Average-Consensus in a Two-Time-Scale Markov System

by

Kevin James Topley
B.Sc., Dalhousie University, 2003
M.Math., Waterloo University, 2005

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

The Faculty of Graduate and Postdoctoral Studies
(Electrical and Computer Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA
(Vancouver)

November, 2014

© Kevin James Topley 2014
Abstract

In a spatially distributed network of sensors or mobile agents it is often required to compute the average of all local data collected by each member of the group. Obtaining the average of all local data is, for example, sufficient to conduct robust statistical inference; identify the group center of mass and direction of motion; or evenly assign a set of divisible tasks among processors. Due to the spatial distribution of the network, energy limitations and geographic barriers may render a data fusion center to be infeasible or highly inefficient in regard to averaging the local data. The problem of distributively computing the network average - also known as the average-consensus problem - has thus received significant attention in the signal processing and control research communities. Efforts in this direction propose and study distributed algorithms that allow every agent in the network to compute the global average via communication with only a subset of fellow agents.

The thesis will present a framework in which to analyze distributed algorithms for both dynamic and static consensus formation. For dynamic consensus we consider a two-time-scale Markov system wherein each sensor node can observe the state of a local Markov chain. Assuming each Markov chain has a stationary distribution with slowly switching regime, we show that a local stochastic approximation algorithm in conjunction with linear distributed averaging can imply that each node estimate converges weakly to the current average of all stationary distributions. Each node can thus track the average of all stationary distributions, provided the regime switching is sufficiently slow. We then consider static consensus formation when the inter-node communication pattern is a priori unknown and signals possess arbitrarily long time-delays. In this setting, four distributed algorithms are proposed and shown to obtain average-consensus under a variety of general communication conditions.
Preface

The work in this thesis is based primarily on the results contained in the following three journal papers:


All three of the above papers analyze algorithms for consensus formation in a network of sensor nodes. The first paper deals with a dynamic consensus based on local stochastic approximation algorithms operating in parallel at each node. The second two papers consider static consensus formation when the inter-node communication pattern is \textit{a priori} unknown and arbitrary (but finite) communication time-delays may occur.

In the current presentation of these results we have combined the framework of all three papers into a single comprehensive model. This approach allows the reader to see each algorithm as a special case of a larger algorithmic class and observation model. Furthermore, our approach sheds light on the variety of trade-offs that may exist between communication resource costs, signal pattern assumptions, and the convergence rate at which consensus formation occurs.

# Table of Contents

Abstract ......................................................... ii

Preface .......................................................... iii

Table of Contents ............................................... iv

List of Tables ................................................... vii

List of Figures ................................................... viii

List of Acronyms ............................................... ix

Acknowledgments ............................................... x

1 Introduction and Overview .................................... 1
  1.1 Introduction ............................................... 1
    1.1.1 An Illustrative Example ............................... 2
  1.2 Thesis Statement and Objectives .......................... 6
    1.2.1 Thesis Statement ..................................... 6
    1.2.2 Thesis Objectives .................................... 6
  1.3 Thesis Contributions and Organization .................. 7
    1.3.1 Summary of Contributions ............................ 7
    1.3.2 Thesis Organization .................................. 7

2 Average-Consensus Problem Descriptions .................... 9
  2.1 Preliminary Notations, Assumptions, and Definitions .... 9
    2.1.1 Consensus Protocols .................................. 10
    2.1.2 Communication Patterns .............................. 11
  2.2 The Average-Consensus Problems .......................... 12
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.1.1</td>
<td>Proof of Theorem 3.4.1</td>
<td>79</td>
</tr>
<tr>
<td>A.1.2</td>
<td>Proof of Theorem 3.4.2</td>
<td>82</td>
</tr>
<tr>
<td>A.1.3</td>
<td>Proof of Theorem 3.4.4</td>
<td>84</td>
</tr>
<tr>
<td>A.1.4</td>
<td>Proof of Theorem 3.4.6</td>
<td>84</td>
</tr>
<tr>
<td>A.1.5</td>
<td>Proof of Lemma 3.4.8</td>
<td>85</td>
</tr>
<tr>
<td>A.1.6</td>
<td>Proof of Lemma 3.4.9</td>
<td>88</td>
</tr>
<tr>
<td>A.1.7</td>
<td>Proof of Lemma 3.4.10</td>
<td>88</td>
</tr>
<tr>
<td>A.1.8</td>
<td>Proof of Corollary 3.4.11</td>
<td>88</td>
</tr>
<tr>
<td>A.2</td>
<td>Proof of Results Contained in Chapter 4</td>
<td>88</td>
</tr>
<tr>
<td>A.2.1</td>
<td>Proof of Theorem 4.4.9</td>
<td>89</td>
</tr>
<tr>
<td>A.2.2</td>
<td>Proof of Theorem 4.4.10</td>
<td>103</td>
</tr>
<tr>
<td>A.3</td>
<td>Data Dissemination Results</td>
<td>110</td>
</tr>
<tr>
<td>A.4</td>
<td>Preliminary Definitions</td>
<td>111</td>
</tr>
<tr>
<td>A.5</td>
<td>Fixed Ring Graph Theorem</td>
<td>115</td>
</tr>
<tr>
<td>A.5.1</td>
<td>Statement of Theorem A.5.1</td>
<td>115</td>
</tr>
<tr>
<td>A.5.2</td>
<td>Proof of Theorem A.5.1</td>
<td>115</td>
</tr>
<tr>
<td>A.5.3</td>
<td>Supporting Lemmas for Theorem A.5.1</td>
<td>117</td>
</tr>
<tr>
<td>A.6</td>
<td>Corollary A.6.1.</td>
<td>118</td>
</tr>
<tr>
<td>A.6.1</td>
<td>Statement of Corollary A.6.1</td>
<td>118</td>
</tr>
<tr>
<td>A.6.2</td>
<td>Proof of Corollary A.6.1</td>
<td>118</td>
</tr>
<tr>
<td>A.6.3</td>
<td>Supporting Lemmas for Corollary A.6.1</td>
<td>119</td>
</tr>
<tr>
<td>A.7</td>
<td>Conjecture A.7.1.</td>
<td>119</td>
</tr>
<tr>
<td>A.7.1</td>
<td>Statement of Conjecture A.7.1</td>
<td>120</td>
</tr>
<tr>
<td>A.8</td>
<td>Extensions to Time-Delayed and Asynchronous Communication Sequences</td>
<td>120</td>
</tr>
<tr>
<td>A.8.1</td>
<td>Statement of Extensions to Cor.A.6.1 and Conj.A.7.1</td>
<td>120</td>
</tr>
<tr>
<td>A.8.2</td>
<td>Proof of Theorem A.8.1</td>
<td>120</td>
</tr>
<tr>
<td>A.8.3</td>
<td>Proof of Conjecture A.8.2</td>
<td>121</td>
</tr>
<tr>
<td>A.8.4</td>
<td>Supporting Lemmas for Thm.A.8.1 and Conj.A.8.2</td>
<td>121</td>
</tr>
</tbody>
</table>
List of Tables

3.1 Numerical results of the ratio comparisons between Models 1 and 2 . . . . . . . . 39
List of Figures

1.1 Underwater sensor nodes estimating the ambient concentration of biohazards. . . 3
1.2 A centralized underwater communication network. . . . . . . . . . . . . . . . . . 4
1.3 Time-varying Consensus with \textit{a priori} Unknown Communication Links. . . . . . 5
3.1 Sensor estimation of the average-consensus under both Models 1 and 2 . . . . . . 37
3.2 The absolute scaled error, plotted consecutively over the simulation time for each state in $S$ . ............................................................. 38
3.3 The analytic and sample variance of the scaled tracking error under both Models 1 and 2 . ............................................................. 39
3.4 Unordered Sensor Diffusion Coefficients, Absolute Error, and Functions of Weights 40
4.1 Illustrations of the $S\forall CC$ and $S\forall C\tilde{C}$ communication conditions . . . . 59
4.2 Convergence Under Example 3 in [1] . . . . . . . . . . . . . . . . . . . . . . . . . . . . 61
4.3 Convergence under Asynchronous Communication with Time-Varying Asymmetrical Time-Delays; Gossip, Random, and OH Protocols . . . . . . . . . . . . . 62
4.4 Convergence under Asynchronous Communication with Time-Varying Asymmetrical Time-Delays; IA, DA, and DDA Protocols . . . . . . . . . . . . . . . . . . . 63
4.5 Convergence under Asynchronous Communication with Time-Varying Asymmetrical Time-Delays and Unknown Network Size . . . . . . . . . . . . . . . . . . . . 64
A.1 A diagram of the Signal Notation in Def.A.4.10 . . . . . . . . . . . . . . . . . . . . . 113
A.2 A diagram of Thm.A.8.1 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 121
A.3 A diagram of Conj.A.8.2 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 121
List of Acronyms

trcp  time-repecting communication path
ntrcp non-time-repecting communication path
trcs  strongly-connected time-repecting communication sequence
ntrcs strongly-connected non-time-repecting communication sequence
rmcs  regime-modulated communication sequence
SVSC  singly $\mathcal{V}$ strongly connected
I$\mathcal{V}$SC infinitely $\mathcal{V}$ strongly connected
SVCC  singly $\mathcal{V}$ centrally connected
I$\mathcal{V}$CC infinitely $\mathcal{V}$ centrally connected
SV$\hat{\mathcal{C}}$C singly $\mathcal{V}$ dual-centrally connected
ACP  average-consensus problem
DLA  distributed linear averaging
DA  distributed-averaging algorithm
DDA  discretized distributed-averaging algorithm
IA  ideal distributed algorithm
OH  one-hop distributed algorithm
RG  ring-graph
RGF  fixed ring-graph
RG$\text{SC}$ strongly-connected ring-graph
RGF$\text{SC}$ strongly-connected fixed ring-graph
Acknowledgments

Firstly, I would to thank my supervisor Dr. V. Krishnamurthy, my supervisory committee, as well as the studies coordinator Dr. L. Lampe, for their helpful suggestions.

