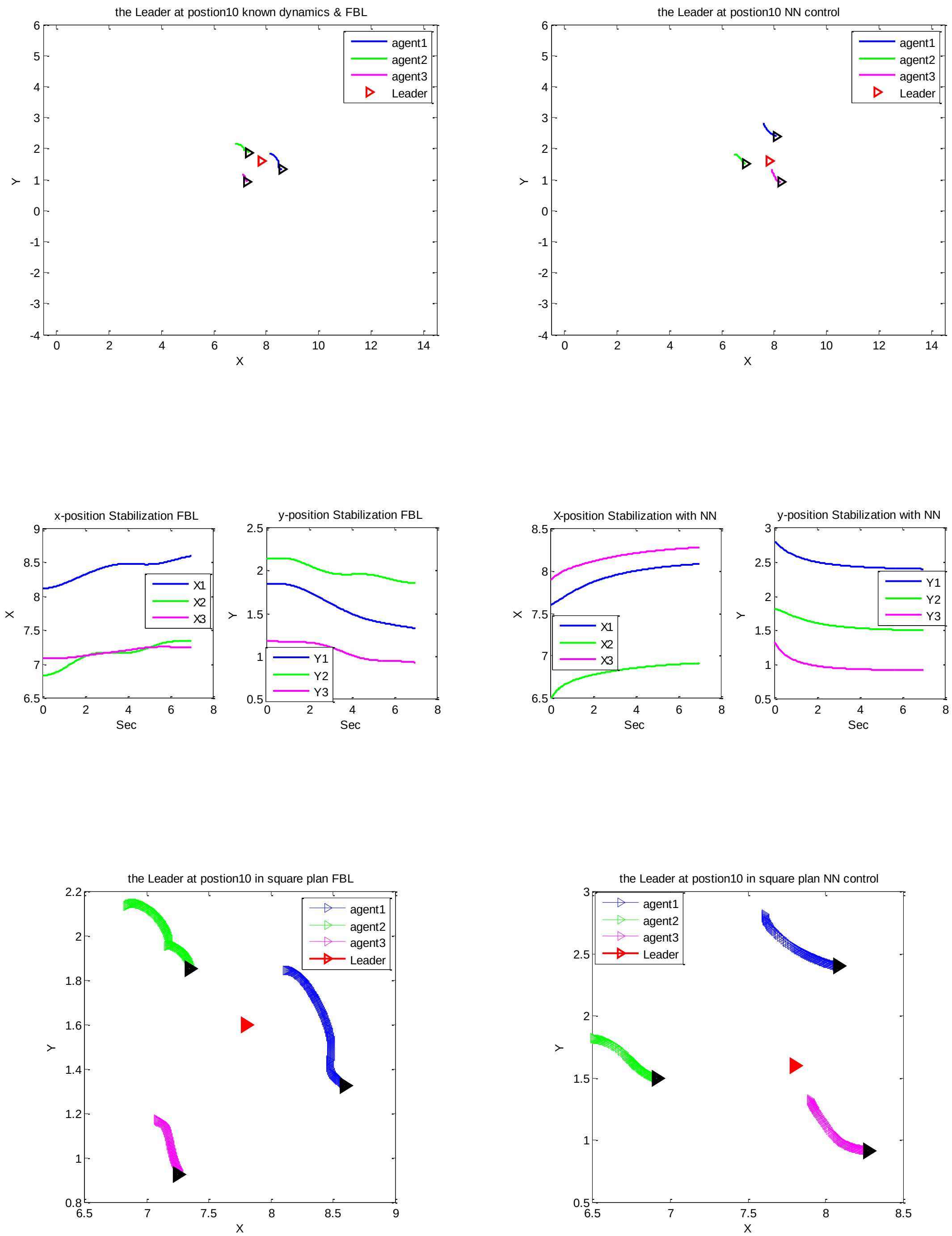


Figure (4.8.10). The leader at the postion Pc = (7.8,1.6)

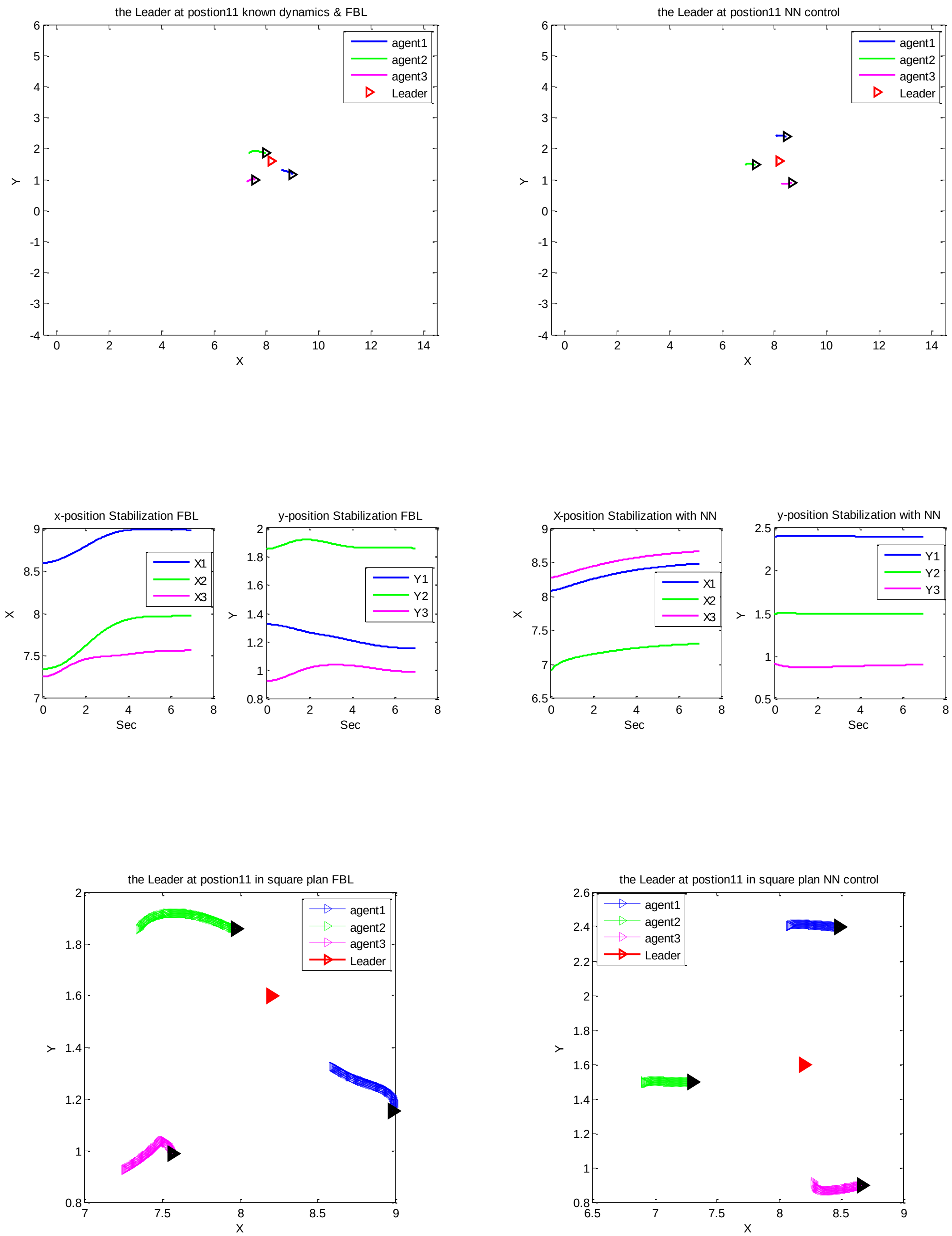


Figure (4.8.11). The leader at the postion Pc = (8.2,1.6)

