M.S. THESIS

Decentralized Formation Tracking Control of Multiple Homogeneous Agents

다 개체 동종 시스템의 분산 편대 추종 제어

BY

SANGHOON KIM

August 2008

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

공학석사 학위논문

Decentralized Formation Tracking Control of Multiple Homogeneous Agents

다 개체 동종 시스템의 분산 편대 추종 제어

2008년 8월

서울대학교 대학원

전기컴퓨터 공학부

김 상 훈

Decentralized Formation Tracking Control of Multiple Homogeneous Agents

다 개체 동종 시스템의 분산 편대 추종 제어

지도교수 심 형 보

이 논문을 공학석사 학위논문으로 제출함

2008년 4월

서울대학교 대학원

전기컴퓨터 공학부

김 상 훈

김 상 훈의 공학석사 학위논문을 인준함

2008년 6월

위 원 장 이 범 희 (인)

부 위 원 장 심 형 보 (인)

위 원 서 진 헌 (인)

학위논문 원문제공 서비스에 대한 동의서

본인의 학위논문에 대하여 서울대학교가 아래와 같이 학위논문을 제공하는 것에 동의합니다.

1. 동의사항
   1. 본인의 논문을 보존이나 인터넷 등을 통한 온라인 서비스 목적으로 복제할 경우 저작물의 내용을 변경하지 않는 범위 내에서의 복제를 허용합니다.
   2. 본인의 논문을 디지털화하여 인터넷 등 정보통신망을 통한 논문의 일부 또는 전부의 복제, 배포 및 전송 시 무료로 제공하는 것에 동의합니다.
2. 개인(저작자)의 의무
   1. 본 논문의 저작권을 타인에게 양도하거나 또는 출판을 허락하는 등 동의 내용을 변경하고자 할 때는 소속대학(원)에 공개의 유보 또는 해지를 즉시 통보하겠습니다.
3. 서울대학교의 의무
   1. 서울대학교는 본 논문을 외부에 제공할 경우 저작권 보호장치(DRM)를 사용하여야 합니다.
   2. 서울대학교는 본 논문에 대한 공개의 유보나 해지 신청 시 즉시 처리해야 합니다.

제목 : Decentralized Formation Tracking Control of Multiple Homogeneous Agents

학 위 구 분 : 공학석사
학 과(부) : 전기컴퓨터공학부
학 번 : 2006-23150
연 락 처 : wizest@gmail.com
저 작 자 : 김 상 훈 (인)

제 출 일 : 2008 년 8 월 일

서울대학교총장 귀하

Abstract

Decentralized Formation Tracking Control of Multiple Homogeneous Agents

SANGHOON KIM

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Formation control of multiple agents has been studied by many researchers in recent years. Since lots of new applications have arisen, there is growing interest in this area for military systems, mobile sensor network, intelligent transportation systems and so on. Formation control has been exploited by a variety of methods while many of formation approaches utilize the graph theory such as Laplacian adjacency matrix to describe the interconnection of the agents. This representation also appears in consensus and synchronization problems with a similar analysis scheme. In this thesis, we focus on the formation tracking of multiple homogeneous nonlinear systems by a decentralized manner. For time-varying smooth formation references, the collection of the agents keeps the formation while the whole group can freely move anywhere regardless of the existence of leader of the group. One of the main goals is to achieve such a controller and to find the stability condition for formation tracking. And additionally, we discuss a permutation of the agents to get a benefit of the homogeneity of the collection.

Keywords: output tracking, formation, decentralized control, feedback linearization, permutation, multiple homogeneous systems
Student Number: 2006-23150

Contents

List of Figures

Notation and Symbols

$\mathbb{N}$	the set of all natural numbers
$\mathbb{R}$	the set of all real numbers
$\mathbb{R}^n$	the n-dimensional Euclidean space
$\mathbb{R}^{n\times m}$	the space of $m \times n$ matrices with real entries
$\mathbb{C}$	the set of all complex numbers
$\mathbb{C}^{n\times m}$	the space of $m \times n$ matrices with complex entries
$\triangleq$	defined as
$L_f h(x)$	the Lie derivative of h with respect to the vector field f, $(= \frac{\partial h}{\partial x} f(x))$
$L_{f_1} L_{f_2} h$	$L_{f_1}(L_{f_2} h)$
$L_f^i h$	$L_f L_f^{i-1} h$
$L_f^1 h$	$L_f h$
$L_f^0 h$	h
$y^{(i)}$	the i-th derivative of y with respect to time
$\therefore$	therefore
$\because$	because
$\forall$	for all
$\exists$	there exists
$s.t.$	such that
$\in$	belong to
$\subset$	subset of
$[1, N]$	$\{1, 2, 3, \cdots, N\}$
$\prod_{i\in A} f(i)$	product of the members $f(i)$
$\lvert A\rvert$	the cardinality of the set A
$B \setminus A$	the set of all elements which are members of B, but not of A,

	$(B-A)$
$\|x\|$	the norm of vector x
$\|A\|$	the norm of matrix A
$\|A\|_F$	the Frobenius norm of matrix A
$A \circ B$	the elementwise product of the matrices
$f : S_1 \rightarrow S_2$	a function f mapping a set S_1 into a set S_2
$f^{-1}(\cdot)$	the inverse of f
$diag(A_1, \cdots, A_n)$	a block diagonal matrix with diagonal blocks A_1 to A_n
I_n	the identity matrix of size n
$\mathbf{1}_n$	$[1, \cdots, 1]^T$
$P > 0$	a positive definite matrix P
$P \geq 0$	a positive semidefinite matrix P
A^T	the transpose of A
A^H	the Hermitian adjoint of A, $\bar{A}^T$
$A \bigotimes B$	the Kronecker product with A by B
$tr(A)$	the trace of A
$A = \{a_{ij}\}$	the matrix A in which its (i, j) element is a_{ij}
$\square$	the end of proof
[xx]	see reference number xx in the bibliography

Chapter 1

Introduction

Formation control of multiple agents has been studied by many researchers in recent years. Since lots of new applications have arisen, there is growing interest in this area. These applications include formation flight, cooperative classification, attack and rendezvous in military systems. In mobile sensor network, environmental sampling and distributed observing are good examples as the application of the formation control. Moreover, Intelligent transportation systems like automated highway system(AHS) and air traffic control have been investigated by many groups of the academy and the industry. The recent survey [20] presents a readable overview of those applications.

Formation control has been exploited by a variety of methods. In [29], decentralized overlapping scheme was proposed for a formation of unmanned aerial vehicles which uses the mathematical framework of the inclusion principle and the expansion of the dimensional space for interconnected systems. In [18], Linear matrix inequalities (LMI) and graph theory were combined to deal with formation flying control of multiple spacecraft. Coordination of groups using nearest neighbors was studied in [13] where each agent's heading is updated by a local rule based on the average of ones of its neighbors. In [28], cooperative control of steered agents was discussed in which the developed methodology stabilizes isolated relative equilibria in identical systems moving in the plane. And a tracking control for formation was researched in [34] through a sliding mode framework for a certain class of nonlinear systems. Moreover, a distributed control scheme was suggested for formation of multiple robots without supervisor in [33]. And behavior-based formation control was explained in

[4]. In addition, cooperation among a collection of vehicles using intervehicle communication was analyzed in [8, 15] which consider a formation problem as stabilization and provide a stability condition and a relation between the connectivity of agents and the stability.

Many of formation approaches utilize the graph theory such as Laplacian adjacency matrix to describe the interconnection of the agents. This representation also appears in consensus and synchronization problems. In [24, 23, 22], consensus under time varying interaction topologies was mentioned where dynamical agents with fixed and switching network topology reach an agreement regarding a certain quantity of interest. And synchronization of coupled systems was investigated in [32, 31, 3, 19].

In this thesis, we focus on a formation tracking of multiple homogeneous nonlinear systems by a decentralized manner. For time-varying smooth formation references, the collection of the agents keeps the formation while the whole group can freely move anywhere regardless of the existence of leader of the group. One of the main goals is to achieve such a controller and to find the stability condition for formation tracking. And additionally, we discuss a permutation of the agents to get a benefit of the homogeneity of the collection. The rest of this thesis is organized as follows.

In chapter 2, we discuss the output tracking control for nonlinear systems. State feedback input-output linearization is employed to deal with the nonlinearity for both single-input single-output (SISO) and multi-input multi-output (MIMO) systems. And a simple static feedback controller is proposed for tracking a reference input.

In chapter 3, we extend the tracking controller to multiple homogeneous systems. The first part of the chapter is briefly mentioned about tracking absolute references which is an easy case where the each agent of the group tracks individually its own reference. The second part of the chapter is devoted to tracking relative references. The output tracking problem of multiple homogeneous agents by relative references is to design a controller which drives the differences of the outputs among the agents so as to track given references as shown below.

$$\lim_{t\to\infty} \| (y_i(t) - y_j(t)) - y_{ijr}(t)\| = 0, \qquad 1 \le \forall i, j \le N \tag{1.1}$$

where N is the number of the agents, y_i, y_j are respectively the outputs of the agent i, j and y_{ijr} is the reference for the difference of the outputs between the agent i and j. To simplify complexity to express equations for the collection of multiple homogeneous agents, we introduce Kronecker product and graph theory which includes Laplacian adjacency matrix. The eigenvalues of Laplacian matrix play a key role for the stability of tracking references which is analyzed by Perron Frobenius theorem and the property of the null space of Laplacian matrix. In the last section of the chapter, the decentralized output tracking controller for relative references is developed where each agent cannot sense all outputs of the agents but ones of only its neighborhoods.

Chapter 4 is composed of two parts about the formation tracking problem and the permutation invariant formation tracking problem. Before we discuss the formation tracking control, the common concept of formations is described. And the formation is mathematically defined as the matrix and the vector about relative outputs of the agents in the group with several examples. The former formation tracking problem is to design a controller such that the relative output between the arbitrary two agents follows the formation reference, which is resolved by applying the output tracking by relative references in the earlier chapter. The latter permutation invariant formation tracking problem is to utilize the homogeneity of the agents. Because the agents have the identical dynamics and we assume that all agents have the same ability to achieve a certain mission, each agent can be replaced by others. For taking this advantage of homogeneity, a weighted graph matching problem is introduced with respect to the nearness in the sense of the included angle matching. And we finally propose a formation tracking controller combined with the assignment dynamics by which each agent goes through the nearest path to be in the formation.

In chapter 5, 6 agents of multiple homogeneous nonholonomic mobile robots are considered as an example. And we verify the proposed controller well designed by simulating the collection of the mobile robots.

Chapter 2

Output Tracking of Nonlinear System

In this chapter, we discuss the output tracking control for nonlinear systems. State feedback input-output linearization is employed to deal with the nonlinearity for both single-input single-output and multi-input multi-output systems. And a simple static feedback controller is proposed for tracking a reference input.

2.1 Single-Input Single-Output Case (SISO)

Definition 2.1 (Output Tracking Problem (SISO))**.** Consider the following single input single output systems with no uncertainties

$$\begin{aligned} \dot{x} &= f(x) + g(x)u, \qquad && x \in \mathbb{R}^n \\ y &= h(x), && y \in \mathbb{R} \end{aligned} \tag{2.1}$$

with smooth f and g vector fields over $\mathbb{R}^n$ and h: $\mathbb{R}^n \to \mathbb{R}$ a smooth function s.t. $h(0) = 0$.

The **output tracking problem** is to design a controller with the property that, for given a smooth bounded reference signal $y_r(t)$, the output of system satisfies

$$\lim_{t\to\infty} (y(t) - y_r(t)) = 0$$

for any initial condition of the closed loop system.

2.1.1 Input-Output Feedback Linearization

Definition 2.2 (Input-Output Feedback Linearization)**.** System (2.1) is locally (globally) **state feedback input-output linearizable** in a neighborhood of the origin U_0 (in $\mathbb{R}^n$) if there exists a state feedback

$$u = k(x) + \beta(x)v \tag{2.2}$$

with k and β smooth functions, $\beta(x) \neq 0, \forall x \in U_0 (\forall x \in \mathbb{R}^n)$, such that the input-output dynamics of the closed loop system

$$\begin{aligned} \dot{x} &= f(x) + g(x)(k(x) + \beta(x)v) \\ y &= h(x) \end{aligned} \tag{2.3}$$

are given in $U_0(\mathbb{R}^n)$ by

$$\frac{d^r y}{dt^r} = v$$

with $1 \leq r \leq n$.

Remark 2.3. The output y of the system (2.3) is chained to the input u with the ρ-th degree of integrators. In the aspect of input v to output y, the system (2.3) is linear.

Definition 2.4 (Relative Degree ρ)**.** The **(global) relative degree** ρ of system (2.1) is defined as the integer s.t.

$$\begin{aligned} L_g L_f^i h(x) &= 0, & &\forall 0 \leq i \leq \rho - 2 \\ L_g L_f^{\rho-1} h(x) &\neq 0, & &\forall x \in U_0 \quad (\forall x \in \mathbb{R}^n) \end{aligned} \tag{2.4}$$

where U_0 is a neighborhood of the origin.[1]

Remark 2.5. There might be a case where the relative degree cannot be defined.

[1] Lie derivative of h with respect to $f = L_f h(x) = \frac{\partial h}{\partial x} f(x), \quad L_{f_1} L_{f_2} h = L_{f_1}(L_{f_2} h), \quad L_f^i h = L_f L_f^{i-1} h, \quad L_f^1 h = L_f h, \quad L_f^0 h = h.$

Theorem 2.6 (State Feedback Linearized System)**.** *Assume that the relative degree ρ is (globally) well defined. Then system* (2.1) *is locally (globally) partially state feedback linearizable with index ρ and is locally (globally) state feedback input-output linearizable, i.e. locally (globally) feedback equivalent to:*

$$\begin{aligned} \dot{\xi} &= \phi(\xi, z), & \xi &\in \mathbb{R}^{n-\rho} \\ \dot{z}_i &= z_{i+1}, & 1 &\leq i \leq \rho - 1 \\ \dot{z}_\rho &= v \\ y &= z_1 \end{aligned} \tag{2.5}$$

if and only if $\rho \leq n$. Moreover, the feedback controller u is to be

$$u = \frac{-L_f^\rho h(x)}{L_g L_f^{\rho-1} h(x)} + \frac{1}{L_g L_f^{\rho-1} h(x)} v \triangleq k(x) + \beta(x) v \tag{2.6}$$

Proof. See the proof in [16]. □

Remark 2.7. System (2.1) is globally state feedback input-output linearizable if, and only if the vector fields

$$\tilde{f} \triangleq f + \frac{-L_f^\rho h(x)}{L_g L_f^{\rho-1} h(x)} g, \qquad \tilde{g} \triangleq \frac{1}{L_g L_f^{\rho-1} h(x)} g \tag{2.7}$$

are complete[2].

Definition 2.8 (Zero Dynamics)**.** Assume that $\rho \leq n$ in U_0 for system (2.1). Let $z_i = L_f^{i-1} h(x), 1 \leq i \leq \rho$. Define the $(n-\rho)$-dimensional manifold $M = \{x \in U_0 : h(x) = 0, \cdots, L_f^{\rho-1} h(x) = 0\}$. The dynamics of system (2.1) constrained in M are called the **zero dynamics**.

Remark 2.9. The zero dynamics of system (2.1) are given with equation (2.5) by

$$\dot{\xi} = \phi(\xi, 0), \qquad \xi \in \mathbb{R}^{n-\rho} \tag{2.8}$$

Definition 2.10 (Minimum Phase)**.** System (2.1) with $\rho \leq n$ is called **minimum**

[2] A vector field $f(x)$ is said to be **compelete** if the solutions to the differential equation $\dot{x} = f(x)$ may be defined for all $t \in \mathbb{R}$

phase if the origin $\xi = 0$ is an asymptotically stable equilibrium point for the zero dynamics. A system which is not minimum phase is said to be **non-minimum phase**.

Remark 2.11. If the relative degree ρ is equal to the dimension n of x, then there are no ξ terms in (2.5). (i.e. no zero dynamics), in which the set of functions $z_i = L_f^{i-1}h(x) \quad (1 \le i \le n)$ completely define a coordinates transformation.

2.1.2 Output Tracking

Definition 2.12 (Tracking Dynamics)**.** Assume that it is $\rho \le n$ in U_0 for system (2.1) and there exists an initial condition $x_0 \in U_0$ which is compatible[3] with the reference signal $y_r(t)$. Let

$$M_t = \{x \in U_0 : h(x) = y_r(t), \cdots, L_f^{\rho-1}h(x) = y_r^{(\rho-1)}(t)\} \tag{2.9}$$

be the time-varying $(n-\rho)$ -dimensional integral manifold called the tracking manifold. The dynamics of system (2.1) which are subjected to constraints M_t are said to be the **tracking dynamics**.

Theorem 2.13 (Tracking Control [16])**.** *For system* (2.1)*, assume that the global relative degree is well defined with $\rho \le n$, the vector fields* (2.7) *are complete, and the tracking dynamics are bounded input bounded state stable. If*

$$s^\rho + k_\rho s^{\rho-1} + \cdots + k_1 \tag{2.10}$$

is a Hurwitz polynomial[4]*, then the tracking problem is globally solvable by static state feedback controller u in* (2.6) *with*

$$v = -k_1(z_1 - y_r(t)) - \cdots - k_\rho(z_\rho - y_r^{(\rho-1)}(t)) + y_r^{(\rho)}(t) \tag{2.11}$$

Proof. If system (2.1) has a well defined relative degree ρ, then it satisfies the conditions of Theorem 2.6. thus, the control (2.6) is globally well defined. In that case,

[3]The initial condition $x(0) = x_0 \in U_0$ is said to be **compatible** with the reference signal $y_r(t)$ for system (2.1) if $y_r^{(i)}(0) = L_f^i h(x_0), 0 \le i \le \rho - 1$

[4]A polynomial is called a **Hurwitz polynomial** if all its roots have negative real parts.

one takes v as the equation (2.11), we obtain

$$\begin{bmatrix} \dot{e_1} \\ \vdots \\ \dot{e_{\rho-1}} \\ \dot{e_\rho} \end{bmatrix} = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ -k_1 & -k_2 & \cdots & -k_\rho \end{bmatrix} \begin{bmatrix} e_1 \\ \vdots \\ e_{\rho-1} \\ e_\rho \end{bmatrix} \tag{2.12}$$

with $e_i \triangleq z_i - y_r^{(i-1)} = y^{(i-1)} - y_r^{(i-1)}$. In the assumption, we have the Hurwitz polynomial (2.10), finally asymptotic tracking is achieved for any initial error $e(0)$. Since $y_r, \cdots, y_r^\rho$ are bounded, it follows that $z_1, \cdots, z_\rho$ are bounded. Because of the property of the bounded input bounded state stability of the tracking dynamics, ξ from (2.5) is bounded as well, $z_1(t), \cdots, z_\rho(t)$ as bounded references. $\square$

2.2 Multi-Input Multi-Output Case (MIMO)

Definition 2.14 (Output Tracking Problem (MIMO))**.** Consider the following multi-input multi-output systems with no uncertainties

$$\begin{aligned} \dot{x} &= f(x) + g(x)u, & x \in \mathbb{R}^n \\ y &= h(x), & y \in \mathbb{R}^m \end{aligned} \tag{2.13}$$

where

$$\begin{aligned} u &= [u_1, \cdots, u_m]^T, & u \in \mathbb{R}^m \\ g(x) &= \begin{bmatrix} | & & | \\ g_1(x), & \cdots, & g_m(x) \\ | & & | \end{bmatrix}, & g(\cdot) \in \mathbb{R}^{n\times m} \\ h(x) &= [h_1(x), \cdots, h_m(x)]^T, & h(\cdot) \in \mathbb{R}^m \end{aligned}$$

with smooth f and $g_1, \cdots g_m$ vector fields over $\mathbb{R}^n$ and h_i: $\mathbb{R}^n \to \mathbb{R}$ a smooth function s.t. $h_i(0) = 0 \quad (0 \le i \le m)$.

The **output tracking problem** is to design a controller with the property that, for a given smooth bounded reference signal $y_r(t) \in \mathbb{R}^m$, the output of system satisfies

$$\lim_{t \to \infty} \|y(t) - y_r(t)\| = 0$$

for any initial condition of the closed loop system.

2.2.1 Input-Output Feedback Linearization

Definition 2.15 (Input-Output Feedback Linearization)**.** System (2.13) is locally (globally) **state feedback input-output linearizable** in a neighborhood of the origin U_0 (in $\mathbb{R}^n$) if there exists a state feedback

$$u = k(x) + \beta(x)v \tag{2.14}$$

where $v = [v_1, \cdots, v_m]^T \in \mathbb{R}^m$ with a smooth vector field $k(x)$ and smooth non-singular $\beta(x) \in \mathbb{R}^{m\times m}$ matrix for $\forall x \in U_0 (\forall x \in \mathbb{R}^n)$, such that the input-output dynamics of the closed loop system

$$\begin{aligned} \dot{x} &= f(x) + g(x)(k(x) + \beta(x)v) \\ y &= h(x) \end{aligned} \tag{2.15}$$

are given in $U_0(\mathbb{R}^n)$ by

$$\frac{d^{r_i} y_i}{dt^{r_i}} = v_i$$

with $1 \le r_1 + \cdots + r_m \le n$ for $1 \le \forall i \le m$.