The main results presented in this thesis are derived from three journal papers, co-written by the author. The contribution to each journal paper by the respective authors is now described here.

The work *Consensus Formation in a Two-Time-Scale Markovian System* is a research paper based on collaboration with Dr. Yin from Wayne State University and Dr. Krishnamurthy. The conception and presentation of the project is credited to Dr. Krishnamurthy, whereas Dr. Yin developed all of the weak-convergence proofs. The author’s contributions to the project were the development of the dynamic consensus algorithm analyzed in the work; the connectivity conditions on the network graph shown to be necessary and sufficient for average-consensus under this algorithm; and finally all of the simulations, which illustrate a numerical implementation of the algorithm in a two-time-scale Markov system.

The algorithms and convergence proofs presented in the two papers *Average-Consensus Algorithms in a Deterministic Framework: Part 1 Strong Connectivity* and *Average-Consensus Algorithms in a Deterministic Framework: Part 2 Central Connectivity* were collectively developed by the author. Dr. Krishnamurthy aided substantially in the presentation of the works.
Chapter 1

Introduction and Overview

1.1 Introduction

Sensor networks are comprised of wireless networks of low-cost sensors that can be embedded in an environment for purposes such as security surveillance, monitoring and data aggregation. Over the past few decades, both sensor networks and ad hoc networks of mobile agents have attracted many researchers from diverse backgrounds in computer science, signal processing, robotics, and control theory [2–11]. Information processing in sensor networks is also closely related to traditional areas of research in systems and control such as multi-target tracking and sensor fusion [12–15].

The introduction of sensor, wireless ad hoc and peer-to-peer networks has motivated the design of distributed and fault-tolerant computation algorithms. This is largely due to the fact that distributed networks are characterized by numerous operational constraints. Such constraints include a lack of knowledge at each node in regard to the network topology [1,12,16]; the lack of a central node to facilitate the required computations [3,17–19]; or, in the case of sensor networks, the computational power and energy resources of each node may be very limited [20–22]. These constraints necessitate the design of distributed algorithms for computation wherein each node can send information to only a sparse subset of fellow nodes within any given time period. The purpose of such algorithms in this setting is to quickly determine at each node the value of a global computation using minimum exchange of local data. Examples of such global computations include finding the minimum value, mode value, and average value among all nodes.

A particularly useful instance of the above distributed problem is that of computing the average input at each node. A considerable amount of work in recent years has been devoted to ascertaining the communication conditions under which the average of node input values can be accurately estimated by all nodes in a sparsely connected network (a sample of the literature includes [16,17,19,23–26]). The distributed linear averaging protocol has been a primary focal point in this area of research due to its analytic tractability, decentralized nature, and minimum memory storage required at each node [8,27]. The basic paradigm of this protocol parameterizes the network communication at any given time instant by a graph $G(t) = \{V, E(t), W(t)\}$ with
1.1. Introduction

directed (possibly time-delayed) edge set $E(t) \subseteq V \times V$, node set $V = \{1, \ldots, n\}$, and weighted adjacency matrix $W(t)$ with elements $W_{ij}(t) = 0$ if $(j, i) \notin E(t)$. The weighted adjacency matrix $W(t)$ determines how each node consensus estimate is updated as a linear average of the estimates received from neighboring nodes. Under various conditions on the communication graph $G(t)$, research in this area have proven convergence results in regard to the distributed linear averaging of all local data. This computation can be used as a primary for more complex distributed calculations such as hypothesis testing in Bayesian networks, maximum likelihood estimation, solving general convex optimization problems, and load balancing within a network of processors. In this thesis we will provide convergence results under a general collection of “consensus protocols” that lead to an average-consensus estimate at all nodes. The nodes thereby obtain average-consensus by only a decoupled interaction among autonomous sets of agents that are collectively connected within a multi-agent system.

The convergence to a common value among a collection of autonomous agents was first technically analyzed in the seminal works [28, 29]. Although these papers preempted an entire line of research (i.e. [23, 25]) that expanded on these topics in various directions, a key difference between this line of work and the present thesis is that we focus on convergence to the average of each node’s input values, rather than simply convergence to an arbitrary common value. The latter is of little use in many practical applications that require utilization of the average value for purposes such as maximum likelihood estimation, hypothesis testing, or load balancing. The average-consensus problem was largely revitalized by the work [18], which emphasized the reality that in many applications an average of node initial values is required. In this thesis we focus on average-consensus, and thus our results have a direct applicability to the various practical situations wherein the average initial value, rather than merely an arbitrary agreement value, is obtained. In the proofs regarding the network weak-convergence to average-consensus in a two-time-scale Markov system we require a detailed martingale formulation of the problem in conjunction with a well established matrix eigen-decomposition. With regard to average-consensus in a priori unknown communication sequences we rely primarily on extensive algebraic manipulation, the Cauchy-Schwarz inequality, as well as the triangle inequality.

1.1.1 An Illustrative Example

As a simple example, consider a set of 4 sensor nodes dispersed under water with the intent of monitoring the amount of ambient biohazardous material in their vicinity. In this case it is reasonable to assume that at each time instant each sensor node should take a local measurement and average this to obtain the current ambient concentration of biohazards (see Fig.1.1).

Suppose at each time instant $t \in \{0, 1, 2, \ldots\}$ each node $i \in \{1, 2, 3, 4\}$ transmits a local consensus estimate $\hat{s}_i(t)$ to their neighboring node as shown in Fig.1.1. It is easy to show that if
1.1. Introduction

Figure 1.1: Underwater sensor nodes estimating the ambient concentration of biohazardous material by means of a distributed linear averaging protocol.

Each node initializes their consensus estimate by the measurement $s_i(0)$, and each receiving node $j$ updates their local consensus estimate as $\hat{s}_j(t+1) = (1-p)\hat{s}_j(t) + p\hat{s}_i(t)$, then for any $p \in (0,1)$ each node consensus estimate will asymptotically converge to the average $\bar{s}(0) = \frac{1}{4}\sum_{i=1}^{4}s_i(0)$, thereby obtaining at each node the ambient concentration of hazardous material in the water.

Unfortunately however, the communication sequence and weighting scheme depicted in Fig.1.1 requires a number of operational constraints, such as synchronization of the sensor nodes clocks, an a priori agreed upon update weight $p$, as well as an initial node positioning such that (i) each node sends only one other node its consensus estimate and (ii) the union of communicated signals constitute a strongly-connected network. These constraints could be by-passed if a centralized communication network was assumed, as shown in Fig.1.2. However, if the number of sensing nodes was large (say, 100 or 1000), then a centralized node in such circumstances would be clearly infeasible due to physical limitations of the network, such as node transmission power and geological barriers. Furthermore, any fixed scheme to route each node’s data to a fusion center could be compromised if the network communication graph was to change with time. With these problems in mind, we nonetheless note that a centralized node as shown in Fig.1.2 could solve the average-consensus problem trivially by simply aggregating its received data and dividing by the total number of nodes.

In this thesis we will generalize the basic average-consensus problem in two diametrically
1.1. Introduction

In these terms, we thus refer to the average-consensus problem depicted in Fig.1.1 as a “static” consensus problem with \textit{a priori} known communication links. The motivation for analyzing an average-consensus protocol with time-varying (dynamic) inputs is natural in any real-time setting, such as security surveillance and environmental monitoring. The motivation for average-consensus using \textit{a priori} unknown communication links is again natural, since most wireless networks are deployed in dangerous and/or unknown environmental conditions (i.e. the swells in the ocean may disable certain communication links and enable other links at unpredictable time intervals).

Both the static and dynamic average-consensus problem has been analyzed in the past and applied in a variety of practical settings such as formation control (i.e. UAV co-ordination \cite{22,30}); distributed message routing \cite{31,32}; the analysis of swarming behaviors \cite{23}; oscillator synchronization \cite{33}; and network load balancing \cite{16}.

Obtaining the average of all local data is useful for theoretical network operations such as
1.1. Introduction

Figure 1.3: Time-varying Consensus with a priori Unknown Communication Links.

solving a set of local convex optimization problems [34, 35]; distributed Kalman filtering [8];
conducting maximum likelihood estimation [16, 36]; and machine learning [37]. The coordina-
tion of an average-consensus within a multi-agent system has thus been considered under a wide-
range of assumptions regarding the inter-agent communication capabilities [4, 38–48]. The
spatial distribution of the network, energy limitations and geographic barriers may all render
a data fusion center to be infeasible with respect to the averaging of local data. The problem
of distributively computing the network average has received particularly significant attention.
The majority of research in this area propose and study distributed algorithms that allow every
agent in the network to compute the global average via communication with only a subset of
fellow agents. In this thesis we will focus on a specific class of consensus algorithms under
which a network consensus is obtained by the linear averaging of data held at neighboring
nodes. Our contribution to this body of literature includes an extension of the input consensus
dynamics to a two-time-scale Markov system, as well as an extension to arbitrarily time-delayed,
asynchronous, and time-varying network graph topologies.

This thesis will employ a general paradigm within which the same average-consensus prob-
lems as studied in e.g. [17, 26] have been considered. Specifically, we will investigate how each
node can obtain the average of all initial node values by means only of averaging communicated
values under a stochastically switching network communication graph $G(t)$. In addition, we
consider a general communication model that eliminates the possibility of average-consensus
by any linear averaging algorithm with fixed averaging weights. In the former case, to obtain
the desired average-consensus value we propose a stochastic approximation (SA) implemented independently at each node. The SA acts as an on-line “learning” algorithm and can be used to make local weight adjustments that will asymptotically eliminate the distance between each node state-value and the collective average when employing the distributed linear averaging protocol. In the latter case, we are required to employ “normal” consensus estimates that maintain a known linear dependence of each consensus estimate on the initial set of node values. The drawback to using the normal consensus estimates is that they require an $O(n)$ memory storage at each node, regardless of the initial consensus input dimensions. We motivate our contributions by providing numerous practical examples from the past literature that necessitate $O(n)$ consensus estimates, thus nullifying the cost of using the normal consensus estimates when network communication is asynchronous, time-delayed, and a priori unknown at every node.