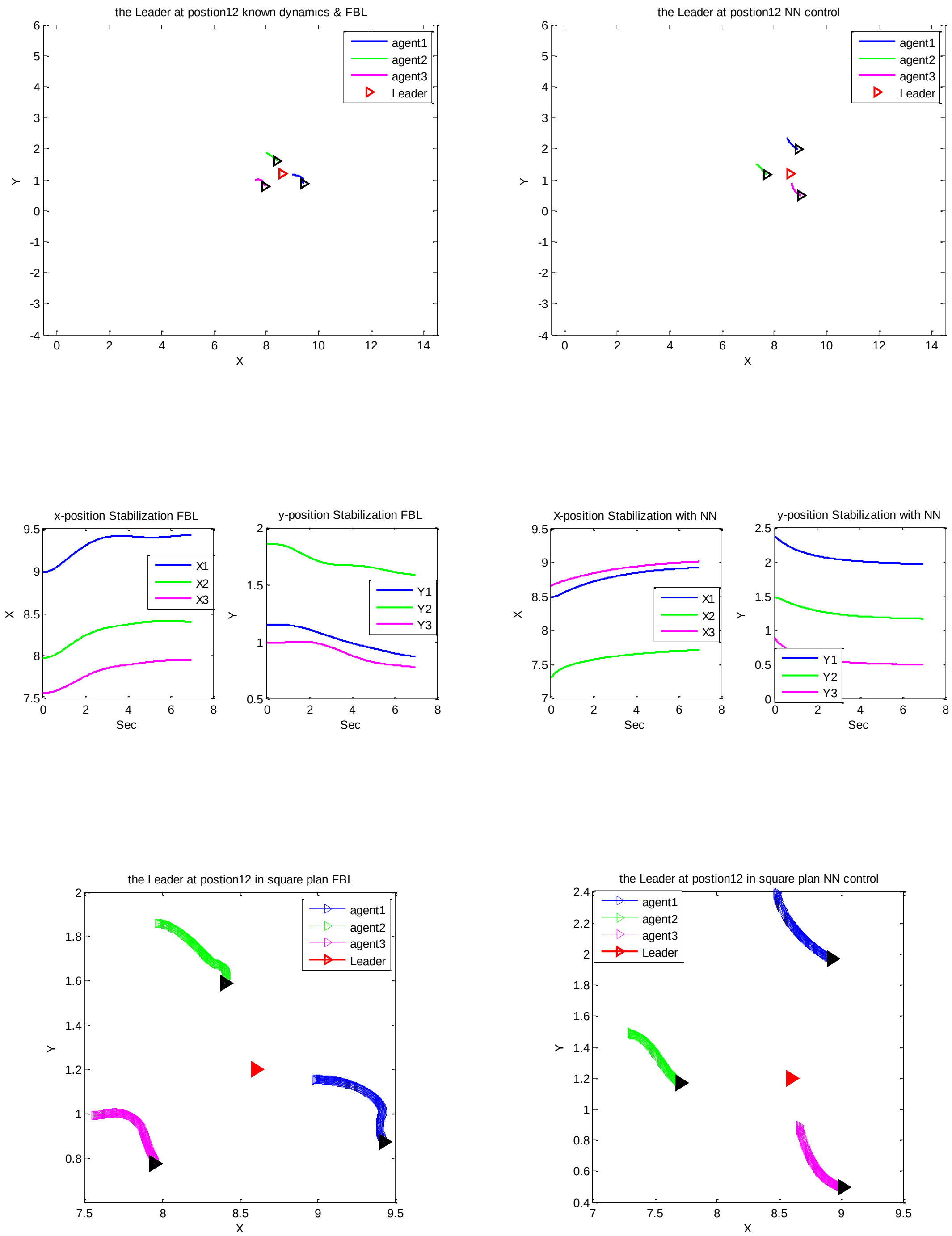


Figure (4.8.12). The leader at the postion Pc = (8.6,1.2)