Definition 2.16 (Vector Relative Degree ρ)**.** The **(global) vector relative degree** ρ of system (2.13) is defined as the vector $\rho = [\rho_1, \cdots, \rho_m]^T \in \mathbb{N}^m$ s.t.

$$\begin{aligned} &L_{g_j} L_f^k h_i(x) = 0, \\ &\mathcal{A}(x) = \begin{bmatrix} L_{g_1} L_f^{\rho_1 - 1} h_1(x) & \cdots & L_{g_m} L_f^{\rho_1 - 1} h_1(x) \\ L_{g_1} L_f^{\rho_2 - 1} h_2(x) & \cdots & L_{g_m} L_f^{\rho_2 - 1} h_2(x) \\ \vdots & \ddots & \vdots \\ L_{g_1} L_f^{\rho_m - 1} h_m(x) & \cdots & L_{g_m} L_f^{\rho_m - 1} h_m(x) \end{bmatrix}, \end{aligned} \tag{2.16}$$

for $1 \le \forall j \le m, \quad 0 \le \forall k \le \rho_i - 2, \quad 1 \le \forall i \le m, \quad \forall x \in U_0 \, (\forall x \in \mathbb{R}^n)$
where U_0 is a neighborhood of the origin and
$\mathcal{A}(\cdot) \in \mathbb{R}^{m\times m}$ is nonsingular at $x \in U_0$ (nonsingular at $x \in \mathbb{R}^n$).

Theorem 2.17 (State Feedback Linearized System). *Assume that the vector relative degree ρ is (globally) well defined. Then system (2.13) is locally (globally) partially state feedback linearizable with indices $\{\rho_1, \cdots, \rho_m\}$ and is locally (globally) state feedback input-output linearizable, i.e. locally (globally) feedback equivalent to:*

$$
\begin{aligned}
\dot{\xi} &= \phi(\xi, z, v), & & \xi \in \mathbb{R}^{n-\rho^*} \\
\dot{z_{1i_1}} &= z_{1i_1+1}, \quad \dot{z_{1\rho_1}} = v_1, & & 1 \le i_1 \le \rho_1 - 1 \\
\dot{z_{2i_2}} &= z_{2i_2+1}, \quad \dot{z_{2\rho_2}} = v_2, & & 1 \le i_2 \le \rho_2 - 1 \\
&\vdots \\
\dot{z_{mi_m}} &= z_{mi_m+1}, \quad \dot{z_{m\rho_m}} = v_m, & & 1 \le i_m \le \rho_m - 1 \\
y &= [z_{11}, z_{21}, \cdots, z_{m1}]^T
\end{aligned}
\tag{2.17}
$$

if and only if $\rho^* \triangleq \rho_1 + \rho_2 + \cdots + \rho_m \le n$. Moreover, the feedback controller u is to be

$$
\begin{aligned}
u &= -\mathcal{A}^{-1}(x) \begin{bmatrix} L_f^{\rho_1} h_1 \\ L_f^{\rho_2} h_2 \\ \vdots \\ L_f^{\rho_m} h_m \end{bmatrix} + \mathcal{A}^{-1}(x) v \\
&\triangleq k(x) + \beta(x) v, \qquad k(\cdot) \in \mathbb{R}^m, \quad \beta(\cdot) \in \mathbb{R}^{m \times m}
\end{aligned}
\tag{2.18}
$$

where $v = [v_1, \cdots, v_m]^T$.

Proof. See the proof in [14, 12, 16]. □

Remark 2.18. Equation (2.17) has the Brunovsky canonical form. i.e.

$$
\begin{aligned}
\dot{z} &= Az + Bv \\
y &= Cz
\end{aligned}
\tag{2.19}
$$

where

$$
\begin{aligned}
&z = [z_{11}, z_{12}, \cdots, z_{1\rho_1}, \cdots\cdots, z_{m1}, z_{m2}, \cdots, z_{m\rho_m}]^T \\
&A = diag(A_1, \cdots, A_m) \in \mathbb{R}^{\rho^* \times \rho^*}, \qquad B = diag(B_1, \cdots, B_m) \in \mathbb{R}^{\rho^* \times m}, \\
&C = diag(C_1, \cdots, C_m) \in \mathbb{R}^{m \times \rho^*},
\end{aligned}
\tag{2.20}
$$

$$A_i = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{\rho_i \times \rho_i}, \quad B_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \in \mathbb{R}^{\rho_i},$$
$$C_i = [1, 0, \cdots, 0] \in \mathbb{R}^{1 \times \rho_i}$$
$$\text{for} \quad 1 \leq i \leq m$$

Remark 2.19. System (2.13) is globally state feedback input-output linearizable if, and only if the vector fields

$$\tilde{f} \triangleq f + gk(x), \qquad \tilde{g_i} \triangleq g\beta_i(x), \qquad 1 \leq i \leq m \tag{2.21}$$

are complete where $\beta_i(x)$ is the i-th column vector of $\beta(x)$.

Definition 2.20 (Zero Dynamics)**.** Assume that $\rho^* \leq n$ in U_0 for system (2.13). Let $z_{1i_1} = L_f^{i_1-1}h_1(x), \cdots\cdots, z_{mi_m} = L_f^{i_m-1}h_m(x)$, $1 \leq i_j \leq \rho_j$. Define the $(n-\rho^*)$-dimensional manifold $M = \{x \in U_0 : h_1(x) = 0, \cdots, L_f^{\rho_1-1}h_1(x) = 0, \cdots, h_m(x) = 0, \cdots, L_f^{\rho_m-1}h_m(x) = 0\}$. The dynamics of system (2.13) constrained in M are called the **zero dynamics**.

Remark 2.21. The zero dynamics of system (2.13) are given with equation (2.17) by

$$\dot{\xi} = \phi(\xi, 0, 0), \qquad \xi \in \mathbb{R}^{n-\rho^*} \tag{2.22}$$

Definition 2.22 (Minimum Phase)**.** System (2.13) with $\rho^* \leq n$ is called **minimum phase** if the origin $\xi = 0$ is an asymptotically stable equilibrium point for the zero dynamics. A system which is not minimum phase is said to be **non-minimum phase**.

2.2.2 Output Tracking

Definition 2.23 (Tracking Dynamics)**.** Assume that $\rho^* \leq n$ in U_0 for system (2.13) and that there exists an initial condition $x_0 \in U_0$ which is compatible[5] with the

[5] The initial condition $x(0) = x_0 \in U_0$ is said to be **compatible** with the reference signals $y_{1r}(t), \cdots, y_{mr}(t)$ for system (2.13) if $y_{1r}^{(i_1)}(0) = L_f^{i_1}h_1(x_0), \cdots, y_{mr}^{(i_m)}(0) = L_f^{i_m}h_m(x_0) \quad 0 \leq i_j \leq \rho_j - 1$

reference signals $y_{1r}(t), \cdots, y_{mr}(t)$. Let

$$\begin{aligned} M_t = \{x \in U_0 : & h_1(x) = y_{1r}(t), \cdots, L_f^{\rho_1 - 1} h_1(x) = y_{1r}^{(\rho_1 - 1)}(t), \cdots\cdots, \\ & h_m(x) = y_{mr}(t), \cdots, L_f^{\rho_m - 1} h_m(x) = y_{mr}^{(\rho_m - 1)}(t)\} \end{aligned} \tag{2.23}$$

be the time-varying $(n - \rho^*)$ -dimensional integral manifold called the tracking manifold. The dynamics of system (2.13) which are subjected to constraints M_t are said to be the **tracking dynamics**.

Theorem 2.24 (Tracking Control). *For the system* (2.13)*, assume that the global vector relative degree ρ is well defined with $\rho^* \leq n$, the vector fields* (2.21) *are complete, and the tracking dynamics are bounded input bounded state stable. If*

$$\begin{gathered} s^{\rho_1} + k_{1\rho_1} s^{\rho_1 - 1} + \cdots + k_{11}, \\ \vdots \\ s^{\rho_m} + k_{m\rho_m} s^{\rho_m - 1} + \cdots + k_{m1} \end{gathered} \tag{2.24}$$

are Hurwitz polynomials, then the tracking problem is globally solvable by static state feedback controller u in (2.18) *with*

$$\begin{aligned} v_1 &= -k_{11}(z_{11} - y_{1r}(t)) - \cdots - k_{1\rho_1}(z_{1\rho_1} - y_{1r}^{(\rho_1 - 1)}(t)) + y_{1r}^{(\rho_1)}(t) \\ &\vdots \\ v_m &= -k_{m1}(z_{m1} - y_{mr}(t)) - \cdots - k_{m\rho_m}(z_{m\rho_m} - y_{mr}^{(\rho_m - 1)}(t)) + y_{mr}^{(\rho_m)}(t) \end{aligned} \tag{2.25}$$

Proof. If system (2.13) has a well defined vector relative degree $\rho = [\rho_1, \cdots, \rho_m]^T$, then it satisfies the conditions of Theorem 2.17. Thus, the control (2.18) is globally well defined. In that case, one takes v as the equation (2.25), then we obtain

$$\begin{bmatrix} \dot{e_{i1}} \\ \vdots \\ \dot{e_{i\rho_{i-1}}} \\ \dot{e_{i\rho_i}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ -k_{i1} & -k_{i2} & \cdots & -k_{i\rho_i} \end{bmatrix} \begin{bmatrix} e_{i1} \\ \vdots \\ e_{i\rho_{i-1}} \\ e_{i\rho_i} \end{bmatrix}, \quad 1 \leq \forall i \leq m \tag{2.26}$$

with $e_{ij} \triangleq z_{ij} - y_{ir}^{(j-1)} = y_i^{(j-1)} - y_{ir}^{(j-1)}$. In the assumptions, we have Hurwitz polynomials (2.24), asymptotic tracking is finally achieved for any initial error $e(0)$.

Since $y_{ir}, \cdots, y_{ir}^{\rho_i}$ are bounded, it follows that $z_{i1}, \cdots, z_{i\rho_i}$ are bounded. Because of the property of the bounded input bounded state stability of the tracking dynamics, ξ from (2.17) is bounded as well, $z_{i1}(t), \cdots, z_{i\rho_i}(t)$ as bounded references. $\square$

Chapter 3

Output Tracking of Multiple Homogeneous Agents

In this chapter, we extend the tracking controller discussed in the earlier chapter to multiple homogeneous systems. The first part of the chapter is briefly mentioned about the output tracking by absolute references where the each agent of the group tracks individually its own reference. And the second part of the chapter is devoted to the output tracking by relative references which is utilized for the formation tracking control in the next chapter.

Consider the following N systems of multiple homogeneous[1] agents.

$$\begin{aligned} \dot{x_1} &= f(x_1) + g(x_1)u_1, & y_1 &= h(x_1), \\ \dot{x_2} &= f(x_2) + g(x_2)u_2, & y_2 &= h(x_2), \\ &\vdots \\ \dot{x_N} &= f(x_N) + g(x_N)u_N, & y_N &= h(x_N), \end{aligned} \tag{3.1}$$

[1] They have an identical dynamics.

where

$$
\begin{aligned}
u_i &= [u_{i1}, \cdots, u_{im}]^T, & u_i \in \mathbb{R}^m \\
g(x_i) &= \begin{bmatrix} | & & | \\ g_1(x_i), & \cdots, & g_m(x_i) \\ | & & | \end{bmatrix}, & g(\cdot) \in \mathbb{R}^{n\times m} \\
h(x_i) &= [h_1(x_i), \cdots, h_m(x_i)]^T, & h(\cdot) \in \mathbb{R}^m
\end{aligned}
$$

with $x_i \in \mathbb{R}^n$, smooth f and $g_1, \cdots, g_m$ vector fields over $\mathbb{R}^n$ and h_j: $\mathbb{R}^n \to \mathbb{R}$ a smooth function s.t. $h_j(0) = 0$ for $1 \leq \forall i \leq N$ and $0 \leq \forall j \leq m$.

3.1 Output Tracking by Absolute References

At first, we consider the case that every references are given by absolute values. In such cases, straightforwardly, one can get the controller which lets the outputs of the corresponding agent track its own reference signals by the method in which an individual agent independently tracks its reference.

Definition 3.1. The **output tracking problem of multiple homogeneous agents by absolute references** is to design a controller with the property that, for given a smooth bounded reference signal $y_{ir}(t) \in \mathbb{R}^m$, the outputs of systems (3.1) satisfies

$$\lim_{t\to\infty} \|y_i(t) - y_{ir}(t)\| = 0, \qquad 1 \ \leq \forall i \leq N \tag{3.2}$$

for any initial condition of the closed loop system.

Theorem 3.2. *For each agent system of multiple homogeneous agent systems* (3.1)*, assume that the global vector relative degree ρ is well defined with $\rho^* \leq n$, and the tracking dynamics are bounded input bounded state stable. If the references for systems* (3.1) *are given by absolute values, then the tracking problem is globally solvable.*

Proof. For each agent system in systems (3.1), choose k_{ij} such as (2.24). By Theorem 2.24, we can independently solve the tracking problem for each agent. Thus, the case of multiple agents is also solved. □

Remark 3.3. For a tracking dynamics, bounded input bounded state stabilty can be replaced with the term of the input-to-state stability (ISS).

3.2 Output Tracking by Relative References

Now, we consider the references that are given by relative values.

Definition 3.4. The **output tracking problem of multiple homogeneous agents by relative references** is to design a controller with the property that, for given a smooth bounded relative reference signal $y_{(ij)r}(t) \in \mathbb{R}^m$, the outputs of systems (3.1) satisfies

$$\lim_{t\to\infty} \|(y_i(t) - y_j(t)) - y_{(ij)r}(t)\| = 0, \qquad 1 \ \leq \forall i, j \leq N \tag{3.3}$$

for any initial condition of the closed loop system.

For multiple systems (3.1), if the references are given by relative values, bearing in mind of relativeness, one can redefine each output of the agent system in (3.1) as follows,

$$y_i^* = \sum_{j\in\mathcal{N}_i} w_{ij}(y_i - y_j), \qquad 1 \leq \forall i \leq N \tag{3.4}$$

where $\mathcal{N}_i \subset [1, N] \setminus \{i\}$ is the index set that represents the set of agents which the agent i can sense, and w_{ij} is a scalar for weighting.

For each system of multiple systems (3.1), if its own output of the agent can be absolutely measurable by itself[2] and the vector relative degree is well defined, (i.e. for i-th agent, the well defined vector relative degree with respect to u_i and y_i), each agent system can be state feedback input output linearizable.

Using the new outputs of each agents and the input-output linearizability, the whole system dynamics are represented as follows

Theorem 3.5. *For the collection of multiple homogeneous agents systems* (3.1)*, assume that the global vector relative degree ρ is well defined with $\rho^* \leq n$, the vector fields* (2.21) *are complete, and the references for systems* (3.1) *are given by relative*

[2]Someone in a moving elevator could not get one's absolute position by oneself without an additional information from the elevator.

values. If the output of each agent system can be absolutely measurable by itself, then systems (3.1) *can be represented with new outputs such that* (3.4) *as follows.*

$$
\begin{aligned}
\dot{\xi}_i &= \phi(\xi_i, z_i, v_i), \\
\dot{z}_i &= A z_i + B v_i, \\
y_i^* &= \sum_{j \in \mathcal{N}_i} w_{ij}(C z_i - C z_j), \qquad 1 \le \forall i \le N
\end{aligned}
\tag{3.5}
$$

where A, B, C *are defined as* (2.20)*.*

Proof. By the assumption, one can apply Theorem 2.17 to each agent of systems (3.1), and replace each output with new output (3.4). □

Before we deal with more detailed parts of tracking of relative references for multiple agents, we introduce convenient tools to analyze a stability and express equations.

3.3 Kronecker Product

The Kronecker product is useful to describe the collection of multiple homogeneous systems. It makes the whole dynamics collection be able to be one single simple form like examples at the end of the paragraph.

Definition 3.6 ($A \otimes B$)**.** If A is an m by n matrix and B is a p by q matrix, then the **Kronecker product** $A \otimes B$ is the mp by nq block matrix

$$
A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}
\tag{3.6}
$$

with properties as follows

$$\begin{aligned} A \otimes (B + C) &= A \otimes B + A \otimes C \\ (A + B) \otimes C &= A \otimes C + B \otimes C \\ (kA) \otimes B &= A \otimes (kB) = k(A \otimes B) \\ (A \otimes B) \otimes C &= A \otimes (B \otimes C) \\ (A \otimes B)(C \otimes D) &= AC \otimes BD \\ (A \otimes B)^{-1} &= A^{-1} \otimes B^{-1} \\ (A \otimes B)^T &= A^T \otimes B^T \end{aligned} \tag{3.7}$$

Example 1.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \otimes \begin{bmatrix} 0 & 5 \\ 6 & 7 \end{bmatrix} = \begin{bmatrix} 1 \cdot 0 & 1 \cdot 5 & 2 \cdot 0 & 2 \cdot 5 \\ 1 \cdot 6 & 1 \cdot 7 & 2 \cdot 6 & 2 \cdot 7 \\ 3 \cdot 0 & 3 \cdot 5 & 4 \cdot 0 & 4 \cdot 5 \\ 3 \cdot 6 & 3 \cdot 7 & 4 \cdot 6 & 4 \cdot 7 \end{bmatrix} \tag{3.8}$$

Example 2 ($I_N \otimes A$)**.** *Consider the agent system s.t. $\dot{x}_i = Ax_i$ where $A \in \mathbb{R}^{n \times n}$. The collection of N systems of multiple homogeneous agents is that*

$$\dot{x} = (I_N \otimes A)x, \qquad x = [x_1, x_2, \cdots, x_N]^T$$

Example 3 ($K \otimes I_n$)**.** *Consider the agent system s.t. $\dot{x}_i = Ax_i + u_i$ where $A \in \mathbb{R}^{n \times n}$. Output manipulation among the N systems is represented by $K \in \mathbb{R}^{N \times N}$ as follows.*

$$\begin{aligned} \dot{x} &= (I_N \otimes A)x + u, & x &= [x_1, x_2, \cdots, x_N]^T, \\ u &= (K \otimes I_n)x, & u &= [u_1, u_2, \cdots, u_N]^T \end{aligned}$$

3.4 Graph Theory

In this section, we study the graph theory. This is helpful to express relative outputs for tracking relative references, and useful to analyze the stability by the property of Graph Laplacian matrix. Now, we introduce some concepts and basic notation of graph theory. Many part of description and explanation here comes from [6, 2].

3.4.1 Definitions

A **directed graph** G is an ordered pair $G \triangleq (V, E)$ where V is a set, whose elements are called **vertices**, E is a set of ordered pairs of vertices, called **directed edges**. An directed edge $e = (x, y)$ is considered to be directed from $x \in V$ to $y \in V$. y is called the **head** and x is called the **tail** of the edge. A graph in which, if $\forall(x, y) \in E$, then $(y, x) \in E$, is said to be an **undirected graph**.

The **degree** of a vertex is the number of other vertices it is connected to by edges. **In-degree** of $x \in V$ is the number of edges with x as its head, **Out-degree** of $x \in V$ is analogously the number of edges with x as its tail.

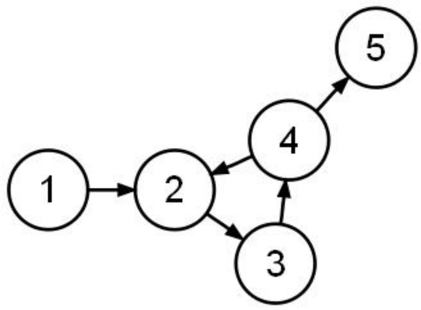


Figure 3.1: Directed graph

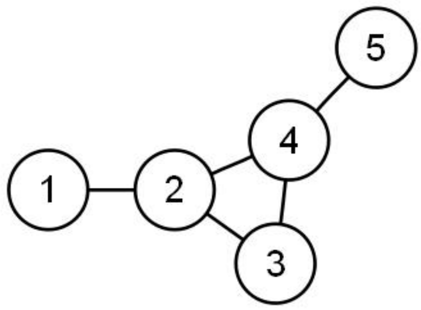


Figure 3.2: Undirected graph

3.4.2 Path

A **path** on G is a sequence of vertices such that from each of its vertices there is an edge to the next vertex in the sequence. i.e. $\{v_0, v_1, \cdots, v_N\}$ such that $(v_{i-1}, v_i) \in A$

for $\forall i \in [1, N]$. A **cycle** is a path such that the start vertex and end vertex are the same. **N-cycle** is a cycle that has a length N of path such that $v_0 = v_N$. If the set of all cycles that have less length than common divider k, then that graph is said to be **k-periodic**. **Acycle** graph is a graph that has no cycle. Sometimes, this graph is called **aperiodic** or **primitive**. For aperiodic graph, there is necessarily a vertex which has no path from the others or to the others. A vertex which is not head of any edge is **initial** or **follower**, and a vertex which is not tail of any edge is **final** or **leader**.