1.2 Thesis Statement and Objectives

1.2.1 Thesis Statement

This thesis addresses two diametrically opposed aspects of the average-consensus problem. The first aspect concerns average-consensus in a two-time-scale Markov system when inter-node communication is synchronous, instantaneous, and adheres to a fixed network graph. The second aspect concerns average-consensus with node input variables that do not vary over time, yet the inter-node communication may be asynchronous, time-delayed, and adhere to no fixed network graph. Although in the Appendix we attempt to combine the difficulties of both aspects by proposing a collection of data dissemination results that permit average-consensus in a two-time-scale Markov system under general time-delayed inter-node communication patterns, the main efforts in this thesis tackle the two aforementioned contrasting sides of the average-consensus problem.

1.2.2 Thesis Objectives

Based on the description in Sec.1.1, it is evident that average-consensus can in general be a multi-faceted and intricate network optimization problem. Its applications involve significant research abridging stochastic optimization, communication theory, and network science. Moreover, the involved difficulties vary with the underlying network infrastructure that support the data transmitted by any given consensus protocol. Therefore, the goals of this research include the following:

1. efficient data aggregation given a priori knowledge of the underlying communication patterns.
1.3 Thesis Contributions and Organization

2. stochastic and non-stochastic solutions to average-consensus problems in noisy and noiseless settings.

3. distributed algorithms that solve combinations of the above problem features.

To address these objectives we employ techniques in graph theory, stochastic approximation, and convex optimization.

1.3 Thesis Contributions and Organization

1.3.1 Summary of Contributions

The Chapters 3, 4, and 5 detail our contributions relative to the current literature. Here we summarize the main contributions of this thesis at a more abstract level:

- distributed protocols that weakly-converge to average-consensus in a two-time-scale Markov system.
- average-consensus protocols that converge under *a priori* unknown communication patterns.
- data dissemination protocols that can operate under various communication sequence assumptions.

By utilizing the appropriate data dissemination results, we attempt to combine all 3 of the above contributions by suggesting efficient consensus protocols that will weakly-converge to average-consensus under *a priori* unknown communication patterns.

1.3.2 Thesis Organization

In addition to this introductory chapter, the thesis includes two chapters which were originally prepared for journal publication and have been slightly modified in order to offer a logical progression in the thesis. The final chapter discusses the conclusions and directions for future work.

The remaining chapters in this thesis are organized as follows:

Chapter 2 introduces the notation and general distributed algorithm assumptions that will be made throughout the thesis. We also define in general terms the 2 average-consensus problems that will be addressed in the subsequent 2 chapters.

In Chapter 3 we define a two-time-scale Markov system and describe a distributed consensus protocol that weakly-converges to the current average stationary distribution. Subsequently,
1.3. Thesis Contributions and Organization

the network graph weights in this setting are analyzed with respect to the conditions necessary and sufficient for asymptotic convergence to average-consensus.

Chapter 4 considers a static average-consensus with \textit{a priori} unknown communication patterns. Four consensus protocols are defined and the various communication conditions necessary and/or sufficient for each protocols’ convergence are defined. In the Appendix we prove various data dissemination results (yet unpublished) that are required for the latter set of protocols to provide weak-convergence results when the consensus input variables adhere to the two-time-scale Markov system defined in Chapter 3.

Finally, we draw conclusions in Chapter 5 thereby wrapping up the contributions of this thesis and proposing several suggestions for future research on the discussed topics.
Chapter 2

Average-Consensus Problem Descriptions

In this chapter we define two diametrically opposed average-consensus problems that will be addressed in this thesis. The first problem (ACP1) assumes each node is associated with an input parameter observed in a slowly changing Markov noise process; the average-consensus problem then consists of having all nodes weakly-converge to the average of all current input noise distributions. The second problem (ACP2) assumes each node is associated with a fixed initial consensus input; the average-consensus problem then consists of having all nodes converge to the average of all the initial node inputs. Although the second problem is a special case of the first in regard to the node input parameters, we will consider the second consensus problem under far more general inter-node communication assumptions than the first.

To formalize these problems we require some preliminary notation and definitions. These are presented in Sec.2.1. Subsequently in Sec.2.2 we define the two aforementioned consensus problems.

2.1 Preliminary Notations, Assumptions, and Definitions

Let $\mathbb{N}$ denote the non-negative integers and $t \in \mathbb{N}$ denote discrete time. At initial time $t = 0$, consider a finite set of arbitrarily numbered nodes $\mathcal{V} = \{1, \ldots, n\}$ and a set of $d$-dimensional vectors $\{s_i(0) \in \mathbb{R}^{d \times 1} : i \in \mathcal{V}\}$. The set $\{s_i(0) \in \mathbb{R}^{d \times 1} : i \in \mathcal{V}\}$ is referred to as the set of “initial input consensus vectors”. We generally will consider the input consensus vectors $\{s_i(t) \in \mathbb{R}^{d \times 1} : i \in \mathcal{V}, t \in \mathbb{N}\}$ to have the ability to vary with time. Throughout the thesis, we define $\bar{s}(t)$ as the “current average-consensus value”,

$$\bar{s}(t) = \frac{1}{n} \sum_{i=1}^{n} s_i(t). \quad (2.1)$$

Assume each node $i \in \mathcal{V}$ can locally store a “knowledge set” $\mathcal{K}_i(t)$ that consists of an ordered set of elements each with a distinct meaning.
• (A1): (knowledge set assumption) At any time \( t \in \mathbb{N} \), each node \( i \in \mathcal{V} \) is equipped with a device that can update and store a “knowledge set” \( \mathcal{K}_i(t) \). For each \( i \in \mathcal{V} \), the knowledge set \( \mathcal{K}_i(t) \) may have a time-varying cardinality.

The algorithms proposed in this thesis will assume that every knowledge set \( \mathcal{K}_i(t) \) contains a local “consensus estimate” \( \hat{s}_i(t) \in \mathbb{R}^{d \times 1} \) that represents the “belief” of node \( i \) in regard to the current average-consensus vector \( \bar{s}(t) \) defined in (2.1).

• (A2): \( \mathcal{K}_i(t) \supseteq \{ \hat{s}_i(t) \} \), \( \forall \, i \in \mathcal{V} \), \( \forall \, t \in \mathbb{N} \), where \( \hat{s}_i(t) \in \mathbb{R}^{d \times 1} \) represents the local estimate of \( \bar{s}(t) \) at node \( i \) and time \( t \).

Next we assume that an ordered set \( S^{ij}(t_0^{ij}, t_1^{ij}) \) can be transmitted from node \( j \) at a time denoted \( t_0^{ij} \), and received at node \( i \) at a time denoted as \( t_1^{ij} \), where due to causality \( t_1^{ij} \geq t_0^{ij} \). Throughout this work we will for notational convenience denote \( t_0 = t_0^{ij} \) and \( t_1 = t_1^{ij} \), since the superscripts are necessary only when multiple signals are considered in a single equation. We refer to \( S^{ij}(t_0, t_1) \) as a “signal”, or “signal set”.

• (A3): (signal set assumption) At any time \( t_0 \in \mathbb{N} \), each node \( j \in \mathcal{V} \) has the ability to transmit a “signal set” \( S^{ij}(t_0, t_1) \subseteq \mathcal{K}_j(t_0) \) that will be received at some node \( i \in \mathcal{V} \) at time \( t_1 \in \mathbb{N} \), where \( t_1 \geq t_0 \).

The signal \( S^{ij}(t_0, t_1) \) has a time-delay \( (t_1 - t_0) \geq 0 \). As our final condition, we assume that at any time \( t \in \mathbb{N} \), each node \( i \in \mathcal{V} \) will “know” its unique node identifier value \( i \), the network size \( n \), the current input consensus vector \( s_i(t) \), and the local consensus estimate vector \( \hat{s}_i(t) \). This is formalized as,

• (A4): At any time \( t \in \mathbb{N} \), the knowledge set \( \mathcal{K}_i(t) \) of each node \( i \in \mathcal{V} \) satisfies \( \mathcal{K}_i(t) \supseteq \{i, n, \hat{s}_i(t), s_i(t)\} \).

In [27] we explain how the proposed distributed algorithms can be modified if the network size \( n \) is not initially known at any node. We assume \( \mathcal{K}_i(t) \supseteq \{n\} \) for all \( i \in \mathcal{V} \) and \( t \in \mathbb{N} \) only for simplicity of presentation.

### 2.1.1 Consensus Protocols

Given (A1)-(A4), we define a “consensus protocol” (P) in terms of its “knowledge set updating rule” \( f^P_K \{\cdot\} \) together with a “signal specification rule” \( f^P_S \{\cdot\} \). The knowledge set updating rule \( f^P_K \{\cdot\} \) defines the effect that a set of signals has on the knowledge set \( \mathcal{K}_i(t_1) \) at the receiving node \( i \). The signal specification rule \( f^P_S \{\cdot\} \) specifies the elements contained in the signal \( S^{ij}(t_0, t_1) \) given as a function of the knowledge set \( \mathcal{K}_j(t_0) \) at the transmitting node \( j \).
2.1. Preliminary Notations, Assumptions, and Definitions

### The Consensus Protocol (P) under (A1)-(A4):

Knowledge Set Updating Rule: \( f^P_K : K_i(t_1) \bigcup S^{ij}(t_0, t_1) \rightarrow K_i(t_1+1) \) (2.2)

Signal Specification Rule: \( f^P_S : K_j(t_0) \rightarrow S^{ij}(t_0, t_1) \) (2.3)

Any given consensus protocol \( P \) may or may not solve a particular average-consensus problem, as we shall see in each subsequent chapter.