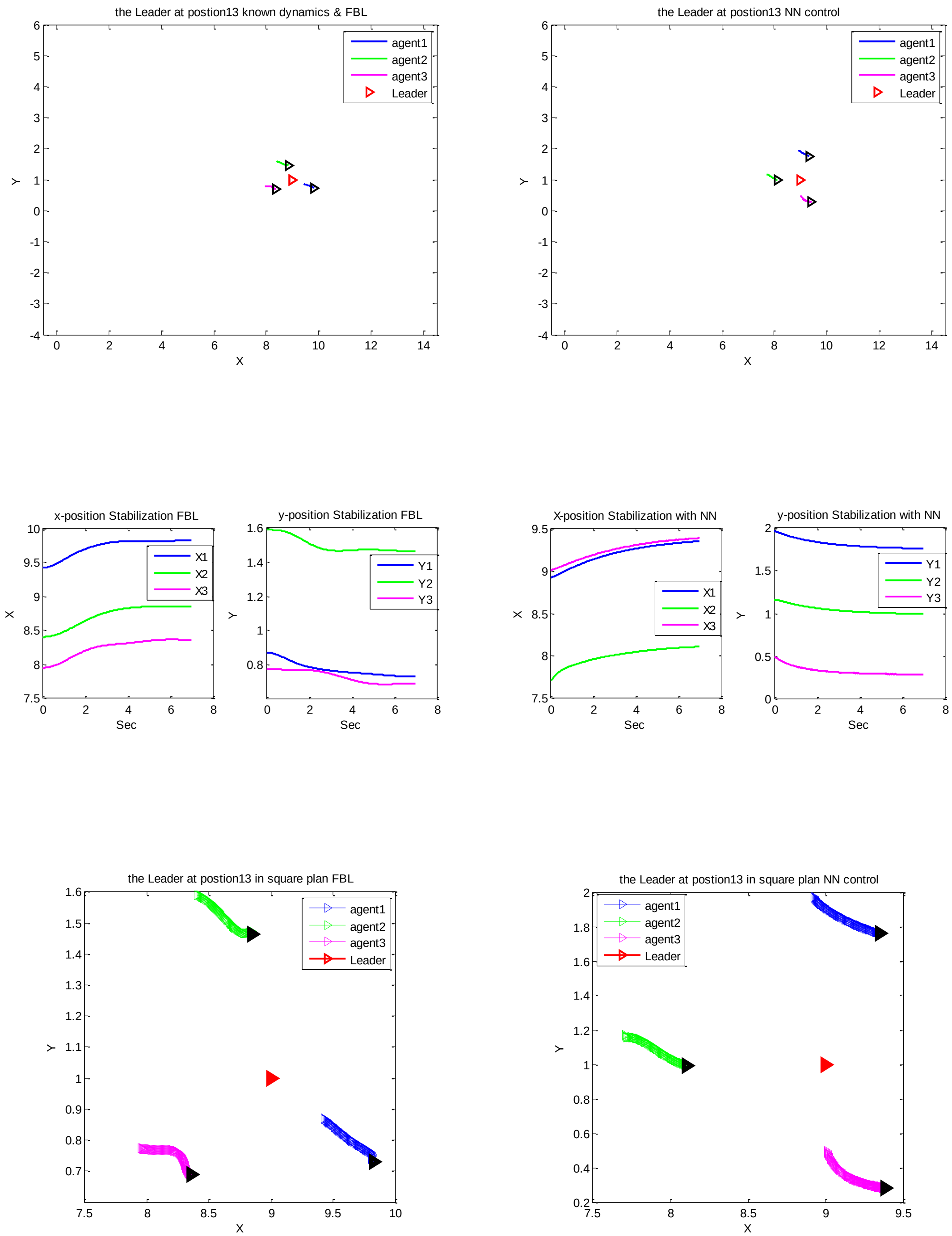


Figure (4.8.13). The leader at the postion Pc = (9,1)

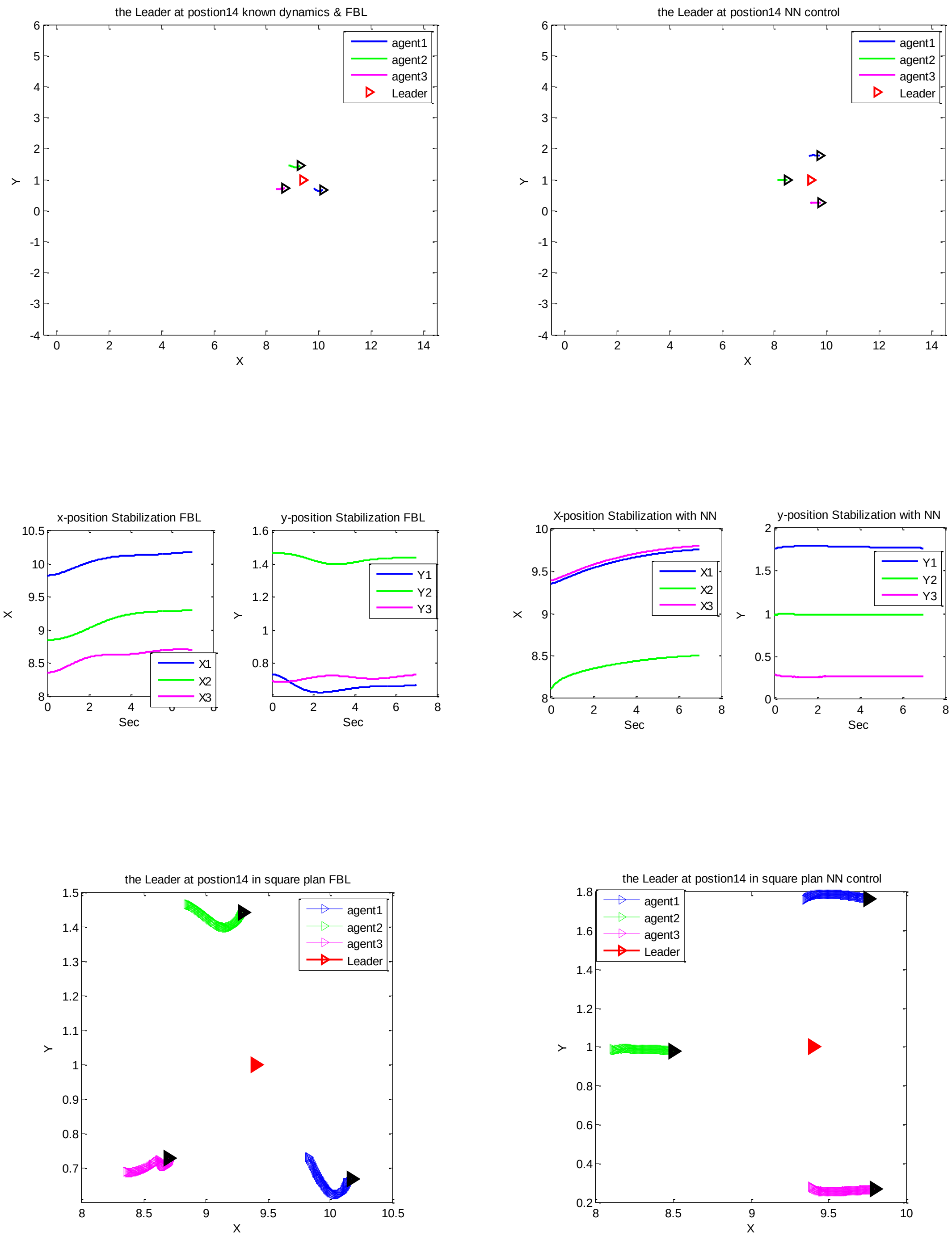


Figure (4.8.14). The leader at the postion Pc = (9.4,1)