Example 4 (Leader and Follower)**.** *In figure 3.1, the vertex 1 is follower, and the vertex 5 is leader.*

3.4.3 Connectivity

In an undirected graph G, two vertices u and v are called **connected** if G contains a path from u to v. Otherwise, they are called **disconnected**. A directed graph is called **weakly connected** if replacing all of its directed edges with undirected edges produces a connected (undirected) graph. (i.e. one way connection) It is **strongly connected** if it contains a directed path from u to v for every pair of vertices $u, v \in V$. (i.e. two-way connection) A graph is **complete** if every pair of distinct vertices is connected by an edge, in which each pair of vertices is joined by an edge, that is, the graph contains all possible edges.

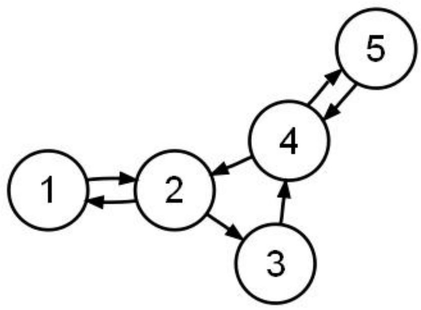


Figure 3.3: Strongly connected graph (Irreducible)

Example 5 (Strongly and Weakly Connected)**.** *The graph in figure 3.3 is strongly connected. All vertices are connected to each other. The graph in figure 3.1 is weakly* *connected due to vertices* 1*,* 5*. The vertex 1 has no path from the others, and the vertex 5 has no path to the others.*

3.4.4 Component

For a directed graph, **components** are its maximal strongly connected subgraphs which are said to be strongly connected components(SCC). These form a partition of the graph. If each strongly connected component is contracted to a single vertex, the resulting graph is a directed acyclic graph. Such as the previous case of the vertex, A component in which all vertex is not head of any edge is initial or follower, and a component in which all vertex is not tail of any edge is final or leader.

Example 6 (Component)**.** *In figure 3.1, there are three components as shown below.*

1. *A subgraph with vertices* $\{1\}$ *as follower,*
2. *A subgraph with vertices* $\{2, 3, 4\}$*,*
3. *A subgraph with vertices* $\{5\}$ *as leader.*

3.4.5 Graph Representation

The **adjacency matrix** denoted as A is the representation for a graph. For given finite graph G on n vertices, it is the $n \times n$ matrix where the nondiagonal entry a_{ij} is the number of edges from vertex i to vertex j, and the diagonal entry a_{ii} is the number of loops. If the graph is undirected, the adjacency matrix is symmetric. The **normalized adjacency matrix** of a graph on n vertices is a variation of an adjacency matrix, of which the sum of each row entries is normalized to be one. This matrix will be denoted as G. Now, we introduce the **Laplacian matrix** denoted as L. For a graph without loops from vertex i to i or multiple edges from one vertex to another, the Laplacian matrix is defined as $L = \{l_{ij}\}$

$$l_{ij} = \begin{cases} 1 & \text{if } i = j \text{ and } d_i \neq 0, \\ -\frac{1}{d_i} & \text{if } i \neq j \text{ and } \exists edge(i, j), \\ 0 & \text{otherwise.} \end{cases} \tag{3.9}$$

for $i, j = 1, ..., N$ where d_i is the out-degree of vertex i. More explanation of the Laplacian matrix is covered in [17].

Example 7 (Graph Representation)**.** *In Figure 3.2, Normalized adjacency matrix G and Laplacian matrix L are like that*

$$G = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad L = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -\frac{1}{3} & 1 & -\frac{1}{3} & -\frac{1}{3} & 0 \\ 0 & -\frac{1}{2} & 1 & -\frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{3} & 1 & -\frac{1}{3} \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \tag{3.10}$$

Remark 3.7.

$$L = I_n^\star - G \tag{3.11}$$

$$I_n^\star = diag(i_1, \cdots, i_n), \qquad i_j = \begin{cases} 1 & \text{if } d_j > 0, \\ 0 & \text{otherwise.} \end{cases} \tag{3.12}$$

Moreover, if all out-degree of vertices is non-zero,

$$L = I_n - G \tag{3.13}$$

Theorem 3.8. *The Laplacian matrix $L \in \mathbb{R}^{n \times n}$ always has the vector $\mathbf{1}_n \triangleq [1, \cdots, 1]^T \in \mathbb{R}^n$ as a member of the null space of L.*

$$L\mathbf{1}_n = 0 \tag{3.14}$$

Proof. By multiplication $\mathbf{1}_n$ to a right side of L, we immediately know $L\mathbf{1}_n = 0$. □

3.4.6 Irreducible

A matrix $A \in \mathbb{R}^{n \times n}$ is said to be **reducible** if either $n = 1$ and $A = 0$; or $n \geq 2$, there is a permutation matrix $P \in M^{n \times n}$ and there is some integer r with $1 \leq r \leq n - 1$ such that

$$P^T AP = \begin{bmatrix} B & C \\ 0 & D \end{bmatrix}$$

where $B \in \mathbb{R}^{r \times r}$, $D \in \mathbb{R}^{n-r \times n-r}$, $C \in \mathbb{R}^{r \times n-r}$, and $0 \in \mathbb{R}^{n-r \times r}$ is a zero matrix. A matrix $A \in \mathbb{R}^{n \times n}$ is **irreducible** if it is not reducible.

A directed graph is **irreducible** if, given any two vertices, there exists a path from the first vertex to the second. A directed graph is irreducible if and only if its Laplacian matrix is irreducible.

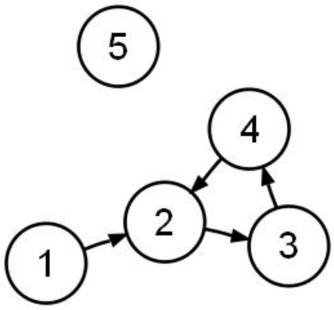


Figure 3.4: Reducible graph (Disconnected)

Theorem 3.9. *A directed graph G is strongly connected if and only if its adjacency matrix is irreducible.*

Proof. See Theorem 6.2.24 in [11]. □

3.5 Nonnegative Matrix

From here, we would study nonnegative matrix theory and especially we are interested in the relation between the null space of L and its properties. Many of materials described here come from [11].

Definition 3.10 (Nonnegative Matrix and Vector)**.** Consider $A = \{a_{ij}\} \in \mathbb{R}^{n \times m}$. If all $a_{ij} \geq 0$, then A is said to be a **nonnegative matrix** denoted as $A \geq 0$.

Similarly, assume that $v \in \mathbb{R}^n$ is a vector of which all elements are nonnegative. v is called **nonnegative vector**.

Definition 3.11 (Spectral Radius $r(A)$)**.** The set of all λ that are eigenvalues of $A \in \mathbb{C}^{n \times n}$ is called the spectrum of A and is denoted by $\sigma(A)$. The **spectral radius of** A is the nonnegative real number $r(A) = \max\{|\lambda| : \lambda \in \sigma(A)\}$. This is just the radius of the smallest disc centered at the origin in the complex plane that includes all the eigenvalues of A.

Theorem 3.12. *If $\|\cdot\|$ is any matrix norm, then*

$$r(A) \leq \|A\|, \qquad A \in \mathbb{C}^{n\times n} \tag{3.15}$$

Proof. Assume that $|\lambda| = r(A)$ and X is all the columns of which are equal to the eigenvector x such that $Ax = \lambda x$, $x \neq 0$.

$$|\lambda|\|X\| = \|\lambda X\| = \|AX\| \leq \|A\|\|X\| \Rightarrow r(A) \leq \|A\|$$

□

Theorem 3.13 (Perron-Frobenius Theorem for Nonnegative Irreducible Matrices)**.** *Let $A \in \mathbb{C}^{n\times n}$ and suppose that A is irreducible and nonnegative. Then*

1. $r(A) > 0$.
2. *$r(A)$ is an eigenvalue of A.*
3. *There is a positive vector x such that $Ax = r(A)x$.*
4. *$r(A)$ is an algebraically simple eigenvalue*[3] *of A.*

Proof. See Theorem 8.4.4 in [11]. □

Remark 3.14. Assume that A has $r(A)$ as the spectral radius, then A^T has also. Moreover, if A is irreducible and nonnegative, then A^T is so, and it is also true that there is a positive vector y such that $A^T y = r(A)y$ as well as a positive vector x such that $Ax = r(A)x$.

Definition 3.15 (Stochastic Matrix)**.** Consider a $n \times n$ matrix A. The matrix A is said to be a **row stochastic matrix** if each of whose rows consists of nonnegative real numbers with each row summing to 1. Analogously, it is a **column stochastic matrix** if whose columns consist of nonnegative real numbers whose sum is 1. And a **doubly stochastic matrix** if the matrix A is a row and column stochastic matrix.

[3] An algebraically simple eigenvalue has the algebraic multiplicity which is equal to 1.

Remark 3.16. The normalized adjacency matrix G is a stochastic one.

Theorem 3.17. *The spectrum of a stochastic matrix is contained in the unit circle in the complex plane.*

Proof. Assume the matrix A is stochastic. We can easily prove the theorem by investigating the norm of a stochastic matix A with the property $r(A) \leq \|A\|$ of Theorem 3.12. If A is row stochastic, $\|A\|_\infty = \max_i \left(\sum_j |a_{ij}|\right) = 1$ which is the maximum row sum, and if A is column stochastic, $\|A\|_1 = \max_j \left(\sum_i |a_{ij}|\right) = 1$ which is the maximum column sum. □

Theorem 3.18. *Assume that $r = r(G)$ where $G \in \mathbb{R}^{n\times n}$ is a normalized adjacency matrix. If G is irreducible or its directed graph is strongly connected, then there exist positive vectors $x, y^T \in \mathbb{R}^n$ such that $Gx = rx$ and $G^T y = ry$. Moreover, r is an algebraically simple eigenvalue of G, and r is 1.*

Proof. Because of the normalized adjacency matrix's property that $G\mathbf{1}_n = 1\mathbf{1}_n$, we have the eigenvalue 1 corresponding the eigenvector $\mathbf{1}_n$ of G. By Theorem 3.17, 1 is the largest one of eigenvalues of G. Thus, $r = 1$.

Now, under the given assumption that G is irreducible and nonnegative. By Theorem 3.13, we straightforwardly have positive vectors x, y^T such that $Gx = x$ and $G^T y = y$, and the eigenvalue $r = 1$ of G has the multiplicity 1. □

Theorem 3.19. *Assume that $L \in \mathbb{R}^{n\times n}$ is the Laplacian matrix of a directed graph which is strongly connected. The left and right null space of L are individually spanned by a single positive vector.*

Proof. At first, we claim that an eigenvalue λ of L corresponds to $1 - \lambda$ of G. By virtue of strongly connected graph, there is no vertex which has 0 as its out-degree. Thus, $L = I_n - G$. Assume that v is the eigenvector of L associated with λ.

$$Lv = (I_n - G)v = \lambda v \quad \Leftrightarrow \quad Gv = (1 - \lambda)v \tag{3.16}$$

Therefore, an eigenvalue $\lambda_i(L)$ of L is defined as

$$\lambda_i(L) = 1 - \lambda_i(G) \tag{3.17}$$

where $\lambda_i(G)$ is an eigenvalues of G and $i \in [1, n]$. Moreover, the matrices G and L share the same eigenvectors.

Now, we already know that G has $r(G) = 1$ which is simple, and corresponding left and right eigenvectors x, y^T such that $Gx = 1x$ and $G^T y = 1y$ are positive via Theorem 3.18. Hence, L has $1 - r(G) = 0$ as a simple eigenvalue and $Lx = 0x = 0, L^T y = 0y = 0$. Consequently, the left and right null space of L are spanned by a single positive vector. □

Theorem 3.20. *Assume that $L \in \mathbb{R}^{n\times n}$ is the Laplacian matrix of a directed graph which has a single leader component. The left null space of L includes a nonnegative vector except the zero.*

Proof. When L has a single leader component, we can write the normalized adjacency matrix $G = I_n - L$ as follows with a permutation matrix P.

$$PGP^T = \begin{bmatrix} G_{11} & 0 & \cdots & 0 \\ G_{21} & G_{22} & 0 & 0 \\ \vdots & & \ddots & \vdots \\ G_{m1} & G_{m2} & \cdots & G_{mm} \end{bmatrix} \tag{3.18}$$

where G_{ii} are irreducible which stand for strong connected components, and m is the number of components of the graph represented by G. Now, consider G_{11}. In the same manner of Theorem 3.19, there is a vector y which spans the left nullspace of $I - G_{11}$, that is $y(I - G_{11}) = 0$. If we choose $y^\star \in \mathbb{R}^{1\times n}$ such that $y^\star = [y, 0, \cdots, 0]$, then,

$$\begin{gathered} y^\star(I_n - PGP^T) = y^\star(PP^T - PGP^T) = y^\star P(I_n - G)P^T = 0 \\ \Rightarrow y^* L = 0, \qquad y^* = y^\star P \end{gathered} \tag{3.19}$$

As a result, we have the nonnegative vector y^* included in the left nullspace of L. □

Before closing this section, we additionally introduce a result about the relation among the dimension of nullspace of L, the number of leader components and non-negative vectors from [7] and [25].

Theorem 3.21. *The multiplicity m of the zero eigenvalue of the Laplacian matrix L is equal to the number of leader components of its graph. The dimension of the* *nullspace of L is also equal to m, and is spanned by a basis of m nonnegative vectors.*

Proof. See Proposition 4.5 in [7]. $\square$

Remark 3.22. If a graph has multiple leader components, then the the dimension of the nullspace of its Laplacian matrix might be greater than 1.

3.6 Control Analysis and Design

Now, we consider systems (3.5) over again. One can put the systems of N agents into one single system together by the Kronecker product and the Laplacian matrix. If we set the weight w_{ij} of system (3.5) to $\frac{1}{|\mathcal{N}_i|}$[4], then we have a very simple form of (3.4) together by virtue of the Laplacian matrix as the next theorem. See [8] as a consensus algorithm[5] in cooperative control.

Before proceeding our analysis, we would redefine $\frac{a}{|A|}$ as 0 if $|A| = 0$, to avoid the division by 0 where A and a are arbitrary set and number. This definition does not conflict with a conventional inverse of a number in our theorem.

Theorem 3.23. *For the following collection of multiple homogeneous systems,*

$$\begin{aligned} \dot{z}_i &= Az_i + Bv_i, \\ y_i &= \sum_{j \in \mathcal{N}_i} w_{ij}(Cz_i - Cz_j), \\ &A \in \mathbb{R}^{n\times n}, \quad B \in \mathbb{R}^{n\times m}, \quad C \in \mathbb{R}^{m\times n}, \quad w_{ij} \in \mathbb{R} \end{aligned} \tag{3.20}$$

for $1 \le \forall i \le N$ where $\mathcal{N}_i \subset [1, N] \setminus \{i\}$ is an index set that represents the set of agents which agent i can sense. If

$$w_{ij} = \frac{1}{|\mathcal{N}_i|}, \tag{3.21}$$

then the above collection of systems can be rewritten as one system

$$\begin{aligned} \dot{z} &= A_N z + B_N v, \\ y &= L_{(m)} C_N z \end{aligned} \tag{3.22}$$

[4] $|\mathcal{N}_i|$ is the cardinality of $\mathcal{N}_i$, that is the number of agents which the agent i can sense.

[5] The consensus problem is to have a group of agents reach a common assessment based on distributed information.

where

$$\begin{aligned}
z &= [z_1, z_2, \cdots, z_N]^T, \\
v &= [v_1, v_2, \cdots, v_N]^T, \quad y = [y_1, y_2, \cdots, y_N]^T, \\
A_N &= I_N \otimes A, \quad B_N = I_N \otimes B, \quad C_N = I_N \otimes C, \\
L_{(m)} &= L \otimes I_m, \quad L = \{l_{ij}\} \in \mathbb{R}^{N\times N}, \\
l_{ij} &= \begin{cases} 1 & \textit{if } i = j, \\ -\frac{1}{|\mathcal{N}_i|} & \textit{if } i \neq j \textit{ and } j \in \mathcal{N}_i, \\ 0 & \textit{otherwise.} \end{cases}
\end{aligned} \tag{3.23}$$

Proof. Straightforwardly, the collection of z_i can be written as follows

$$\begin{aligned}
\begin{bmatrix} \dot{z_1} \\ \dot{z_2} \\ \vdots \\ \dot{z_N} \end{bmatrix} &= \begin{bmatrix} A & 0 & \cdots & 0 \\ 0 & A & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & A \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix} + \begin{bmatrix} B & 0 & \cdots & 0 \\ 0 & B & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & B \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{bmatrix} \\
&= (I_N \otimes A)z + (I_N \otimes B)v = A_N z + B_N v = \dot{z}
\end{aligned} \tag{3.24}$$

Next, we now investigate the collection of y_i. Let $G \in \mathbb{R}^{N\times N}$ be the matrix as shown below. Note that G is a normalized adjacency matrix.

$$\begin{aligned}
G &= diag(\frac{1}{|\mathcal{N}_1|}, \frac{1}{|\mathcal{N}_2|}, \cdots, \frac{1}{|\mathcal{N}_N|}) \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1N} \\ g_{21} & g_{22} & \cdots & g_{2N} \\ \vdots & & \ddots & \vdots \\ g_{N1} & g_{N2} & \cdots & g_{NN} \end{bmatrix} \\
g_{ij} &= \begin{cases} 1 & \text{if } i \neq j \text{ and } j \in \mathcal{N}_i, \\ 0 & \text{otherwise.} \end{cases} \\
G_i &= \text{the } i\text{-th row of G}
\end{aligned} \tag{3.25}$$

$$
\begin{aligned}
y_i &= \sum_{j\in\mathcal{N}_i} w_{ij}(Cz_i - Cz_j) = \sum_{j\in\mathcal{N}_i} \frac{1}{|\mathcal{N}_i|}(Cz_i - Cz_j) \\
&= \frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i}(Cz_i - Cz_j) = \frac{1}{|\mathcal{N}_i|}(|\mathcal{N}_i|Cz_i - \sum_{j\in\mathcal{N}_i} Cz_j) \\
&= Cz_i - \frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i} Cz_j \\
&= Cz_i - \frac{1}{|\mathcal{N}_i|}([g_{i1}, g_{i2}, \cdots, g_{iN}] \otimes I_m)\begin{bmatrix} Cz_1 \\ Cz_2 \\ \vdots \\ Cz_N \end{bmatrix} \\
&= Cz_i - (\frac{1}{|\mathcal{N}_i|}[g_{i1}, g_{i2}, \cdots, g_{iN}] \otimes I_m)(I_N \otimes C)z \\
&= Cz_i - (G_i \otimes I_m)(I_N \otimes C)z
\end{aligned}
\tag{3.26}
$$

Thus, the collection of y_i is to be such that

$$
\begin{aligned}
\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} &= \begin{bmatrix} C & 0 & \cdots & 0 \\ 0 & C & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & C \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix} \\
&- \begin{bmatrix} G_1 \otimes I_m & 0 & \cdots & 0 \\ 0 & G_2 \otimes I_m & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & G_N \otimes I_m \end{bmatrix}\begin{bmatrix} C & 0 & \cdots & 0 \\ 0 & C & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & C \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix} \\
&= (I_N \otimes C)z - (G \otimes I_m)(I_N \otimes C)z \\
&= (I_{Nm} - G \otimes I_m)(I_N \otimes C)z \\
&= ((I_N - G) \otimes I_m)(I_N \otimes C)z \\
&= (L \otimes I_m)(I_N \otimes C)z \\
&= L_{(m)}C_N z = y
\end{aligned}
\tag{3.27}
$$

□

Corollary 3.24. *For the collection of input-output feedback linearized systems of homogeneous agents* (3.5)*, If*

$$w_{ij} = \frac{1}{|\mathcal{N}_i|}, \tag{3.28}$$

then they can be simplified to

$$\begin{aligned} \dot{\xi} &= \phi(\xi, z, v), \\ \dot{z} &= A_N z + B_N v, \\ y^* &= L_{(m)} C_N z \end{aligned} \tag{3.29}$$

where $y^* = [y_1^*, \cdots, y_N^*]^T$, $A_N, B_N, C_N, L_{(m)}$ *are such that equations* (3.23) *in Theorem 3.23.*

Theorem 3.25. *Consider system* (3.29)*, Assume that the tracking dynamics of the agent are bounded input bounded state stable,* $y_{(ij)r} \in \mathbb{R}^m$ *are the relative references*[6] *of* (3.3)*, and* $\lambda(L)$ *is the set of eigenvalues of* L.
Choose y_{ir} *such that*

$$y_{(ij)r} = y_{ir} - y_{jr} \qquad \textit{for } 1 \leq \forall i, j \leq N. \tag{3.30}$$

If

$$\begin{gathered} s^{\rho_1} + \lambda_i k_{1\rho_1} s^{\rho_1 - 1} + \cdots + \lambda_i k_{11}, \\ \vdots \\ s^{\rho_m} + \lambda_i k_{m\rho_m} s^{\rho_m - 1} + \cdots + \lambda_i k_{m1} \end{gathered} \tag{3.31}$$

are Hurwitz polynomials for $\forall \lambda_i \in \lambda(L)$ *such that* $\lambda_i \neq 0$*, and the multiplicity of the eigenvalue* λ_i *such that* $\lambda_i = 0$ *is*[7] 1*, furthermore, a nonnegative vector except the zero is included in the left null space of* L*, then*

$$\lim_{t \to \infty} y^*(t) = y_r^*(t) \tag{3.32}$$

[6] $(Cz_i(t) - Cz_j(t)) \to y_{(ij)r}(t)$ as $t \to \infty$.
[7] The dimension of nullspace of L.

with the controller $v = [v_{11}, \cdots, v_{1m}, \cdots\cdots, v_{N1}, \cdots, v_{Nm}]^T$ *as follows*

$$\begin{aligned}
v_{i1} &= -k_{11}(y^*_{i1} - y^*_{i1r}(t)) - \cdots - k_{1\rho_1}({y^*_{i1}}^{(\rho_1-1)} - {y^*_{i1r}}^{(\rho_1-1)}(t)) + {y_{i1r}}^{(\rho_1)}(t) \\
&\vdots \\
v_{im} &= -k_{m1}(y^*_{im} - y^*_{imr}(t)) - \cdots - k_{m\rho_m}({y^*_{im}}^{(\rho_1-1)} - {y^*_{imr}}^{(\rho_m-1)}(t)) + {y_{imr}}^{(\rho_m)}(t)
\end{aligned} \tag{3.33}$$

where

$$\begin{aligned}
y_r &= [y_{1r}, y_{2r}, \cdots, y_{Nr}]^T, \\
y^*_r &= L_{(m)} y_r, \\
y^*_{ijr} &\ \textit{is the } ((i-1) \times m + j)\textit{-th element of } y^*_r, \\
y^*_{ij} &\ \textit{is the } ((i-1) \times m + j)\textit{-th element of } y^*
\end{aligned} \tag{3.34}$$

Proof. We can prove the above theorem in a similar manner with the proof of Theorem 2.24.

