2012

Consensusability of discrete-time multi-agent systems

Abhishek Pandey
Louisiana State University and Agricultural and Mechanical College, apande3@tigers.lsu.edu

CONSENSUSABILITY OF DISCRETE-TIME MULTI-AGENT SYSTEMS

A Thesis

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering

in

The Department of Electrical and Computer Engineering

by
Abhishek Pandey
Bachelor of Engineering, Visvesvaraya Technological University, 2008
August 2012

ACKNOWLEDGEMENT

I would like to express my never ending gratitude to my advisor and committee chair, Professor Guoxiang Gu for his guidance, and constant motivation towards the completion of this thesis. His technical advice and suggestions helped me to overcome hurdles and kept me enthusiastic and made this work a wonderful learning experience. In addition, I would like to thank my committee members Professor Kemin Zhou and Professor Lu Peng for taking time out of their busy schedule and agreeing to be a part of my committee and for their valuable feedback. I also take this opportunity to thank Professor James J. Spivey for his moral and financial support throughout my studies at LSU.

I would like to thank the faculty members of the Department of Electrical and Computer Engineering, whose classes I had the privilege of attending while completing my studies and making my study at Louisiana State University a pleasant experience.

Finally, I would like to thank my family and friends without whom I would not have made it to this point.

TABLE OF CONTENTS

LIST OF FIGURES

NOTATION AND SYMBOLS

$\mathbb{R}$	set of real numbers
$\mathbb{C}$	set of complex numbers
$\in$	belong to
$:=$	defined as
I_n	$n \times n$ identity matrix
$[a_{i,j}]$	a matrix with $a_{i,j}$ as the element on the ith row and jth column
$diag(a_1, \ldots, a_n)$	an $n \times n$ diagonal matrix with a_i as the ith element on the diagonal
A^T	transpose of matrix A
A^*	transpose conjugate of matrix A
A^{-1}	inverse of matrix A
$\det(A)$	determinant of matrix A
$\lambda_i(A)$	ith eigenvalue of matrix A
$\sigma_i(A)$	ith singular value of matrix A
$\overline{\sigma}(A)$	largest singular value of matrix A
$\otimes$	Kronecker product operator
$G(s)$	transfer function (continuous time)
$G(z)$	transfer function (discrete time)
$\left[\begin{array}{c\|c} A & B \\ \hline C & D \end{array}\right]$	shorthand for state space realization $C\,(sI - A)^{-1}\,B + D$ (continuous time) $C\,(zI - A)^{-1}\,B + D$ (discrete time)

ACRONYMS

MAS – Multi Agent Systems

SISO – Single Input Single Output

LTI – Linear Time Invariant

SVD – Singular Value Decomposition

ARE – Algebraic Riccati Equation

LQR – Linear Quadratic Regulator

MIMO – Multi-Input Multi-Output

ABSTRACT

The study of multi-agent systems (MAS) focuses on systems in which many intelligent agents interact within an environment. The agents are considered to be autonomous entities. MAS can be used to solve problems that are difficult or impossible for an individual agent to solve. The main feature which is achieved when developing MAS, if they work, is flexibility, since MAS can be added to, modified and reconstructed, without the need for detailed rewriting of the application. MAS can manifest self-organization as well as self-steering related complex behaviors even when the individual strategies of all their agents are simple. The goal of MAS research is to find methods that allow us to build complex systems composed of autonomous agents who, while operating on local knowledge and possessing only limited abilities, are nonetheless capable of enacting the desired global behaviors. We want to know how to take a description of what a system of agents should do and break it down into individual agent behaviors.

This thesis investigates the problem when discrete-time MAS are consensusable under undirected graph. A discussion is provided to show how the problem differs from continuous time system. Then a consensusability condition is derived in terms of the Mahler measure of the agent system for single input single out systems (SISO) and result shows that there is an improved consensusability by a power of two. An algorithm is proposed for distributed consensus feedback control law when the consensusability holds. Also the case of output feedback is considered in which the consensusability problem becomes more complicated. To solve this we decompose the problem into two parts i.e. state feedback and state estimation. Simulation results demonstrate the effectiveness of the established results.

CHAPTER 1
INTRODUCTION

This chapter introduces the consensus problem and the motivation for studying it. The contribution of this thesis is discussed briefly. Some basic concepts and important terminologies are provided as background material.

1.1 Motivation

Consider N homogenous dynamic agents described by the following state space model:

$$x_i(k+1) = A\,x_i(k) + B\,u_i(k)$$

$$y_i(k) = C\,x_i(k) \qquad (1.1)$$

where, $x_i(k) \in \mathbb{R}^n$, $u_i(k) \in \mathbb{R}^m$, and $y_i(k) \in \mathbb{R}^p$ represent the state vector, control input and measured output of agent i, respectively, for $i \in \mathcal{N} = \{1,2,3, \dots N\}$. Only single-input single-output (SISO) agents will be considered in this thesis, and thus $m = p = 1$. We call $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^n$ the state and input matrices respectively. The transfer matrix of the ith agent is $(z) = C\,(zI - A)^{-1}\,B$, i.e., all agents have the same plant model. The N discrete-time dynamic agents are networked together under the communication topology represented by either directed or undirected graph. An individual agent updates its state by local communications with its neighbors so that all agents asymptotically reach an agreement. The consensus control aims at designing distributed feedback control protocols such that

$$\lim_{k\to\infty} \left\| x_i\,(k) - x_j(k) \right\| = 0 \quad \forall\, i,j \in N$$

The consensus problem is a fundamental research topic in the field of distributed computing. It has attracted great attention from many research communities, ever since the theoretical framework of the consensus problem for multiagent systems (MAS) was proposed and analyzed by Olfati-Saber and Murray in [1] and [2] built upon on the earlier work of Fax and Murray [3]. It leads to the research field of consensus control.

The main objective of the consensus control is to develop algorithms for MAS such that the group of dynamic agents reaches an agreement regarding a certain quantity of interest by communicating information with neighboring agents and itself. The MAS differs from traditional control systems because it requires the convergence of control theory, communications and computer science.

The challenges to MAS lie in the design of control systems that achieve robust cooperation, despite disconnections of some agents, inherent to most distributed environments. Had no notion of distributed computing evolved, each agent would be running separately, utilizing more resources and increasing the cost. It would not be able to utilize the availability of several agents in a distributed environment. It is the need for the cooperation which reveals many problems which otherwise would have been undiscovered. The field has interested researchers because of its broad applications to formation control [1, 3-6], sensor networks [7, 8], cooperative control [3, 9], flocks [10-12] and synchronization of coupled chaotic oscillators [13-16].

Distributed Computing and Consensusability in Computer Science

Distributed computing includes a wide range of algorithms that can be classified by a variety of attributes like shared memory, message passing, dataflow, timing models, resource allocation, communication, database concurrency control, agreement, deadlock detection etc. Some of the major intended applications platforms are:

- Communication systems
- Shared-memory multiprocessor computation
- Distributed operating systems
- Distributed database systems
- Digital circuits and
- Real-time process-control systems

These algorithms have a high degree of uncertainty and are more dependent on the activities. Some of the uncertainties can be unknown shape of networks, independent input at different locations, processor failures, unknown number of processors, etc. Because of all these uncertainties, no component of distributed systems knows the complete state.

Consensus is one of the problems in distributed computing that has a long history in computer science [10]. It encapsulates the task of group agreement in the presence of faults. An important subject in the social sciences involves models of how opinions change over time, until hopefully some consensus is reached. These models have begun to be used in computer science applications in distributed computing. In such applications, the values of processors in a network are updated until all the processors have the same value. Valid consensus protocols must ensure that all processors reach the same value and do not fail at any point during its execution. A consensus protocol is an interaction rule that specifies the information exchange between an agent and all its neighbors on the network. The consensus problem is challenging primarily because one or more of the processes involved might fail at any time. It may fail because of one of the following reasons: a failed process might stop participating in the protocol or it might collaborate with other failed processes in order to deliberately subvert the operations.

1.2 Scope of the Thesis

This thesis studies the consensus problem for MAS in both continuous time and discrete time systems. We begin with continuous time MAS systems in [13], which relate the consensusability of MAS to synchronization of complex networks. It shows the important role of eigen-ratio that is the ratio of the largest eigenvalue to the smallest non-zero eigenvalue of the communication graph. Consensusability condition derived in [13] requires that the complementary sensitivity of each individual agent be positive real for which LQR control can be employed.

The positive real condition can be met with static feedback control laws as studied in [17]. Consider a group of N identical agents with general linear dynamics, which may be regarded as the linearized model of some nonlinear systems. The dynamics of the ith agent are described by

$$\dot{x}_i = Ax_i + Bu_i \; ; \qquad\qquad y = Cx_i$$

where $x_i = [x_{i,1}, \dots \dots \dots, x_{i,n}]^T \in \mathbb{R}^n$ is the state, $u_i \in \mathbb{R}$ is the control input, and $y_i \in \mathbb{R}$ is the measured output. It is assumed that (A, B, C) is stabilizable and detectable.

The state feedback control law $u = -Kx$ can be designed to stabilize the closed loop system and minimize the quadratic cost function

$$J = \int_0^{\infty} [x(t)'Q\, x(t) + u(t)'R\, u(t)]dt \,, \quad Q \geq 0, \qquad R > 0$$

Such an optimal control law can be obtained by solving the unique stabilizing solution of the continuous time algebraic Riccati equation (ARE)

$$A'X + XA - XB_{\lambda_2}R^{-1}B'_{\lambda_2} + Q = 0$$

and setting $K = R^{-1}B'_{\lambda_2}X$. Then $u_{opt}(t) = Kx(t)$ minimizes J over all stabilizing control laws.

Consider $T_{\lambda_2}(z) = K(sI - A - B_{\lambda_2}K)^{-1}B_{\lambda_2}$ with $B_{\lambda_2} = B\lambda_2$. Then $T_{\lambda_2}(z)$ is positive real in the sense that

$$T_{\lambda_2}(j\omega) + T_{\lambda_2}(j\omega)^* > 0 \;\; \forall\omega \in \mathbb{R}$$

As shown in [13], the consensusability is equivalent to stability of $A - \lambda_i BK$ for $i \geq 2$ that is equivalent to

$$1 + \left(\frac{\lambda_i}{\lambda_2} - 1\right)T_{\lambda_2}(s) \neq 0 \qquad \forall \;\; Re(s) \geq 0$$

Therefore as long as the system is stabilizable, and is void of poles on the imaginary axis, a positive real condition can be met under state feedback. Since output can be decomposed into state feedback and its dual using distributed observers proposed in [13]. The problem of consensusability becomes trivial in the case of the continuous time systems.

However the same positive real condition cannot be applied to the discrete-time system and has been pointed out in [18]. In this paper, the authors studied the consensusability for linear discrete-time MAS. The joint effects of agent dynamic and network topology are considered. The authors propose an observer based distributed control protocol. A necessary and sufficient condition under this type of protocol is given which shows that the eigen-ratio of undirected communication graph affects consensusability. The eigen-ratio is referred to the ratio of the largest eigenvalue to the second smallest eigenvalue of the undirected graph Laplacian matrix. It is shown in [18] that eigen-ratio needs to satisfy the following condition.

$$\frac{\lambda_N}{\lambda_2} < \frac{\mu(A)+1}{\mu(A)-1},$$

where $\mu(A)$ is termed as Mahler defined by

$$\mu(A) = \prod_{i=1}^{n} \max\{|\lambda_i(A)|, 1\}$$

If A is stable, then

$$\mu(A) = 1 \Rightarrow \frac{\lambda_N}{\lambda_2} < \infty$$

for which the consensusability condition always holds. An improvement over the above condition has been provided in [19] where a stable filter $F(z)$ is employed to improve the consensus condition by a power of 2, leading to

$$\frac{\lambda_N}{\lambda_2} < \left(\frac{\mu(A)+1}{\mu(A)-1}\right)^2$$

In this thesis, we consider the problem of consensusability for discrete time multi agent systems under undirected communication graph. The consensusability condition is derived which depends only on $\mu(A)$, the Mahler measure, of the MAS. An algorithm is proposed for consensus control to design the distributed feedback control protocol based on local information, when the consensusability holds. Simulation examples are worked out to illustrate the results.

1.3 Background

This section reviews some of the basic concepts in linear algebra, graph theory and block diagram reduction method that are going to be used throughout this thesis.

1.3.1 Linear Algebra

Basic linear algebra concepts are discussed below taking references from [20] and [21].