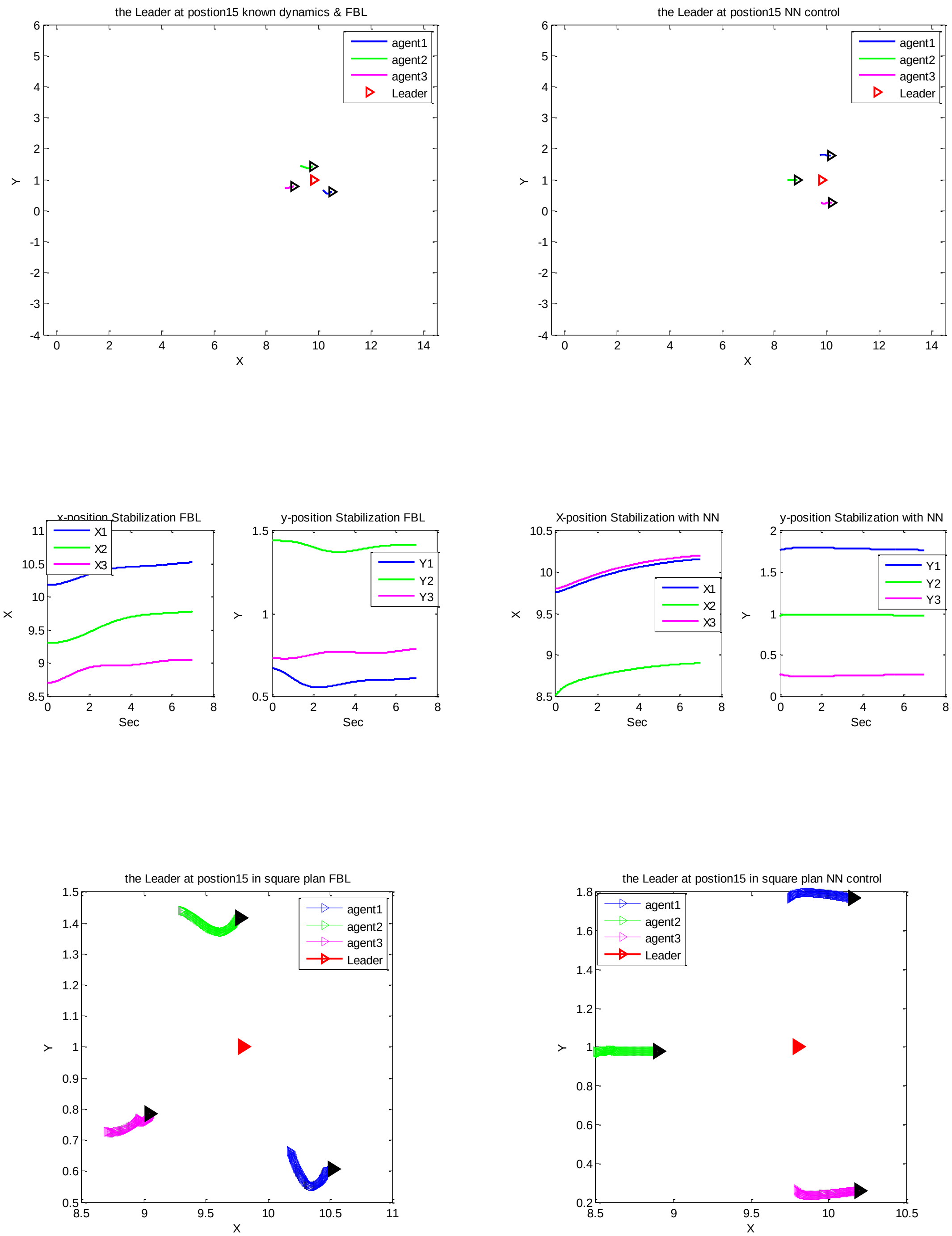


Figure (4.8.15). The leader at the postion Pc = (9.8,1)

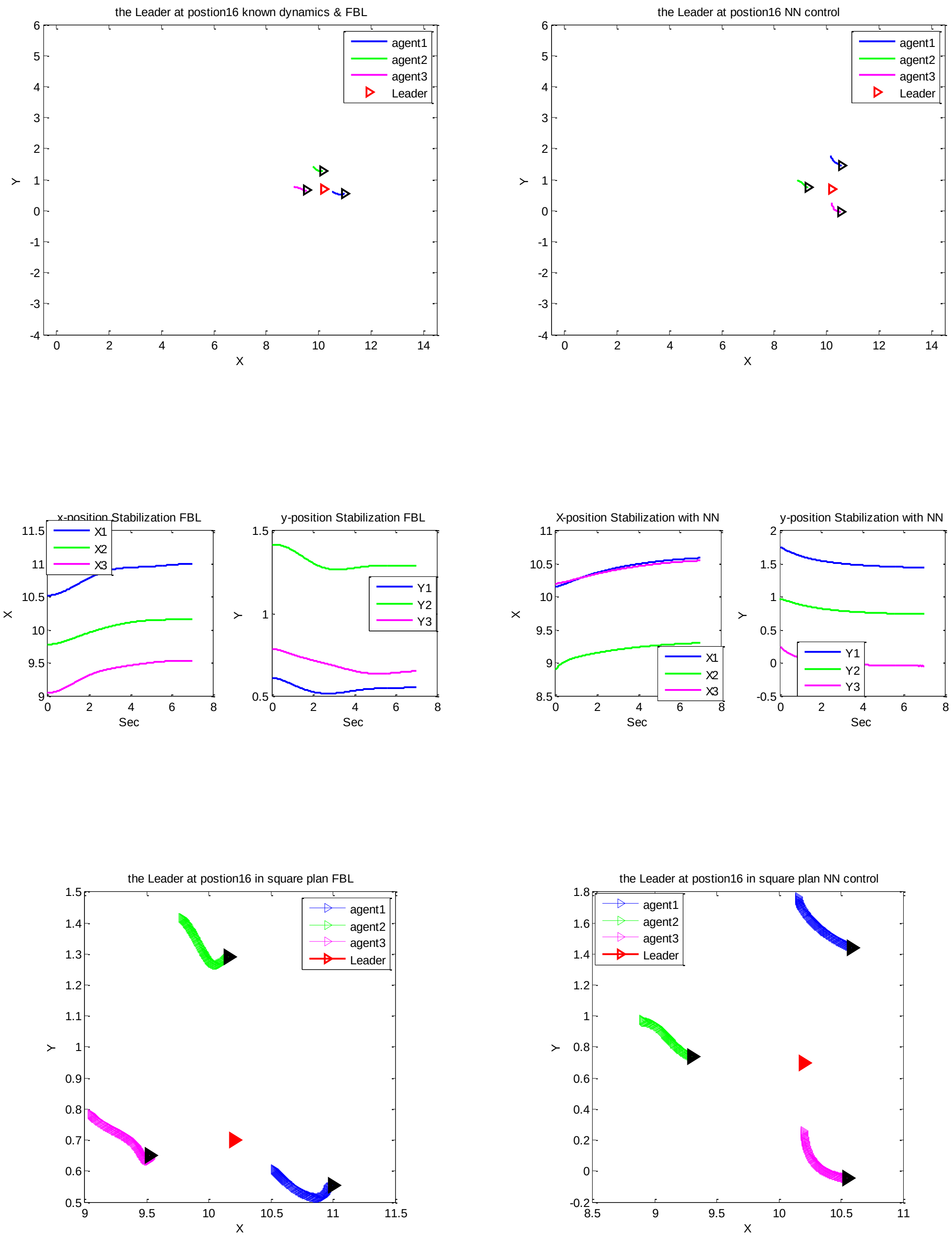


Figure (4.8.16). The leader at the postion Pc = (10.2,0.7)