For each $j \in [1, m]$, Let $e_{ijk} \triangleq {y^*_{ij}}^{(k-1)} - {y^*_{ijr}}^{(k-1)}$. One can then take $v_i =$

$[v_{i1}, \cdots, v_{im}]^T$ as the equation (3.33). We obtain

$$
\begin{aligned}
\dot{e}_{ijk} &= e_{ij(k+1)}, \qquad 1 \le k \le \rho_{j-1}, \\
\dot{e}_{ij\rho_j} &= {y_{ij}^*}^{(\rho_j)} - {y_{ijr}^*}^{(\rho_j)} \\
&= \frac{1}{\|\mathcal{N}_i\|} \sum_{l \in \mathcal{N}_i} (\dot{z_{ij\rho_j}} - \dot{z_{lj\rho_j}}) - \frac{1}{\|\mathcal{N}_i\|} \sum_{l \in \mathcal{N}_i} ({y_{ijr}}^{(\rho_j)} - {y_{ljr}}^{(\rho_j)}) \\
&= \frac{1}{\|\mathcal{N}_i\|} \sum_{l \in \mathcal{N}_i} (v_{ij} - v_{lj} - {y_{ijr}}^{(\rho_j)} + {y_{ljr}}^{(\rho_j)}) \\
&= \frac{1}{\|\mathcal{N}_i\|} \sum_{l \in \mathcal{N}_i} ((v_{ij} - {y_{ijr}}^{(\rho_j)}) - (v_{lj} - {y_{ljr}}^{(\rho_j)})) \\
&= (v_{ij} - {y_{ijr}}^{(\rho_j)}) - \frac{1}{\|\mathcal{N}_i\|} \sum_{l \in \mathcal{N}_i} (v_{lj} - {y_{ljr}}^{(\rho_j)}) \\
&= -k_{j1}(y_{ij}^* - y_{ijr}^*) - \cdots - k_{j\rho_j}({y_{ij}^*}^{(\rho_1 - 1)} - {y_{ijr}^*}^{(\rho_j - 1)}) + {y_{ijr}}^{(\rho_j)} - {y_{ijr}}^{(\rho_j)} \\
&\quad - \frac{1}{\|\mathcal{N}_i\|} \sum_{l \in \mathcal{N}_i} (-k_{j1}(y_{lj}^* - y_{ljr}^*) - \cdots - k_{j\rho_j}({y_{lj}^*}^{(\rho_1 - 1)} - {y_{ljr}^*}^{(\rho_j - 1)}) \\
&\qquad\qquad + {y_{ljr}}^{(\rho_j)} - {y_{ljr}}^{(\rho_j)}) \\
&= -k_{j1} e_{ij1} - \cdots - k_{j\rho_j} e_{ij\rho_j} - \frac{1}{\|\mathcal{N}_i\|} \sum_{l \in \mathcal{N}_i} (-k_{j1} e_{lj1} - \cdots - k_{j\rho_j} e_{lj\rho_j})
\end{aligned}
$$

for $1 \le \forall i \le N$. Thus, as a simple form,

$$
\begin{aligned}
\dot{e}_{ij} &= H_j e_{ij} - \frac{1}{\|\mathcal{N}_i\|} \sum_{l \in \mathcal{N}_i} F_j H_j e_{ij} \\
&= H_j e_{ij} - F_j H_j (G_i \otimes I_{\rho_j}) e_j \qquad\qquad 1 \le \forall j \le m
\end{aligned}
\tag{3.35}
$$

where

$$H_j = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ -k_{j1} & -k_{j2} & \cdots & -k_{j\rho_j} \end{bmatrix}, \quad F_j = \begin{bmatrix} 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{\rho_j \times \rho_j},$$
$$e_{ij} = \begin{bmatrix} e_{ij1} \\ \vdots \\ e_{ij\rho_j - 1} \\ e_{ij\rho_j} \end{bmatrix}, \quad e_j = \begin{bmatrix} e_{1j} \\ \vdots \\ e_{N-1j} \\ e_{Nj} \end{bmatrix}, \quad G_i \text{ is that as (3.25)} \tag{3.36}$$

Then, we have the collection e_j of e_{ij}, $\forall i \in [1, N]$, as shown below.

$$\begin{aligned} \dot{e_j} &= (I_N \otimes H_j)e_j - (I_N \otimes F_j H_j)(G \otimes I_{\rho_j})e_j \\ &= (I_N \otimes H_j)e_j - (G \otimes F_j H_j)e_j \\ &= (I_N \otimes H_j)e_j - (G \otimes F_j)(I_N \otimes H_j)e_j \\ &= (I_{N\rho_j} - G \otimes F_j)(I_N \otimes H_j)e_j \\ &= (I_N \otimes I_{\rho_j} - G \otimes F_j)(I_N \otimes H_j)e_j \\ &= (I_N \otimes I_{\rho_j} + (I_N - G) \otimes F_j - I_N \otimes F_j)(I_N \otimes H_j)e_j \\ &= (I_N \otimes (I_{\rho_j} - F_j) + L \otimes F_j)(I_N \otimes H_j)e_j \\ &= ((L \otimes F_j)(I_N \otimes H_j) + I_N \otimes (H_j - F_j H_j))e_j \\ &= (L \otimes F_j H_j + I_N \otimes N_j)e_j \end{aligned} \tag{3.37}$$

where

$$N_j = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \end{bmatrix} \tag{3.38}$$

From now, we want to know the asymptoticity of the matrix $E_j \triangleq L \otimes F_j H_j + I_N \otimes N_j$. If one can transform E_j into an upper triangular form via similarity transformation which preserves the stability of an given system, we are easily able to check a stability of the system (3.37). If the error dynamics (3.37) of output tracking by relative references is asymptotically stable, the statement (3.32) is true.

By Schur's unitary[8] triangularization theorem[11], we can choose a real orthogonal matrix T such that $U = T^{-1}LT$ in which, U is an upper triangular matrix, and the diagonal entries of U are exactly the eigenvalues of L.

By the similarity transformation $\tilde{e_j} = (T^{-1} \otimes I_{\rho_j})e_j$, we will get an upper triangular system like this, which has the same stability property with system (3.37).

$$\begin{aligned} \dot{\tilde{e_j}} &= (T^{-1} \otimes I_{\rho_j})(L \otimes F_j H_j + I_N \otimes N_j)(T \otimes I_{\rho_j})\tilde{e_j} \\ &= (T^{-1}LT \otimes F_j H_j + I_N \otimes N_j)\tilde{e_j} \\ &= (U \otimes F_j H_j + I_N \otimes N_j)\tilde{e_j} \end{aligned} \tag{3.39}$$

Let

$$H_{j\lambda} = \lambda F_j H_j + N_j = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ -\lambda k_{j1} & -\lambda k_{j2} & \cdots & -\lambda k_{j\rho_j} \end{bmatrix} \tag{3.40}$$

The characteristic equation of system (3.39) is

$$\prod_{\lambda_i \in \lambda(L)} det(H_{j\lambda_i} - sI_{\rho_j}) = \prod_{\lambda_i \in \lambda(L)} s^{\rho_j} + \lambda_i k_{j\rho_j} s^{\rho_j - 1} + \cdots + \lambda_i k_{j1} \tag{3.41}$$

by the benefit from an upper triangular shape.

For our assumption (3.31), All roots s of the characteristic equation (3.41) have negative real parts except $s = 0$ when $\lambda_i = 0$. Therefore, at least, system (3.39) is stable and system (3.37) is as well by the similarity. If one has no invariant points except $e_j = 0$ when $s = 0$, we may say that system (3.37) is asymptotic stable by Lasalle's invariance principle[14].

At this time, we will find the invariant points of system (3.37) by setting $\dot{e}_j = 0$. We know the dimension of the null space of $L \otimes F_j H_j + I_N \otimes N_j$ is 1 by the assumption, and propose the vector $\mathbf{1}_N \otimes [1, 0, \cdots, 0]^T$ as the basis of the null space. That is easily

[8]A matrix $T \in \mathbb{C}^{n \times n}$ is said to be **unitary** if $T^H T = I$. In addition, $T \in \mathbb{R}^{n \times n}$ is said to be **real orthogonal**.

checked by substitution. Thus, e_j can be represented as follows.

$$e_j = c(\mathbf{1}_N \otimes [1, 0, \cdots, 0]^T) \tag{3.42}$$

where $c \in \mathbb{R}$. In the above equation, if it is the only possible case of c that $c = 0$, the invariant points of system (3.37) is the origin $e_j = 0$.

Now, assume that c is non-zero. One can get non-zero elements from (3.42) as shown below.

$$\begin{bmatrix} e_{1j1} \\ e_{2j1} \\ \vdots \\ e_{Nj1} \end{bmatrix} = \begin{bmatrix} y^*_{1j} - y^*_{1jr} \\ y^*_{2j} - y^*_{2jr} \\ \vdots \\ y^*_{Nj} - y^*_{Njr} \end{bmatrix} = L(I_N \otimes O_j)(y - y_r) = c\mathbf{1}_N \tag{3.43}$$

where

$$O_j = \{o_{1k}\}, \qquad k \in [1, m], \qquad o_{1k} = \begin{cases} 1 & k = j \\ 0 & otherwise. \end{cases} \tag{3.44}$$

If we multiply a nonnegative vector of our assumption to both sides, which is included in the left null space of L, c should be zero by contradiction.

$$n_l L(I_N \otimes O_j)(y - y_r) = n_l c\mathbf{1}_N \neq 0, \quad \text{Nonnegative } n_l \in \text{Left Null Space of } L \tag{3.45}$$

Consequently, e_j asymptotically approaches the origin. Because we have an asymptotic stable system for every $j \in [1, m]$, one can finally attain that $y^* \to {y_r}^*$ as $t \to \infty$.

Since y^* are bounded, it follows that $z(t)$ are bounded. By the property of the bounded input bounded state stability of the tracking dynamics, $\xi(t)$ from (3.29) is bounded as well, $z(t)$ as bounded references. □

Now, we should ensure that, if

$$\lim_{t \to \infty} y^*(t) = y^*_r(t)$$

then

$$\lim_{t \to \infty} \|(Cz_i(t) - Cz_j(t)) - y_{(ij)r}(t)\| = 0$$

that is our goal of output tracking by relative references.

Theorem 3.26. *Consider system* (3.29) *under the condition of Theorem 3.25. If*

$$\lim_{t\to\infty} y^*(t) = y_r^*(t), \tag{3.46}$$

then

$$\lim_{t\to\infty} \|(Cz_i(t) - Cz_j(t)) - y_{(ij)r}(t)\| = 0 \tag{3.47}$$

Proof.

$$\begin{aligned}
y^* - y_r^* &= L_{(m)}(y - y_r) = 0 \\
&\Downarrow \quad \because \text{The null space of } L \text{ is spanned by } \mathbf{1}_N \\
y_i - y_{ir} = c\mathbf{1}_m \quad &\Leftrightarrow \quad y_i = y_{ir} + c\mathbf{1}_m \\
&\Downarrow \\
Cz_i - Cz_j = y_i - y_j &= y_{ir} + c\mathbf{1}_m - y_{jr} - c\mathbf{1}_m = y_{ir} - y_{jr} = y_{(ij)r}
\end{aligned} \tag{3.48}$$

□

Corollary 3.27. *For system* (3.29)*, if it satisfies the condition of Theorem 3.25, then the problem of the output tracking by relative references is solvable.*

Remark 3.28 (Strongly Connected and Single Leader Component). If the directed graph of the Laplacian matrix L is strongly connected or has a single leader component, then the left null space of L includes a nonnegative vector and its dimension is 1 through Theorem 3.19 and 3.20. Thus, we can replace the condition of a nonnegative vector included in the left null space of L and the nullity 1 of L with the property of strongly connected graph or a single leader component for the Laplacian matrix L.

Remark 3.29. The controller v_i in equation (3.33) is a decentralized one[9]. In the i-th agent, we have the controller v_i as follows.

$$v_i = -\frac{1}{|\mathcal{N}_i|}\sum_{l\in M} K_l \sum_{j\in\mathcal{N}_i} (y_i - y_j - y_{ir}(t) + y_{jr}(t))^{(l-1)} + \begin{bmatrix} y_{i1r}^{(\rho_1)}(t) \\ \vdots \\ y_{imr}^{(\rho_m)}(t) \end{bmatrix} \tag{3.49}$$

[9]The decentralized controller depends on only neighbors of each agents, not all agents.

where

$$M = \{1, 2, \cdots, max(\rho_1, \cdots, \rho_m)\}$$
$$K_l = \begin{bmatrix} k^\star_{1l} & 0 & \cdots & 0 \\ 0 & k^\star_{2l} & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & k^\star_{ml} \end{bmatrix}, \qquad k^\star_{jl} = \begin{cases} k_{jl} & \text{if } l \leq \rho_j, \\ 0 & otherwise. \end{cases} \tag{3.50}$$

Thus, for the collection of systems (3.1), we have the feedback controller u_i with v_i of the i-th agent for the tracking problem as shown below.

$$u_i = -\mathcal{A}^{-1}(x_i)\left(L_f^\rho h(x_i) - \left(\frac{1}{|\mathcal{N}_i|}\sum_{l\in M} K_l \sum_{j\in\mathcal{N}_i}(y_i - y_j - y_{ir} + y_{jr})^{(l-1)} + y_{ir}^{(\rho)}(t)\right)\right) \tag{3.51}$$

where

$$L_f^\rho h(x_i) = \begin{bmatrix} L_f^{\rho_1} h_1(x_i) \\ \vdots \\ L_f^{\rho_m} h_m(x_i) \end{bmatrix}, \qquad y_{ir}^{(\rho)}(t) = \begin{bmatrix} y_{i1r}^{(\rho_1)}(t) \\ \vdots \\ y_{imr}^{(\rho_m)}(t) \end{bmatrix},$$
$$\mathcal{A}(x_i) = \begin{bmatrix} L_{g_1}L_f^{\rho_1-1}h_1(x_i) & \cdots & L_{g_m}L_f^{\rho_1-1}h_1(x_i) \\ L_{g_1}L_f^{\rho_2-1}h_2(x_i) & \cdots & L_{g_m}L_f^{\rho_2-1}h_2(x_i) \\ \vdots & \ddots & \vdots \\ L_{g_1}L_f^{\rho_m-1}h_m(x_i) & \cdots & L_{g_m}L_f^{\rho_m-1}h_m(x_i) \end{bmatrix} \tag{3.52}$$

In this chapter, we have considered the collection of multiple homogeneous systems (3.1). When the global vector relative degree ρ is well defined with $\rho^* \leq n$, the vector fields (2.7) are complete, and the references for systems (3.1) are given by relative values. If the output of each agent system can be absolutely measurable by itself, then we can transform each agent system into the system (3.5) via input-output feedback linearization.

For given relative references, one can set new output form of the system (3.5), and with the assumption of scalars for weights like (3.21), we can rewrite the collection of systems as one big system (3.29) by the Laplacian matrix L and the Kronecker product.

If the directed graph of the Laplacian matrix L is strongly connected or has a single leader component, then the left null space of L includes at least one nonnegative vector except the zero, and its dimension is 1. Therefore, under assumption that the tracking dynamics of the agent is bounded input bounded output stable, we can finally have a tracking controller u_i subject to the condition (3.31). Moreover, u_i is represented by the equation (3.51) as shown above.

Chapter 4

Formation Tracking Control

This chapter is composed of two parts about the formation tracking problem and the permutation invariant formation tracking problem. Before we discuss the formation tracking control, the common concept of formations is described. And the formation is mathematically defined as the matrix and the vector about relative outputs of the agents in the group.

Figure 4.1: Patrolling flight of T-50, Republic of Korea Air Force.

4.1 Formation

In this section, we describe what formation is, and mathematically define it. The term of formation is used in a variety of areas. In military, a formation means the physical deployment of moving military forces like ground forces, aircrafts, and sea vessels. Examples of a formation includes line, square, V form and so on, that is the arrangement for tactical mission.

In sports, for instance, in football, the formation illustrates how the players in a team are positioned on a playground. Various formation can be used relying on more defensive or offensive playing by which a team might be characterized.

Figure 4.2: 4-4-2 Formation, sometimes called a flat-back 4. The midfielders are required to work hard to support both the defence and the attack. Courtesy of Eteamz.com

Concerning transportation system, formation can be considered as platooning of vehicles. Figure 4.3 shows the platooning of cars on a highway. Through formation control, one can achieve to grow the capacity of highway and to enhance the safety without additional lanes and equipments by decreasing the space between cars and increasing the density of traffic.

Generally speaking, Formation commonly indicates a configuration of some agents in the sense of position or location while aiming to coordinate a shared operation among agents. In such formation, each agent is able to freely move to all direction. Hence, formation should not include an absolute location. Thus, it would be a rel-

Figure 4.3: The platoon control demonstration in San Diego by the National Automated Highway System Consortium. See [1].

ative information between the states of the agents. Now, we mathematically define the formation to explain a formation control that is our issue.

4.1.1 Formation Matrix

Definition 4.1 (Formation Matrix F)**.** For the collection of N agents, each agent i has the states $y_i \in \mathbb{R}^m$ that represent a quantity of position or motional traits as its output. **Formation Matrix** F is a $mN \times N$ matrix of smooth vector fields with respect to time as follows.

$$
\begin{aligned}
F(t) &= \begin{bmatrix} f_{11}(t) & \cdots & f_{1N}(t) \\ \vdots & \ddots & \vdots \\ f_{N1}(t) & \cdots & f_{NN}(t) \end{bmatrix}, \quad f_{ij}(\cdot) \in \mathbb{R}^m, \\
f_{ij}(t) &= f_{ik}(t) - f_{jk}(t), \qquad 1 \le \forall k \le N
\end{aligned}
\tag{4.1}
$$

Remark 4.2. The condition $f_{ij}(t) = f_{ik}(t) - f_{jk}(t)$ of (4.1) describes the feasibility of a shape. For instance, in a triangle, the largest edge of the triangle should be shorter than the sum of other edges. Otherwise, the three edges can not make a triangle. See [30] about the feasibility of the formation.

Remark 4.3. If a formation matrix F contains a f_{ij}, then the $-f_{ij}$ is also included in F. Moreover, $f_{ij} = -f_{ji}$. ($\because f_{ij} = f_{ii} - f_{ji} = -f_{ji}$)

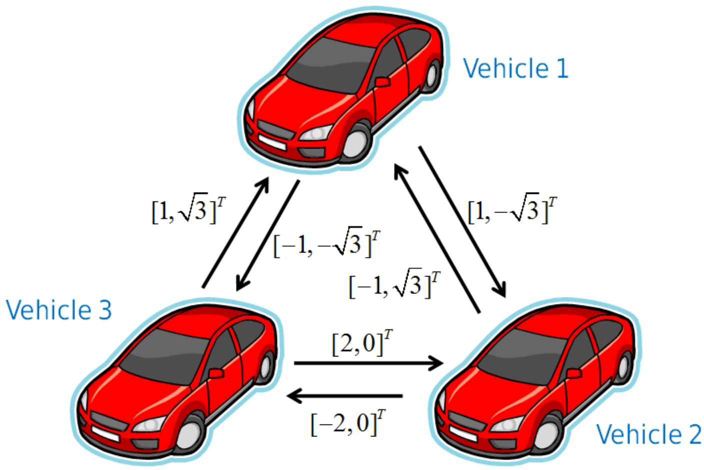


Figure 4.4: The three vehicles in the equilateral triangle formation with 2-dimensional relative outputs.