Eigenvalues and Eigenvectors

Let $A \in \mathbb{R}^{n\times n}$. The eigenvalues of A are the n roots of its characteristic polynomial $p(\lambda) = \det\,(\lambda I - A) = 0$. The maximum modulus of the eigenvalues is called the spectral radius, denoted by

$$\rho(A) := \max_{1\leq i\leq n} |\lambda_i(A)|$$

The real spectral radius of a matrix A, denoted by $\rho_R(\mathrm{A})$, is the maximum modulus of the real eigenvalues of A; that is $\rho_R(\mathrm{A}) := \max_{\lambda_i \in \mathbb{R}} |\lambda_i|$. A nonzero vector $x \in \mathbb{C}^n$ that satisfies

$$Ax = \lambda x$$

is referred to as a right eigenvector of A. Dually, a nonzero vector y is called a left eigenvector of A, if

$$y^* A = \lambda y^*$$

In general, eigenvalues need not be real, neither do their corresponding eigenvectors. However, if A is real and λ is a real eigenvalue of A, then there is a real eigenvector corresponding to λ. In the case that all eigenvectors of matrix A are real, we denote $\lambda_{\max}(A)$ for the largest eigenvalue of A and $\lambda_{\min}(A)$ for the smallest eigenvalue.

Eigenvalues play an important role in situations where the matrix is a transformation from one vector space onto itself. Systems of linear ordinary differential equations are the primary examples. The values of λ can correspond to frequencies of vibration, or critical values of stability parameters, or energy levels of atoms.

Matrix Diagonalization

Matrix diagonalization is a process of taking a square matrix and converting it into a diagonal matrix which shares the properties of the underlying matrix. Matrix diagonalization is equivalent to transforming the underlying system of equations into a special set of coordinate axes in which the matrix takes this canonical form. Diagonalizing a matrix is equivalent to finding the matrix's eigenvalues, which forms the entries of the diagonalized matrix. Similarly, the eigenvectors make up the new set of axes corresponding to the diagonal matrix.

This relationship between diagonalized matrix, eigenvalues and eigenvectors is called eigen decomposition. Let P be a matrix of eigenvectors of a given square matrix A and Λ be a diagonal matrix with the corresponding eigenvalues on the diagonal, i.e.

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix}$$

Then as long as P is a square matrix and eigenvectors are linearly independent, then P^{-1} exists, and A can be written as eigen decomposition

$$A = P\Lambda P^{-1}$$

If the above holds, then it allows us to investigate the properties of A by analyzing the diagonal matrix Λ. For example, repeated matrix powers can be expressed in terms of powers of scalars:

$$A^p = P\Lambda^p P^{-1}$$

If eigenvectors of A are not linearly independent, then such a diagonal decomposition does not exist and the powers of A exhibit more complicated behavior.

In linear algebra, a symmetric matrix is a square matrix that is equal to its transpose. It is a fact that if A is a symmetric matrix, then $P^{-1} = P'$. Then above equation can be written as

$$A = P\Lambda P'$$

In Matlab, the command $[V, D] = eig\,(A)$ produces matrices of eigenvalues (D) and eigenvectors (V) of matrix A. If T is any nonsingular matrix, then

$$A = TBT^{-1}$$

is known as a similarity transformation and A and B are said to be similar. If $Ax = \lambda x$ and $x = Ty$, then $By = \lambda y$. In other words, a similarity transformation preserves eigenvalues. The eigenvalue decomposition is an attempt to find a similarity transformation to the diagonal form.

If A is not a square matrix, then it cannot have an inverse and A does not have eigen decomposition. However if A is $m \times n$, then A can be written into Singular Value Decomposition (SVD) form:

$$A = USV'$$

where, U is an $m \times m$ real or complex unitary matrix, S is a $m \times n$ diagonal matrix with non-negative real numbers on the diagonal, and V, a $n \times n$ real or complex unitary matrix. The following illustrates the case when $m \times n$ with $m > n$:

$$S = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \sigma_n \\ 0 & \cdots & \cdots & 0 \\ 0 & \cdots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 \end{bmatrix}$$

In Matlab, $[U, S, V] = svd\ (A)$ produces a diagonal matrix S of the same dimension as A, with nonnegative diagonal elements in decreasing order, and unitary matrices U and V so that

$$A = USV'$$

Singular values play an important role where the matrix is a transformation from one vector space to a different vector space, possibly with a different dimension. Systems of over- or underdetermined algebraic equations are the primary examples.

Kronecker Product

Given two matrices $A \in \mathbb{C}^{m\times n}$ and $B \in \mathbb{C}^{p\times q}$, the Kronecker product of A and B is defined as

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix} \in \mathbb{C}^{mp\times nq}$$

Some Properties of Kronecker product are listed next:

$$A \otimes (B + C) = A \otimes B + A \otimes C$$

$$A \otimes (B \otimes C) = (A \otimes B) \otimes C$$

$$(\alpha_A A) \otimes (\alpha_B B) = \alpha_A \alpha_B \, (A \otimes B)$$

$$(A \otimes B)^T = A^T \otimes B^T$$

$$(A \otimes B)\,(C \otimes D) = (AC) \otimes (BD)$$

$$(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$$

1.3.2 Communication Graph

Many real-world situations can conveniently be described by means of a diagram consisting of a set of points together with lines joining certain pairs of these points [22]. The ideas from graph theory and control theory help us to study the relation between the communication network and agent dynamics. Therefore a review of some terms and definitions used in graph theory are discussed here using reference [23].

Graph Theory

A graph is an ordered pair $\mathcal{G} = \{\Upsilon, \mathcal{E}\}$ of sets such that, $\mathcal{E} \subset \Upsilon \times \Upsilon$; thus, the elements of $\mathcal{E}$ are 2-element subsets of Υ. The elements of Υ are vertices of graph $\mathcal{G}$; the elements of $\mathcal{E}$ are its edges. A graph with vertex set Υ is said to be on graph Υ. The vertex set of graph $\mathcal{G}$ is referred to as $\Upsilon(\mathcal{G})$, its edge set as $\mathcal{E}(\mathcal{G})$. The number of vertices of a graph $\mathcal{G}$ is its order, written as $|\mathcal{G}|$, its number of edges is denoted by $||\mathcal{G}||$. Graphs are finite, infinite, and countable and so on according to their order. The ends of an edge are said to be incident with the edge and vice versa [22]. Two vertices which are incident with a common edge are adjacent, as are two edges which are incident with a common vertex, and two distinct adjacent vertices are neighbors. An edge with identical ends is called a loop, and an edge with distinct ends a link. Two or more pairs with the same pair of ends are said to be parallel edges. A path is a simple graph whose vertices can be arranged in a linear sequence in such a way that two vertices are adjacent if they are consecutive in the sequence and are non-adjacent otherwise. The length of the path is the number of its edges. A graph is connected if, for every partition of its vertex set into two nonempty sets X and Y, there is an edge with one end in X and one end in Y; otherwise the graph is disconnected.

Although drawings are a convenient means of specifying graphs, they are clearly not suitable for storing graphs in computers, or applying mathematical methods to study their properties. For these reasons, we consider two matrices associated with a graph, its incidence matrix and adjacency matrix.

Let $\mathcal{G}$ be a graph, with vertex set Υ and edge set $\mathcal{E}$. The incidence matrix of $\mathcal{G}$ is the $n \times m$ matrix $M_{\mathcal{G}} \coloneqq (m_{ve})$, where m_{ve} is the number of times (0, 1, or 2) that vertex v and edge e is incident. The adjacency matrix of $\mathcal{G}$ is the $n \times n$ matrix $\mathcal{A} \; := \; (a_{uv})$, where a_{uv} is the number of edges joining vertices u and v, each loop counting as two edges. Because most graphs have many more edges than vertices, the adjacency matrix of a graph is generally much smaller than its incidence matrix and thus requires less storage space.

Although many problems lend themselves to graph-theoretic formulation, the concept of graph is sometimes not adequate. When dealing with communication graph, it is necessary to know in which direction the communication is taking place between the agents. Therefore in such a situation just knowing the graph is not of much use. What we need is a graph in which link has an assigned orientation, namely a directed graph.

A directed graph $\mathcal{G}$ consists of a set of vertices, Υ and a set of edges $\mathcal{E} \subset \Upsilon^2$, where $a = (\alpha, \beta) \in A$ and $v, w \in \Upsilon$. The first element of a is denoted by tail(a), and the second is denoted by head(a). When tail(a) = head(a), it forms a self-loop. In a communication graph we assume that tail(a) $\neq$ head(a), i.e. the graph has no self-loops. A graph with a property that for any $(\alpha, \beta) \in A$, the arc $(\beta, \alpha) \in A$ as well is said to be undirected. A graph in which a path exists from every vertex to every vertex that cannot be joined by any path is termed disconnected.

In a communication graph, an edge from agent i to agent j means that agent i can directly receive information from agent j. Agent i can then refine its own information by learning information that j has, including information acquired by j from another agent.

1.3.3 Block Diagram Reduction Method

Rule 1: Consider $r(t)$ to be an input signal and $y(t)$ to be the output signal.

Error signal, $e(t) = G_2\, r(t) - G_2\, y(t) = G_2\, [r(t) - y(t)]$

Output Signal, $y(t) = G_1\, e(t)$ or $y(t) = G_1\, G_2\, e(t)$

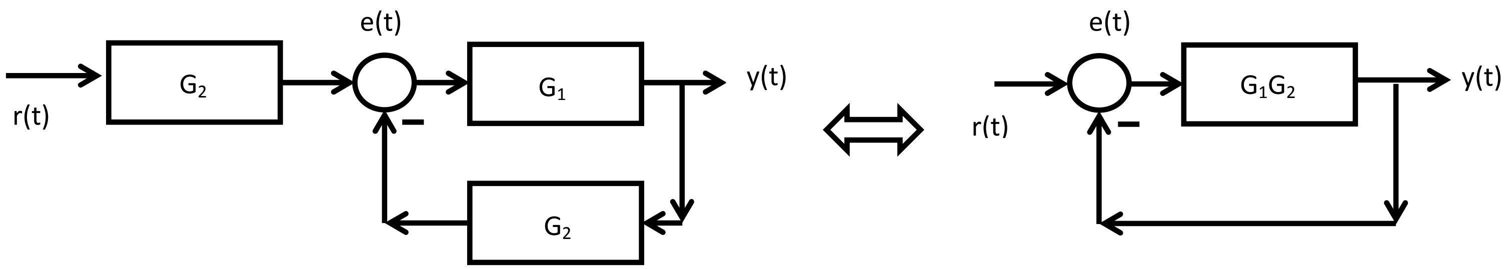

Figure 1.1 Block Diagram Rule 1

The transfer functions of both the block diagrams are equal i.e. $\frac{y(t)}{r(t)} = \frac{G_1 G_2}{1+G_1 G_2}$ and hence are equivalent.

Rule 2:

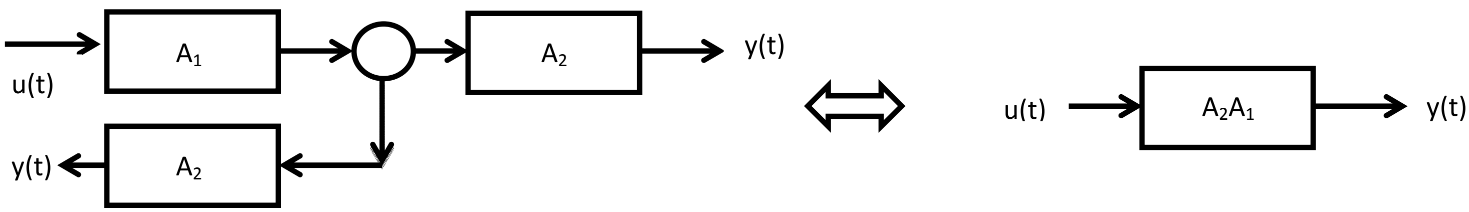

Figure 1.2 Block Diagram Rule 2

CHAPTER 2
LITERATURE REVIEW

In this chapter previous work from the literature is reviewed and known results are outlined here.

2.1 Consensusability in Control Systems

In Control Systems, a network of homogenous dynamic systems is said to be consensusable if all agents converge to a certain quantity of interest that depends on the state of all agents. The issue arising in the development of multi-agent systems is the design of consensus algorithm (or a protocol) which will have distributed control policies based on local information that enables all agents to asymptotically converge to the average of their initial values. Consensus algorithm relies on neighbor-to-neighbor information exchange [24]. The dynamic agents update their states through local interaction between its neighbors under proper communication graph and converge to a common value. Communication graph determines what information is available and how each agent is interacting with another.

The survey paper by Olfati-Saber and Murray [9] summarizes the development of consensus and cooperation in networked multi agent systems. The authors provide a theoretical framework for analysis of consensus algorithms with directed information flow. The paper discusses the concepts of information consensus in networks and methods of convergence and performance analysis for the algorithms provided. A brief introduction to the networked systems with nonlocal information flow is also provided. As stated earlier, the important results and conclusions established in this paper have attracted researchers into the field of consensus control. Some of the work which has been done in the field of consensus of multiagent systems is discussed below.

2.1.1 Formation Control

Distributed formation control is an area of application which has become prominent in research due to its application in multivehicle systems. Technological advances have motivated researchers to study autonomous and adaptable vehicle formations. Due to development of strong control methods for single vehicles, improvement in communication capabilities and the availability of miniaturization technologies have led to interest in vehicles that can interact autonomously with the other vehicles and environment, in presence of uncertainty and perform activities beyond the capability of individual vehicles.