### 2.1.2 Communication Patterns

In addition to (A1)-(A4) and the consensus protocol (2.2) – (2.3), to state the two consensus problems mentioned above we still require four definitions pertaining to the inter-node communication. Based on the definition of a signal as stated in (A3), we define a “communication sequence” as follows.

**Definition 2.1.1 (Communication Sequence)** For any \( 0 \leq t_0 \leq t_1 \), a “communication sequence” \( C_{t_0,t_1} \) is the set of all signals transmitted no earlier than time \( t_0 \) and received no later than time \( t_1 \), that is,

\[
C_{t_0,t_1} = \{S^{ij_1}, S^{ij_2}, \ldots \}
\]

where we have omitted the time indices but it is understood that the transmission time \( t_0 \) and reception time \( t_1 \) of each signal \( S^{ij} \) belong to the interval \([t_0, t_1]\).

Next we define the notion of a “non-time-respecting communication path”, which will be abbreviated as “ntrcp”.

**Definition 2.1.2** A communication sequence \( C_{t_0,t_1} \) contains a “non-time-respecting communication path” (ntrcp) from node \( j \) to node \( i \) if \( C_{t_0,t_1} \) contains a sub-sequence \( C^{ij}_{t_0(t_{ij}),t_1(t_{ij})} \) with the following connectivity property,

\[
C^{ij}_{t_0(t_{ij}),t_1(t_{ij})} = \{S^{\ell_{ij}, S^{\ell_2}, \ldots, S^{\ell_{k(ij)}}, S^{\ell_{k(ij)}+1}} \}
\]

where we have omitted the time indices of each signal but it is understood that the transmission time \( t_0 \) and reception time \( t_1 \) of each signal \( S^{\ell} \) belong to the interval \([t_0(t_{ij}), t_1(t_{ij})]\).

Note that the sub-sequence \( C^{ij}_{t_0(t_{ij}),t_1(t_{ij})} \) in Def.2.1.2 has a finite cardinality \( |C^{ij}_{t_0(t_{ij}),t_1(t_{ij})}| = k(ij) + 1 \geq 1 \).

For the next definition we let \( V_i = V \setminus i \) for each node \( i \in V \). The notion of a “non-time-respecting” strongly-connected communication sequence (ntrcs) is as follows.
2.2. The Average-Consensus Problems

Definition 2.1.3 A communication sequence $C_{t^0, t^1}$ is “non-time-respecting strongly-connected” (ntrcs) if $C_{t^0, t^1}$ contains a ntrcp from each node $i \in V$ to every node $j \in V - i$. □

We let $C_{t^0, t^1} \in \text{ntrcs}$ denote that $C_{t^0, t^1}$ is a ntrcs.

Lastly, we define a “regime-modulated” communication sequence (rmcs).

Definition 2.1.4 Let $\{r_k(t^0(k), t^1(k)) : k \in \mathbb{N}\}$ denote a sequence of regimes over the integers $t \in \mathbb{N}$, with $t^0(k) \leq t^1(k) < t^0(k + 1)$ for all $k \in \mathbb{N}$. A communication sequence $C_{t^0, t^1}$ is “regime-modulated” (rmcs) if $C_{t^0, t^1} = \bigcup_{k \in \mathbb{N}} C_{t^0(k), t^1(k)}$, $S_{ij}^{(r)} \subseteq C_{t^0(k), t^1(k)}$ if and only if (iff) $S_{ij}^{(r)} \in C_{t^0(k), t^1(k)}$ for all $t(k) \in \{t^0(k), t^1(k) + 1, \ldots, t^1(k)\}$, and $C_{t^0(k), t^1(k)} \in \text{ntrcs}$ for each $k \in \mathbb{N}$. □

We let $C_{t^0, t^1} \in \text{rmcs}$ denote that $C_{t^0, t^1}$ is a rmcs. Note that every signal in a regime-modulated communication sequence has a zero time-delay, and that for each particular regime a node $j$ sends a signal to node $i$ at each time instant iff node $j$ sends a signal to node $i$ at any time instant in that regime. In this sense, each communication sequence $C_{t^0(k), t^1(k)}$ is “fixed” over the time span of each regime.

2.2 The Average-Consensus Problems

Given the above definitions, we may now state the two consensus problems that will be considered in this thesis.

Definition 2.2.1 (ACP1) Under (A1)-(A4), assume the communication sequence $C_{0, \infty}$ is regime-modulated and that within each regime $r(k)$ the sequence of each input consensus parameter $\{s_i(t) : t \in [t^0(k), t^1(k)]\}$ is an i.i.d. discrete random variable with finite state-space $S = \{s_1, s_2, \ldots, s_d\}$ and probability distribution $\pi_i(r(k)) \in \mathbb{R}^{d \times 1}$. A consensus protocol (P) (cf. (2.2) – (2.3)) then solves the average-consensus problem (ACP1) if the consensus estimate $\hat{s}_i(t)$ of each node weakly-converges to $\bar{\pi}(r(k)) = \frac{1}{n} \sum_{i \in V} \pi_i(r(k))$ for all $t \in [t^0(k), t^1(k)]$ and each $k \in \mathbb{N}$. □

The above consensus problem requires the consensus estimate $\hat{s}_i(t)$ of each node to weakly-converge to the average of all input variables’ regime-modulated probability distribution. We thus have a consensus in distribution rather than a consensus in regard to a fixed value. Furthermore, the probability distribution of each input variable is regime-modulated and thus can
vary with time. The latter differences are two of the primary distinctions between the consensus problems (ACP1) and (ACP2) defined next.

**Definition 2.2.2 (ACP2)** Under (A1)-(A4), assume the communication sequence $C_{0,\infty}$ is a priori unknown and that the input consensus parameters $\{s_i(t) : i \in \mathcal{V}\}$ remain fixed at their initial values $\{s_i(0) : i \in \mathcal{V}\}$ for all $t \in \mathbb{N}$. A consensus protocol $(P)$ (cf. (2.2) – (2.3)) then solves the average-consensus problem (ACP2) if the consensus estimate $\hat{s}_i(t)$ of each node converges (in the Euclidean norm) to the average initial input parameter $\bar{s}(0) = \frac{1}{n} \sum_{i \in \mathcal{V}} s_i(0)$.

Note that in both (ACP1) and (ACP2) the average-consensus problem is solved when each node consensus estimate converges to the “current average-consensus value” $\bar{s}(t)$ defined in (2.1). The differences between the two problems is that (i) in (ACP1) $\bar{s}(t)$ is the average of a set of time-varying probability distributions whereas in (ACP2) it remains fixed at the initial input value, and (ii) in (ACP1) the communication sequence is regime-modulated (synchronous, without time-delay, and fixed) whereas in (ACP2) it is completely a priori unknown.

In the subsequent chapters we shall present consensus protocols that solve each of these ACPs. In each chapter we first define the local consensus input dynamics as well as the assumed inter-node communication patterns. We then review the literature with respect to the algorithms that researchers in this area have presented. Our own solutions are then described in detail (the proofs of each result can be found in the Appendix). Numerical examples of our algorithms are presented and their performance compared where possible with those of previously proposed algorithms in the literature. Lastly, some concluding comments and potential directions for future work are discussed.
Chapter 3

ACP1: Consensus in a Two-Time-Scale Markov System

Wireless sensor networks are often a preferential candidate to monitor environmental parameters such as moisture degree, presence of contaminants, pressure, and temperature. In many of such applications, the sensors will first sense the environment, and then average their measurements to compute the final global estimator in a completely distributed fashion. However, there is usually no meaningful way to decide when the sensing stage should be terminated and the averaging procedure initiated. In this chapter we present a consensus protocol that consists of both a local stochastic approximation (SA) algorithm that operates in parallel with the distributed linear averaging (DLA) protocol. By utilizing both local SA and strongly-connected DLA we show that an average-consensus can be maintained while simultaneously procuring new observations via the local sensing procedure. We note that if the initial consensus inputs remain fixed (or are drawn from a deterministic source) then there is no need to obtain new observations while simultaneously computing the current average (this particular issue is explored in more detail in Chapter 4).

Specifically, this chapter will propose a consensus protocol that solves a generalized version of ACP1 (cf. Def.2.2.1). We will assume the input consensus variable $X_i$ at each node $i \in V$ obeys a two-time-scale Markov system wherein the stationary distribution $\pi_i(t)$ switches according to a slow Markov chain denoted by the hyper-parameter $\theta(t)$. We propose a scheme where the sensing and the averaging stages are simultaneous: the network continues collecting data while computing on-line the distributed estimator of the current average input value. The asymptotic behavior of the consensus protocol is investigated and compared with that of classical consensus protocols; the impact of the network topology is discussed; and numerical examples are presented.

Motivation:

In many applications of sensor networks, the system is deployed in dangerous areas, subject to some kind of risk such as geomorphological, chemical, radioactive, and so forth. In these cases the wireless architecture is usually preferred to the wired one for the absence of hardware
infrastructures connecting the sensors. The different nodes of the network independently measure various input parameters and the final estimate of the desired quantity is usually obtained in a fully decentralized fashion. Instead of delivering the locally measured data to a common fusion center, the nodes exchange their data by performing decoupled averaging and thereby converging, after a sufficient amount of time, to a common value for all the nodes. The final global estimate is recorded at each node of the network and thus can be recovered from any of the existing nodes.