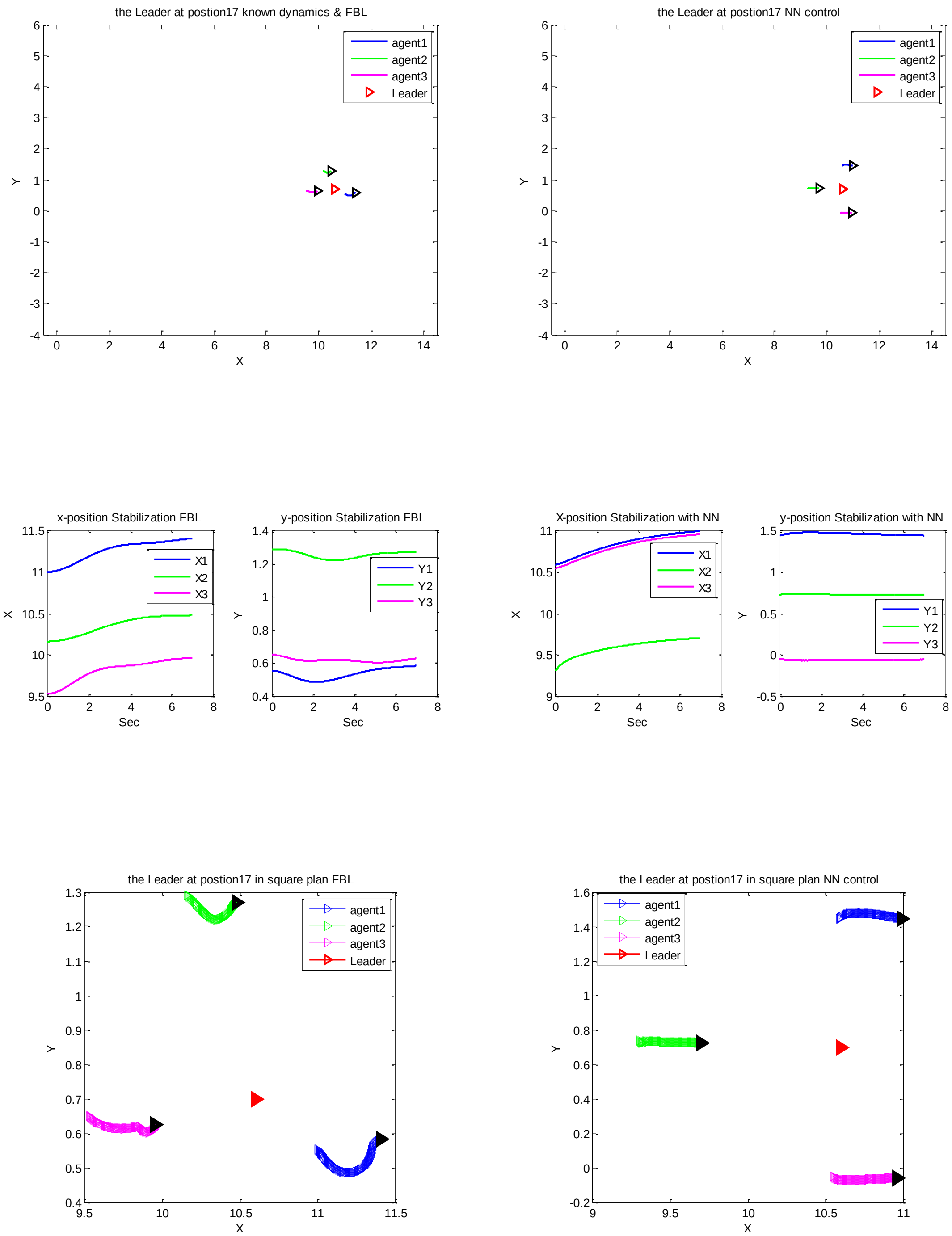


Figure (4.8.17). The leader at the postion Pc = (10.6,0.7)

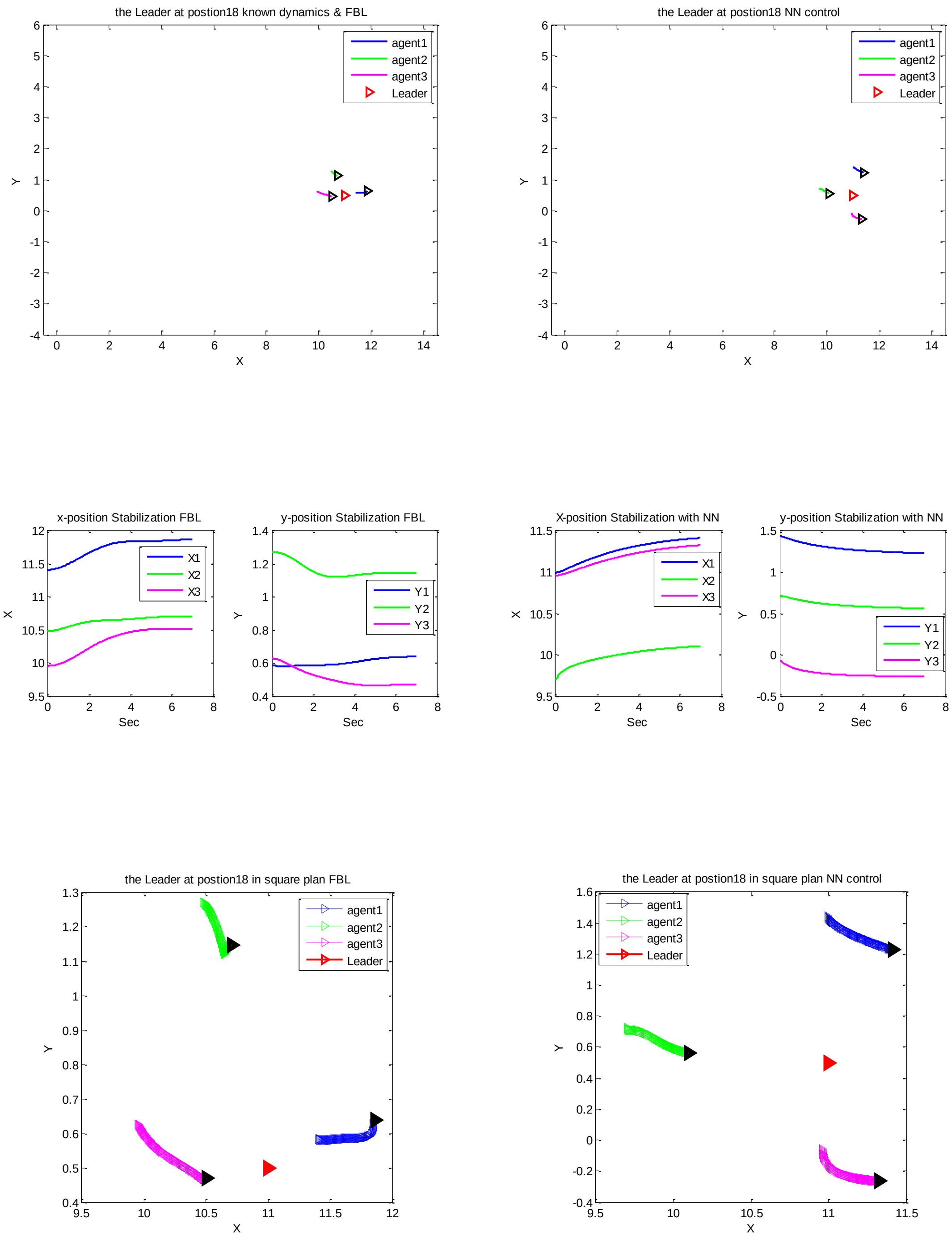


Figure (4.8.18). The leader at the postion Pc = (11,0.5)

Figure (4.8): A complete comparison study between the two designed controllers.