The design goal is to enable decision making and control capability of the vehicles to work cooperatively. Although this area poses its own unique challenges including limited sensing capabilities of the vehicles, network bandwidth limitations, interruptions in communications due to packet loss and physical disruptions to the communication devices of the vehicle. In [9], two approaches are proposed to distributed formation control:

i) Representation of formations as rigid structures and the use of gradient-based controls obtained from their structural potentials, and

ii) Representation of formations using the vectors of relative positions of neighboring vehicles and the use of consensus based controllers with input bias.

$$\dot{x}_i = Ax_i + Bu_i$$

The cooperative behavior discussed in [3] by Fax and Murray is formation control of type ii). The authors categorize the vehicle formation control into two different methods – leader follower method and virtual leader method [25] [26]. The leader follower method is simple as a reference trajectory is defined by the leader. But it suffers from poor disturbance rejection properties. The method depends on the leader, and too much of reliance on one agent in the formation is undesirable. For the virtual leader method, the agents create a fictitious leader vehicle whose trajectory acts as a leader for the group. This method does not have problems in disturbance rejection but at the high cost of communication and computation in finding the agent which acts as a leader and communicates its position in time to support real time control of the other agents. The above two methods use an underlying communication topology that enables the use of a particular formation control methodology.

In cooperative control, the stability and performance of a system depends on the topology of information flow. The authors in [3] derive necessary and sufficient conditions for stability of interconnected systems of identical agents in terms of eigenvalues of the graph Laplacian matrix, and have studied how the information flow affects the performance using Nyquist criterion. A separation principle has been proposed to decompose formation stability into two components – Stability of the achieved information flow for the given graph and stability of the individual vehicle for the given controller. This enables the information flow to be robust to changes in graph and enabling tight formation control despite limitations in intervehicle communication capability.

Consider a group of N vehicles, whose identical linear dynamics are denoted by

$$\dot{x}_i = Ax_i + Bu_i$$

where $x_i \in \mathbb{R}^n$, $u_i \in \mathbb{R}^m$ are the vehicles states and controls and $i \in V = \{1, \dots, N\}$ is the index of the vehicles in the group. Each vehicle receives the following measurements:

$$y_i = C_1 x_i$$

$$z_{ij} = C_2(x_i - x_j), \quad j \in N_i$$

where N_i is the set of vehicles which vehicle i can sense.

Thus, $y_i \in \mathbb{R}^k$ represents internal state measurements, and $z_{ij} \in \mathbb{R}^l$ represents external state measurements relative to other vehicles. Assume that $N_i \neq \emptyset$, meaning that each vehicle can sense at least one other vehicle. A single vehicle cannot drive all the z_{ij} terms to zero simultaneously; the errors must be fused into a single signal error measurement

$$z_i = \frac{1}{|N_i|} \sum_{j \in N_i} z_{ij}$$

where $|N_i|$ is the cardinality of the set $|N_i|$. We also define a distributed controller K which maps y_i, z_i to u_i and has internal states $v_i \in \mathbb{R}^s$, represented in state space form by

$$\dot{v}_i = F v_i + G_1 y_i + G_2 z_i$$

$$u_i = H v_i + D_1 y_i + D_2 z_i$$

Now consider the collective system of all n vehicles. For dimensional capability, we use the Kronecker product to assemble the matrices governing the formation behavior. The collective dynamics of n vehicles can be represented as follows:

$$\begin{pmatrix} \dot{x} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix}$$

where the $\{M_{ij}\}$ are block matrices defined as a function of the normalized graph Laplacian $L_{\mathcal{G}}$ and other matrices as follows:

$$M_{11} = I_n \otimes (A + BD_1C_1) + (I_n \otimes BD_2C_2)(L \otimes I_m)$$

$$M_{12} = I_n \otimes BH$$

$$M_{11} = I_n \otimes G_1C_1 + (I_n \otimes G_2C_2)(L \otimes I_m)$$

$$M_{22} = I_n \otimes F$$

The main stability result on relative position based formation of networked vehicles is due to Fax and Murray [3] and can be stated as follows:

Theorem 2.1: (Fax and Murray, 2004) A local controller K stabilizes the formation dynamics in [27] if and only if it stabilizes the set of N systems.

$$\dot{x_\imath} = Ax_i + Bu_i$$

$$y_i = C_1x_i$$

$$z_{ij} = \lambda_i C_2 x_i$$

where $\{\lambda_i\}_{i=1}^{N}$ is the set of eigenvalues of the normalized graph Laplacian L.

The above theorem reveals that the stability of a formation of N identical vehicles can be verified by stability analysis of a single vehicle with the same dynamics and an output that is scaled by the eigenvalues of the Laplacian of the network. Since λ_i may be complex for MAS over directed graph, it leads to a complex-valued LTI system in the above formulation.

Theorem 2.2: (Fax and Murray, 2004) Suppose P is SISO system. Then K stabilizes the relative dynamics of a formation if and only if the net encirclement of $-1/\lambda_i$ by the Nyquist plot of $-K(s)P(s)$ is zero for all non-zero λ_i.

2.1.2 Flocking Theory

Flocking is a form of collective behavior of large number of interacting agents with a common group objective [11]. This paper by Olfati- Saber introduces the algorithms and theory for flocking in multi agent systems. The deployment of large groups of autonomous vehicles is rapidly becoming possible because of technological advances in networking and in miniaturization of electromechanical systems [10]. Robots are replacing men to perform challenging tasks in adverse environments, exploration, and surveillance etc.

To build a simulated flock, Reynolds introduced three heuristics rules in [12]. Stated briefly in order of decreasing precedence, the behaviors that lead to simulated flocking are:

- Collision Avoidance: avoid collisions with nearby flock mates
- Velocity Matching: attempt to match velocity with nearby flock mates
- Flock Centering: attempt to stay close to nearby flock mates

Building upon these rules, Olfati-Saber has discussed two cases of flocking; in free-space and in presence of multiple obstacles. He has also presented three flocking algorithms: two for free-flocking and one in presence of obstacles. The first algorithm is a gradient based algorithm equipped with velocity consensus protocol and embodies all three rules of Reynolds. It is shown that this algorithm leads to regular fragmentation rather than flocking. The second and third algorithm discussed in this paper leads to flocking. A method for construction of cost functions is provided. A “universal” definition of flocking for particle systems with similarities to Lyapunov stability is given. Several simulation results are provided that demonstrate performing 2-D and 3-D flocking, split/rejoin maneuver, and squeezing maneuver for hundreds of agents using the proposed algorithms.

2.1.3 Synchronization of Complex Networks

Another topic that is closely related to the consensus of multiagent systems is the synchronization of coupled nonlinear oscillators [13]. As cited in [13], the pioneering work in the synchronization phenomenon of the two master-slave chaotic systems was observed and applied to secure communications in [15]. References [14] and [16] discuss the synchronization stability of a network of oscillators by using the master stability function method. The authors in [13] state that, due to nonlinear node dynamics only sufficient conditions can be given for verifying the synchronization. This paper discusses consensus problem of multiagent systems under fixed communication topology. It is shown that there exists a distributed observer-type consensus protocol based on relative output measurements. A framework is proposed which converts the consensus problem of MAS with communication topology having a spanning tree into the stability of set of matrices with the same low dimension. A 3-step approach has been proposed by the authors that allow the designed protocol to achieve consensus over one communication topology that can be directly used to solve the consensus problem for any topology containing a spanning tree where the only task is to select suitable coupling strength. This algorithm decouples the effects of the agent and protocol dynamics on the consensus stability from that of communication topology. The consensus region serves as a measure for the robustness of protocol to parametric uncertainty. It is convenient to design a protocol such that the consensus region is unbounded.

An observer type control protocol is proposed as

$$\dot{v}_i = (A + BK)v_i + F\left(c\sum_{j=1}^{N} a_{ij}C(v_i - v_j) - \varsigma_i\right)$$

$$u_i = Kv_i$$

where $v_i \in \mathbb{R}^n$ is the protocol state, $i = 1, \ldots, N,$ and $F \in \mathbb{R}^{q\times n}$ and $K \in \mathbb{R}^{p\times n}$ are the feedback gain matrices to be determined. The term $\sum_{j=1}^{N} a_{ij}C(v_i - v_j)$ denotes the information exchange between the protocol of agent i and those of its neighbors. This protocol is distributed since it is based only on the relative information of the neighboring agents. The relative measurements of other agents with respect to i are synthesized into single signal using

$$\varsigma_i = c\sum_{j=1}^{N} a_{ij}(y_i - y_j)$$

where $c > 0$ denotes coupling strength, $a_{ii} = 0$ and $a_{ij} = 1$ if agent i can obtain information from agent *j* but 0 otherwise.

Theorem 2.3: For a directed network of agents with communication topology $\mathcal{G}$ that has a directed spanning tree, the proposed protocol solves the consensus problem if and only if all matrices $A - BK$, $A - c\lambda_i FC$, $i = 2, \ldots \ldots.., N$, are Hurwitz, where λ_i, $i = 2, \ldots \ldots.., N$, are nonzero eigenvalues of the Laplacian matrix $L_{\mathcal{G}}$.

The above theorem presents the necessary and sufficient condition for the consensus problem.

2.1.4 Fast Consensus

The development of network design problems for achieving fast consensus algorithms is another research area. Some of the work and methods found in literature are discussed next.

Multi-hop relay in consensus problem is considered in [28], where the control input of each agent depends not only on its neighbors' states, but also on its neighbors' neighbors' state. By introducing more information with second hop, [28] demonstrates that consensus speed is improved. However the tradeoff for introducing the second-hop information is extra communication and large control effort.

In [24] the authors propose a distributed consensus control algorithm that uses both current states and the outdated states stored in memory. In contrast, the standard consensus relies on the current state. This algorithm converges faster than the standard algorithms and improves the performance. It does not require second hop communication. There are consensus algorithms with delays. In such a case delays are considered as a negative factor and the focus is usually on how the delay affects the stability of the consensus algorithms. Whereas the difference between the paper in [24] and the other work in time delay systems is that the outdated state information is considered as a positive factor and is, therefore, applied to multi-agent consensus problems while other papers consider the effect of time delay on stability of single system. This paper shows with both current states and outdated states available, consensus will converge faster than the standard consensus algorithm while requiring identical maximum control effort when the outdated states are chosen properly.

Watts and Strogatz proposed a small world network in [29] which was capable of interpolating between a regular network and a random network using a single parameter. Also known as the random wiring idea as mentioned in [9]. Based on this, Olfati-Saber proposed a randomized algorithm in [30]. His algorithm based on random rewiring of existing links of a network gave rise to a considerably faster consensus algorithm. The author demonstrates that it is possible to dramatically increase the algebraic connectivity of a complex regular network by 1000 times or more without adding new links or nodes to the network. Algebraic connectivity of the graph is defined as the second smallest eigenvalue of its Laplacian matrix and is a measure of speed for solving consensus problems in networks. The paper also shows that the mean of the bulk Laplacian spectrum remains invariant under random wiring.

Another approach was proposed by Xiao and Boyd in [31]. They consider the design of weights for a network and solve the problem of finding the fastest converging linear iteration by using a semi-definite convex programming. There is a slight increase in the speed of convergence of consensus algorithm. To achieve relatively high convergence speed, an alternate way is proposed, i.e., to design the topology of the network while keeping the weights fixed.

Consensus control has received great attention from the research community of control. It is not possible for this thesis to provide an exhaustive survey. However the presentation in this chapter contains some of the important results in this area which are related to our work in this thesis.

CHAPTER 3
CONSENSUS CONTROL

The consensus problem is formally introduced in this chapter. The basic problem of consensusability is first introduced for discrete time MAS under undirected graph. The state feedback and output feedback cases are discussed. We show that the output feedback case can be solved with observer based controllers which decomposes the problem into two parts state feedback and state estimation. Then the consensusability condition for single input MAS is introduced and controller design algorithm is derived.

3.1 Problem Formulation

Consider a multi-agent dynamic system composed of N identical agents described in (1.1). The transfer matrix of the ith agent is $P(z) = C\,(zI - A)^{-1}\,B$, i.e., all agents have the same plant model.

The problem of undirected communication graph for multi agent systems is considered. It is known that such a graph is balanced and its associated adjacent matrix $\mathcal{A}$ is symmetric. Let $\mathcal{N}_i = \{j \mid (j, i) \in \mathcal{E}\}$ and $deg_i = \sum_{j=1}^{N} a_{i,j}$ be the degree of ith agent. Denote $D = diag\,(deg_1, deg_2, \ldots, deg_N)$ and represents the number of neighbors each agent is having. Then $L_{\mathcal{G}} := D - \mathcal{A}$ is symmetric positive and semi-definite, termed as Laplacian matrix of graph $\mathcal{G}$. All Eigen values of $L_{\mathcal{G}}$ are non-negative and can be arranged in ascending order:

$$0 = \lambda_1\,(L_{\mathcal{G}}) \leq \lambda_2\,(L_{\mathcal{G}}) \leq \ldots \leq \lambda_N\,(L_{\mathcal{G}})$$

For simplicity $\lambda_i = \lambda_i(L_{\mathcal{G}})$ will be used. It is assumed that $\lambda_2 > 0$, which holds if the undirected graph is connected.