However, in the above analysis an important scheduling aspect is ignored. Usually the instant of time when a sufficient amount of sensing data has been locally procured is unavailable to any given node, implying that the system designer is not able to decide when the sensing stage should be terminated and the averaging protocol that leads to the final consensus should begin. If the sensing stage is prematurely terminated, many potentially useful observations are lost. On the other hand, prolonging too much the sensing stage in the attempt of collecting more samples entails the risk of duplicate data processing and the collection of uninformative measurements. In this chapter we propose a consensus protocol where the stages of sensing and of averaging are not separated but they evolve simultaneously. We make explicit reference to the gossip algorithms [17], where the final aim is that of making available, to any node, the arithmetic mean of all the observations globally collected by the entire network. In addition to convergence results, our contribution is investigate how the new measurements incorporated in the consensus procedure trade off with the convergence conditions: averaging the observations cuts down the differences between the sensors (consensus), but the occurrences of new observations represent a form of diversity among the nodes.

We generalize the observation input model in the existing literature by considering a two-time-scale Markov system. In such a system it is shown in [49] that the continuous-time interpolation of the iteration

$$\hat{s}_i(k + 1) = (1 - \mu)\hat{s}_i(k) + \mu X_i(\theta(k)) , \quad k \in \mathbb{N}$$

will, under appropriate scaling conditions, weakly-converge to the slowly-switching ODE

$$\frac{ds_i(t)}{dt} = -s_i(t) + \pi_i(\theta(t)).$$

In this chapter we generalize (3.1) from a single node to a multi-agent system of nodes that communicate via an instantaneous, fixed, and synchronous network communication graph. In addition to this generalization of (3.2), we also consider the estimation of the average cumulative distribution function \(\bar{\Pi}(\theta(t))\), the scaled tracking error \(v_i(t)\) at each node \(i \in \mathcal{V}\), as well as the set-valued generalization of (3.2).
The outline of this chapter is as follows. In Sec.3.1 we define the local consensus input dynamics, and in Sec.3.2 we define the inter-node communication patterns that are assumed under ACP1. Also in Sec.3.2 the definition of a consensus protocol as well as the generalized version of the consensus problem ACP1 are re-stated for the readers’ convenience. A literature review with respect to consensus protocols that past research in this area have analyzed is presented in Sec.3.3. The implementation of our proposed protocol is then described in Sec.3.4. Numerical examples of the protocol are subsequently presented (Sec.3.5), and lastly, we provide some concluding comments and potential directions for future work (Sec.3.6).

3.1 ACP1: Local Consensus Input Dynamics

In this section we define a two-time-scale Markov system that independent sensors positioned at each node will observe. This system entails a set of randomly switching input consensus variables observed in Markov noise. By reaching average-consensus each node essentially tracks the average value of the current set of slowly-switching input consensus variables.

At each discrete-time index \( k \in \mathbb{N} \) suppose each node \( i \in \mathcal{V} \) observes the state of an \( S \)-state Markov chain \( X^i(k) \) with state-space \( \{e_1, \ldots, e_S\} \), where each \( e_\ell \) is the \( \ell \)th \( S \times 1 \) standard unit vector. Next consider a continuous time-scale \( t \geq 0 \) such that \( t_k = k\mu \) for a small scale parameter \( \mu \ll 1 \). Assume there exists a hyper-parameter \( \theta \) for which each \( X^i(k) \) is \( \theta \)-dependent for \( \theta \in \mathcal{M} = \{\theta^1, \ldots, \theta^m\} \) such that the transition matrix of \( X^i(k) \) conditioned on \( \theta \) is given by

\[
A^i(\theta) = [a^i_{\ell j}(\theta)] ,
\]

where

\[
a^i_{\ell j}(\theta) = P(X^i(k+1) = e_j|X^i(k) = e_\ell, \theta(k) = \theta).
\]

To proceed, we pose the following three conditions.

(B) 1. For each \( \theta \in \mathcal{M} \), the transition matrix \( A^i(\theta) \) is irreducible and aperiodic for each \( i \in \mathcal{V} \).

2. Parameterize the transition probability matrix of \( \theta \) as

\[
P^\varepsilon = I + \varepsilon Q ,
\]

where \( \varepsilon \) is a small parameter satisfying \( 0 < \varepsilon \ll 1 \) and \( Q \) is the generator of a continuous-time finite-state Markov chain.

3. The process \( \theta(k) \) is slow in the sense \( \varepsilon = O(\mu) \). For simplicity, we take \( \varepsilon = \mu \) henceforth.
3.2. ACP1: Inter-node Communication Pattern

Condition (B) specifies our observation model as a two-time-scale Markov system. As stated in (B), each transition matrix $A^i(\theta)$ is irreducible and aperiodic. Accordingly we will let $\pi^i(\theta(k))$ denote the stationary distribution of $X^i(k)$. Note that since $\varepsilon = O(\mu)$, the hyper-parameter $\theta$ will switch states on a slow time-scale as compared to the state of the input parameter $X^i(k)$, which will switch states on a relatively fast time-scale.

3.2 ACP1: Inter-node Communication Pattern

In this section we define the inter-node communication pattern that will be assumed throughout this chapter. To do so we require the definition of a “non-time-respecting communication path”, which will be abbreviated as “ntrcp”.

**Definition 3.2.1** A communication sequence $C_{t^0,t^1}$ contains a “non-time-respecting communication path” (ntrcp) from node $j$ to node $i$ if $C_{t^0,t^1}$ contains a sub-sequence $C_{t^0,ij,t^1,ij}$ with the following connectivity property,

$$C_{t^0,ij,t^1,ij} = \{S^{f_{ll^1}ij}, S^{f_{ll^2}t_1}, S^{f_{ll^3}t_2}, \ldots, S^{f_{ll^k}t_{k-1}}, S^{f_{ll^k}ij}\}$$

where we have omitted the time indices of each signal. □

Note that the sub-sequence $C_{t^0,ij,t^1,ij}$ in Def.3.2.1 has a finite cardinality $|C_{t^0,ij,t^1,ij}| = k(ij) + 1 \geq 1$. We next define the notion of a “non-time-respecting” strongly-connected communication sequence (ntrcs).

**Definition 3.2.2** A communication sequence $C_{t^0,t^1}$ is “non-time-respecting strongly-connected” (ntrcs) if $C_{t^0,t^1}$ contains a ntrcp from each node $i \in \mathcal{V}$ to every node $j \in \mathcal{V}_{-i}$. □

We let $C_{t^0,t^1} \in \text{ntrcs}$ denote that $C_{t^0,t^1}$ is a ntrcs.

The network communication in this chapter will be assumed to be synchronous, directed, instantaneous, and regime-modulated. Specifically we have the following three conditions.

- (A5) At each time instant $\{t_k = k\mu : k \in \mathbb{N}\}$ every member $i$ in a set of nodes $\mathcal{V}^{out}(t_k) \subseteq \mathcal{V}$ sends an instantaneous signal to at least one member $j$ in a set of nodes $\mathcal{V}^{in}(t_k) \subseteq \mathcal{V}$. We define the $n \times n$ “communication adjacency matrix” $C(t_k)$ to have its $(ij)^{th}$ element equal to 1 if node $i$ sends a signal to node $j$ at time $t_k$ and zero otherwise.

- (A6) At each time instant $\{t_k = k\mu : k \in \mathbb{N}\}$ the matrix $C(t_k)$ determined by the state of the hyper-parameter $\theta(t_k)$.

- (A7) the communication sequence $C_{t_k,t_k}$ is ntrcs for each $t_k \geq 0$. 

17
3.2. ACP1: Inter-node Communication Pattern

Since the inter-node communication is fully defined at time $t_k$ by the communication adjacency matrix $C(t_k)$, conditions (A5) and (A6) together specify a “regime-modulated communication sequence”, with regime dictated by the state of the hyper-parameter $\theta(t_k)$. Under (A5)-(A6), the condition (A7) is necessary for consensus under any consensus protocol, as is shown in Chapter 4 (see Thm.4.4.7).

We now re-state assumptions (A1)-(A4) as well as the definition of a consensus protocol for the reader’s convenience. We will assume,

- (A1): (knowledge set assumption) At any time $t \in \mathbb{N}$, each node $i \in \mathcal{V}$ is equipped with a device that can update and store a “knowledge set” $\mathcal{K}_i(t)$. For each $i \in \mathcal{V}$, the knowledge set $\mathcal{K}_i(t)$ may have a time-varying cardinality.

- (A2): $\mathcal{K}_i(t) \supseteq \{\hat{s}_i(t)\}$, $\forall i \in \mathcal{V}$, $\forall t \in \mathbb{N}$, where $\hat{s}_i(t) \in \mathbb{R}^{d \times 1}$ represents the local estimate of $\bar{s}(t)$ at node $i$ and time $t$.

- (A3): (signal set assumption) At any time $t_k \in \mathbb{N}$, each node $j \in \mathcal{V}$ has the ability to transmit a “signal set” $S^{ij}(t_k, t_k) \subseteq \mathcal{K}_j(t_k)$ that will be received at some node $i \in \mathcal{V}$ at time $t_k \in \mathbb{N}$.

- (A4): At any time $t_k > 0$, the knowledge set $\mathcal{K}_i(t_k)$ of each node $i \in \mathcal{V}$ satisfies $\mathcal{K}_i(t_k) \supseteq \{i, n, \hat{s}_i(t_k), X^i(t_k)\}$.

Given (A1)-(A4), we define a “consensus protocol” (P) in terms of its “knowledge set updating rule” $f^K_P\{\cdot\}$ together with a “signal specification rule” $f^S_P\{\cdot\}$. The knowledge set updating rule $f^K_P\{\cdot\}$ defines the effect that a set of signals has on the knowledge set $\mathcal{K}_i(t_k)$ at the receiving node $i$. The signal specification rule $f^S_P\{\cdot\}$ specifies the elements contained in the signal $S^{ij}(t_k, t_k)$ given as a function of the knowledge set $\mathcal{K}_j(t_k)$ at the transmitting node $j$.