Example 8 (Formation Matrix)**.** *In Figure 4.4, the formation matrix F is defined as shown below.*

$$F = \begin{bmatrix} 0 & -1 & 1 \\ 0 & \sqrt{3} & \sqrt{3} \\ 1 & 0 & 2 \\ -\sqrt{3} & 0 & 0 \\ -1 & -2 & 0 \\ -\sqrt{3} & 0 & 0 \end{bmatrix} \tag{4.2}$$

4.1.2 Formation of Agents

Definition 4.4 (Formation)**.** Consider the collection of N agents. Assume that each agent i has the states $y_i \in \mathbb{R}^m$ that represent a quantity of position or motional traits as its output. The collection of N agents are **in the formation** F if $y_i - y_j = f_{ij}(t)$ for $1 \leq \forall i, j \leq N$ where $f_{ij}(\cdot) \in \mathbb{R}^m$ is the submatrix of the $mN \times N$ formation matrix F like (4.1).

Remark 4.5. In the formation F, the group of agents can freely move anywhere. Information of a formation does not have an absolute value on the outputs of the agents.

4.1.3 Formation Vector

Definition 4.6 (Formation Vector ζ)**.** For a $mN \times N$ formation matrix F, **Formation vector** ζ is a mN-dimensional smooth vector field as shown below.

$$F = \zeta \otimes \mathbf{1}_N^T - \mathbf{1}_N \otimes \zeta^\dagger \tag{4.3}$$

where

$$\begin{aligned} \zeta &= \left[- \zeta_1^T -, \cdots, - \zeta_N^T -\right]^T \in \mathbb{R}^{mN}, \\ \zeta^\dagger &= \begin{bmatrix} | & & | \\ \zeta_1 & \cdots & \zeta_N \\ | & & | \end{bmatrix} \in \mathbb{R}^{m \times N}, \\ \zeta_i(\cdot) &= [\zeta_{i1}(\cdot), \cdots, \zeta_{im}(\cdot)]^T \in \mathbb{R}^m \end{aligned} \tag{4.4}$$

Remark 4.7. If the dimension of the output m is 1, the formation vector ζ satisfies

$$F = \zeta \mathbf{1}_N^T - \mathbf{1}_N \zeta^T \tag{4.5}$$

Remark 4.8. For the given formation matrix F and formation vector ζ in the above definition,

$$f_{ij}(t) = \zeta_i(t) - \zeta_j(t) \tag{4.6}$$

where f_{ij}, ζ_i are defined as (4.1) and (4.4).

Remark 4.9. For the equation (4.3), there are many formation vectors that satisfies the condition. One can easily choose a formation vector ζ by setting ζ_i to the difference between an arbitrary constant vector $h \in \mathbb{R}^m$ and y_i when the agents are in the formation.[7]

$$\zeta_i = y_i - h \tag{4.7}$$

Example 9. *For the formation matrix F in* (4.2) *corresponding Figure 4.4, a fol-*

lowing vector ζ is the formation vector.

$$\begin{aligned} &F = \zeta \otimes \mathbf{1}_3^T - \mathbf{1}_3 \otimes \zeta^\dagger, \\ &\zeta = \left[0, 0, 1, -\sqrt{3}, -1, -\sqrt{3}\right]^T, \quad \zeta^\dagger = \begin{bmatrix} 0 & 1 & -1 \\ 0 & -\sqrt{3} & -\sqrt{3} \end{bmatrix} \end{aligned} \tag{4.8}$$

4.2 Formation Tracking

Consider the collection of multiple systems.

$$\begin{aligned} \dot{x_1} &= f_1(x_1) + g_1(x_1)u_1, & y_1 &= h_1(x_1), \\ \dot{x_2} &= f_2(x_2) + g_2(x_2)u_2, & y_2 &= h_2(x_2), \\ &\vdots \\ \dot{x_N} &= f_N(x_N) + g_N(x_N)u_N, & y_N &= h_N(x_N), \end{aligned} \tag{4.9}$$

where

$$\begin{aligned} u_i &= [u_{i1}, \cdots, u_{im}]^T, & u_i &\in \mathbb{R}^m \\ g_i(x_i) &= \begin{bmatrix} | & & | \\ g_{i1}(x_i), & \cdots, & g_{im}(x_i) \\ | & & | \end{bmatrix}, & g_i(\cdot) &\in \mathbb{R}^{n\times m} \\ h_i(x_i) &= [h_{i1}(x_i), \cdots, h_{im}(x_i)]^T, & h_i(\cdot) &\in \mathbb{R}^m \end{aligned}$$

with $x_i \in \mathbb{R}^n$ $(1 \leq \forall i \leq N)$, smooth f_i and $g_{i1}, \cdots g_{im}$ vector fields over $\mathbb{R}^n$ and h_{ij}: $\mathbb{R}^n \to \mathbb{R}$ a smooth function s.t. $h_{ij}(0) = 0 \quad (0 \leq \forall j \leq m)$.

Definition 4.10 (Formation Tracking Problem)**.** For the collection of multiple systems (4.9), the **formation tracking problem** is to design a controller with the property that, for a formation matrix F like (4.1), the outputs of systems satisfies

$$\lim_{t\to\infty} \| (y_i(t) - y_j(t)) - f_{ij}(t)\| = 0, \qquad 1 \leq \forall i, j \leq N \tag{4.10}$$

for any initial condition of the closed loop system. Therefore, the collection of multiple systems would be in the formation F by the controller.

Remark 4.11. For a formation tracking problem, One can think about the rotation of the group in the formation. But, the formation matrix changes under the rotation. Actually, a formation matrix is invariant under the translation of the group but not under the its rotation. One might deal with such a rotation invariant formation with introducing a rotation matrix where we will not consider the case.

Remark 4.12. For a formation tracking problem, one can replace the formation matrix F with a formation vector ζ like (4.3) as follows.

$$\lim_{t\to\infty} \| (y_i(t) - y_j(t)) - (\zeta_i(t) - \zeta_j(t)) \| = 0, \qquad 1 \leq \forall i, j \leq N \tag{4.11}$$

Remark 4.13. For the collection of the system (4.9), if f_i, g_i and h_i are respectively identical for $1 \leq \forall i \leq N$, then we can deal with a formation tracking problem as an output tracking of multiple homogeneous agents by relative references. In that case, relative references are to be f_{ij} of a formation matrix F.

Theorem 4.14 (Formation Tracking Control)**.** *Consider the collection of the agent systems* (4.9)*. Assume that f_i, g_i and h_i are respectively identical for $1 \leq \forall i \leq N$ that means homogeneous systems, and the Laplacian matrix L stands for an output sensing graph $\mathcal{G}$ of the agents in which each agent i can sense the outputs of its neighbors $\mathcal{N}_i$. For a formation vector ζ like* (4.4)*, if*

- *the agent is globally state feedback input-output linearizable,*
- *each output of agents is absolutely measurable by itself,*
- *the tracking dynamics of the agent is bounded input bounded state stable,*
- *the directed graph $\mathcal{G}$ of the Laplacian L is strongly connected or has a single leader component,*
- *and $s^{\rho_j} + \lambda_i k_{j\rho_j} s^{\rho_j - 1} + \cdots + \lambda_i k_{j1}$ are Hurwitz polynomials for $\forall \lambda_i \in \lambda(L)$ such that $\lambda_i \neq 0$, $1 \leq \forall j \leq m$,*

then, the formation tracking problem is solvable with the controller u_i as shown below.

$$u_i = -\mathcal{A}^{-1}(x_i) \left(L_f^{\rho} h(x_i) - \left(\frac{1}{|\mathcal{N}_i|} \sum_{l \in M} K_l \sum_{j \in \mathcal{N}_i} (y_i - y_j - \zeta_i + \zeta_j)^{(l-1)} + \zeta_i^{(\rho)}(t) \right) \right) \tag{4.12}$$

where

$$
\begin{aligned}
&\rho = [\rho_1, \cdots, \rho_m]^T \in \mathbb{N}^m \ \textit{as the vector relative degree of the agent,} \\
&M = \{1, 2, \cdots, max(\rho_1, \cdots, \rho_m)\}, \\
&K_l = \begin{bmatrix} k^\star_{1l} & 0 & \cdots & 0 \\ 0 & k^\star_{2l} & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & k^\star_{ml} \end{bmatrix}, \qquad k^\star_{jl} = \begin{cases} k_{jl} & \textit{if } l \le \rho_j, \\ 0 & \textit{otherwise.} \end{cases} \\
&L^\rho_f h(x_i) = \begin{bmatrix} L^{\rho_1}_f h_1(x_i) \\ \vdots \\ L^{\rho_m}_f h_m(x_i) \end{bmatrix}, \qquad \zeta^{(\rho)}_i(t) = \begin{bmatrix} \zeta^{(\rho_1)}_{i1}(t) \\ \vdots \\ \zeta^{(\rho_m)}_{im}(t) \end{bmatrix}, \\
&\mathcal{A}(x_i) = \begin{bmatrix} L_{g_1} L^{\rho_1 - 1}_f h_1(x_i) & \cdots & L_{g_m} L^{\rho_1 - 1}_f h_1(x_i) \\ L_{g_1} L^{\rho_2 - 1}_f h_2(x_i) & \cdots & L_{g_m} L^{\rho_2 - 1}_f h_2(x_i) \\ \vdots & \ddots & \vdots \\ L_{g_1} L^{\rho_m - 1}_f h_m(x_i) & \cdots & L_{g_m} L^{\rho_m - 1}_f h_m(x_i) \end{bmatrix}
\end{aligned}
\tag{4.13}
$$

Proof. If we respectively choose ζ_i and ζ_j as y_{ir} and y_{jr} in Theorem 3.25, then, one can straightforwardly prove the given theorem. See Theorem 3.5, 3.19, 3.20, 3.25 and Corollary 3.27. □

4.3 Permutation of Agents

In this section, we think about an permutation of the agents in the formation. Consider the collection of multiple homogeneous agents. Each of agents is identical to the others which means that every agents have the same ability of not only the movement but also the other features in the group. In the formation, a certain agent can equivalently play a role even though its position might be changed. This concept is one that the formation is unchanged by permuting the agents in the group, which is described by the term of permutation invariant formation in [26].

Definition 4.15 (Permutation Invariant Formation)**.** Consider the collection of N agents. Assume that each agent i has the states $y_i \in \mathbb{R}^m$ that represent a quantity of position or motional traits as its output. The collection of N agents are **in the**

permutation invariant formation F if there exists a bijective function[1] $p : A \to A$ such that

$$y_{p(i)} - y_{p(j)} = f_{ij}(t), \qquad 1 \le \forall i, j \le N, \\ A = \{1, 2, \cdots, N\} \subset \mathbb{N} \tag{4.14}$$

where $f_{ij}(\cdot) \in \mathbb{R}^m$ is the submatrix of the $mN \times N$ formation matrix F like (4.4).

Definition 4.16 (Formation Assignment and Virtual Agent)**.** For the equation (4.14), the function p does the **formation assignment on** F for the **virtual agent** in the formation F. One might be aware of the mapping p from i to $p(i)$ as the assignment from the i-th virtual agent to the $p(i)$-th actual agent.

Remark 4.17*.* A bijective function p in Definition 4.15 can be rewritten by a $N \times N$ permutation matrix P and a formation vector ζ as shown below.

$$y = (P \otimes I_m)\, \zeta \tag{4.15}$$

where $y = [y_1, y_2, \cdots, y_N]^T$.

Definition 4.18 (Permutation Invariant Formation Tracking Problem)**.** Consider the collection of multiple systems (4.9) for a formation matrix F like (4.1). Assume that f_i, g_i and h_i are respectively identical for $1 \le \forall i \le N$, and every agent can equivalently play each role of the agents regardless of the index of the agent. The **permutation invariant formation tracking problem** is to design a controller with the property that, the outputs of systems satisfies

$$\lim_{t\to\infty} \| \left(y_{p(i)}(t) - y_{p(j)}(t)\right) - f_{ij}(t)\| = 0, \qquad 1 \ \le \forall i, j \le N \tag{4.16}$$

for any initial condition of the closed loop system where the function p is bijective from $[1, N]$ to $[1, N]$ which does the formation assignment on F. Therefore, the collection of multiple systems would be in the permutation invariant formation F by the controller.

Remark 4.19*.* Concerning a permutation invariant formation tracking problem, the position of an agent in the formation is not deterministic in the stage of designing

[1] A bijective function is one-to-one and onto between the domain and range.

the formation matrix. That means that one does not explicitly know which agent in the group goes where in the formation.

4.4 Permutation Invariant Formation Tracking

This section is devoted to the discussion of how to solve a permutation invariant formation tracking problem. In that problem, The permutation matrix P representing a formation assignment is essential to characterize the feature of the formation tracking. Now, we focus on a configuration of P to minimize the distance for reaching the permutation invariant formation in the aspect of the weighted graph matching problem.

4.4.1 Weighted Graph Matching Problem

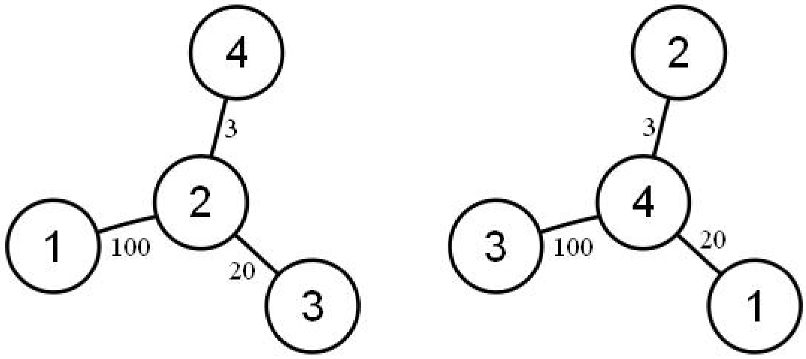
The weighted graph matching problem is to find a permutation of vertices of one graph such that the rearranged vertices make the distance minimized from the edge weights. Graph match problem is a fundamental one that arises in a lot of fields like distributed control, computer vision and facility allocation. Graph matching problem in this section is that the collection of multiple homogeneous agents reach a permutation invariant formation while minimizing the distance to getting the formation.

Definition 4.20 (Weighted Graph Matching Problem)**.** Consider two weighted undirected graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ with the same n vertices. Assume that A_1 and A_2 are respectively $n \times n$ weighted adjacency matrices of the graph $\mathcal{G}_1$ and $\mathcal{G}_2$. The **weighted graph matching problem** is defined to find the permutation matrix P such that

$$\min_{P \in P_n} \|A_1 - P^T A_2 P\| \tag{4.17}$$

where P_n denote the set of all $n \times n$ permutation matrices, and $\|\cdot\|$ is a matrix norm.

Remark 4.21. If a permutation matrix P makes the value of (4.17) equal to 0, the two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ have the same structure with simple relabelling of the vertices. In such case, we say that the two graphs are **isomorphic**.



(a) Graph $\mathcal{G}_1$ (b) Graph $\mathcal{G}_2$

Figure 4.5: The graph $\mathcal{G}_1$ and $\mathcal{G}_2$ are isomorphic while sharing the same structure. By the permutation of the vertices, two graphs come to be identical.

4.4.2 Dynamical Approach to Formation Assignment

Generally, a weighted graph matching problem is known as NP-complete[2]. See [10, 9]. It has received a lot of attention due to its hardness. Thus, there are many solutions to tackle this problem. Now, we introduce one method to solve the weighted graph matching problem. That is the dynamical system approach which is described in [35]. They utilize Theorems of [5] that is about the dynamical systems to sort lists, diagonalize matrices, and solve linear programming problems. This approach has the solution as the trajectory of the system which is dynamically evaluated. Hence, we can take the advantage of not only the final exact solution of the problem but also the transient and approximated solution that will be used to our problem, the permutation invariant formation tracking by rearrangement of the formation vector, later on.

Dynamical Formation Assignment

In the paper [35], they reformulate the graph matching problem as follows

$$\min_{P \in O_n \cap N_n} \|PA_1 - A_2P\|_F^2 \tag{4.18}$$

[2]The complexity NP-complete is a subset of NP which is the set of non-deterministic polynomial time problems. A decision problem C is NP-complete if C is in NP, and every problem in NP is reducible to C.

where O_n denotes the set of $n \times n$ orthogonal matrices, N_n denotes the set of $n \times n$ element-wise nonnegative matrices, and $\| \cdot \|_F$ is the Frobenius norm[3] of matrix. They claim P_n is equivalent to $O_n \cap N_n$, and construct the dynamical system on the manifold of orthogonal matrices. That dynamical system has two gradient flows. The first one minimizes the cost of weighted graph matching (4.18) and the second one forces the matrix P to be a permutation matrix which is the subset of orthogonal matrices.

Theorem 4.22. *Assume that A_1 and A_2 are the adjacency matrices corresponding to the weighted undirected graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, and $P(0) \in O_n$. If $P(t)$ is the solution of the following matrix differential equation*

$$\dot{P} = P\left(P^T A_2 P A_1 - A_1 P^T A_2 P\right) - kP\left((P \circ P)^T P - P^T (P \circ P)\right) \tag{4.19}$$

for all $t \geq 0$ where k is a positive constant and $A \circ B$ denotes the element-wise product of the matrices[4]. Then, for sufficiently large k, $lim_{t \to \infty} P(t) = P_\infty$ exists and approximates a permutation matrix that also minimizes the distance $\|PA_1 - A_2 P\|_F^2$. Moreover, the larger k yields the better P_∞ that approximates a permutation matrix.

Proof. See Theorem 3.9 in [35]. $\square$

Example 10. *In Figure 4.5, the adjacency matrices A_1 and A_2 corresponding the graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are as follows. And if we set the initial $P(0)$ to the identical matrix, one can have the response of P at time T as shown below.*

$$A_1 = \begin{bmatrix} 0 & 100 & 0 & 0 \\ 100 & 0 & 20 & 3 \\ 0 & 20 & 0 & 0 \\ 0 & 3 & 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 & 0 & 20 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 100 \\ 20 & 3 & 100 & 0 \end{bmatrix},$$
$$P(0) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; P(T) = \begin{bmatrix} 0.0002 & 0.0000 & 0.9997 & 0.0000 \\ -0.0002 & -0.0000 & -0.0000 & 0.9997 \\ 1.0000 & 0.0000 & 0.0001 & -0.0000 \\ 0.0000 & 1.0001 & 0.0000 & 0.0000 \end{bmatrix} \tag{4.20}$$

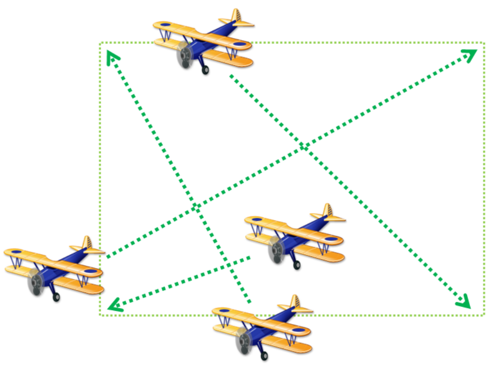
[3]The Frobenius norm is defined as $\|A\|_F = tr(AA^T)^{\frac{1}{2}}$ for $X \in \mathbb{R}^{n \times n}$.
[4]Suppose that $A = \{a_{ij}\}$ and $B = \{b_{ij}\}$. Then, $A \circ B = \{a_{ij} b_{ij}\}$.

where k is 100, and T is 1.

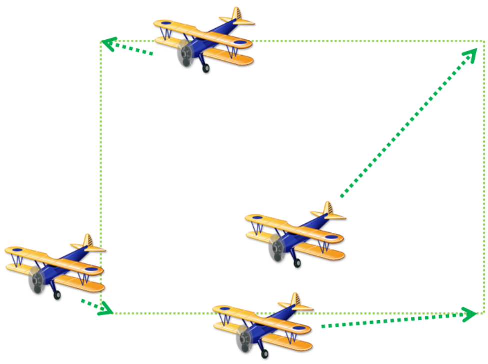
Remark 4.23. The first part of the equation (4.19), $P\left(P^T A_2 P A_1 - A_1 P^T A_2 P\right)$ is to minimize the graph matching cost. And the second part, $-kP\left((P \circ P)^T P - P^T (P \circ P)\right)$ is to make the matrix P be a permutation.

How to configure the graph $\mathcal{G}_1$ and $\mathcal{G}_2$

From this moment, we talk about the organization of A_1 and A_2 in the differential equation (4.19). Our goal is to take the benefit of the homogeneity of multiple agents in which all agents have the same ability in not only the motional features but also other features in the group. Hence, every agent can be replaced by others. While keeping this perspective, one can think about an idea of the nearness which is the formation assignment for the closest direction from the initial direction of an agent in the group to the final direction in the formation. Even though the agents in the formation can move freely to any direction so that the initial and final direction of the agent are not directly comparable due to the different references for the directions, the idea of the nearness is meaningful. Because, if each agent initially goes to the nearest direction in the sense of the minimizing the difference of included-angle matrices that would be discussed later, it will be eliminated needlessly wandering of each agent where the agent takes the further path in spite of the existence of the nearer path, and the collision of the agents will be decreasing.



(a) Bad



(b) Good

Figure 4.6: Assignments of aircrafts on the rectangular formation.