Consensus control aims at designing distributed feedback control protocol based on local information such that

$$\lim_{k\to\infty}\left\|x_i(k) - x_j(k)\right\| = 0 \quad \forall\, i,j \in N \tag{3.1}$$

State feedback control is given by (for $C = I$)

$$u_i(k) = K\sum_{j=1}^{N} a_{i,j}\left[x_j(k) - x_i(k)\right],\ i \in \mathcal{N} \tag{3.2}$$

where, $K \in \mathbb{R}^{m\times n}$

The multi agent system is said to be consensusable if there exists a control protocol in the form of (3.2) that achieves (3.1).

The case of output feedback ($C \neq I$) can be solved by observer based controllers to divide output feedback into dual problems of state feedback and state estimation.

3.2 State Feedback

Lemma 3.2.1 Consider state feedback control, i.e. $C = I$. The discrete time multi-agent systems in (1.1) are consensusable under the control protocol (3.3), if and only if there exists a common stable filter $F(z)$ of size $m \times m$ and control gain $K \in \mathbb{R}^{m\times n}$ such that $\hat{A} - \lambda_i \hat{B}\hat{K}$ is a stability matrix for $i = 2,3,\ldots,N$.

The control protocol in the form of (3.3) is inadequate in the case of state feedback. Denote q as the unit advance operator, i.e. $q^{-1}\, s(k) = s(k-1)$.

The following distributed feedback control protocol is proposed in [19]:

$$u_i(k) = F(q)\, K \sum_{j=1}^{N} a_{i,j} \left[x_j(k) - x_i(k)\right],\ i \in \mathcal{N} \qquad (3.3)$$

where, $F(z)$ is a transfer matrix of some stable filter, and K is the constant state feedback gain. Let $x_{iF}(k) \in \mathbb{R}^{n_f}$ be the state vector associated with $F(z)$ for the i th agent and

$$F(z) = D_F + C_F\, (zI - A_F)^{-1} B_F \qquad (3.4)$$

Define the augmented state vector $x_{i\kappa}(k) := x_i(k) \otimes x_{iF}(k) \in \mathbb{R}^{n+n_f}$.

$$x_{iF}(k) = A_F x_{iF}(k) + B_F u_i(k)$$

$$y_{iF}(k) = C_F x_{iF}(k) + D_F u_i(k)$$

$$x_i(k+1) = Ax_i(k) + By_{iF}(k) = Ax_i(k) + BC_F x_{iF}(k) + BD_F u_i(k)$$

$$y_i(k+1) = Cx_i(k)$$

The above equations result in:

$$\begin{bmatrix} x_i(k+1) \\ x_{iF}(k+1) \end{bmatrix} = \begin{bmatrix} A & BC_F \\ 0 & A_F \end{bmatrix} \begin{bmatrix} x_i(k) \\ x_{iF}(k) \end{bmatrix} + \begin{bmatrix} BD_F \\ B_F \end{bmatrix} u_i(k)$$

$$u_i(k) = Kx_i(k) = [K \quad 0] \begin{bmatrix} x_i(k) \\ x_{iF}(k) \end{bmatrix} = K[I_n \quad 0] \begin{bmatrix} x_i(k) \\ x_{iF}(k) \end{bmatrix} = K\hat{C} \begin{bmatrix} x_i(k) \\ x_{iF}(k) \end{bmatrix}$$

Denote

$$\hat{A} = \begin{bmatrix} A & BC_F \\ 0 & A_F \end{bmatrix}, \hat{B} = \begin{bmatrix} BD_F \\ B_F \end{bmatrix},$$

$$\hat{C} = [I_n \quad 0], \hat{K} = K\hat{C} \qquad (3.5)$$

where I_κ is the identity matrix of the size $\kappa \times \kappa$. It can then be verified that

$$\hat{C}\,(zI - \hat{A})^{-1}\,\hat{B} = P(z)F(z) \qquad (3.6)$$

Recall that $P(z) = C\,(zI - A)^{-1}\,B$ and $F(z)$ in (3.5). With $u_i(t) = F(q)\,v_i(t),$ the dynamic equations in (1.1) and (3.3) for the ith agent can be equivalently converted into

$$x_{i\kappa}(k+1) = \hat{A}\,x_{i\kappa}(k) + \hat{B}\,v_i(k) \qquad (3.7)$$

$$v_i(k) = \hat{K} \sum_{j=1}^{N} a_{i,j} \left[x_{j\kappa}(k) - x_{i\kappa}(k)\right] \qquad (3.8)$$

for each $i \in \mathcal{N}$. The consensusability and consensus control in [19] aim at designing both stable filter $F(z)$ and constant state feedback gain K such that

$$\lim_{k\to\infty} \left\| x_{i\kappa}(k) - x_{j\kappa}(k) \right\| = 0 \quad \forall\, i, j \in N \qquad (3.9)$$

If the above holds, the multi agent systems described in (1.1) are said to be consensusable under the control protocol (3.3), which is equivalent to the consensusability of the augmented multi agent systems in (3.7) under the control protocol in (3.8).

3.3 Output Feedback

In the case of output feedback (i.e. $C \neq I \in \mathbb{R}^{p\times n}$) the stable filter $F(z)$ needs to be employed at the output rather than input, of the ith agent for each i. The plant model $P(z)$ can be described by the following state space model:

$$x_i(k+1) = A\, x_i(k) + B\, u_i(k)$$

$$y_i(k) = C\, x_i(k) \qquad (3.10)$$

Let $x_{i_F}(k) \in \mathbb{R}^{n_f}$ be the state vector associated with $F(z)$ for the ith agent and is given by

$$F(z) = D_F + C_F\,(zI - A_F)^{-1} B_F \qquad (3.11)$$

It can be easily verified that:

$$\hat{C}_o (zI - \hat{A}_o)^{-1} \hat{B}_o = F(z)P(z) \qquad (3.12)$$

where

$$\hat{A}_o = \begin{bmatrix} A & 0 \\ B_F C & A_F \end{bmatrix},\ \hat{B}_o = \begin{bmatrix} B \\ 0 \end{bmatrix},\ \hat{C}_o = \begin{bmatrix} D_F C & C_F \end{bmatrix} \qquad (3.13)$$

The output feedback controller is distributed observer given by:

$$\tilde{x}_{i_k}(k+1) = \left(\hat{A}_o - \hat{B}_o \widehat{K}_c\right) \tilde{x}_{i_k}(k) - \widehat{K}_f \sum_{j=1}^{N} a_{i,j}\, \hat{C}_o\, e_{i,j}(k),$$

$$u_i(k) = -\widehat{K}_c \tilde{x}_{i_k}(k),\ \widehat{K}_f = \begin{bmatrix} I_n \\ 0 \end{bmatrix} K_f \qquad (3.14)$$

for $i \in \mathcal{N}$ where $K_f \in \mathbb{R}^{p\times n}$ and

$$e_{i,j}(k) = \left[\tilde{x}_{i_k}(k) - \tilde{x}_{j_k}(k)\right] - \left[x_{i_k}(k) - x_{j_k}(k)\right]$$

$$= -\left[x_{i_k}(k) - \tilde{x}_{i_k}(k)\right] + \left[x_{j_k}(k) - \tilde{x}_{j_k}(k)\right] \qquad (3.15)$$

Define the augmented state vector $x_{i_k}(k) = x_i(k) \otimes x_{i_F}(k) \in \mathbb{R}^{n+n_f}$ with $x_i(k)$ the state for $P(z)$ and $x_{i_F}(k)$ for $F(z)$. Then using (3.13) we can write:

$$x_{i_k}(k+1) = \hat{A}_o x_{i_k}(k) + \hat{B}_o u_i(k) = \hat{A}_o x_{i_k}(k) - \hat{B}_o \hat{K}_c \tilde{x}_{i_k}(k)$$

$$y_F(k) = \hat{C}_o x_{i_k}(k)$$

Rearranging (3.14) we can obtain the following equation:

$$\tilde{x}_{i_k}(k+1) = \hat{A}_o \hat{x}_{i_k}(k) + \hat{K}_f \sum_{j=1}^{N} a_{i,j}\, \hat{C}_o e_{i,j}(k) + \hat{B}_o u_i(k) \qquad (3.16)$$

Substituting (3.15) into (3.16) results in

$$\tilde{x}_{i_k}(k+1) = \left(\hat{A}_o - \hat{B}_o \hat{K}_c\right)\hat{x}_{i_k} + \hat{K}_f \sum_{j=1}^{N} a_{i,j}\, \hat{C}_o [I \quad -I] \begin{bmatrix} x_{i_k}(k) \\ \tilde{x}_{i_k}(k) \end{bmatrix}$$

$$- \hat{K}_f \sum_{j=1}^{N} a_{i,j}\, \hat{C}_o [I \quad -I] \begin{bmatrix} x_{j_k}(k) \\ \tilde{x}_{j_k}(k) \end{bmatrix}$$

Under the distributed observer-based consensus control protocol in (3.14), the composite system of (3.12) and (3.14) can be written in the state space form as follows:

$$\chi_i(k+1) = \underline{A}\,\chi_i(k) + \sum_{j=1}^{N} l_{i,j}\,\widehat{H}\,\chi_j(k), \quad i \in \mathcal{N} \tag{3.17}$$

where $\chi_i(k) = x_{i_k}(k) \otimes \widetilde{x}_{i_k}(k)$ is the augmented vector, i.e., $\chi_i(k) = \begin{bmatrix} x_{i_k}(k) \\ \widetilde{x}_{i_k}(k) \end{bmatrix}$. Hence we obtain

$$\chi_i(k+1) = \begin{bmatrix} \hat{A}_o & -\widehat{B}_o\widehat{K}_c \\ 0 & \hat{A}_o - \widehat{B}_o\widehat{K}_c \end{bmatrix} \chi_{i_k}(k) + \begin{bmatrix} 0 & 0 \\ \widehat{K}_f\hat{C}_o & -\widehat{K}_f\hat{C}_o \end{bmatrix} \sum_{j=1}^{N} a_{i,j}\ \chi_{i_k}(k)$$

$$- \begin{bmatrix} 0 & 0 \\ \widehat{K}_f\hat{C}_o & -\widehat{K}_f\hat{C}_o \end{bmatrix} \sum_{j=1}^{N} a_{i,j}\ \chi_{j_k}(k)$$

For convenience, denote

$$\underline{A} = \begin{bmatrix} \hat{A}_o & -\widehat{B}_o\widehat{K}_c \\ 0 & \hat{A}_o - \widehat{B}_o\widehat{K}_c \end{bmatrix}, \qquad \widehat{H} = \begin{bmatrix} 0 & 0 \\ \widehat{K}_f\hat{C}_o & -\widehat{K}_f\hat{C}_o \end{bmatrix}$$

Alternatively we can write

$$\chi_i(k+1) = \underline{A}\,\chi_i(k) + \widehat{H} d_i\,\chi_i(k) - [a_{i,1}\widehat{H} \quad a_{i,2}\widehat{H} \quad \cdots \quad \cdots \quad a_{i,N}\widehat{H}]\,\chi_i(k)$$

where

$$d_i = \sum_{j=1}^{N} a_{i,j}$$

Define the augmented state vector

$$\underline{\chi}(k+1) = \begin{bmatrix} \chi_1(k+1) \\ \chi_2(k+1) \\ \vdots \\ \vdots \\ \chi_N(k+1) \end{bmatrix} \quad \text{or} \quad \underline{\chi}(k) = \begin{bmatrix} \chi_1(k) \\ \chi_2(k) \\ \vdots \\ \vdots \\ \chi_N(k) \end{bmatrix}$$

Then we have

$$\underline{\chi}(k+1) == \begin{bmatrix} \underline{A} & 0 & 0 & 0 & 0 & 0 \\ 0 & \underline{A} & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \underline{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & \underline{A} \end{bmatrix} \underline{\chi}(k) + \begin{bmatrix} \widehat{H}d_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \widehat{H}d_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \widehat{H}d_N \end{bmatrix} \underline{\chi}(k)$$

$$- \begin{bmatrix} a_{11}\widehat{H} & a_{12}\widehat{H} & a_{13}\widehat{H} & \cdots & \cdots & a_{1N}\widehat{H} \\ a_{21}\widehat{H} & a_{22}\widehat{H} & a_{23}\widehat{H} & \cdots & \cdots & \vdots \\ a_{31}\widehat{H} & a_{32}\widehat{H} & a_{33}\widehat{H} & \cdots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N1}\widehat{H} & \cdots & \cdots & \cdots & \cdots & a_{NN}\widehat{H} \end{bmatrix} \underline{\chi}(k) \qquad (3.18)$$

By the definition of the Laplacian matrix, we have a compact form below:

$$\underline{\chi}(k+1) = (I_N \otimes \underline{A} + \mathcal{L}_G \otimes \widehat{H})\, \underline{\chi}(k) \qquad (3.19)$$

Let $\eta = n + n_f$ with n and n_f the state dimensions of $x_i(k)$ and $x_{i_F}(k)$, respectively. Then $\underline{A} \in \mathbb{R}^{2\eta\times 2\eta}$. Denote $\mathbf{1} \in \mathbb{R}^N$ as vector of elements 1 and $\mathbf{0} \in \mathbb{R}^N$ as vector of elements 0.