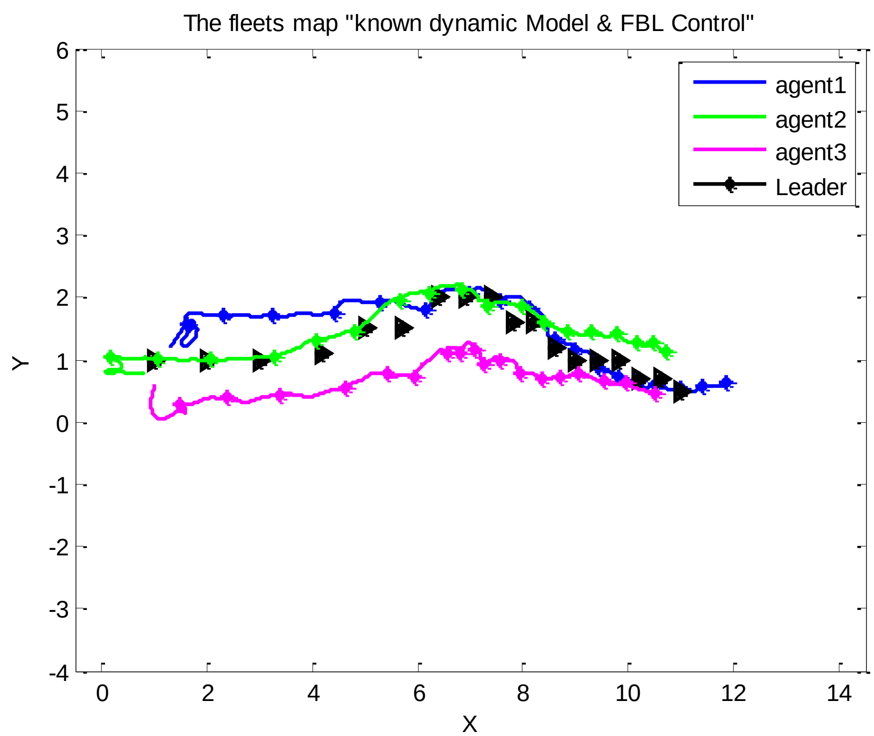


Figure (4.9). The map of fleet of three agents along its navigation.with their leader under the FBL control

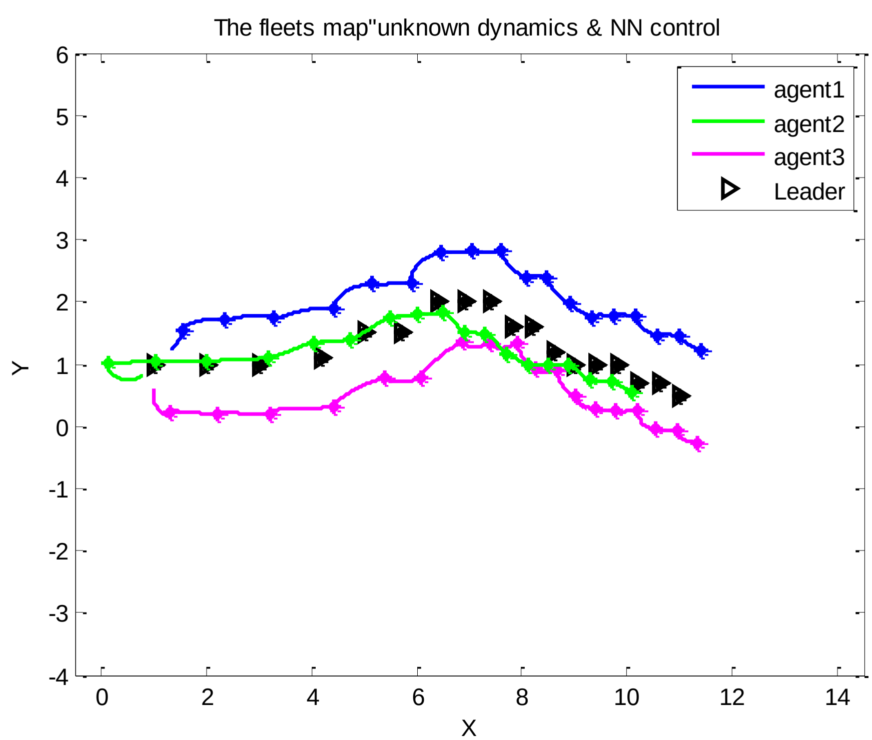


Figure (4.10). The map of fleet of three agents along its navigation with their leader under NN control.

4.8 A Comparison Study

In this thesis we implement two controllers, the first one is assuming a complete knowledge of robot dynamics. The second one is tracking error dynamics using neural network controller, which is used when we do not have knowledge about the robot dynamics. The simulation results of these two controllers are presented in the previous section.

4.8.1 Controller with Known Dynamics and Feedback Linearization.

In this controller a feedback linearization is used to cancel the inner loop nonlinearities of the robot and the potential field is used to control the robots formation. The system response of the implemented controller is shown in figure (4.6) , figure (4.6.a) in the case of three agents, figure (4.6.b) for four agents and figure (4.6.c) for five agents. In the figures on the left handside in figure (4.8) where the figure in the top of each page presents the postions of the leader and its followers in the whole navigation map. The figure in the middle of each page presents how the designed controller can stabilize (x, y) positions. The complete navigations map along the desired path is presented in figure (4.9). It is appearing that the formation response of the system under this kind of control is good but it still needs for appropriate controller to give a perfect response.