Example 11. *In Figure 4.6, the left picture illustrates a bad case in which each aircraft takes the poor path compared to the path for the case of the right picture. We can guess that this case causes increasing unwanted collisions and more time needed in order to be in the formation.*

Definition 4.24 (Included-Angle Graph)**.** Consider the collection of N homogeneous agents. For N position vectors $\psi_i \in \mathbb{R}^m$ for the agents, The weighted undirected graph $\mathcal{G}$ with N vertices is the **included-angle graph** if the graph $\mathcal{G}$ is complete and the each weight θ_{ij} of the edge between two vertices i and j is the included-angle of two vectors ψ_i^c and ψ_j^c as follows.

$$\begin{aligned}
\psi_i^c &= \psi_i - c, \\
\psi_j^c &= \psi_j - c, \\
\theta_{ij} &= \begin{cases} s\cos^{-1}\left(\frac{\psi_i^{cT}\psi_j^c}{\|\psi_i^c\|\|\psi_j^c\|}\right) & \text{if } \|\psi_i^c\|\|\psi_j^c\| \neq 0, \\ 0 & \text{otherwise} \end{cases}
\end{aligned} \tag{4.21}$$

where c is the average of $\psi_i, i \in [1, N]$ which will be the center of position vectors, and s is a scalar.

Definition 4.25 (Included-Angle Matrix Θ)**.** Consider a graph with N vertices. A $N \times N$ weighted adjacency matrix Θ of the included-angle graph is the **included-angle matrix** if

$$\begin{aligned}
\Theta =& \{\theta_{ij}\} \in \mathbb{R}^{N\times N}, \\
\theta_{ij} =& \text{the weight of the edge} \\
& \text{between vertices } i \text{ and } j \text{ like (4.21)}\ .
\end{aligned} \tag{4.22}$$

Example 12. *For the formation vector ζ in Example 9 corresponding Figure 4.4, if we select each ζ_i as the position vector ψ_i, then the included-angle matrix Θ is as follows.*

$$\Theta = \begin{bmatrix} 0 & 120 & 120 \\ 120 & 0 & 120 \\ 120 & 120 & 0 \end{bmatrix} \tag{4.23}$$

where $s = \frac{180}{\pi}$, and $c = \left[0, \frac{-2\sqrt{3}}{3}\right]^T$.

For the equation (4.19), let a graph $\mathcal{G}_1$ be the included-angle graph for the N subvectors ζ_i like (4.4) of the formation vector ζ as position vectors, and analogously let a graph $\mathcal{G}_2$ be the included-angle graph for the initial outputs of the agents as position vectors. If we respectively choose A_1 and A_2 as the included-angle matrices of the graph $\mathcal{G}_1$ and $\mathcal{G}_2$, then, we can get the dynamical system to achieve its solution P such that P_∞ approximates the permutation matrix that minimizes $\|PA_1 - A_2P\|_F^2$ through the equation (4.19). Therefore, once the error from an approximation of the permutation matrix is acceptable[5], the steady state solution P_∞ of the above system can be used for the permutation invariant formation tracking problem. Moreover, the matrix P_∞ minimizes the differences of the included angles.

Formation Vector for Actual Agents

From Theorem 4.22, because of a dynamical approach, the solution P has the transient response which is not exactly a permutation, and even in the steady state response, P could not be a permutation since P_∞ is an approximation of a permutation matrix. Thus, we should think about how to apply the matrix P for the formation tracking. Now, we simply suggest the formation vector ζ_v, which is designed for virtual agents, multiplied by P as the new formation vector ζ for actual agents as shown below.

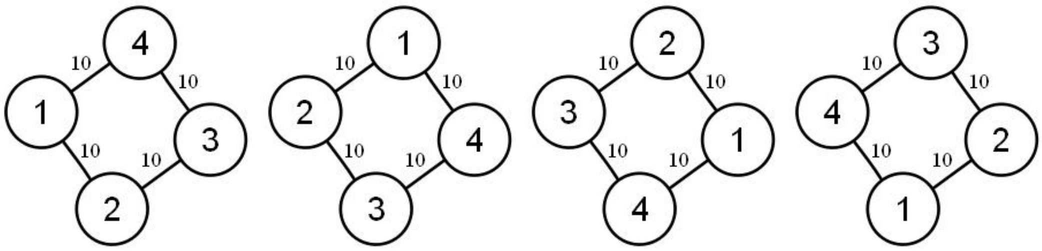
Definition 4.26 (Assignment Dynamics)**.** Consider the collection of N homogeneous agents, of which each has the m-dimensional output. Suppose that A_1 and A_2 are respectively the $N \times N$ included-angle matrices with respect to a graph $\mathcal{G}_1$ and $\mathcal{G}_2$ with N vertices. The graph $\mathcal{G}_1$ is the included-angle graph for the N subvectors $\zeta_{vi}(0)$ like (4.4) of the formation vector $\zeta_v(t)$ as position vectors, and the graph $\mathcal{G}_2$ is the included-angle graph for the initial outputs of the agents as position vectors. The following dynamical system is the **assignment dynamics** which maps the formation vector $\zeta_v(t)$ for virtual agents to the formation vector $\zeta(t)$ for actual ones.

$$\begin{aligned} \dot{P} &= P\left(P^T A_2 P A_1 - A_1 P^T A_2 P\right) - kP\left((P \circ P)^T P - P^T (P \circ P)\right) \\ \zeta(t) &= (P \otimes I_m)\,\zeta_v(t) \end{aligned} \tag{4.24}$$

[5] If P_∞ is not exactly a permutation matrix, we cannot actually say that it is a permutation invariant formation with P_∞ due to the approximation error which makes the assignment on a formation be confused. But, one can consider the relaxed matrix which is almost a permutation matrix such that its error can be arbitrarily reduced.

where $P(0) = I_N$ and k is a positive constant. Furthermore, we say that the matrix P minimizes **the nearness in the sense of the included-angle matching** of the agents in the group.

Remark 4.27. The nearness in the sense of the included-angle matching is distinguished from the physical closeness for an agent to get a position in the formation. In Figure 4.7, Graphs $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{G}_3$ and $\mathcal{G}_4$ are identical. For a weighted graph matching problem, these four graphs are the same class so that each graph trivially has the identical matrix as permutation matrix to the others. But, the formation vectors corresponding the graphs might be different from each other because a formation vector is not invariant under the rotation. Consequently, for a permutation invariant tracking problem, we can say that the permutation matrix P from the assignment dynamics (4.24) sequentially assigns the virtual agents in the group to the actual agents through one case of its class of which all formation vectors have the same included-angle matrices. In other words, we have the permutation matrix P as one of the matrices in the same class, and we don't know which one will be selected in the class by the assignment dynamics.



(a) Graph $\mathcal{G}_1$ (b) Graph $\mathcal{G}_2$ (c) Graph $\mathcal{G}_3$ (d) Graph $\mathcal{G}_4$

Figure 4.7: Identical graphs for a weighted graph matching problem.

Remark 4.28 (Transient Response of Assignment Dynamics). For the system (4.24), a Lyapunov function is given in [35] as follows.

$$V(P) = \frac{1}{2}\|PA_1 - A_2P\|_F^2 + k\frac{2}{3}tr\left(P^T\left(P - (P \circ P)\right)\right) \tag{4.25}$$

In such cases, $V(\dot{P})$ is non-increasing which means that P will converge to a local minimum that will be P_∞. In other words, P is continuously changing forward to minimize the cost such as (4.18). Therefore, the transient response of the system is still valuable to assign virtual agents on the formation besides the steady state response. This transient assignment will be shown as the morphing of the formation while the matrix P comes into effect as the weight for changing shape.

4.4.3 Integrated Tracking Control

Theorem 4.29. *Consider the collection of the agent systems* (4.9)*. Assume that f_i, g_i and h_i are respectively identical for $1 \leq \forall i \leq N$ that means homogeneous systems, and the Laplacian matrix L stands for an output sensing graph $\mathcal{G}$ of the agents in which each agent i can sense the outputs of its neighbors $\mathcal{N}_i$. For a formation vector ζ_v, if*

- *the agent is globally state feedback input-output linearizable,*
- *each output of agents is absolutely measurable by itself,*
- *the tracking dynamics of the agent is bounded input bounded state stable,*
- *the directed graph $\mathcal{G}$ of the Laplacian L is strongly connected or has a single leader component,*
- *and $s^{\rho_j} + \lambda_i k_{j\rho_j} s^{\rho_j - 1} + \cdots + \lambda_i k_{j1}$ are Hurwitz polynomials for $\forall \lambda_i \in \lambda(L)$, such that $\lambda_i \neq 0$, $1 \leq \forall j \leq m$,*

then, the permutation invariant formation tracking problem is approximately solvable with the additional assignment dynamics. And, the controller u_i would be as shown below.

$$
\begin{aligned}
&\dot{P} = P\left(P^T A_2 P A_1 - A_1 P^T A_2 P\right) - kP\left((P \circ P)^T P - P^T (P \circ P)\right) \\
&\zeta = (P \otimes I_m)\,\zeta_v \\
&u_i = -\mathcal{A}^{-1}(x_i)\left(L_f^{\rho} h(x_i) - \left(\frac{1}{|\mathcal{N}_i|}\sum_{l\in M} K_l \sum_{j\in\mathcal{N}_i} (y_i - y_j - \zeta_i + \zeta_j)^{(l-1)} + \zeta_i^{(\rho)}(t)\right)\right)
\end{aligned}
\tag{4.26}
$$

where the matrix A_1 and A_2 are defined as Definition 4.26, $P(0) = I_N$, k is a positive constant. And, ρ, M, $L_f^{\rho} h(\cdot)$, K_l, $\mathcal{A}$ and $\zeta_i^{(\rho)}$ are like (4.13)*. Moreover,* *each agent goes through the nearest path in the sense of the included-angle matching of the agents.*

Proof. By Theorem 4.14, the collection of multiple agents would be in the formation which is represented by the formation vector ζ. And, $\lim_{t\to\infty} P = P_\infty$ approximates the permutation matrix which maps the formation vector ζ_v to ζ. From Theorem 4.22, we can arbitrarily reduce the approximation error by choosing larger k. Consequently, we can resolve the permutation invariant formation problem remaining a permutation error which appears as a distortion of the formation in the steady states while minimizing the nearness in the sense of the included-angle matching by Definition 4.26. □

Remark 4.30. The controller u_i in (4.26) is a decentralized one that means it is sufficient that each agent i only knows the information about its neighborhoods $\mathcal{N}_i$. For a permutation invariant formation tracking, however, every agents should be aware of the global knowledge. For instance, each agent builds A_1 and A_2 matrices to be assigned to the virtual agent at initial time. But, these kinds of global information are disposable after setup of an assignment dynamics. It is undesired that every agents keep the maintenance of the global knowledge. Therefore, the decentralized control is also valid for a permutation invariant formation tracking problem as well as a formation tracking problem.

Until now, we have studied about a formation. The word of the formation appears in a variety of fields. Generally, the formation indicates the shape of a group to collaborate a common mission. In this chapter, we mathematically defined what the formation is, and we introduced the formation tracking problem. To resolve this problem, we quantified the formation by the formation matrix and vector. And the output tracking by relative references presented in the previous chapter was applied to this problem. We also discussed the permutation of agents. If the agents in the group are homogeneous, each agent can be replaced by others. For taking the advantage of the homogeneity, a weighted graph matching problem was employed and a dynamical approach was introduced as its solver. Finally, we described the nearness with the included-angle concept, and integrated the formation tracking problem and the weighted graph matching problem into the permutation invariant formation tracking problem.

Chapter 5

Example

5.1 Multiple Homogeneous Nonholonomic Mobile Robots

5.1.1 Modeling a Mobile Robot

Consider the following simple kinematic model of a mobile robot described in Figure 5.1 that shows a vehicle with two tires subject to the constraint of the wheels which is allowed to roll and spin, but not to slip.

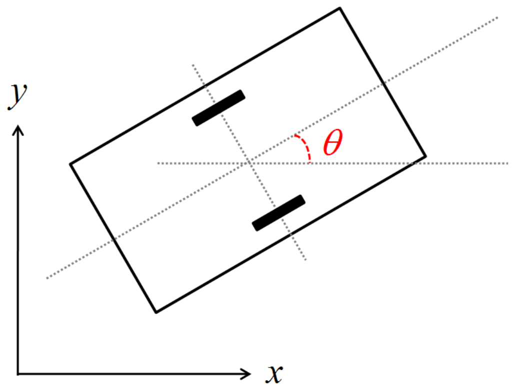


Figure 5.1: Kinematic model of a mobile robot.

The constraint restricts the wheels such that the velocity of their sideways is to be zero as shown below.

$$\dot{x}\sin\theta - \dot{y}\cos\theta = 0 \tag{5.1}$$

where x and y are the configuration of the robot by the xy location of the wheels, θ is the angle of the body with respect to the horizontal.

The equation (5.1) is a kind of **Pfaffian constraint** that is composed of velocity conditions expressed by $A(q)\dot{q} = 0$ where $q \in$ configuration space $\mathbb{Q}$, $A(q) \in \mathbb{R}^{k\times n}$ represents a set of k velocity constraints. It is said to be **integrable** if there exists a vector-valued function $h : \mathbb{Q} \to \mathbb{R}^k$ such that $A(q)\dot{q} = 0 \iff \frac{\partial h}{\partial q}\dot{q} = 0$. This h can be represented locally as **algebraic constraints** on the configuration space like $h_i(q) = 0$, $i \in [1, k]$ that is said to be **holonomic constraint**; the motion of system is restricted to **a smooth hypersurface** in the (unconstrained) $\mathbb{Q}$. Otherwise, it is **not integrable** if there is no h satisfying the above condition. In such cases, the constraint is **nonholonomy**; the instantaneous velocities of the system are constrained to an $n - k$ dimensional subspace, but the set of reachable configuration is not restricted to some $n - k$ dimensional hypersurface[1].

From a Pfaffian constraints (5.1), we can make it possible that converting the constraints to the state space equation (5.2) through taking an adequate variable substitution. Generally, the conversion for a control system is to select null vectors v_i from the null space of $A(q)$ such that $A(q)\dot{q} = 0$, with which we have the differential equation like $\dot{q} = v_1u_1 + \ldots + v_ku_k$.

For the constraints (5.1) of the mobile robot, suppose that u_1 and u_2 is input variables for a control system. If we respectively choose u_1 and u_2 as the driving velocity and the steering velocity, then, one can have the dynamical system as shown below.

$$\begin{aligned} \dot{x} &= u_1\cos\theta \\ \dot{y} &= u_1\sin\theta \\ \dot{\theta} &= u_2 \end{aligned} \tag{5.2}$$

where u_1 is the velocity of the wheels, and u_2 is the velocity of the angle of the wheels.

[1]For more detailed explanation, see [21, 27].

5.1.2 Input-output feedback linearization

For state feedback input-output linearization, we cannot directly get the well defined the vector relative degree of the system (5.2). In order to achieve the relative degree, we employ a dynamic extension [12] while introducing the acceleration of the wheels u_3 as new input variable as follows.

$$\begin{aligned}\dot{x} &= u_1 \cos\theta \\ \dot{y} &= u_1 \sin\theta \\ \dot{\theta} &= u_2 \\ \dot{u_1} &= u_3\end{aligned} \tag{5.3}$$

Consider the dynamical system (5.3). We investigate the vector relative degree. Let

$$\begin{aligned} f(\mathbf{x}) &= \begin{bmatrix} u_1\cos\theta \\ u_1\sin\theta \\ 0 \\ 0 \end{bmatrix}, \quad g(\mathbf{x}) = \begin{bmatrix} | & | \\ g_1(\mathbf{x}) & g_2(\mathbf{x}) \\ | & | \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \\ h(\mathbf{x}) &= [h_1(\mathbf{x}), h_2(\mathbf{x})]^T = [x, y]^T \end{aligned} \tag{5.4}$$

where $\mathbf{x} = [x, y, \theta, u_1]^T$ and $\mathbf{u} = [u_2, u_3]^T$.

Remark 5.1. The system (5.3) is rewritten as the simple form as shown below.

$$\begin{aligned}\dot{\mathbf{x}} &= f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} \\ \mathbf{y} &= h(\mathbf{x})\end{aligned} \tag{5.5}$$

Now, we have the global vector relative degree $\rho = [\rho_1, \rho_2]^T = [2, 2]^T$ if $u_1 \neq 0$. Because,

$$\begin{aligned} & L_{g_j} h_k(\mathbf{x}) = 0, \qquad 1 \leq \forall j, k \leq 2, \\ & \mathcal{A}(\mathbf{x}) = \begin{bmatrix} L_{g_1} L_f h_1(\mathbf{x}) & L_{g_2} L_f h_1(\mathbf{x}) \\ L_{g_1} L_f h_2(\mathbf{x}) & L_{g_2} L_f h_2(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} -u_1 \sin\theta & \cos\theta \\ u_1 \cos\theta & \sin\theta \end{bmatrix} \end{aligned} \tag{5.6}$$

where $\mathcal{A}(\cdot)$ is nonsingular except $u_1 = 0$. When $u_1 = 0$ with the proposed controller discussed in earlier chapters, the mobile robot is only stuck at the position where the robot is in the formation. Thus, we don't need to worry about this singularity via no control input at that position.

The system (5.3) is globally state feedback input-output linearizable without zero dynamics since the sum of ρ_1, ρ_2 is the dimension of $\mathbf{x}$. Therefore, we can keep away the boundedness of the tracking dynamics for the condition of Theorem 4.14 and 4.29.

5.1.3 Multiple Homogeneous Mobile Robots

At this moment, we consider $N = 6$ agents of multiple homogeneous mobile robots of which each agent has the identical dynamical system as shown below, and the output of the agent is absolutely measurable by itself[2].

$$\begin{aligned} \dot{\mathbf{x}}_i &= f(\mathbf{x}_i) + g(\mathbf{x}_i)\mathbf{u}_i \\ \mathbf{y}_i &= h(\mathbf{x}_i) \end{aligned} \tag{5.7}$$

for all $i \in [1, 6]$ where $\mathbf{x}_i = [x_i, y_i, \theta_i, u_{i1}]^T$, $\mathbf{u}_i = [u_{i2}, u_{i3}]^T$ and the vector fields f, g, h are like (5.4).

5.1.4 Formation Configuration

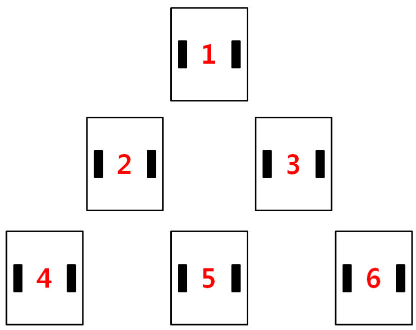


Figure 5.2: Triangular formation of 6 multiple homogeneous agents. The number on body is the index of the agent.

[2] It is nothing but the guarantee of state feedback input-output linearization for the agent without help from the others

Formation Matrix and Vector

$$F(t) = 50 \begin{bmatrix} 0 & 2+\sin(t) & -2-\sin(t) & 4+2\sin(t) & 0 & -4-2\sin(t) \\ 0 & 4+2\sin(t) & 4+2\sin(t) & 8+4\sin(t) & 8+4\sin(t) & 8+4\sin(t) \\ -2-\sin(t) & 0 & -4-2\sin(t) & 2+\sin(t) & -2-\sin(t) & -6-3\sin(t) \\ -4-2\sin(t) & 0 & 0 & 4+2\sin(t) & 4+2\sin(t) & 4+2\sin(t) \\ 2+\sin(t) & 4+2\sin(t) & 0 & 6+3\sin(t) & 2+\sin(t) & -2-\sin(t) \\ -4-2\sin(t) & 0 & 0 & 4+2\sin(t) & 4+2\sin(t) & 4+2\sin(t) \\ -4-2\sin(t) & -2-\sin(t) & -6-3\sin(t) & 0 & -4-2\sin(t) & -8-4\sin(t) \\ -8-4\sin(t) & -4-2\sin(t) & -4-2\sin(t) & 0 & 0 & 0 \\ 0 & 2+\sin(t) & -2-\sin(t) & 4+2\sin(t) & 0 & -4-2\sin(t) \\ -8-4\sin(t) & -4-2\sin(t) & -4-2\sin(t) & 0 & 0 & 0 \\ 4+2\sin(t) & 6+3\sin(t) & 2+\sin(t) & 8+4\sin(t) & 4+2\sin(t) & 0 \\ -8-4\sin(t) & -4-2\sin(t) & -4-2\sin(t) & 0 & 0 & 0 \end{bmatrix},$$

$$\zeta(t) = \begin{bmatrix} \zeta_1(t) \\ \zeta_2(t) \\ \zeta_3(t) \\ \zeta_4(t) \\ \zeta_5(t) \\ \zeta_6(t) \end{bmatrix}, \qquad \begin{aligned} \zeta_1(t) &= [0, 0]^T, \\ \zeta_2(t) &= [-100 - 50\,\sin(t), -200 - 100\,\sin(t)]^T, \\ \zeta_3(t) &= [100 + 50\,\sin(t), -200 - 100\,\sin(t)]^T, \\ \zeta_4(t) &= [-200 - 100\,\sin(t), -400 - 200\,\sin(t)]^T, \\ \zeta_5(t) &= [0, -400 - 200\,\sin(t)]^T, \\ \zeta_6(t) &= [200 + 100\,\sin(t), -400 - 200\,\sin(t)]^T \end{aligned} \tag{5.8}$$

Our goal is that the collection of the 6 agents will be in the formation described by (5.8). As shown above, the formation is varying with respect to time and the shape of which is triangular.