In consensusability each component should reach its average value. We define

$$\overline{\chi}(k) := \frac{1}{N}\sum_{i=1}^{N}\chi_i(k) = \frac{1}{N}\,[I \quad I \quad \cdots \quad \cdots \quad I]\, \underline{\chi}(k) \qquad (3.20)$$

Multiplying $\frac{1}{N}$ $[I \quad I \quad \cdots \quad \cdots \quad I]$ to (3.19) on both sides yields

$$\bar{\chi}(k+1) = \frac{1}{N}\,[\underline{A} \quad \underline{A} \quad \cdots \quad \cdots \quad \underline{A}]\,\underline{\chi}(k)$$

$$+\frac{1}{N}\,[\underline{A} \quad \underline{A} \quad \cdots \quad \cdots \quad \underline{A}]\begin{bmatrix} l_{11}\hat{H} & l_{12}\hat{H} & l_{13}\hat{H} & \cdots & \cdots & l_{1N}\hat{H} \\ l_{21}\hat{H} & \ddots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{N1}\hat{H} & \cdots & \cdots & \cdots & \cdots & l_{NN}\hat{H} \end{bmatrix}\underline{\chi}(k)$$

$$= \frac{1}{N}\sum_{i=1}^{N}\underline{A}\,\chi_i(k) + \frac{1}{N}\sum_{i=1}^{N}\underline{A}\,l_{ij}\hat{H}\,\chi_i(k) = \underline{A}\,\frac{1}{N}\sum_{i=1}^{N}\chi_i(k) + \underline{A}\,\frac{1}{N}\sum_{i=1}^{N}l_{ij}\hat{H}\,\chi_i(k)$$

where l_{ij} is the (i,j)th element of $\mathcal{L}_G$. Hence

$$\sum_{i=1}^{N} l_{ij} = 0 \qquad (3.21)$$

leads to $\bar{\chi}(k+1) = \underline{A}\,\bar{\chi}(k)$ (3.22)

Also we can write (3.22) in the form below:

$$\begin{bmatrix} \bar{\chi}(k+1) \\ \vdots \\ \vdots \\ \vdots \\ \bar{\chi}(k+1) \end{bmatrix} = \begin{bmatrix} \underline{A} & 0 & 0 & 0 & 0 & 0 \\ 0 & \underline{A} & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \underline{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & \underline{A} \end{bmatrix}\begin{bmatrix} \bar{\chi}(k) \\ \vdots \\ \vdots \\ \vdots \\ \bar{\chi}(k) \end{bmatrix} + \begin{bmatrix} l_{11}\hat{H} & l_{12}\hat{H} & l_{13}\hat{H} & \cdots & \cdots & l_{1N}\hat{H} \\ l_{21}\hat{H} & \ddots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{N1}\hat{H} & \cdots & \cdots & \cdots & \cdots & l_{NN}\hat{H} \end{bmatrix}\begin{bmatrix} \bar{\chi}(k) \\ \vdots \\ \vdots \\ \vdots \\ \bar{\chi}(k) \end{bmatrix}$$

The above holds in the light of (3.21). So we write (3.22) in the following form:

$$\begin{bmatrix} \overline{\chi}(k+1) \\ \vdots \\ \vdots \\ \vdots \\ \overline{\chi}(k+1) \end{bmatrix} = (I_N \otimes \underline{A} + \mathcal{L}_G \otimes \widehat{H}) \begin{bmatrix} \overline{\chi}(k) \\ \vdots \\ \vdots \\ \vdots \\ \overline{\chi}(k) \end{bmatrix}$$

Now setting $\delta\chi(k) = \begin{bmatrix} \chi_1(k) \\ \vdots \\ \vdots \\ \vdots \\ \chi_N(k) \end{bmatrix} - \begin{bmatrix} \overline{\chi}(k) \\ \vdots \\ \vdots \\ \vdots \\ \overline{\chi}(k) \end{bmatrix}$ yeilds

$$\delta\underline{\chi}(k+1) = (I_N \otimes \underline{A} + \mathcal{L}_G \otimes \widehat{H})\delta\,\underline{\chi}(k) \tag{3.23}$$

Consensusability depends on the stability of $(I_N \otimes \underline{A} + \mathcal{L}_G \otimes \widehat{H})$. Let $\mathcal{L}_G = U\Lambda U'$ be eigenvalue decomposition. The symmetry of $\mathcal{L}_G$ implies that $U' = U^{-1}$. It follows that $U'\mathcal{L}_G U = diag(\lambda_1, \lambda_2, \dots \dots .., \lambda_N)$ with $\lambda_1 = 0$. Set the similarity transform matrix by

$$S = U \otimes I_\eta = \begin{bmatrix} U_{11}I & \cdots & U_{1N}I \\ \vdots & \ddots & \vdots \\ U_{N1}I & \cdots & U_{NN}I \end{bmatrix}.$$

Then it can be shown that

$$S'(I_N \otimes \underline{A} + \mathcal{L}_G \otimes \widehat{H})S = \begin{bmatrix} \underline{A} & 0 & 0 & 0 & 0 & 0 \\ 0 & \underline{A} & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \underline{A} & 0 \\ 0 & 0 & 0 & 0 & 0 & \underline{A} \end{bmatrix} + \begin{bmatrix} \lambda_1\widehat{H} & 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_2\widehat{H} & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda_N\widehat{H} \end{bmatrix}$$

$$= diag(\underline{A}, \underline{A} + \lambda_2\widehat{H}, \dots \dots \dots .., \underline{A} + \lambda_N\widehat{H})$$

Let u_1 be the eigenvector corresponding to $\lambda_1 = 0$. Then

$$u_1 = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \frac{1}{\sqrt{N}}$$

After transformation $\widehat{\delta\,\underline{\chi}}(k) = S'\delta\,\underline{\chi}(k)$, the state space system decomposes into N subsystems.

The first one is governed by $\delta\widehat{\underline{\chi}_1}\,(k+1) = \underline{A}\delta\widehat{\underline{\chi}_1}\,(k)$. There holds $\widehat{\delta\,\underline{\chi}}(k) = 0$ by

$$\frac{1}{\sqrt{N}}[I \quad \cdots \quad \cdots \quad I]\,\delta\,\underline{\chi}(k+1) = \frac{1}{\sqrt{N}}[I \quad \cdots \quad \cdots \quad I]\begin{bmatrix} \chi_1(k) \\ \vdots \\ \vdots \\ \vdots \\ \chi_N(k) \end{bmatrix} - \begin{bmatrix} \overline{\chi}(k) \\ \vdots \\ \vdots \\ \vdots \\ \overline{\chi}(k) \end{bmatrix} = 0$$

The ith subsystem is $\widehat{\delta\underline{\chi}_{1\iota}}\,(k+1) = \left[\,\underline{A} + \lambda_i\,\widehat{H}\,\right]\delta\widehat{\underline{\chi}_\iota}\,(k)$ for $2 \le i \le N$.

For consensusability $(\underline{A} + \lambda_i\,\widehat{H})$ is required to be a should be a stability matrix for $i = 2,3,\ldots\ldots..,N$. We note that

$$(\underline{A} + \lambda_i\,\widehat{H}) = \begin{bmatrix} \hat{A}_o & -\hat{B}_o\widehat{K}_c \\ \lambda_i\widehat{K}_f\hat{C}_o & \hat{A}_o - \hat{B}_o\widehat{K}_c - \lambda_i\widehat{K}_f\hat{C}_o \end{bmatrix}$$

A similarity transformation is applied to the above matrix as follows

$$\begin{bmatrix} I & -I \\ 0 & I \end{bmatrix}\begin{bmatrix} \hat{A}_o & -\hat{B}_o\widehat{K}_c \\ \lambda_i\widehat{K}_f\hat{C}_o & \hat{A}_o - \hat{B}_o\widehat{K}_c - \lambda_i\widehat{K}_f\hat{C}_o \end{bmatrix}\begin{bmatrix} I & -I \\ 0 & I \end{bmatrix}^{-1}$$

Note that, $\begin{bmatrix} I & -I \\ 0 & I \end{bmatrix}^{-1} = \begin{bmatrix} I & I \\ 0 & I \end{bmatrix}$. Then the above is the same as

$$\begin{bmatrix} \hat{A}_o - \lambda_i\widehat{K}_f\hat{C}_o & -\hat{A}_o + \lambda_i\widehat{K}_f\hat{C}_o \\ \lambda_i\widehat{K}_f\hat{C}_o & \hat{A}_o - \hat{B}_o\widehat{K}_c - \lambda_i\widehat{K}_f\hat{C}_o \end{bmatrix}\begin{bmatrix} I & I \\ 0 & I \end{bmatrix} = \begin{bmatrix} \hat{A}_o - \lambda_i\widehat{K}_f\hat{C}_o & 0 \\ \lambda_i\widehat{K}_f\hat{C}_o & \hat{A}_o - \hat{B}_o\widehat{K}_c \end{bmatrix}$$

It follows that the consensusability under the control protocol (3.14) is equivalent to stability of $\hat{A}_o - \hat{B}_o\widehat{K}_c$ and of $\hat{A}_o - \lambda_i\widehat{K}_f\hat{C}_o$ for $i = 2, \ldots\ldots, N$.

In summary, we can state that the discrete-time MAS in (3.9) is consensusable under the control protocol (3.14), if and only if there exists a common filter $F(z)$ of size $p \times p$ and stabilizing state feedback $\widehat{K}_c$ and state estimation gain $\widehat{K}_f$ such that $\hat{A}_o - \lambda_i\widehat{K}_f\hat{C}_o$ is a stability matrix for $i = 2, \ldots\ldots, N$.

The design procedure is discussed in the following section.

3.4 Controller Design Algorithm for Single Input Systems

Recall the Mahler measure defined in Chapter 1. The following result is known from [19].

Lemma 3.4.1 Let $T_0(z) = K(zI - A + BK)^{-1}B$ be the complementary sensitivity under the state feedback where $K', B \in \mathbb{R}^n$. If (A, B) is stabilizable, then

$$\gamma_{opt} = inf_{K\in\mathbb{R}^{1\times n}} \|T_0\|_{H_\infty} = \mu(A)$$

For each $\gamma > \gamma_{opt}$, a stabilizing state feedback gain achieving $\|T_0\|_{H_\infty} < \gamma$ is given by

$$K = [I + (1 - \gamma^{-2})B'_{\lambda_2}XB_{\lambda_2}]^{-1}B'_{\lambda_2}XA$$

$$B'_{\lambda_2}X\, B_{\lambda_2} < \gamma^2$$

where $X \geq 0$ is the stabilizing solution to the following discrete-time ARE:

$$X = A'X[I + (1 - \gamma^{-2})B_{\lambda_2}B'_{\lambda_2}X]^{-1}A$$

The following is the main result of this section.

Theorem 3.4.1 Let $C = I$ and $F(z)$ is a stable filter. The discrete time multi agents in (1.1) are consensusable under the protocol (3.3), if

$$\frac{\lambda_N}{\lambda_2} < \left(\frac{\mu(A)+1}{\mu(A)-1}\right)^2 \tag{3.24}$$

Before giving the proof, we provide the algorithm for design of K and $F(z)$ that achieve consensusability under the above condition.