4.8.2 Controller with Unknown Dynamics and Neural Network

In this controller the potential field is used to control the robots formation and the neural network is used to estimate the system dynamics and tracking the inner loop errors. The neural networks also deals with unmodeled disturbance. The system response of the implemented controller is shown in figure (4.7) , figure (4.7.a) in the case of three agents, figure (4.7.b) for four agents and figure (4.7.c) for five agents. In the figures on the right handside in figure (4.8) where the figure in the top of each page presents the postions of the leader and its followers in the whole navigation map. The figure in the middle of each page presents how the designed controller can stabilize (x, y) positions. Also the complete navigations map along the group desired path is shown in figure (4.10). It is appearing that if it is compared with the previous one, the formation response of the system under this kind of control is improved and give us the desired formation.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 Conclusion

In this thesis, a framework for controlling leader-follower nonholonomic robots group's formations is presented. In the case where all robot dynamics are known, the leader-follower group formation and navigation controller is developed based on artificial potential field formation strategy and SLAM navigation. The attractive and repulsive potential fields are used to control nonholonomic robots' positions and hold them to their desired paths with respect to their leader. The group's leader generates its navigation map using the simultanous localization and mapping (SLAM) approach. Also the designed framework can not only be used for a group of three robots, but can also be used effectively for formation control for any n nonholonomic robots. The formation stability has been presented based on Lyapunov analysis.

In the case of unknown robot dynamics, the on-line NN weight tuning algorithms are used to guarantee tracking error dynamics and to follow the desired position in the following path. The designed controllers are verified and supported by numerical Matlab simulations.

5.2 Thesis Contributions

This work contributes to the literature on many fronts:

1) The new framework extends the work in Lorenzo et al. [12], from point mass holonomic agents to nonholonomic known dynamics vehicles.
2) This framework allows one of the nonholonomic robots to lead the others in an unknown environment, assuming that the leader configuration is used as the group field center. At the same time, all the agents in the fleet will keep their formation shape based on the potential fields.
3) The cost function $\tilde{V}$ equation (3.42) is used for analyzing the formation Lyapnov stability.
4) The potential field is developed to obtain many shape (star, square, and pentagon, etc.) formations for a group of nonholonomic mobile robots.
5) The feedback linearization control is used to cancel the nonlinearities of the nonholonomic vehicles.
6) In the case of unknown robot dynamics, the controller is developed using on-line NN weight tuning algorithms which guarantee path following of the follower for its desired trajectory.

5.3 Future Work

1) The designed system presented in this thesis only considers the nonholonomic robots leader-follower formation and navigation in 2-D space. This system would

be extended to 3-D space. In this case, the control will be more complicated because the size of the degree of freedom will be increased.

2) The obstacle avoidance will be addressed in future to give a good contribution for the designed framework. It will be treated as part of the repulsive forces if the postion of the obastacle is known. On the opposite case sensors for range and bearing can be used to determaine of the obstacle needed in the repulsive forces.

3) Future work can also consider moving landmarks for the leader navigation. For instance submarines do not have known landmarks and SLAM as it is described here can not be used for the leader. An another example when the robots share the same environmentwith otherteams of moving robots. For instance unmanned ground vehicle (UGV) in the urban traffic.

REFERENCES

[1] Jensfelt, P., and H.I. Christensen. “Laser Based Pose Tracking.” In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference On*, 4:2994–3000, 1999.

[2] Guivant, J., E. Nebot, and S. Baiker. “Localization and Map Building Using Laser Range Sensors in Outdoor Applications.” *Journal of Robotic Systems* 17, no. 10 (2000): 565–583.

[3] http://prism2.mem.drexel.edu/~billgreen/slam/

[4] Smith, R., M. Self, and P. Cheeseman. “A Stochastic Map for Uncertain Spatial Relationships.” In *Proceedings of the 4th International Symposium on Robotics Research*, 467–474, 1988.

[5] Smith, R.C., and P. Cheeseman. “On the Representation and Estimation of Spatial Uncertainty.” *The International Journal of Robotics Research* 5, no. 4 (1986): 56–68.

[6] Durrant-Whyte, H.F. “Uncertain Geometry in Robotics.” *Robotics and Automation, IEEE Journal Of* 4, no. 1 (1988): 23–31.

[7] N. Ayache and D. Faugeras, “Building, Registrating and Fusing Noisy Visual Maps” *The International Journal of Robotics Research* 7, 6 (1988) 45-65

[8] Crowley, J.L. “World Modeling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging.” In *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference On*, 674–680, 1989.

[9]. Durrant-Whyte, H., and T. Bailey. “Simultaneous Localisation and Mapping (SLAM): Part I the Essential Algorithms.” *Robotics and Automation Magazine* 13, no. 99 (2006): 80.

[10]. Bailey, T., and H. Durrant-Whyte. "Simultaneous Localization and Mapping (SLAM): Part II." *IEEE Robotics & Automation Magazine* 13, no. 3 (September 2006): 108–117.

[11] Wang, C.C., and C. Thorpe. "Simultaneous Localization and Mapping with Detection and Tracking of Moving Objects." In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference On*, 3:2918–2924, 2002.

[12] Rogers, JG, A.J.B. Trevor, C. Nieto-Granda, and H.I. Christensen. "SLAM with Expectation Maximization for Moveable Object Tracking." In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference On*, 2077–2082, 2010

[13] Wu, M., and J.Y. Sun. "Moving Object Detecting and Tracking with Mobile Robot Based on Extended Kalman Filter in Unknown Environment." In *Machine Vision and Human-Machine Interface (MVHI), 2010 International Conference On*, 64–67, 2010.

[14] Russell, R.A. "Robotic Location of Underground Chemical Sources." *Robotica* 22, no. 01 (2004): 109–115.

[15] Csorba, M. "Simultaneous Localisation and Map Building". University of Oxford, 1998.

[16] J.J. Leonard and H.J.S. Feder. "A computationally efficient method for large-scale concurrent mapping and localization". *Robotics Research*, 2000.

[17] Castellanos, J. A, J. M. Martnez, J. Neira, and J. D Tardos. "Experiments in Multisensor Mobile Robot Localization and Map Building." *Proc. 3rd IFAC Sym. Intell. Auton. Vehicles* (1998): 173–178.

[18] Castellanos, JA, JD Tardós, and G. Schmidt. "Building a Global Map of the Environment of a Mobile Robot: The Importance of Correlations." In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference On*, 2:1053–1059, 1997.

[19] Wolf, D.F., and G.S. Sukhatme. "Mobile Robot Simultaneous Localization and Mapping in Dynamic Environments." *Autonomous Robots* 19, no. 1 (2005): 53–65.

[20] Zhao, H., M. Chiba, R. Shibasaki, X. Shao, J. Cui, and H. Zha. "Slam in a Dynamic Large Outdoor Environment Using a Laser Scanner." In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference On*, 1455–1462, 2008.

[21] Y. Wang, Y. Feng, and D. Hung "Detection and Tracking of Moving Objects in SLAM Using Vision Sensors" (2011).

[22] Lu, F., and E. Milios. "Robot Pose Estimation in Unknown Environments by Matching 2d Range Scans." *Journal of Intelligent & Robotic Systems* 18, no. 3 (1997): 249–275.

[23] Besl, P.J., and N.D. McKay. "A Method for Registration of 3-D Shapes." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14, no. 2 (1992): 239–256.

[24] Chatila, R., and J. Laumond. "Position Referencing and Consistent World Modeling for Mobile Robots." In *Robotics and Automation. Proceedings. 1985 IEEE International Conference On*, 2:138–145, 1985.