5.1.5 Connectivity of Agents

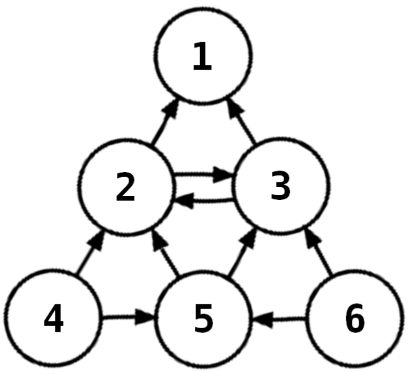


Figure 5.3: Output sensing graph of the agents for Laplacian matrix (5.10). This graph represents that the agent i can sense the outputs of its neighbors $\mathcal{N}_i$. The agent 1 is the single leader of the group by the connectivity of the graph.

Figure 5.3 describes the connectivity of the mobile robot group. Each agent i can only sense the outputs of its neighbor $\mathcal{N}_i$ defined in (5.9). Thus, the individual agent has not to know all outputs of the agents which induces the formation controller to be decentralized. And Figure 5.3 shows that there is one leader component which is determined by the connectivity of the graph corresponding its Laplacian matrix (5.10) as we already discussed in Section 3.4.

$$\begin{aligned} &\mathcal{N}_1 = \{\}, &&\mathcal{N}_2 = \{1,3\}, &&\mathcal{N}_3 = \{1,2\}, \\ &\mathcal{N}_4 = \{2,5\}, &&\mathcal{N}_5 = \{2,3\}, &&\mathcal{N}_6 = \{3,5\} \end{aligned} \tag{5.9}$$

$$L = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{2} & 1 & -\frac{1}{2} & 0 & 0 & 0 \\ -\frac{1}{2} & -\frac{1}{2} & 1 & 0 & 0 & 0 \\ 0 & -\frac{1}{2} & 0 & 1 & -\frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & -\frac{1}{2} & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{2} & 0 & -\frac{1}{2} & 1 \end{bmatrix}, \qquad \lambda(L) = \{0, 0.5, 1, 1.5\} \tag{5.10}$$

5.1.6 Decentralized Formation Controller

Now, we examine the gain of the controller. If we choose

$$k_{11} = 300, \quad k_{12} = 50, \quad k_{21} = 300, \quad k_{22} = 50, \tag{5.11}$$

then the following equations are guaranteed to be Hurwitz polynomials.

$$s^{\rho_j} + \lambda_i k_{j\rho_j} s^{\rho_j - 1} + \cdots + \lambda_i k_{j1} \tag{5.12}$$

where $\forall \lambda_i \in \lambda(L)$ such that $\lambda_i \neq 0$ and $1 \leq \forall j \leq m = 2$. Therefore, by Theorem 4.14, we finally achieve the decentralized formation controller $\mathbf{u}_i$ as shown below.

$$\mathbf{u}_i = -\mathcal{A}^{-1}(\mathbf{x}_i) \left(L_f^{\rho} h(\mathbf{x}_i) - \left(\frac{1}{|\mathcal{N}_i|} \sum_{l \in M} K_l \sum_{j \in \mathcal{N}_i} (\mathbf{y}_i - \mathbf{y}_j - \zeta_i + \zeta_j)^{(l-1)} + \zeta_i^{(\rho)}(t) \right) \right) \tag{5.13}$$

where

$$
\begin{aligned}
\mathcal{A}^{-1}(\mathbf{x}_i) &= \begin{bmatrix} -\frac{1}{u_{i1}}\sin\theta_i & \frac{1}{u_{i1}}\cos\theta_i \\ \cos\theta_i & \sin\theta_i \end{bmatrix}, \\
L_f^{\rho} h(\mathbf{x}_i) &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \qquad \zeta_i^{(\rho)}(t) = \begin{bmatrix} \zeta_{i1}^{(2)}(t) \\ \zeta_{i2}^{(2)}(t) \end{bmatrix}, \\
K_1 &= \begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}, \qquad K_2 = \begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}, \\
M &= \{1, 2\}.
\end{aligned}
\tag{5.14}
$$

5.1.7 Permutation Invariant Formation

In this subsection, we consider the permutation invariant formation control talked about in Section 4.4. For the formation (5.8), if the initial states of the agents are given like (5.16), we have the included angle matrices A_1 and A_2 as follows.

$$
\begin{aligned}
A_1 &= \begin{bmatrix}
0 & 56.3099 & 56.3099 & 123.6901 & 180.0000 & 123.6901 \\
56.3099 & 0 & 112.6199 & 67.3801 & 123.6901 & 180.0000 \\
56.3099 & 112.6199 & 0 & 180.0000 & 123.6901 & 67.3801 \\
123.6901 & 67.3801 & 180.0000 & 0 & 56.3099 & 112.6199 \\
180.0000 & 123.6901 & 123.6901 & 56.3099 & 0 & 56.3099 \\
123.6901 & 180.0000 & 67.3801 & 112.6199 & 56.3099 & 0
\end{bmatrix}, \\
A_2 &= \begin{bmatrix}
0 & 42.1485 & 153.9344 & 147.1317 & 31.1422 & 179.0182 \\
42.1485 & 0 & 163.9171 & 104.9832 & 73.2907 & 138.8333 \\
153.9344 & 163.9171 & 0 & 58.9338 & 122.7923 & 25.0838 \\
147.1317 & 104.9832 & 58.9338 & 0.0000 & 178.2739 & 33.8501 \\
31.1422 & 73.2907 & 122.7923 & 178.2739 & 0.0000 & 147.8760 \\
179.0182 & 138.8333 & 25.0838 & 33.8501 & 147.8760 & 0.0000
\end{bmatrix}
\end{aligned}
\tag{5.15}
$$

where $s = \frac{180}{\pi}$ and the initial states of the mobile robots are like

$$
\begin{aligned}
\mathbf{x}_1(0) &= \left[0, 0, \frac{90}{180}\pi, 150\right]^T, & \mathbf{x}_2(0) &= \left[-150, -900, \frac{-70}{180}\pi, 70\right]^T, \\
\mathbf{x}_3(0) &= \left[250, -1600, \frac{-60}{180}\pi, 100\right]^T, & \mathbf{x}_4(0) &= \left[-300, -1500, \frac{30}{180}\pi, 70\right]^T, \\
\mathbf{x}_5(0) &= \left[150, -800, \frac{90}{180}\pi, 20\right]^T, & \mathbf{x}_6(0) &= \left[-5, -1550, \frac{-120}{180}\pi, 80\right]^T
\end{aligned}
\tag{5.16}
$$

Consequently, we can resolve the permutation invariant tracking control problem by Theorem 4.29 with the additional assignment dynamics as shown below while ζ_v is substituted with ζ of (5.8) and the new ζ of (5.17) is used as the new formation vector of the controller (5.13).

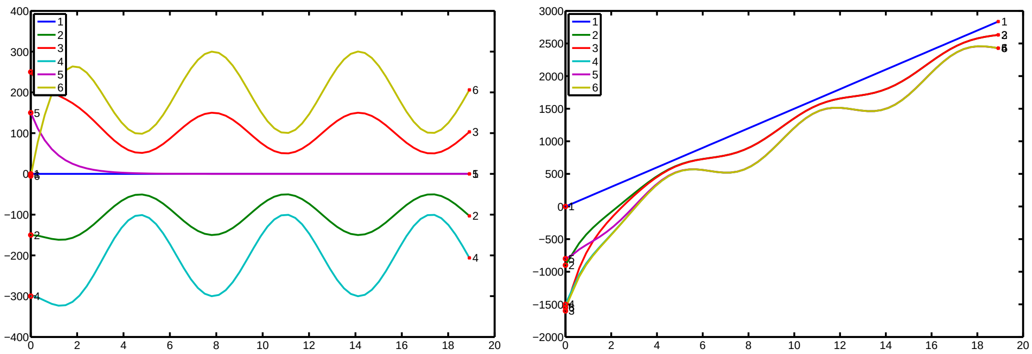
$$\begin{aligned}\dot{P} &= P\left(P^T A_2 P A_1 - A_1 P^T A_2 P\right) - kP\left((P \circ P)^T P - P^T (P \circ P)\right) \\ \zeta &= \left(P \otimes I_m\right)\zeta_v\end{aligned} \tag{5.17}$$

where $P(0) = I_6$ and $k = 1000$.

5.1.8 Simulation Results

All results in this subsection were simulated with the initial values of (5.16) on MATLAB.

Decentralized Formation Tracking Control



(a) x trajectories (b) y trajectories

Figure 5.4: x, y trajectories of the agents. ‘o’ indicates the initial positions of the agents and ‘x’ does the final positions. The number of each line is the index of the agent.

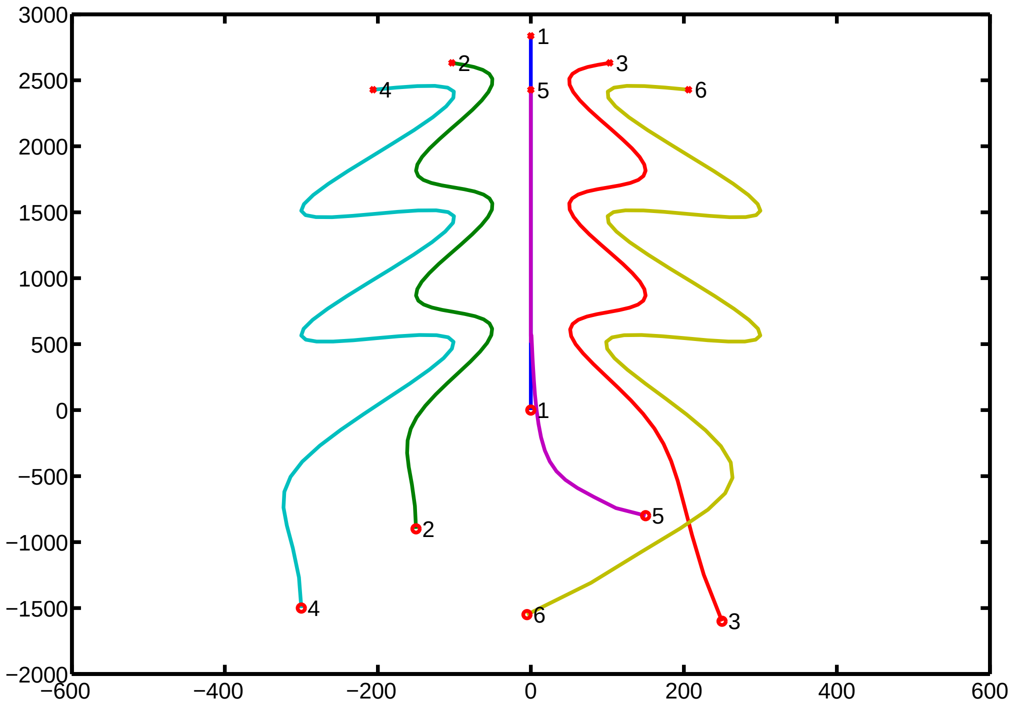


Figure 5.5: Decentralized formation tracking control for (5.13). 'o' indicates the initial positions of the agents and 'x' does the final positions. The number of each line is the index of the agent.

Permutation Invariant Formation Tracking Control

The output of assignment dynamics (5.17) at time 50 was as shown below.

$$P(50) = \begin{bmatrix} 1.0019 & -0.0259 & -0.0495 & 0.0434 & 0.0008 & 0.0816 \\ 0.0284 & 1.0048 & -0.0207 & 0.0199 & -0.0273 & -0.0245 \\ -0.0678 & 0.0153 & -0.0299 & 0.0181 & 0.0455 & 1.0020 \\ -0.0353 & -0.0114 & 0.0173 & 0.9776 & 0.0460 & -0.0264 \\ 0.0504 & 0.0265 & 1.0030 & -0.0048 & -0.0616 & 0.0264 \\ 0.0081 & 0.0235 & 0.0720 & -0.0360 & 1.0219 & -0.0386 \end{bmatrix} \tag{5.18}$$

which approximated the following matrix P_∞.

$$P_\infty = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{5.19}$$

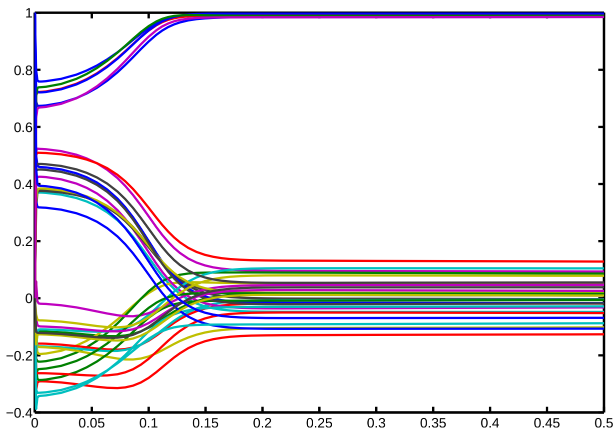
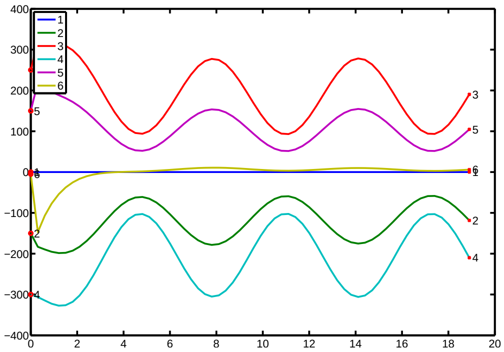
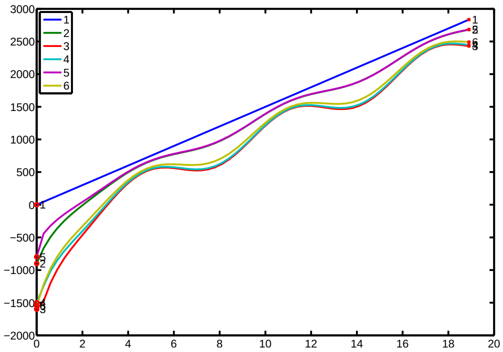


Figure 5.6: Evaluation of the elements of the output matrix P for the assignment dynamics (5.17) during 0.5 time.



(a) x trajectories



(b) y trajectories

Figure 5.7: x, y trajectories of the agents. 'o' indicates the initial positions of the agents and 'x' does the final positions. The number of each line is the index of the agent.

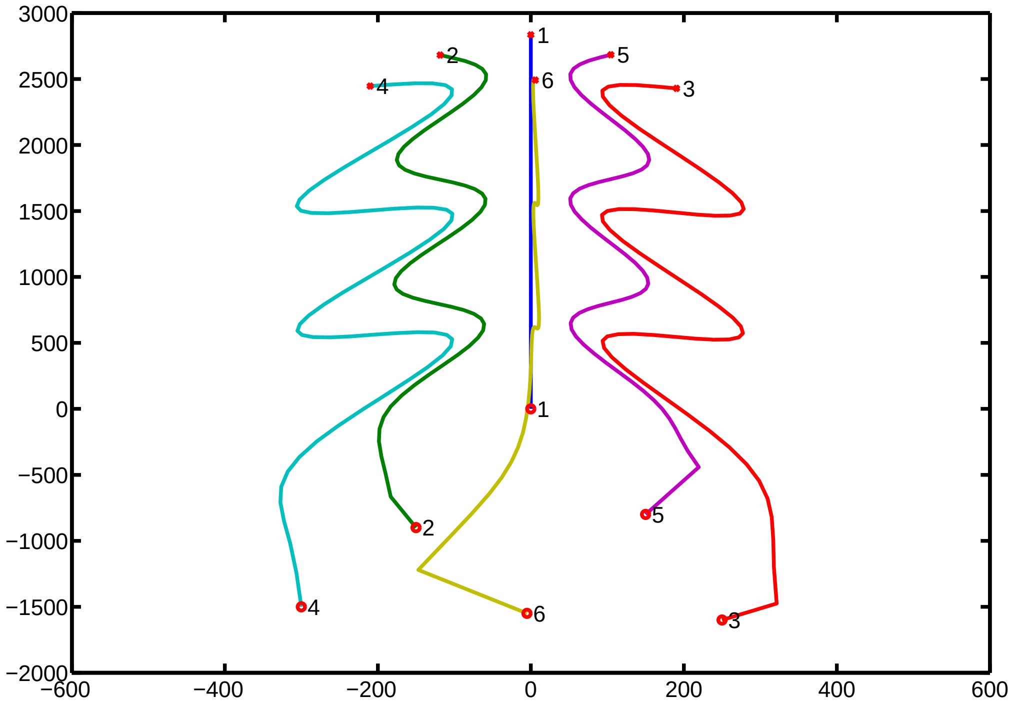


Figure 5.8: Permutation invariant formation tracking control. 'o' indicates the initial positions of the agents and 'x' does the final positions. The number of each line is the index of the agent.

Chapter 6

Conclusion

In this thesis, we have considered the decentralized formation tracking control of multiple homogeneous agents. By the proposed control scheme, the collection of the agents keeps the formation while the flock of agents can freely move anywhere.

We first discussed the output tracking problem and extended it for multiple homogeneous systems. The output tracking problem of multiple agents by relative references is that the differences of the outputs among the agents keep tracking given references. In order to resolve it, we introduced graph theory. The properties of the eigenvalue and the null space of Laplacian adjacency matrix play a key role to explain the stability of tracking control. Thus, we analyzed the condition of stability with those properties and developed the tracking controller where Laplacian matrix corresponding the connectivity of the agents is strongly connected or has a single leader component.

Formation generally are described as a configuration of multiple agents with respect to positions and locations while aiming to coordinate a shared task among the agents. To deal with the formation, we mathematically defined formation matrix and vector, and we formulated the formation tracking problem in which we utilized the output tracking framework for multiple agents as the control scheme and the formation vector as the relative references. We also mentioned the permutation of the agents to take the advantage of homogeneity of the agents. And the weighted graph matching problem was considered to characterize the permutation with the concept of the nearness in the sense of the included angle matching. Finally, we verified our studies by the simulation of 6 agents of homogeneous nonholonomic mobile robots.

Throughout this thesis, we have treated a fixed number of the agents and time-invariant interconnection of the agents. The ultimate purpose of this research effort remains to be challenged in the case of undetermined number of the agents. For instance, there are many of going in and out cars in highway systems. Hence, we cannot guarantee our approach to be directly applicable to this kind of problems. To tackle this case, we might think about tiling of the road and deal with the hierarchical collection of the tiles. And for another example, we can suppose the bundle of cheap robots which are easily damageable. Because each robot is defective from unexpected environment, we should take into account supplement of new robots.

So far, we have examined the formation on multiple agents. While we close the last page, it is to be hoped that this thesis will contribute to reveal the coordination of complicated multiple systems in the nature.