1) Select $\gamma > \mu(A)$ according to

$$\frac{\gamma+1}{\gamma-1} = \sqrt{\frac{\lambda_N}{\lambda_2}} \Rightarrow \gamma = \left(\frac{\sqrt{\frac{\lambda_N}{\lambda_2}}+1}{\sqrt{\frac{\lambda_N}{\lambda_2}}-1}\right) > \mu(A)$$

2) Solve the stabilizing solution $X = A'X[I + (1-\gamma^{-2})B_{\lambda_2}B'_{\lambda_2}X]^{-1}A\ ;\ X \geq 0$

   Set the state feedback gain according to

$$K = [I + (1-\gamma^{-2})B'_{\lambda_2}XB_{\lambda_2}]^{-1}B'_{\lambda_2}XA$$

   That achieves $\|T_{\lambda_2}\|_{H_\infty} < \gamma$. Note that

$$\frac{\lambda_N}{\lambda_2} = \left(\frac{\gamma+1}{\gamma-1}\right)^2 < \left(\frac{\mu(A)+1}{\mu(A)-1}\right)^2 \tag{3.25}$$

3) Find the complementary sensitivity under state feedback,

$$T_{\lambda_2}(z) = K(zI - A + B_{\lambda_2}K)^{-1}B_{\lambda_2}$$

4) After K and $T_{\lambda_2}(z)$ are available, calculate the stable filter $F(z)$ according to

$$F(z) = \frac{(1-\gamma^{-1})^2}{1-\gamma^{-2}T_{\lambda_2}(z)}$$

The above stable filter can further be written as

$$F(z) = (1-\gamma^{-1})^2[1-\gamma^{-2}K\left(zI-A+B_{\lambda_2}K\right)^{-1}B_{\lambda_2}]^{-1}$$

$$= (1-\gamma^{-1})^2[1+\gamma^{-2}K(zI-A+(1-\gamma^{-2})B_{\lambda_2}K)^{-1}B_{\lambda_2}]$$

For this reason, state realization of $F(z)$ can be obtained as

$$A_F = A-(1-\gamma^{-2})B_{\lambda_2}K \qquad B_F = B_{\lambda_2}$$

$$C_F = (1-\gamma^{-1})^2\,\gamma^{-2}K; \qquad D_F = (1-\gamma^{-1})^2$$

Proof: As per Lemma 3.2.1 the consensusability problem is solvable, if and only if $\hat{A}-\lambda_i\hat{B}\hat{K}$ is a stability matrix for $i = 2,3,\ldots,N$. Construct K and $F(z)$ to solve the consensusability problem under the condition in Theorem 3.4.1. We assume that A has at least one eigenvalue outside the unit circle. Otherwise the consensusability becomes trivial.

The facts that $\left\|T_{\lambda_2}\right\|_{H\infty} < \gamma$ for $\gamma > \gamma_{opt} = \mu(A) > 1$ imply that $F(z)$ is a stable transfer function, by small gain theorem. Next we show that $\hat{A}-\lambda_i\hat{B}\hat{K}$ is a stability matrix for $i \geq 2$ under the condition in (3.24) that ensures (3.25).

Stability condition is given by:

$$\det(zI-(\hat{A}-\lambda_i\hat{B}\hat{K})) \neq 0 \;\forall\; |z| \geq 1 \; and \; i \geq 2$$

$$\Leftrightarrow \det((zI - \hat{A}) + \lambda_i \hat{B}\hat{K}) \neq 0 \ \forall \ |z| \geq 1 \ and \ i \geq 2$$

By root locus argument it can be shown that,

$$\Leftrightarrow \det(I + \lambda_i (zI - \hat{A})^{-1}\hat{B}\hat{K}) \neq 0 \ \forall \ |z| \geq 1 \ and \ i \geq 2$$

$$\Leftrightarrow 1 + \lambda_i \hat{K}(zI - \hat{A})^{-1}\hat{B}) \neq 0 \ \forall \ |z| \geq 1 \ and \ i \geq 2$$

From (3.5), $\hat{K} = K\hat{C}$

$$\Leftrightarrow 1 + \lambda_i K\hat{C}(zI - \hat{A})^{-1}\hat{B}) \neq 0 \ \forall \ |z| \geq 1 \ and \ i \geq 2$$

From (3.6),

$$\hat{C}\ (zI - \ \hat{A})^{-1}\ \hat{B} = P(z)F(z)$$

$$\Leftrightarrow 1 + \lambda_i KP(z)F(z) \ \neq 0 \ \forall \ |z| \geq 1 \ and \ i \geq 2$$

Denote,

$$G(z) = K(zI - A)^{-1}\mathrm{B}_{\lambda_2} \text{ and } \lambda = \frac{\lambda}{\lambda_2}\lambda_2 = (1 + \Delta)\lambda_2 \quad ; \Delta \geq 0$$

Then the above is equivalent to

$$\alpha(z) = 1 + (1 + \Delta)\mathrm{K}(zI - A)^{-1}\mathrm{B}_{\lambda_2}F(z) \neq 0 \ \forall \ |z| \geq 1$$

$$= 1 + \mathrm{K}(zI - A)^{-1}\mathrm{B}_{\lambda_2}F(z) + \Delta\mathrm{K}(zI - A)^{-1}\mathrm{B}_{\lambda_2}F(z) \neq 0 \ \forall \ |z| \geq 1$$

$$= 1 + \frac{\Delta\mathrm{K}(zI - A)^{-1}\mathrm{B}_{\lambda_2}F(z)}{1 + \mathrm{K}(zI - A)^{-1}\mathrm{B}_{\lambda_2}F(z)} \neq 0 \ \ \forall \ |z| \geq 1$$

$$= 1 + \frac{\Delta \mathrm{G(z)F(z)}}{1 + G(z)F(z)} \neq 0 \; \forall \; |z| \geq 1$$

$$= 1 + \frac{\Delta \mathrm{G(z)} \; \frac{(1-\gamma^{-1})^2}{1-\gamma^{-2}\mathrm{T}_{\lambda_2}(\mathrm{z})}}{1 + G(z) \; \frac{(1-\gamma^{-1})^2}{1-\gamma^{-2}\mathrm{T}_{\lambda_2}(\mathrm{z})}}$$

$$= 1 + \frac{\Delta \mathrm{G(z)}(1-\gamma^{-1})^2}{1-\gamma^{-2}\mathrm{T}_{\lambda_2}(\mathrm{z}) + G(z)(1-\gamma^{-1})^2}$$

We know that,

$$\mathrm{T}_{\lambda_2}(\mathrm{z}) = K(zI - A)^{-1}(I + \mathrm{B}_{\lambda_2}K(zI - A)^{-1})^{-1}\mathrm{B}_{\lambda_2}$$

$$= K(zI - A)^{-1}\mathrm{B}_{\lambda_2}(I + K(zI - A)^{-1}\mathrm{B}_{\lambda_2})^{-1} = \frac{G(z)}{1 + G(z)}$$

Then,

$$\alpha(z) = 1 + \Delta \mathrm{T(z)}$$

Consider

$$\mathrm{T(z)} = \frac{\mathrm{G(z)}(1-\gamma^{-1})^2}{1-\gamma^{-2}\frac{G(z)}{1+G(z)} + G(z)(1-\gamma^{-1})^2}$$

$$= \frac{(1 + G(z))G(z)(1-\gamma^{-1})^2}{1 + G(z) - \gamma^{-2}G(z) + G(z)(1-\gamma^{-1})^2 + G(z)^2(1-\gamma^{-1})^2}$$

For convenience, denote the denominator of T(z) by $den(z)$. Then

$$den(z) = 1 + G(z) - \gamma^{-2}G(z) + G(z)(1-\gamma^{-1})^2 + G(z)^2(1-\gamma^{-1})^2$$

$$= 1^2 + [G(z)(1-\gamma^{-1})]^2 + G(z)[1-\gamma^{-2} + (1-\gamma^{-1})^2]$$

$$= 1^2 + [G(z)(1-\gamma^{-1})]^2 + 2G(z)(1-\gamma^{-1})$$

$$= [1 + G(z)(1-\gamma^{-1})]^2 = [1 + G(z) - \gamma^{-1}G(z)]^2$$

$$= \left(1 + G(z)\right)^2 \left[1 - \frac{\gamma^{-1}G(z)}{1+G(z)}\right]^2$$

Substituting the above into the expression of T(z) gives

$$\mathrm{T(z)} = \frac{G(z)(1-\gamma^{-1})^2}{(1+G(z))\left[1 - \frac{\gamma^{-1}G(z)}{1+G(z)}\right]^2} = \frac{\mathrm{T}_{\lambda_2}(\mathrm{z})(1-\gamma^{-1})^2}{\left[1-\gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})\right]^2} \quad \in \mathcal{H}_\infty$$

$$\alpha(z) = 1 + \Delta \frac{\mathrm{T}_{\lambda_2}(\mathrm{z})(1-\gamma^{-1})^2}{\left[1-\gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})\right]^2}$$

Next we claim that: $\alpha(z) \coloneqq 1 + \Delta\mathrm{T(z)} \neq 0 \ \ \forall\, |z| \geq 1$

To prove the above claim we have at $\lambda = \lambda_2$; $\alpha(z) = 1 \neq 0 \quad \forall\, |z| \geq 1$. At $\lambda = \lambda_N$,

$$\Delta = \frac{\lambda_N - \lambda_2}{\lambda_2} = \frac{\lambda_N}{\lambda_2} - 1$$

Using (3.24),

$$\Delta = \left(\frac{\gamma+1}{\gamma-1}\right)^2 - 1 = \frac{4\gamma^{-1}}{(1-\gamma^{-1})^2}$$

Hence we can obtain

$$\alpha(z) = 1 + \frac{4\gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})}{\left[1 - \gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})\right]^2} = \frac{\left[1 - \gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})\right]^2 + 4\gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})}{\left[1 - \gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})\right]^2} = \left[\frac{1 + \gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})}{1 - \gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})}\right]^2$$

The fact that $\|T_{\lambda_2}\|_{H_\infty} < \gamma$ implies that

$$\gamma^{-1}\left|T_{\lambda_2}(z)\right| < 1 \qquad \forall\, |z| \geq 1$$

By a known property of bilinear transform, the above equation implies that

$$Re\left[\frac{1 + \gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})}{1 - \gamma^{-1}\mathrm{T}_{\lambda_2}(\mathrm{z})}\right]^2 > 0 \;\; \forall\, |z| \geq 1$$

Thus, Theorem 3.4.1 provides a procedure for designing the stable filter $F(z)$ and the state feedback gain K for consensus control when the following consensusability condition holds:

$$\frac{\lambda_N}{\lambda_2} < \left(\frac{\mu(A) + 1}{\mu(A) - 1}\right)^2$$

In conclusion to this chapter, we summarize as follows:

1) A distributed feedback control protocol is proposed.
2) State feedback and output feedback control for discrete-time MAS is studied. We show that the MAS are consensusable under output feedback control.
3) A step by step procedure for designing a stable filter $F(z)$ and the state feedback gain K is given such that (3.9) holds true.

CHAPTER 4
SIMULATION RESULTS

This chapter employs a simulation example to illustrate consensus control, and the results presented in the previous chapter. It assumes that the agent plant is modeled by the transfer function described by

$$P(z) = \frac{2}{z-2} = C\,(zI - A)^{-1}\,B$$

with $A = 2,\ B = 2\ and\ C = 1$. The undirected feedback graph is assumed to have the following adjacent matrix:

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Number of agents each neighbor is having is given by

$$D = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

Then the Laplacian Matrix, $\mathcal{L}_{\mathcal{G}} = D - \mathcal{A} = USU'$ is obtained as

$$\mathcal{L}_{\mathcal{G}} = \begin{bmatrix} 2 & -1 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & 0 & -1 & 2 \end{bmatrix}$$

It can be verified that $\lambda_2 = 1$ and $\lambda_N = 4$.

For the agent model $P(z)$, it is easy to see that $\mu(A) = 2$. We set γ according to (3.25) yielding $\gamma = 3 > \mu(A)$ and satisfying (3.24). Solving the stabilizing solution to ARE

$$X = A'X[I + (1-\gamma^{-2})B_{\lambda_2}B'_{\lambda_2}X]^{-1}A \; ; \; X \geq 0$$

yields $X = \frac{27}{32}$. The state feedback gain K such that $\|T_{\lambda_2}\|_{H_\infty} < \gamma$ is given by

$$K = [I + (1-\gamma^{-2})B'_{\lambda_2}XB_{\lambda_2}]^{-1}B'_{\lambda_2}XA = 0.8438$$

The complementary sensitivity under state feedback is given by

$$T_{\lambda_2}(z) = K(zI - A + B_{\lambda_2}K)^{-1}B_{\lambda_2}$$

Direct calculation yields

$$T_{\lambda_2}(z) = \frac{1.688}{z + 0.3125}$$

It can be verified that $\|T_{\lambda_2}\|_{H_\infty} = 2.4545 < \gamma$ =3. Recall the stable filter

$$F(z) = \frac{(1-\gamma^{-1})^2}{1-\gamma^{-2}T_{\lambda_2}(z)}$$

Substituting the expression of $T_{\lambda_2}(z)$ yields

$$F(z) = \frac{0.4444z - 0.1389}{z - 0.5}$$

that is indeed stable.

We now construct a multi-agent system with $N = 6$ agents. Define:

$$G = I \otimes [F(z)\ K\ P(z)]$$

Performing Singular Value Decomposition of $\mathcal{L}_{\mathcal{G}}$ yields

$$U = \begin{bmatrix} 0.4082 & -0.5774 & 0 & -0.5774 & 0 & -0.4082 \\ 0.4082 & -0.2887 & -0.5 & 0.2887 & 0.5 & 0.4082 \\ 0.4082 & 0.2887 & -0.5 & 0.2887 & -0.5 & -0.4082 \\ 0.4082 & 0.5774 & 0 & -0.5774 & 0 & 0.4082 \\ 0.4082 & 0.2887 & 0.5 & 0.2887 & 0.5 & -0.4082 \\ 0.4082 & -0.2887 & 0.5 & 0.2887 & -0.5 & 0.4082 \end{bmatrix};$$

$$S = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix}$$

It is important to note that the consensusability does not ensure stability of the underlying MAS, if the agent model is unstable. That is, the average of the state can diverge to infinity. For this reason, a reference signal $r(t)$ is introduced with $y(t)$ as the output signal. The error of the system may be defined as $e(t) = r(t) - y(t)$ where the reference signal is the signal that the output $y(t)$ needs to track. The reference signal applied is a step input in our simulation.