[25] Esaka, Y., H. Enomoto, M. Hashimoto, and K. Takahashi. "Platooning Method for Multiple Mobile Robots Using Laser-based SLAM." In *SICE Annual Conference (SICE), 2011 Proceedings Of*, 251–254, 2011.

[26] Elfes, A. "Using Occupancy Grids for Mobile Robot Perception and Navigation." *Computer* 22, no. 6 (1989): 46–57.

[27] Dissanayake, M.W.M.G., P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba. "A Solution to the Simultaneous Localization and Map Building (SLAM) Problem." *Robotics and Automation, IEEE Transactions On* 17, no. 3 (June 2001): 229 –241.

[28] Miller, P.A., J.A. Farrell, Y. Zhao, and V. Djapic. "Autonomous Underwater Vehicle Navigation." *Oceanic Engineering, IEEE Journal Of* 35, no. 3 (2010): 663–678.

[29] P. Song and V. Kumar, "A Potential Feld Based Approach to Multi-Robot Manipulation*," in Proc. IEEE Intl. Conf. On Robotics and Automation,*Washington, DC, May 2002.

[30]. Chaimowicz, L., N. Michael, and V. Kumar. “Controlling Swarms of Robots Using Interpolated Implicit Functions.” In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference On*, 2487–2492, 2005.

[31]. Hsieh, M. A., and V. Kumar. “Pattern Generation with Multiple Robots.” In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference On*, 2442–2447, 2006.

[32]. Sabattini Lorenzo, Cristian Secchi, and Cesare Fantuzzi. “Arbitrarily Shaped Formations of Mobile Robots: Artificial Potential Fields and Coordinate Transformation.” *Autonomous Robots* 30, no. 4 (2011): 385–397.

[33] Huang, S., and G. Dissanayake. “Convergence and Consistency Analysis for Extended Kalman Filter Based SLAM.” *Robotics, IEEE Transactions On* 23, no. 5 (2007): 1036–1049.

[34] Frese, U. “A Proof for the Approximate Sparsity of SLAM Information Matrices.” In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference On*, 329–335, 2005.

[35] Fierro, R., and F. L. Lewis. “Control of Nonholonomic Mobile Robot: Backstepping Kinematics into Dynamics.” *Journal of Robotic Systems* 14, no. 3 (1997): 149–164.

[36] Lindegaard, K. P. “Acceleration Feedback in Dynamic Positioning.” *Department of Engineering Cybernetics, Norwegian University of Science and Technology* (2003).

[37] Fossen, T. I. Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles. Marine Cybernetics AS. Trondheim, Norway. (2002).

[39] Walter, M. R., R. M. Eustice, and J. J. Leonard. “Exactly Sparse Extended Information Filters for Feature-based SLAM.” *The International Journal of Robotics Research* 26, no. 4 (2007): 335–359.

[40] Wang, Z., S. Huang, and G. Dissanayake. "D-SLAM: Decoupled Localization and Mapping for Autonomous Robots." *Robotics Research* (2007): 203–213.

[41] Andrade-Cetto, J., and A. Sanfeliu. "The Effects of Partial Observability in SLAM." In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference On*, 1:397–402, 2004.

[42] Song, T. L. "Observability of Target Tracking with Bearings-only Measurements." *Aerospace and Electronic Systems, IEEE Transactions On* 32, no. 4 (1996): 1468–1472.

[43] Song, T. L. "Observability of Target Tracking with Range-only Measurements." *Oceanic Engineering, IEEE Journal Of* 24, no. 3 (1999): 383–387.

[44] Lee, K. W., W. S. Wijesoma, and I. G. Javier. "On the Observability and Observability Analysis of SLAM." In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference On*, 3569–3574, 2006.

[45] Sim, R., and N. Roy. "Global A-optimal Robot Exploration in Slam." In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference On*, 661–666, 2005.

[46] Makarenko, A. A., S. B. Williams, F. Bourgault, and H. F. Durrant-Whyte. "An Experiment in Integrated Exploration." In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference On*, 1:534–539, 2002.

[47] Stachniss, C., D. Hahnel, and W. Burgard. "Exploration with Active Loop-closing for FastSLAM." In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference On*, 2:1505–1510, 2004.

[48] Wijesoma, W. S., L. D. L. Perera, M. D. Adams, and S. Challa. "An Analysis of the Bias Correction Problem in Simultaneous Localization and Mapping." In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference On*, 747–752, 2005.

[49] Martinelli, A., N. Tomatis, and R. Siegwart. “Simultaneous Localization and Odometry Self Calibration for Mobile Robot.” *Autonomous Robots* 22, no. 1 (2007): 75–85.

[50] Perera, L. D. L., W. S. Wijesoma, and M. D. Adams. “The Estimation Theoretic Sensor Bias Correction Problem in Map Aided Localization.” *The International Journal of Robotics Research* 25, no. 7 (2006): 645–667.

[51] Desai, J. P., J. Ostrowski, and V. Kumar. “Controlling Formations of Multiple Mobile Robots.” In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference On,* 4:2864–2869, 1998.

[52] Dierks, T., and S. Jagannathan. “Control of Nonholonomic Mobile Robot Formations: Backstepping Kinematics into Dynamics.” In *Control Applications, 2007. CCA 2007. IEEE International Conference On*, 94–99, 2007.

[53] Dierks, T., and S. Jagannathan. “Control of Nonholonomic Mobile Robot Formations: Backstepping Kinematics into Dynamics.” In *Control Applications, 2007. CCA 2007. IEEE International Conference On*, 94–99, 2007.

[54] Fierro, R., A. K. Das, V. Kumar, and J. P. Ostrowski. “Hybrid Control of Formations of Robots.” In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference On*, 1:157–162, 2001.

[55] Kowdiki, K. H., R. K. Barai, and S. Bhattacharya. “Leader-follower Formation Control Using Artificial Potential Functions: A Kinematic Approach.” In *Advances in Engineering, Science and Management (ICAESM), 2012 International Conference On*, 500–505, 2012.

[56] Lewis, F. W., S. Jagannathan, and A. Yesildirak. *Neural Network Control Of Robot Manipulators And Non-Linear Systems*. 1st Ed. CRC Press, 1998.

[57] Lewis, F. L. “Neural Network Control of Robot Manipulators.” *IEEE Expert* 11, no. 3 (1996): 64–75.

Vita

Name : OMAR ALBURAIKI

Nationality : YEMENI

Date of Birth : 9/22/1979

Email : alburaiki.omer@gmail.com

Address : KFUPM main Campus Bld 808 Room 220

Academic Background : Received Bachelor's Degree with honor in Electronics and Communication Engineering from Hadramout University of Science and Technology, Mukalla, Yemen in 2007. Completed Master's degree requirements at King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia in December, 2012.