<table>
<thead>
<tr>
<th>The Consensus Protocol (P) under (A1)-(A4):</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge Set Updating Rule: $f^K_P: \mathcal{K}_i(t_k) \bigcup S^{ij}(t_k, t_k) \mapsto \mathcal{K}<em>i(t</em>{k+1})$ (3.3)</td>
</tr>
<tr>
<td>Signal Specification Rule: $f^S_P: \mathcal{K}_j(t_k) \mapsto S^{ij}(t_k, t_k)$ (3.4)</td>
</tr>
</tbody>
</table>

The consensus problem ACP1 is now re-stated in terms (B), (A1)-(A7) and (3.3) - (3.4).
Definition 3.2.3 (ACP1) Under (B) and (A1)-(A7), a consensus protocol (P) (cf. (3.3)–(3.4)) solves the average-consensus problem (ACP1) if the consensus estimate \( \hat{s}_i(t_k) \) of each node weakly-converges to \( \bar{\pi}(\theta(t_k)) = \frac{1}{n} \pi_i(\theta(t_k)) \) for all \( t_k > 0 \) as \( \mu \to 0 \).

To solve ACP1 (cf. Def.3.2.3) we will propose a consensus protocol that satisfies (3.3)–(3.4) under the communication assumptions (A1)-(A7). We define this protocol in Sec.3.4.1, and present the formal convergence results in Sec.3.4.2. Before doing so, we review the past literature related to the results presented in this chapter.

3.3 ACP1: Algorithm Description and Relation to Past Literature

The coordination of a consensus within a multi-agent system is a problem that has been considered in a variety of general settings and solved under a wide-range of assumptions regarding the inter-agent communication capabilities. Algorithms that result in “consensus” have thus in general taken on a variety of forms, for some examples see [4, 22, 38–48, 50]. Here we will focus on a specific consensus algorithm under which a network consensus is obtained by linear averaging the local data held at neighboring sensor nodes.

Linear averaging has been a particular research focal point regarding consensus formation [19, 24, 51, 52]. In these and many other works on consensus, the multi-agent system is defined as a collection of sensors with no extensive memory base, hence the linear averaging scheme is analyzed with respect to a group of distributed agents. This particular network paradigm and consensus mechanism will be together referred to here as distributed linear averaging (DLA). Under a DLA algorithm, every sensor has a direct linear effect on the estimates of each of its neighbors. Thus the DLA algorithm is scalable and can be robust under a variety of network communication conditions [16,18,25,53]. This is in contrast to consensus protocols for specialized data fusion problems that have complex algorithmic forms that cannot be simplified to mere averaging [19,24,39,40,48].

In precise terms, the DLA consensus protocol is an iterative procedure wherein each sensor node linearly averages the information it holds with the information it receives from other sensors whom with it directly communicates. This computation is performed unanimously by all sensors, thus DLA requires minimum data storage, labeling of data, or specialization of individual sensors, i.e. use of base nodes. For a network of \( n \) sensors, the DLA protocol can be parameterized by a weighted digraph \( G = \{V, E, W\} \) where \( V = \{1, \ldots, n\} \) denotes the set
of sensor nodes, \( E \subseteq V \times V \) is a set of directed edges specifying which sensors are connected (i.e. can transmit or receive information), and the weight matrix \( W \in \mathbb{R}^{n \times n} \) that denotes the weights with which neighboring sensors average their held information. We will by definition let the \((i, j)^{th}\) element of \( W \) equal zero if \((i, j) \notin E\).

**Distributed Consensus-Tracking Algorithm.** In this chapter, a discrete-time DLA algorithm is assumed to operate together with a linear stochastic approximation (SA) tracking algorithm based locally at each sensor \( i \in V \). Each SA tracking algorithm aims to estimate the stationary distribution \( \pi^i \in \mathbb{R}^S \) of a fast \( S \)-state ergodic \( X^i \in \mathbb{R}^S \) that is observed uniquely by sensor node \( i \). Motivated by applications in adaptive tracking in sensor networks, we assume the transition matrix of each Markov chain \( \{X^i : i \in V\} \) jump changes slowly with time. More specifically, we assume that the transition matrix of each Markov chain \( \{X^i : i \in V\} \) is conditioned on a hyper-parameter \( \theta \) taking values in a finite set \( M \). The dynamics of \( \theta \) are determined by a slowly changing \( m \)-state Markov chain with state-space \( M = \{\theta_1, \ldots, \theta_m\} \) and transition matrix \( P^\epsilon = I + \epsilon Q \), where \( \epsilon > 0 \) is a small parameter and \( Q = (q_{ij}) \in \mathbb{R}^{m \times m} \) is a generator of a continuous-time Markov chain. Since \( \epsilon \) is small, \( \theta \) is a slowly varying process and will jump infrequently among different states relative to the state of each Markov chain \( X^i(k), i \in V \).

For each \( \theta \in M \) and \( i \in V \), \( X^i_k \) is an \( S \)-state Markov chain with a transition matrix modulated by the slow Markov chain \( \theta_k \), thus we describe the sequence \( \{\theta_k, X^i_k : i \in V, k \in \mathbb{N}\} \) as a switched Markov system. As a result, each stationary distribution \( \pi^i(\theta) \) is piece-wise fixed but with a jump change at every time switch in \( \theta_k \). When \( \theta_k \) switches its value from one state to another within \( M \), the stationary distribution \( \pi^i(\theta_k) \) switches accordingly, thus it follows that the sensor network must track these time-varying distributions.

We next give an intuitive explanation of consensus formation. Given the averaging weights \( W \) and graph edge set \( E \) of the sensor network described above, each node \( i \in V \) computes via the distributed consensus tracking algorithm (the iteration (3.16) below) a state-value, denoted \( \hat{s}^i(t_k) \in \mathbb{R}^{S \times 1} \).

We will prove that for a suitably scaled continuous-time, the state-value \( \hat{s}^i(t) \) converges weakly to a linear combination \( \hat{\pi}^i(\theta(t)) = \sum_{l=1}^n \psi^i_l \pi^l(\theta(t)) \in \mathbb{R}^S \) of the stationary distributions \( \{\pi^l(\theta(t)) : i \in V\} \). Here the coefficients \( \psi^i_l \) are determined by the choice of \( W \) and \( E \). Consensus formation deals with the assumptions on the information exchange between sensors required to obtain a specific structure of \( \hat{\pi}^i(\theta(t)) \) in the above expression. The sensor network achieves a consensus if for any given \( l \in V \), \( \psi^i_l = \psi^j_l \) for all \( i, j \in V \). The average-consensus occurs if \( \psi^i_l = \frac{1}{n} \) for all \( i, l \). Due to the dependence of \( \pi^l \) on \( \theta \), the values of \( \hat{\pi}^i(\theta) \) stochastically switch among a finite set of values, each corresponding to a particular \( \theta \in M \). The average-consensus in this setting is thus a random process. To supplement the weak convergence results, we
derive conditions on the averaging weights required for each sensor to obtain the stochastic average-consensus $\bar{\pi}(\theta(t))$.

Assuming both the DLA and local tracking algorithms use a constant step-size $\mu$ at each discrete-time iteration $k \in \mathbb{N}$, it will be seen that in order for all sensors to track the average distribution $\bar{\pi}(\theta)$ each sensor should distributively average both its observed signal, denoted $X^i_k$, and its state-value $\hat{s}^i_k$. Define the observation exchange graph $\mathcal{G}^o = \{\mathcal{V}, \mathcal{E}^o, \mathcal{W}^o\}$ and state-value exchange graph $\mathcal{G}^v = \{\mathcal{V}, \mathcal{E}^v, \mathcal{W}^v\}$ that respectively determine how each type of data, $X^i$ (observation) and $\hat{s}^i$ (state), is distributively averaged. The distributed consensus-tracking algorithm we then consider is as follows:

$$\hat{s}_{k+1} = (I - \mu D^v + \mu W^o) \hat{s}_k + \mu W^o X_k = (I - \mu H) \hat{s}_k + \mu W^o X_k$$

$$\hat{s}(0) = X(0), \quad D^p = \text{diag}(W^p)1 \quad p \in \{o,v\}, \quad H = D^o + D^v - W^v, \quad (3.5)$$

where $\hat{s}_k = [\hat{s}^1_k, \ldots, \hat{s}^n_k]' \in \mathbb{R}^{Sn \times 1}$, $X_k = [X^1_k, \ldots, X^n_k]' \in \mathbb{R}^{Sn \times 1}$, the term $1$ denotes a column vector of ones with appropriate dimension, and we let $\text{diag}(r)$ be the $d \times d$ diagonal matrix with elements corresponding to those of the vector $r \in \mathbb{R}^d$. We will henceforth let each element of the weight matrices $\{W^o, W^v\}$ represent an $S \times S$ identity matrix scaled by the respective element, that is we let $W = W \otimes I_{S \times S}$, where $\otimes$ denotes the Kronecker product [54]. As a final note, throughout this chapter we let subscript $k$ indicate dependence on a discrete-time in $\mathbb{N}$ (e.g., $\theta_k$), whereas dependence on continuous-time $t \geq 0$ will be denoted $(t)$ (e.g. $\theta(t)$). The single exception to this rule is the initial data, which is always denoted $(0)$ for both discrete-time and continuous-time systems.