We propose the following block diagram for implementing the MAS.

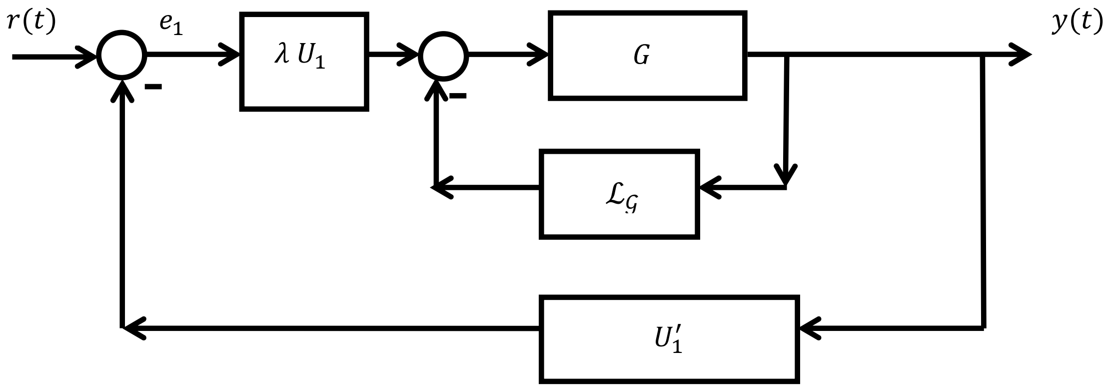

Figure 4.1 Proposed block diagram for implementing the MAS

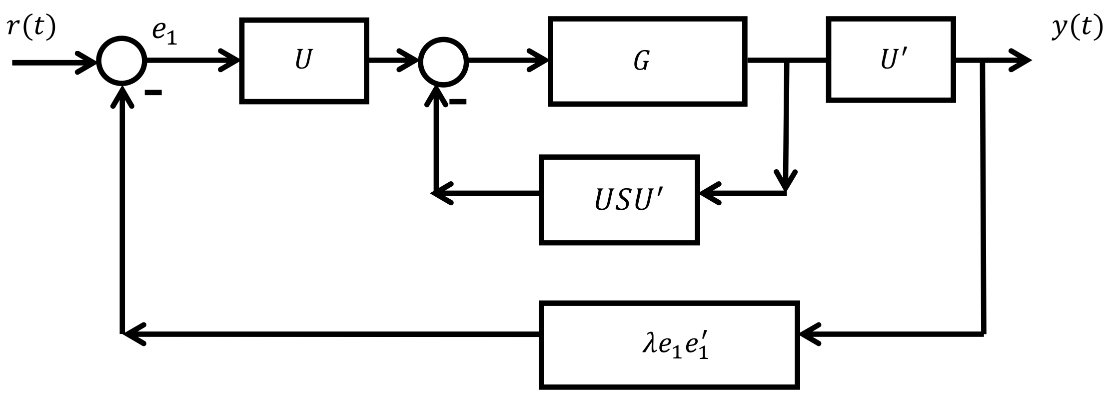

Figure 4.2 Equivalent block diagram

where $e_1' = [1 \quad 0 \quad \cdots \quad 0 \quad 0]$

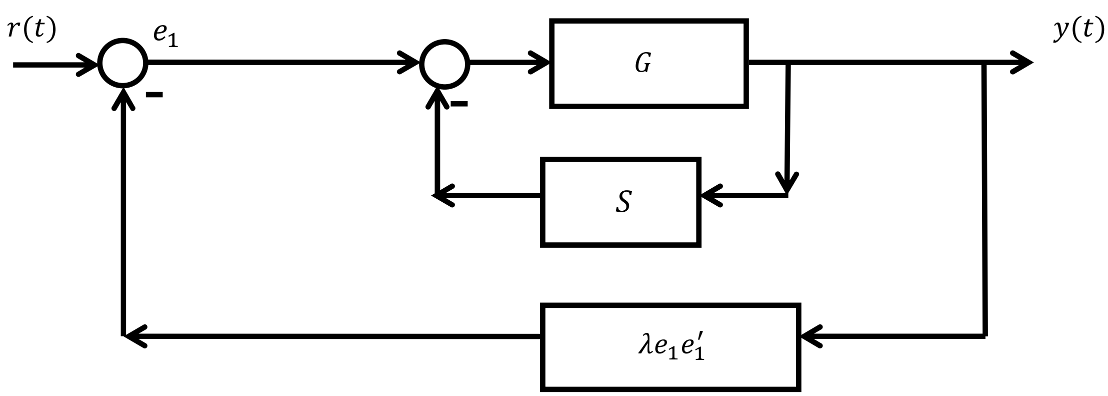

Figure 4.3 Equivalent block diagram

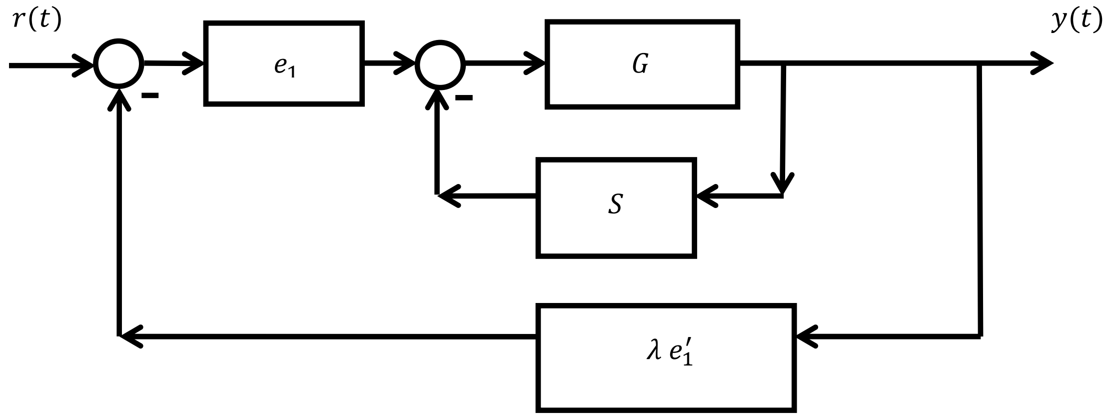

Figure 4.4 Final equivalent block diagram for implementing the MAS

Note that we now have effectively changed the Laplacian matrix to $\hat{\mathcal{L}} = U\hat{S}U'$ with the $\hat{S} = diag(\lambda, 1,1,3,3,4)$. By taking $1 \leq \lambda \leq 4$, stability of the MAS is ensured, and $y(t) \rightarrow r(t)$ asymptotically.

The simulink model for $N = 6$ MAS is shown below. A step input is applied as refernce.

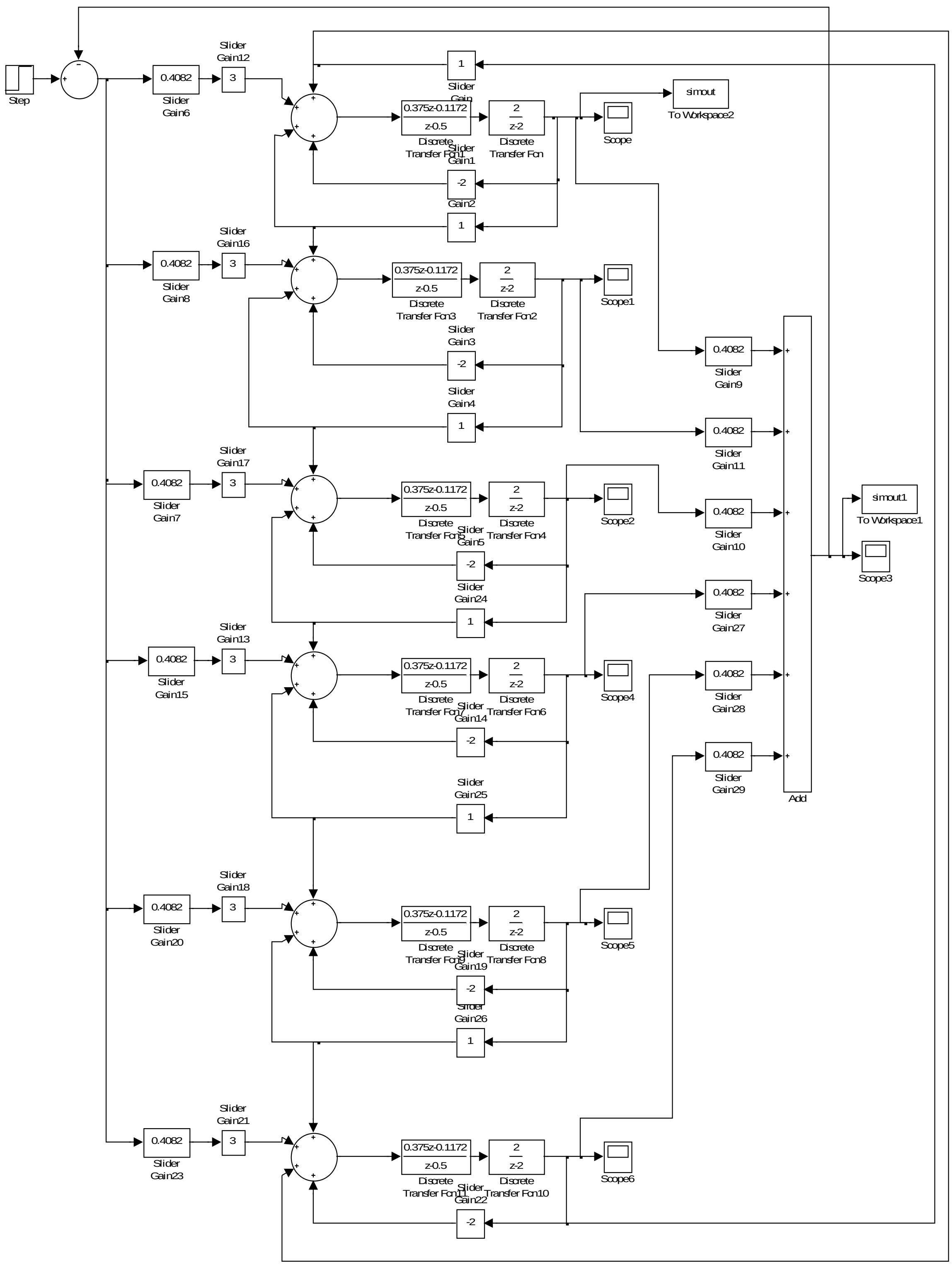

Figure 4.5 Simulink Model for $N = 6$ Multi Agent Systems

The time response characteristics after introducing K and $F(z)$ at one the agents is shown below:

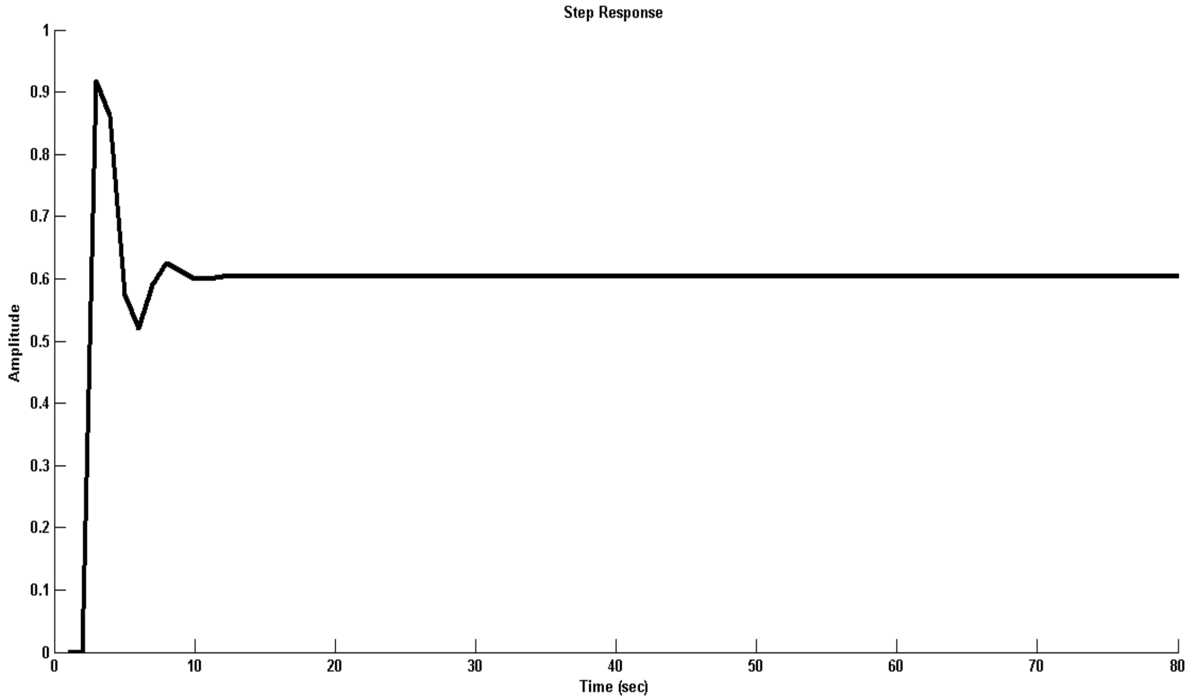

Figure 4.6 Time Response characteristics at the output of one of the agents after introducing K and $F(z)$

The Simulink model for $N = 6$ multi-agent systems is shown below in Figure 4.5. The block diagram rules are used for the implementation. Each agent can communicate only with its only two of its neighbors. The step response characteristic at one of the agent is shown in Figure 4.6 which tends to be stable. Hence, a consensus is achieved between the 6 agents.