Appendix A

Matlab Code

A.1 Example in Chapter 5

linearization.m

```
syms X x y th u1 u2 u3
f=[u1*cos(th), u1*sin(th) , 0, 0].';
g1=[0 0 1 0].';     g2=[0 0 0 1].';
h1=x;               h2=y;       h=[h1 h2].';
X=[x y th u1].';

Lg1Lfh1 = jacobian((jacobian(h1,X)*f),X)*g1;
Lg2Lfh1 = jacobian((jacobian(h1,X)*f),X)*g2;
Lg2Lfh2 = jacobian((jacobian(h2,X)*f),X)*g2;
Lg1Lfh2 = jacobian((jacobian(h2,X)*f),X)*g1;
A=[Lg1Lfh1 Lg2Lfh1; Lg1Lfh2 Lg2Lfh2];

L2fh1 = jacobian((jacobian(h1,X)*f),X)*f;
L2fh2 = jacobian((jacobian(h2,X)*f),X)*f;
Lfh=[L2fh1; L2fh2];
```

feedbackLinearizer.m

```
function feedbackLinearizer(block)
    setup(block);

function setup(block)
    %% Register number of dialog parameters
    block.NumDialogPrms = 1;
    %% Register number of input and output ports
    block.NumInputPorts  = 1;
    block.NumOutputPorts = 1;
    %% Setup functional port properties to dynamically
    %% inherited.
    block.SetPreCompInpPortInfoToDynamic;
    block.SetPreCompOutPortInfoToDynamic;
    block.InputPort(1).Dimensions        = 6;
    block.InputPort(1).DirectFeedthrough = false;
    block.OutputPort(1).Dimensions       = 2;
    %% Set block sample time to continuous
```

```
    block.SampleTimes = [0 0];
    %% Setup Dwork
    block.NumContStates = 0;
    %% Register methods
    block.RegBlockMethod('Outputs', @Output);

function Output(block)
    v1 = block.InputPort(1).Data(1);  v2 =  block.InputPort(1).Data(2);
    x =  block.InputPort(1).Data(3);  y =  block.InputPort(1).Data(4);
    th = block.InputPort(1).Data(5);  u1 = block.InputPort(1).Data(6);

    if (u1 == 0)
        invA= [0 0;0 0];
    else
        invA =[  -sin(th)/u1    cos(th)/u1;     cos(th)   sin(th)];
    end
    Lfh=[0;0];
    block.OutputPort(1).Data =  -invA*Lfh+ invA*[v1; v2];
```

mobileRobot.m

```
function mobileRobot(block)
    setup(block);

function setup(block)
    %% Register number of dialog parameters
    block.NumDialogPrms = 1;
    %% Register number of input and output ports
    block.NumInputPorts  = 1;
    block.NumOutputPorts = 1;
    %% Setup functional port properties to dynamically
    %% inherited.
    block.SetPreCompInpPortInfoToDynamic;
    block.SetPreCompOutPortInfoToDynamic;
    block.InputPort(1).Dimensions        = 2;
    block.InputPort(1).DirectFeedthrough = false;
    block.OutputPort(1).Dimensions       = 4;
    %% Set block sample time to continuous
    block.SampleTimes = [0 0];
    %% Setup Dwork
    block.NumContStates = 4;
    %% Register methods
    block.RegBlockMethod('InitializeConditions',    @InitConditions);
    block.RegBlockMethod('Outputs',                 @Output);
    block.RegBlockMethod('Derivatives',             @Derivative);

function InitConditions(block)
    %% Initialize Dwork
    block.ContStates.Data = block.DialogPrm(1).Data;

function Output(block)
    x=block.ContStates.Data(1);
    y=block.ContStates.Data(2);
    th=block.ContStates.Data(3);
    u1=block.ContStates.Data(4);
    block.OutputPort(1).Data = [ x y th u1 ]';

function Derivative(block)
    u2 =  block.InputPort(1).Data(1);
    u3 =  block.InputPort(1).Data(2);
    th =  block.ContStates.Data(3);
    u1 =  block.ContStates.Data(4);
    block.Derivatives.Data = [  u1*cos(th);   u1*sin(th);     u2;   u3];
```

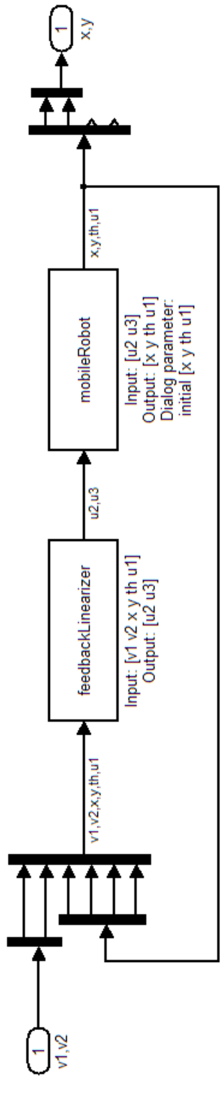


Figure A.1: State Feedback Input-Output Linearized Mobile Robot.

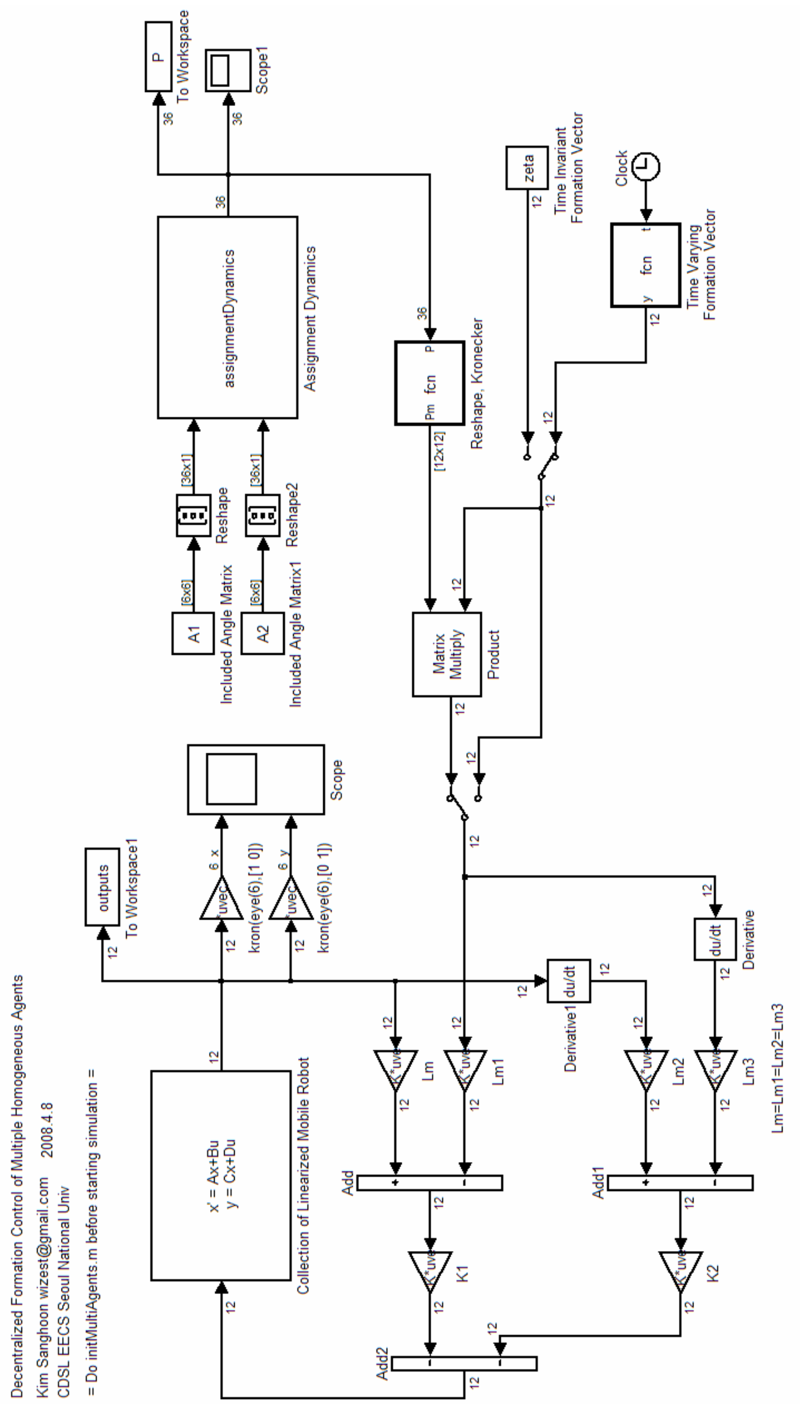


Figure A.2: Matlab Implementation of Formation Tracking Control.

makeIncludedAngleMatrix.m

```
function A = makeIncludedAngleMatrix(v)
v = reshape(v,2,length(v)/2);
c = mean(v.').';    % center
l=length(v);
A=zeros(l,l);
s=180/pi;
for i=1:l
    for j=1:l
        psi=v(:,i)-c;   psj=v(:,j)-c;
        if (norm(psi) * norm(psj) == 0)
            A(i,j) = 0;
        else
            val= (psi'*psj) / (norm(psi)*norm(psj));
            A(i,j)=s*real(acos(val));
        end
    end
end
```

toMatrix.m

```
function X=toMatrix(x)
n=sqrt(length(x));
X=reshape(x,[n n]);
```

toVector.m

```
function x=toVector(X)
[i j]=size(X);
x=reshape(X,[i*j 1]);
```

initMultiAgents.m

```
clear all;

zeta=[0 0 -100 -200 100 -200 -200 -400 0 -400 200 -400].';
k11=300;    k12=50;     k21=300;    k22=50;
K1=kron(eye(6),[k11 0;0 k12]);
K2=kron(eye(6),[k21 0;0 k22]);

L=[ 0 0 0 0 0 0;            -.5 1 -.5 0 0 0;   -.5 -.5 1 0 0 0;
    0 -.5 0 1 -.5 0         0 -.5 -.5 0 1 0    0 0 -.5 0 -.5 1];
Lm=kron(L,eye(2));

Ai=[0 0 1 0; 0 0 0 1; 0 0 0 0; 0 0 0 0];    Bi=[0 0; 0 0; 1 0; 0 1];
Ci=[1 0 0 0; 0 1 0 0];                      Di=[0 0; 0 0];
A=kron(eye(6),Ai);  B=kron(eye(6),Bi);
C=kron(eye(6),Ci);  D=kron(eye(6),Di);

% initial values of mobile robots
z01 =[ 0 0 (150)*cos(90/180*pi) (150)*sin(90/180*pi)];
z02 =[ -150 -900 (70)*cos(-70/180*pi) (70)*sin(-70/180*pi)];
z03 =[ 250 -1600 (100)*cos(-60/180*pi) (100)*sin(-60/180*pi)];
z04 =[ -300 -1500  (70)*cos(30/180*pi) (70)*sin(30/180*pi)];
z05 =[ 150 -800  (20)*cos(90/180*pi) (20)*sin(90/180*pi)];
z06 =[ -5 -1550  (80)*cos(-120/180*pi) (80)*sin(-120/180*pi)];
initZ=[z01, z02, z03, z04, z05, z06].';

% assignment dynamics
% A1 from formation vector, A2 from initial outputs
A1=makeIncludedAngleMatrix(zeta);
```

```
    tmp=reshape(initZ,4,6);
    tmp=reshape(tmp(1:2,:),12,1);
A2=makeIncludedAngleMatrix(tmp);
```

assignmentDynamics.m

```
function assignmentDynamics(block)
    setup(block);

function setup(block)
    %% Register number of dialog parameters
        block.NumDialogPrms = 0;
    %% Register number of input and output ports
    block.NumInputPorts  = 2;
    block.NumOutputPorts = 1;
    %% Setup functional port properties to dynamically
    %% inherited.
    block.SetPreCompInpPortInfoToDynamic;
    block.SetPreCompOutPortInfoToDynamic;
    block.InputPort(1).Dimensions        = 6^2;
    block.InputPort(1).DirectFeedthrough = false;
    block.InputPort(2).Dimensions        = 6^2;
    block.InputPort(2).DirectFeedthrough = false;
    block.OutputPort(1).Dimensions       = 6^2;
    %% Set block sample time to continuous
    block.SampleTimes = [0 0];
    %% Setup Dwork
    block.NumContStates = 6^2;
    %% Register methods
    block.RegBlockMethod('InitializeConditions',    @InitConditions);
    block.RegBlockMethod('Outputs',                 @Output);
    block.RegBlockMethod('Derivatives',             @Derivative);

function InitConditions(block)
    block.ContStates.Data = toVector(eye(6));

function Output(block)
    block.OutputPort(1).Data = block.ContStates.Data;

function Derivative(block)
    A1= toMatrix(block.InputPort(1).Data);
    A2= toMatrix(block.InputPort(2).Data);
    P = toMatrix(block.ContStates.Data);
    k= 1000;
    dP = P*(P'*A2*P*A1 -A1*P'*A2*P) -k*P*((P.*P)'*P -P'*(P.*P));
    block.Derivatives.Data = toVector(dP);
```

Block: Time Varying Formation Vector

```
function y = fcn(t)
zeta=[  0;                    0;
        -100-50*sin(t);     -200-100*sin(t);
        100+50*sin(t);      -200-100*sin(t);
        -200-100*sin(t);     -400-200*sin(t);
        0;                   -400-200*sin(t);
        200+100*sin(t);      -400-200*sin(t)];
y = zeta;
```

Bibliography

[1] *California partners for advenced transit and highways*, `http://www.path.berkeley.edu`, accessed on Feb 2008.

[2] *Graph theory*, wikipedia.org `http://en.wikipedia.org/wiki/Graph_theory`, accessed on Feb 2008.

[3] M. Amano, Z. W. Luo, and S. Hosoe, *Graph dependent sufficient conditions for synchronization of dynamic network system with time-delay*, Proc. IFAC Workshop on Time Delay Systems (2003).

[4] Tucker Balch and Ronald C. Arkin, *Behavior-based formation control for multi-robot teams*, IEEE Trans. on Robotics and Automation (1998).

[5] R. W. Brockett, *Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems*, Linear Algebra and Its Applications (1991).

[6] Diestel and Reinhard, *Graph theory 2nd edition*, Springer, 2000.

[7] J. Alexander Fax, *Optimal and cooperative control of vehicle formations*, Ph.D. thesis, California Institute of Technology, 2002.

[8] J. Alexander Fax and Richard M. Murray, *Information flow and cooperative control of vehicle formations*, IEEE Trans. on Automatic Control (2004).

[9] M. R. Garey and D. S. Johnson, *Computers and intractability: A guide to the theory of np-completeness*, (1979).

[10] Steven Gold and Anand Rangarajan, *A graduated assignment algorithm for graph matching*, IEEE Trans. on Pattern Analysis and Machine Intelligence (1996).

[11] Roger A. Horn and Charles R. Johnson, *Matrix analysis*, Cambridge, 1985.

[12] Albert Isidori, *Nonlinear control systems 3rd ed.*, Springer, 1995.

[13] Ali Jadbabaie, Jie Lin, and A. Stephen Morse, *Coordination of groups of mobile autonomous agents using nearest neighbor rules*, IEEE Trans. on Automatic Control (2003).

[14] Hassan K. Khalil, *Nonlinear systems 3rd ed.*, Prentice Hall, 2002.

[15] G. Lafferriere, A. Williams, J. Caughman, and J.J.P Veerman, *Decentralized control of vehicle formations*, System and Control Letters (2005).

[16] Riccardo Marino and Patrizio Tomei, *Nonlinear control design*, Prentice Hall, 1995.

[17] Russell Merris, *Laplacian matrices of graphs: A survey*, Linear Algebra and Its Applications (1994).

[18] M. Mesbahi and F. Y. Hadaegh, *Formation flying control of multiple spacecraft via graphs, matrix inequalities, and switching*, IEEE Conf. on Control Applications (1999).

[19] Luc Moreau, *Stability of multiagent systems with time-dependent communication links*, IEEE Trans. on Automatic Control (2005).

[20] Richard M. Murray, *Recent research in cooperative control of multi-vehicle systems*, ASME Journal of Dynamic Systems, Measurement, and Control (2006).

[21] Richard M. Murray, Z. Li, and Shanker Sastry, *A mathematical introduction to robotic manipulation*, CRC Press, 1993.

[22] Reza Olfati-Saber and Richard M. Murray, *Consensus problems in networks of agents with switching topology and time-delays*, IEEE Trans. on Automatic Control (2005).

[23] Wei Ren, *Consensus strategies for cooperative control of vehicle formations*, IET Control Theory Appl. (2007).

[24] Wei Ren and Randal W. Beard, *Consensus seeking in multiagent systems under dynamically changing interaction topologies*, IEEE Trans. on Automatic Control (2005).

[25] U. G. Rothblum, *Algebraic eigenspaces of nonnegative matrices*, Linear Algebra and Its Applications (1975).

[26] Kloder S. and Hutchinson S., *Path planning for permutation-invariant multi-robot formations*, IEEE Conf. on Robotics and Automation (2005).

[27] Shanker Sastry, *Nonlinear systems analysis, stability, and control*, Springer, 1999.

[28] Sepulchre, Rodolphe, Paley, Derek, Leonard, and Naomi Ehrich, *Group coordination and cooperative control of steered particles in the plane*, Group Coordination and Cooperative control (K. Y. Pettersen, J. T. Gravdahl, and H. Nijmeijer, eds.), Lecture Notes in Control and Information Sciences, Spinger-Verlag, London, 2006, pp. 217–232.

[29] Dusan M. Stipanovic, Gokhan Inalhan, Rodney Teo, and Claire J. Tomlin, *Decentralized overlapping control of a formation of unmanned aerial vehicles*, Automatica (2004).

[30] Paulo Tabuada, George J. Pappas, and Pedro Linda, *Feasible formations of multi-agent systems*, Proceedings of the American Control Conference (2001).

[31] Chai Wah Wu, *Synchronization in arrays of coupled nonlinear systems with delay and nonreciprocal time-varying coupling*, IEEE Trans. on Circuit and Systems (2005).

[32] Chai Wah Wu and Leon O. Chua, *Synchronization in an array of linearly coupled dynamical systems*, IEEE Trans. on Circuit and Systems (1995).

[33] Hiroaki Yamaguchi, Tomio Arai, and Gerardo Beni, *A distributed control scheme for multiple robotic vehicles to make group formations*, Robotics and Autonomous Systems (2001).

[34] Hsi-Han Yeh, Eric Nelson, and Andrew Sparks, *Nonlinear tracking control for satellite formations*, IEEE Conf. on Decision and Control (2000).

[35] Michael M. Zavlanos and George J. Pappas, *A dynamical systems approach to weighted graph matching*, IEEE Conf. on Decision and Control (2006).

국문 초록

다 개체 동종 시스템의 분산 편대 추종 제어

다 개체 동종 시스템의 분산 편대 추종 제어는 최근 몇 년간 많은 연구가 이루어져 왔다. 이러한 관심은 군사 분야, 모발 센서 네트워크, 지능형 교통 시스템과 같은 새롭고 흥미로운 응용 사례가 대두됨에 따라 집중되었다. 분산 편대 제어는 여러 가지 이론과 방법으로 연구되어 왔는데 대부분의 경우 여러 동종 시스템의 상호 연결성을 토대로 라플라스 인접 행렬의 성질로 설명하고 있다. 이러한 접근법은 합의 문제와 동기화 문제와도 관련 깊으며 이 경우 편대 제어와 유사한 방식으로 안정도를 설명하며 설계 고려 사항을 도출한다. 본 논문에서, 우리는 분산된 방법으로 여러 대의 동종 비선형 시스템을 다루며 이 시스템들은 시간에 대해 부드럽게 변하는 상대 기준 위치에 따라 편대를 이룰 수 있고 또한 전체 그룹은 모든 방향으로 자유롭게 움직이는 것이 가능하다. 우리의 주요 목표는 위와 같은 편대 추종 제어의 안정도를 해석하고 제어기를 설계하는 것이다. 그리고 우리는 추가적으로 시스템의 순서 교환을 통해 여러 시스템간 동질성을 이용하는 방법에 대해 논한다.

주요어휘: 추종 제어,편대 제어,분산 제어,궤한 선형화,다 개체 동종 시스템
학번: 2006-23150

감사의 글

부족한 제자를 관심과 사랑으로 아낌 없이 지도해 주신 심형보, 서진헌 교수님께 깊은 감사의 말씀을 드립니다. 더불어 학위 논문 심사를 맡아 주신 이범희 교수님께도 심심한 감사의 말씀을 올립니다. 지난 2년간 격려와 애정을 베풀어 주신 제어 및 동역학 연구실의 많은 선후배님께도 이 기회를 빌어 고마운 마음을 전하고자 합니다. 특히 301동 7층의 연구실을 함께 환히 밝혔던 상보, 홍근, 영준, 태규, Artem 에게 그리고 윗방의 종욱형, 용운형, 재성형, 한성, 현철형에게 또한 사랑스런 후배 진영, 찬화, 진우, 수범, 성훈에게 덧붙여 졸업한 원민형, 세진, 병인에게 마지막으로 백주훈, 김정수 박사님께 한 분 한 분 감사의 말씀을 드립니다.

그간 학창 시절을 보람차고 즐겁게 할 수 있었던 커다란 원동력이었고 많은 경험과 기회의 터전이었던 전공학회 핸즈의 선후배님께도 감사의 마음을 표현하고 싶습니다. 마이크로 로봇을 함께 만들며 저의 우상이 되어 주었던 창현형, 즐겁고 재미있는 일이 가득했던 병수형, 항상 밝은 표정으로 묵묵히 따라주었던 환주, 지호 그리고 언제나 친근한 말동무가 되어주었던 승구, 진형에게 특별한 마음을 표현하고 싶습니다. 또한 지금은 각자의 길에서 열심히 노력하고 있는 99학번의 모든 동기와 이름을 모두 열거하기에 자리가 모자랄 정도로 수많은 선후배님께 감사의 마음을 전합니다.

그 외에도 언제나 우울할 틈이 없도록 만들어 주었던 신나는 광하, 항상 든든한 버팀목이 되어주는 상일 그리고 변치 않는 오랜 나의 친구 재교, 함께 지내어도 잘 살펴주지 못해 미안한 사촌동생 민우에게도 고마운 마음을 보냅니다. 덧붙여 완전 소중한 나의 누나와 믿음직한 매형, 이쁜이 조카 현서에게 감사의 마음을 전하며 손자의 건승을 기도하시는 할머니께도 고마움을 올립니다. 끝으로 언제나 곁을 함께 하며 목표와 의지를 잃지 않게 도와 주었던 사랑하는 소연에게 고마움을 전합니다.

이제 오랜 학업을 뒤로 하고 사회인이 되려 합니다. 많은 분들의 염려와 걱정에 보답하며 자랑스런 상훈이 되도록 항상 처음 그대로의 마음으로 열심히 노력하겠습니다.

마지막 페이지를 덮으며, 이 모든 저를 있게 하고 그 무엇으로도 대신할 수 없는 더할 수 없이 사랑하는 나의 부모님께 이 논문을 바칩니다. 어머니, 아버지 사랑합니다.

2008년 7월의 어느 깊은 밤에 김상훈 올림

새로운 삶의 경계를 향해, 나는 걸음을 뗀다.