The above algorithm can be viewed as comprising two DLA protocols together with a family of local stochastic approximations; each DLA protocol averages with the local state-value either with the state value at each neighboring node (as determined by $\mathcal{G}^v$), or the observations at each neighboring node (as determined by $\mathcal{G}^o$). For a sufficiently small step-size $\mu$, the iteration (3.16) implies that at each discrete time $k \in \mathbb{N}$, the state-value $\hat{s}^i_k$ computed at any sensor $i \in \mathcal{V}$ is updated linearly in three ways as follows (the superscripts * and ** below are intermediate variables for illustration only and will not be used subsequently):

1. by its own local observation $X^i_k$,

$$\hat{s}^*_k = \hat{s}^i_k + \mu W^o_{ii} (X^i_k - \hat{s}^i_k), \quad (3.6)$$

2. by the local observation $X^j_k$ of each neighboring sensor $j \in \{j : (i,j) \in \mathcal{E}^o\}$,

$$\hat{s}^{**}_k = \hat{s}^*_k + \mu \sum_{j \neq i} W^o_{ij} (X^j_k - \hat{s}^*_k),$$
3.3. ACP1: Algorithm Description and Relation to Past Literature

3. by the state-value $s_k^j$ of each neighboring sensor $j \in \{j : (i,j) \in E\}$,

$$
\hat{s}_{k+1}^i = (\hat{s}_k^i)^{**} + \mu \sum_{j \neq i} W_{ij}^v (\hat{s}_j^i - (\hat{s}_k^i)^{**}).
$$

As $\mu$ vanishes these three nested steps may be written as the single iteration (3.16) because in this limit all second and third order terms in $\mu$ become negligible compared to the first order terms that are retained in (3.16). We also note that due to the similarity between steps 1 and 2, each row $i \in V$ of the exchange graph $G^o$ may either specify which sensors communicate their observed information $X_j^i$ to sensor $i$, or equivalently which Markov chains $X_j^i$ sensor $i$ can actually observe, assuming that the same SA algorithm (3.6) with update weights $W_{ij}^o$ are applied to each Markov chain $X_j^i$.

**Context.** The tracking algorithm (3.16) was first introduced in [8] as the distributed consensus filter. However, in [8] it is assumed (i) $W^v = W^v + I$, (ii) the diagonal of $W^v$ is zero, and (iii) each non-zero averaging weight is unity. Furthermore, the observation variables $\{X_1^i, \ldots, X_n^i\}$ are reduced to a single time-varying scalar observed in i.i.d. Gaussian noise by all sensors. In contrast to [8], recall that we assume a Markov-modulated dynamic model where each sensor $i \in V$ observes the Markov chain $X_i^j$. Our observation model involves a two-time-scale formulation with parameters $\mu$ and $\varepsilon$. The step-size $\mu$ is used in the recursive tracking and averaging algorithms, whereas the step-size $\varepsilon$ represents the transition rate of the Markov chain $\theta$.

To analyze the continuous-time limit of (3.16), it is assumed in [8] that the observed scalar has a uniformly bounded derivative as the step-size $\mu$ approaches zero. Here, we assume $\varepsilon = O(\mu)$, indicating that (3.16) has a tracking rate on the same time-scale as the Markov modulating process. Our main result shows that under suitable conditions on the network exchange graphs $\{G^v, G^o\}$, the assumption $\varepsilon = O(\mu)$ implies the limit of a piece-wise constant continuous-time interpolation of the sequence of each sensors estimates $\hat{s}^i(t_k)$ converges weakly to the solution of a Markov modulated ODE as $\mu \to 0$. Considering also the sequence of sensor tracking errors, we demonstrate under similar conditions that an interpolation of the normalized errors converges weakly to a switching diffusion. Under additional weight constraints it then follows as a special case of the above results, that the estimates $\hat{s}^i(t_k)$ converge weakly to the average-consensus $\bar{\pi}(\theta)$.

Within this general framework we also address two modifications of the consensus-tracking algorithm (3.16). One modification assumes that rather than a single chain $X_i^j$ each sensor $i$ observes a family of Markov chains $X_{ik}^j \in X_k^i$, thus implying the sensor state-values converge weakly to solutions of a switched differential inclusion. The second modification considers sensor estimation of the cumulative distribution function (CDF) $\Pi^i(\theta(t))$, $i \in V$, $\theta \in M =$
A. Related Work

We briefly review here some of the related literature on consensus formation. The distributed consensus-tracking algorithm (3.16) can be viewed as the combination and extension of two sub-algorithms: a local sensor adaptive SA tracking algorithm,

\[ s_{i,k+1}^i = s_{i,k}^i + \mu(X_{i,k}^i - s_{i,k}^i), \quad s_{i,0}^i = X_{i,0}^i, \quad 0 < \mu < 1 \ll 1 \]

operating in parallel with the (DLA) or so-called “Laplacian” consensus algorithm,

\[ s_{k+1} = (I - \mu L)s_k, \quad L = D - W. \tag{3.7} \]

The consensus algorithm (3.7) has been explored in works such as [16, 18] regarding its ability to achieve an average-consensus on node initial state-values \( s(0) = [s^1(0), \ldots, s^n(0)] \), that is \( \bar{s}(0) = \frac{1}{n} \sum_{i=1}^{n} s^i(0) \). It is clear the average-consensus \( \bar{s}(0) \) is constant at all times, thus the algorithm (3.7) by itself can result in a static consensus that is not driven by some external parameters, such as the observed Markov chains \( X_{i,k}^i \) considered here (see also [51] and references therein).

As proposed in [8], we extend the Laplacian consensus algorithm to include not only averaging of the sensor state-values \( s^i(0) \), but also of the sensor observations \( X_{i,k}^i \). This extension can be very useful for DLA to ensure average-consensus when considered within the generality of the current setting, specifically when each sensor observes and estimates a unique parameter. Unlike certain works on distributed averaging such as [8, 24], we allow \( \{G^v, G^o\} \) to be both directed and weighted, such as those considered for instance in [18, 55, 56]. By directed it is meant \( (i,j) \in E \implies (j,i) \notin E \), and by weighted it is meant \( W \) need not have each element belong to the set \( \{0, w\} \) for some fixed \( w \in (0,1) \). We also note that \( W_{ij} \neq 0 \) implies \( (j,i) \in E \), and thus knowledge of \( \{W^v, W^o\} \) is sufficient to completely define \( \{G^v, G^o\} \).

The Laplacian consensus algorithm (3.7) derives its name due to its asymptotic equivalence with the gradient system \( \dot{s}(t) = -\nabla \Phi_G(s(0)) \) associated with the Laplacian potential

\[ \Phi_G(s(0)) = \frac{1}{2} \sum_{i,j=1}^{n} W_{ij}^v (s^i(0) - s^j(0))^2 = \frac{1}{2} s(0)'Ls(0), \]

where \( L = \text{diag}(W^v1) - W^v \) is defined as the Laplacian matrix of an undirected graph \( G = \{V, E^v, W^v\} \) and each node \( i \in V \) has an initial state-value \( s^i(0) \in \mathbb{R} \) [18]. Many works [19, 24, 25, 57, 58] consider the Laplacian consensus dynamics (3.7) in isolation from any tracking model,
such works are thus concerned with static, rather than time-varying, consensus formation. The general theme addressed in these works regard the network graph conditions under which (3.7) solves the static average-consensus problem, that is, having all sensors reach \( \bar{s}(0) \) given that \( \mathcal{E}^v \) has time-varying or stochastic properties. These concerns are closely related to the issues of consensus under time-delayed or asynchronous communication within sensor networks and other multi-agent systems, we refer the reader to works such as [18, 26, 43, 56, 57, 59–62] for detailed consideration.

In contrast, [17] assumes the communication edge set \( \mathcal{E}^v \) is fixed, undirected, and strongly-connected (that is, each node can be reached from all other nodes by traversing the edges in \( \mathcal{E}^v \)). Under these conditions it is shown in [17] that the matrix \( W = I - \mu \mathcal{L} \) satisfying,

\[
\begin{align*}
\text{minimize} \quad & \| W - \frac{1}{n} 11' \|_2 \\
\text{subject to} \quad & W \in \mathcal{E}^v, \quad W 1 = 1, \quad 1' W = 1',
\end{align*}
\]

will imply the maximum asymptotic per-step convergence factor \( r_{\text{step}}(W) \) of the sensor state-value estimates \( s_k \) to the average-consensus value \( \bar{s}(0) \), where

\[
\begin{align*}
r_{\text{step}}(W) = \sup_{s_k \neq \bar{s}(0)} \frac{\| s_{k+1} - \bar{s}(0) \|_2}{\| s_k - \bar{s}(0) \|_2}.
\end{align*}
\]

In (3.8), the operation \( \| \cdot \|_2 \) indicates spectral norm, thus (3.8) is a convex optimization problem and in [17] cast as a semi-definite program. The relation of [17] to the present work is emphasized since the same constraints of (3.8) are sufficient to ensure sensor estimation of the average-consensus \( \bar{\pi}(\theta) \), and thus the weights \( W \) that solve (3.8) may also be argued as the optimal weights for fast convergence to \( \bar{\pi}(\theta) \).

The consensus-tracking framework of our paper is also similar to recent works of [7,8,61,63]. These papers deploy (3.7) as part of a distributed Kalman filter or in conjunction with local sensor estimation of a time-varying parameter, thus these works also deal with a time-varying input consensus value. For example, assuming that each sensor node \( i \) observes in continuous-time the \( m \)-dimensional signal data pair \( \{z_i(t), \dot{z}_i(t)\} \), the following result is proven in [7]: for any sequence of weighted matrices \( W_i(t) \) the dynamical system,

\[
\begin{align*}
\dot{x}_i &= W_i^{-1}(x_j - x_i) + \dot{z}_i + W_i^{-1} \dot{W}_i(z_i - x_i), \quad x_i(0) = z_i(0),
\end{align*}
\]

ensures that every sensor estimate \( x_i \) tracks the weighted-average consensus

\[
\lim_{t \to \infty} x_i(t) = \lim_{t \to \infty} \left( \sum_{i=1}^{n} W_i(t) \right)^{-1} (W_i(t) z_i(t)), \quad i = 1, \ldots, n,
\]
3.4. ACP1: Algorithmic Solutions

with zero steady-state error, provided the Laplace transforms of both \( z_i(t) \) and \( W_i(t) \) each have all poles in the left-hand plane with at most one pole at zero. In the switched Markov observation model considered here, we do not assume knowledge of \( \dot{z}_i(t) \) since this would imply direct observation of the parameter \( \{\theta(t_k)\} \). In our framework each sensor observes \( \theta \) only indirectly through changes in the approximated stationary distribution of the observed Markov chains \( X_k^i \), thus an algorithm such as (3.9) cannot be applied to the observation model presumed in [7].