CHAPTER 5
CONCLUSIONS

5.1 Summary

The thesis investigates the problem under what condition, the discrete-time MAS over undirected graph are consensusable. A reason is discussed and shows how the problem differs from continuous-time MAS and why the LQR method cannot be used for discrete-time MAS.

The common motivation behind the study of consensus problem is the rich history of consensus protocols in computer science. Many seemingly different problems that involve interconnection of dynamic agents happen to be closely related to consensus problems of MAS. In this thesis a discussion about the various application areas and results available in literature concerning these areas namely flocking theory, synchronization of complex networks, formation control and fast consensus are provided briefly.

The problem of consensusability has been studied for discrete-time multi-agent systems (SISO) under the undirected graph. The N agent dynamic systems are assumed to be homogenous and admit the same plant model. The consensusability condition is derived in terms of the Mahler measure of the agent system and the result shows that there is an improved consensusability by a power of two. A detailed algorithm is proposed to design distributed consensus feedback control protocols based on local information, when the consensusability holds. Also the case of output feedback is considered in which the consensusability problem becomes more complicated. To solve this problem, we decompose the problem into two parts i.e. state feedback and state estimation.

Simulation results are worked out to demonstrate the effectiveness of the established results as the states of all agents asymptotically converge to the average of their initial values. In fact a reference signal is introduced and the output of each agent is feedback to compare with the reference input. If the feedback gains are designed appropriately, then the stability of MAS is ensured in addition to the consensusability.

5.2 Future Research

For future research, the consensusability problem needs to be studied for MIMO agent systems. This problem is much harder. The existing results do not make use of the MIMO feature, and thus only conservative results are available. It is worth to considering resource allocation for consensus control of MIMO MAS by allocating the resources judiciously among different sub-channels of each agent. This will involve graph and controller co-design. This problem is currently under study.

Most of the existing consensus study is for homogenous agents. But in real world, most systems are heterogonous. In fact for practical systems, the agents coupled with each other have different dynamics because of various restrictions or depending on the common goal which they are trying to achieve together. In [32], the consensus problem of heterogeneous multi-agent system is considered. The authors have proposed a heterogeneous multi-agent systems composed of first-order and second-order integrators. Then the consensus problem of heterogeneous MAS is discussed with linear consensus protocol and saturated consensus protocol. By applying the graph theory and Lyapunov direct method, some sufficient conditions for consensus are established when the communication topologies are undirected connected graphs and leader-following networks. The future work will focus on the more complex consensus problem of heterogeneous multi-agent systems for example; heterogeneous multi-agent systems with delays, heterogeneous multi-agent systems under directed graphs/switching topologies/random networks, discrete-time heterogeneous multi-agent systems etc. It should be pointed out that even if all the agent systems are made by the same manufacturer, the system dynamics may change due to aging and working environments.

For truly heterogonous MAS, the state consensus may not be meaningful due to possible difference in their dynamics and state dimensions. Hence it makes more sense to consider output consensus. Output consensus is discussed in [33] for heterogeneous uncertain linear multi-agent systems limited to linear SISO systems and fixed network topology. Based on the output regulation theory, it is shown that the output consensus is reached if the (state) consensus is achieved within the internal models among the agent's controllers (even though the plant's outputs, rather than the internal model's outputs, are communicated). The internal models can be designed and embedded into the controller, which provides considerable flexibility to designers in terms of the type of signals that are agreed on among the agents. Although the authors have tried to solve the output consensus problem there is scope for future work. The proposed method has limitations which may arise when the relative degree of the plant goes high increasing the implementation complexity of the proposed controller. Also as the order of the high-gain observer increases, the proposed controller becomes sensitive to measurement noise. Finally, consideration of the problem under the time varying network topology as well as multi-output requires attention from researchers. Hence the study of output consensus will be more important than the state consensus.

Another important area for future research is application of consensus control. Although many applications are available, the one attracts our attention is control of the power and temperature of chip multi-processor (CMP). Power and temperature are important design constraints for high performance processors [34]. Existing work on thermal management focuses on open loop search and optimization strategies based on static models or heuristic-based closed loop solutions that rely on oversimplified control algorithms without any theoretical guarantees. The proposed algorithm in this thesis might be useful for controlling power and temperature in chip multi processors. In a network of agents each core is expected to have limited knowledge of both the environment and the state of other cores. These cores can influence their own state and interact with their environment according to their dynamics which determines their behavior. The availability of DVFS (dynamic voltage and frequency scaling) makes it possible to develop advanced management strategies for power and temperature control. In [34], the authors have proposed a design of a chip level power control algorithm which is based on Model Predictive Control (MPC) theory, which is advanced optimal MIMO control theory. In future work, MPC can be replaced by the consensus control which may achieve better results and may help to save energy.

REFERENCES

[1] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *Automatic Control, IEEE Transactions on,* vol. 49, pp. 1520-1533, 2004.

[2] R. Olfati-Saber and R. M. Murray, "Consensus Protocols for Networks of Dynamic Agents," in *Proceeding of the 2003 American Control Conference*, pp. 951-956.

[3] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *Automatic Control, IEEE Transactions on,* vol. 49, pp. 1465-1476, 2004.

[4] R. W. Beard, T. W. McLain, M. A. Goodrich, and E. P. Anderson, "Coordinated target assignment and intercept for unmanned air vehicles," *Robotics and Automation, IEEE Transactions on,* vol. 18, pp. 911-922, 2002.

[5] J. G. Bender, "An overview of systems studies of automated highway systems," *Vehicular Technology, IEEE Transactions on,* vol. 40, pp. 82-99, 1991.

[6] J. Russell Carpenter, "Decentralized control of satellite formations," *International Journal of Robust and Nonlinear Control,* vol. 12, pp. 141-161, 2002.

[7] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *44th IEEE Conf. Decision and Control and Eur. Control Conf.*, 2005, pp. 6698-6703.

[8] D. Spanos, R. Olfati-Saber, and R. M. Murray, "Dynamic consensus on mobile networks," in *16th IFAC World Congr., ,* Prague, Czech, 2005.

[9] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and Cooperation in Networked Multi-Agent Systems," *Proceedings of the IEEE,* vol. 95, pp. 215-233, 2007.

[10] N. A. Lynch, *Distributed Algorithms*. San Francisco, CA: Morgan Kaufmann, 1997.

[11] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: algorithms and theory," *Automatic Control, IEEE Transactions on,* vol. 51, pp. 401-420, 2006.

[12] C. W. Reynolds, "Flocks, herds, and schools: a distributed behavioral model," in *Computer Graphics (ACM SIGGRAPH '87 Conference Proceedings)*, 1987, pp. 25–34.

[13] L. Zhongkui, D. Zhisheng, C. Guanrong, and H. Lin, "Consensus of Multiagent Systems and Synchronization of Complex Networks: A Unified Viewpoint," *Circuits and Systems I: Regular Papers, IEEE Transactions on,* vol. 57, pp. 213-224, 2010.

[14] W. Lu and T. Chen, "New approach to synchronization analysis of linearly coupled ordinary differential systems," *Physica D: Nonlinear Phenomena,* vol. 213, pp. 214-230, 2006.

[15] L. M. Pecora and T. L. Carroll, "Synchronization in chaotic systems," *Physical Review Letters,* vol. 64, pp. 821-824, 1990.

[16] L. M. Pecora and T. L. Carroll, "Master Stability Functions for Synchronized Coupled Systems," *Physical Review Letters,* vol. 80, pp. 2109-2112, 1998.

[17] G. Guoxiang, "On the existence of linear optimal control with output feedback," *SIAM J. Contr and Optimiz.,* vol. 28, pp. 771-719, 1990.

[18] Y. Keyou and X. Lihua, "Consensusability of discrete-time multi-agent systems via relative output feedback," in *Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on*, 2010, pp. 1239-1244.

[19] G. Guoxiang, L. Marinovici, and F. Lewis, "Consensusability of discrete-time multi-agent systems under state feedback control," in *Control Conference (CCC), 2011 30th Chinese*, 2011, pp. 4889-4894.

[20] K. Zhou and J. C. Doyle, *Essentials of Robust Control*. NJ: Prentice-Hall, 1998.

[21] C. T. Chen, *Linear System Theory and Design*. New York: Oxford University Press, 1999.

[22] J. A. Bondy and U. S. R. Murty, *Graph Theory* vol. 244. New York: Springer-Verlag, 2008.

[23] R. Diestel, *Graph Theory* vol. 173. New York: Springer-Verlag, 1997.

[24] Y. Cao and W. Ren, "Multi-Agent Consensus Using Both Current and Outdated States with Fixed and Undirected Interaction," *J. Intell. Robotics Syst.,* vol. 58, pp. 95-106, 2010.

[25] M. Egerstedt, X. Hu, and A. Stotsky, "Control of mobile platforms using a virtual vehicle approach," *Automatic Control, IEEE Transactions on,* vol. 46, pp. 1777-1782, 2001.

[26] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Math. J.,* vol. 23, pp. 298-305, 1973.

[27] D. Swaroop and J. K. Hedrick, "Constant Spacing Strategies for Platooning in Automated Highway Systems," *Journal of Dynamic Systems, Measurement, and Control,* vol. 121, pp. 462-470, 1999.

[28] Z. Jin and R. M. Murray, "Consensus controllabilty for coordinated multiple vehicle control," presented at the The 6th International Conference on Cooperative Control and Optimaization, Gainesville, 2006.

[29] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature,* vol. 393, pp. 440-442, 1998.

[30] R. Olifati-Saber, "Ultrafast consensus in small-world networks," in *Am. Control Conf.* , 2005, pp. 2371-2378.

[31] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters,* vol. 52, pp. 65-78, 2004.

[32] Y. Zheng, Y. Zhu, and L. Wang, "Consensus of heterogeneous multi-agent systems," *Control Theory & Applications, IET,* vol. 5, pp. 1881-1888, 2011.

[33] K. Hongkeun, S. Hyungbo, and S. Jin Heon, "Output Consensus of Heterogeneous Uncertain Linear Multi-Agent Systems," *Automatic Control, IEEE Transactions on,* vol. 56, pp. 200-206, 2011.

[34] Y. Wang, K. Ma, and X. Wang, "Temperature-Constrained Power Control for Chip Multiprocessors with Online Model Estimation," *ISCA,* 2009.

APPENDIX: MATLAB PROGRAM

Matlab function for computing the controller:

```
function final_system_KF = program (gamma,C,A,B,lambda_2)

%% Input Plant Details
%gamma=3;C=1;A=2;B=2;
%lambda_2=1;

%% Solve the Ricatti Equation
B_lambda_2=lambda_2*B;
R=(1-(gamma)^-2)^-1;
[X,L,G] = dare(A,B,0,R);

%% Perfrom Check (< gamma^2)
check=B_lambda_2'*X*B_lambda_2;

%% Calculate K
K=(1+(1-gamma^(-2))*B_lambda_2'*X*B_lambda_2)^(-1)*B_lambda_2'*X*A;
A1=A-B_lambda_2*K;
B1=B_lambda_2;
C1=K;
D1=0;
sys=ss(A1,B1,C1,D1);
m=tf(sys);

%% Calculate F
F_num=(1-inv(gamma))^2;
F_den=1-(gamma)^-2*m;
F=F_num/F_den;
final_system_KF=K*F

%% Plot the output at each agent
hold on
figure(1)
plot(simout.signals.values);
```

VITA

Abhishek Pandey was born in March, 1987, in Dhanbad, India. He completed his undergraduate studies in Electrical and Electronics Engineering at Visvesvaraya Technological University, Belgaum, India, in June 2008. From 2008 to 2009, he worked as an Instructor for the Department of Electrical and Electronics Engineering at Dayananda Sagar College of Engineering, Bangalore, India. In 2009 he joined Wipro Technologies Ltd. as a Software Engineer. After a brief stint as a Software Developer he came to Louisiana State University in August 2010 to pursue graduate studies in Electrical and Computer Engineering Department, focusing on Systems Control. He is currently a candidate for Master's program and will be awarded degree in August 2012.