In light of extensive research conducted regarding multi-agent coordination, there are several practical applications of the DLA algorithm in sensor networks, for instance the synchronization of node clocks [64] or distributed load balancing [65]. The problem of synchronizing coupled oscillators by means of DLA was discussed in [33], whereas in [66] the authors consider DLA as a mechanism to ensure a team of unmanned air vehicles (UAV’s) will approach a steady-state wherein each UAV surveys an equal portion of a one-dimensional perimeter. Conversely [58] models a set of mobile agents in the plane and develops a distributed control law for stable flocking behavior. Similarly [23] applies the decentralized network paradigm of coupled agents to explain the consensus behavior of a group of self-driven particles during phase transition, as was previously considered in [67]. In this chapter we will generalize these average-consensus observation models to incorporate (i) input parameters observed in noise that, (ii) jump-change according to a slowly-switching Markov process.

3.4 ACP1: Algorithmic Solutions

This section will describe a consensus protocol that solves ACP1. The protocol involves a local stochastic approximation algorithm that runs in parallel with a distributed linear averaging protocol. In Sec.3.4.1 we define the specific consensus protocol that will be analyzed, and in Sec.3.4.2 we formally state the convergence results for this protocol. In Sec.3.4.4 we discuss the communication graph weight conditions regarding the average-consensus formation, and numerical examples are presented in Sec.3.5. Lastly, we provide conclusions and directions of future work in Sec.3.6.

3.4.1 ACP1: Stochastic Approximation and Consensus Protocol

We begin by considering the assumption that each node \( i \in \mathcal{V} \) maintains a local stochastic approximation algorithm in relation to the input \( X_k^i \in \mathbb{R}^{S \times 1} \), that is,

\[
\tilde{s}^i(k + 1) = (1 - \mu)\tilde{s}^i(k) + \mu X_k^i, \quad \tilde{s}_0^i = X_0^i.
\]  

(3.10)
The knowledge set $K_i(t)$ of each node is then assumed to satisfy the following modification to (A4),

- (A4'): At any time $t_k > 0$, the knowledge set $K_i(t_k)$ of each node $i \in \mathcal{V}$ satisfies $K_i(t_k) \supseteq \{i, n, s_i(t_k), \tilde{s}_i(t_k), X_i(t_k)\}$.

Next we define the consensus protocol that will be analyzed in Sec. 3.4.2. The protocol must conform to the regime-modulated communication sequence defined by (A5)-(A7) and thus it is characterized by a set of synchronous, directed, instantaneous signals. We propose a “distributed linear averaging” (DLA) scheme in which each node’s consensus estimate $\hat{s}_i(t)$ is updated to a weighted average of its current value and the current values of the consensus estimates held at each node from which it receives a signal. In terms of the consensus protocol $(3.3)-(3.4)$ we then have the specification:

<table>
<thead>
<tr>
<th>Consensus Protocol: Distributed Linear Averaging (DLA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal Specification: $S^{ij}(t_k, t_k) = K_j(t_k) \backslash {j, n, \tilde{s}<em>j(t</em>{k+1})}$, $(i, j) \in \mathcal{V}^2$</td>
</tr>
</tbody>
</table>

$$S^{ij}(t_k, t_k) = K_j(t_k) \backslash \{j, n, \tilde{s}_j(t_{k+1})\}, \quad (i, j) \in \mathcal{V}^2$$

(3.11)  

| Knowledge Set Update: $K_i(t_{k+1}) = \{i, n, \hat{s}_i(t_{k+1}), \tilde{s}_i(t_{k+1}), X_i(t_{k+1})\}$ |

$$K_i(t_{k+1}) = \{i, n, \hat{s}_i(t_{k+1}), \tilde{s}_i(t_{k+1}), X_i(t_{k+1})\}$$

(3.12)  

| Stochastic Approximation Update: $\tilde{s}_i(t_{k+1}) = (1 - \mu) \hat{s}_i(t_k) + \mu X_i(t_k)$ |

$$\tilde{s}_i(t_{k+1}) = (1 - \mu) \hat{s}_i(t_k) + \mu X_i(t_k)$$

(3.13)  

| Consensus Estimate Update: $\hat{s}_i(t_{k+1}) = (1 - \mu (w_{ij}^s(t_k) + w_{ij}^o(t_k))) \hat{s}_i(t_k)$ |

$$\hat{s}_i(t_{k+1}) = (1 - \mu (w_{ij}^s(t_k) + w_{ij}^o(t_k))) \hat{s}_i(t_k)$$

$$+ \mu w_{ij}^s(t_k) \tilde{s}_j(t_k) + \mu w_{ij}^o(t_k) X^j(t_k)$$

(3.14)  

| Estimate Initialization: $\hat{s}_i(0) = X^i(0), \quad \tilde{s}_i(0) = X^i(0)$. |

$$\hat{s}_i(0) = X^i(0), \quad \tilde{s}_i(0) = X^i(0).$$

(3.15)

The coefficients $w_{ij}^s(t_k)$ and $w_{ij}^o(t_k)$ are the averaging weights with which the DLA protocol respectively assigns to the consensus estimate $\tilde{s}_j(t_k)$ and observation $X^j(t_k)$ contained in the received signal $S^{ij}(t_k, t_k)$.

Due to the instantaneous and synchronous nature of the communication sequence, we can write the consensus estimate (3.14) update more compactly via the linear equation,

$$\hat{s}(t_{k+1}) = (I_{Sn} - \mu(D^s(t_k) + D^o(t_k) - W^s(t_k))) \hat{s}(t_k) + \mu W^o(t_k) X(t_k)$$

(3.16)

where $\hat{s}(t_{k+1}) = [\hat{s}_1(t_k), \ldots, \hat{s}_n(t_k)] \in \mathbb{R}^{(Sn) \times 1}, \ X(t_k) = [X^1(t_k), \ldots, X^n(t_k)] \otimes I_S$, $W^o(t_k) = W^o(t_k) \otimes I_S$, $W^s(t_k) = W^s(t_k) \otimes I_S$, $W^o(t_k) = [w_{ij}^o(t_k)]$, $W^s(t_k) = [w_{ij}^s(t_k)]$, $\otimes$ denotes the...
Kronecker product, \( D^o(t_k) = W^o(t_k)1_{Sn}, \) \( D^s(t_k) = W^s(t_k)1_{Sn}, \) \( 1_\ell \) denotes an \( \ell \times 1 \) vector of unit valued elements, and \( I_\ell \) is the \( \ell \times \ell \) identity matrix. Letting \( H(t_k) = (I_{Sn} - \mu(D^s(t_k) + D^o(t_k) - W^s(t_k))) \) we can then re-write (3.16) as,

\[
\hat{s}(t_{k+1}) = H(t_k)\hat{s}(t_k) + \mu W^o(t_k)X(t_k).
\] (3.17)

In Sec.3.4.2 we discuss the coefficient weight conditions necessary and sufficient for average-consensus to be obtained under the DLA protocol (3.11) – (3.15) or equivalently (3.17).

### 3.4.2 ACP1: Convergence Results

This section states the weak-convergence results obtained for the DLA protocol. In addition to the convergence of the iteration (3.17), we also characterize the convergence of an estimate for the cumulative distribution of \( \bar{\pi}(t_k) \), as well as the scaled tracking error and set-valued switching ODE.

To proceed, we pose the following conditions (Condition (B) is re-stated here for the reader’s convenience).

1. For each \( \theta \in M \), the transition matrix \( A^i(\theta) \) is irreducible and aperiodic for each \( i \in \mathcal{V} \).
2. Parameterize the transition probability matrix of \( \theta \) as
   \[
P^\epsilon = I + \epsilon Q,
\]
   where \( \epsilon \) is a small parameter satisfying \( 0 < \epsilon \ll 1 \) and \( Q \) is the generator of a continuous-time finite-state Markov chain.
3. The process \( \theta_k \) is slow in the sense \( \epsilon = O(\mu) \). For simplicity, we take \( \epsilon = \mu \) henceforth.

(C) All eigenvalues of the matrix \( H = (\mathcal{L} + D^o) \) have positive real parts, or non-negative real parts. We denote these exchange graph conditions as \( 0 \prec H \) and \( 0 \preceq H \) respectively.

Condition (B) specifies our observation model as a two-time-scale Markov system. Condition (C) is a constraint on the network exchange graphs \( \{\mathcal{G}^o, \mathcal{G}^s\} \) and ensures bounded stability of (3.16) in the limit as \( \mu \to 0 \). We note that \( 0 \prec H \) implies \( -H \) is a Hurwitz matrix.

**Theorem 3.4.1** Assume conditions (B)-(C) and suppose \( s(0) \) is independent of \( \mu \). Define the continuous-time interpolated sequences of iterates

\[
s^\mu(t) = s_k, \quad \theta^\mu(t) = \theta_k \quad \text{for} \quad t \in [k\mu, (k+1)\mu).
\]
Then as $\mu \to 0$, $(s^\mu(\cdot), \theta^\mu(\cdot))$ converges weakly to $(s(\cdot), \theta(\cdot))$ such that $\theta(\cdot)$ is a continuous-time Markov chain with generator $Q$ and $s(\cdot)$ satisfies

$$\frac{ds(t)}{dt} = -Hs(t) + W^\pi(\theta(t)), \quad t \geq 0, \quad s(0) = X(0). \quad (3.18)$$

Note that if $s(0) = s^\mu(0)$, we require $s^\mu(0)$ converges to $s(0)$ weakly. However, for simplicity, we choose $s(0)$ to be independent of $\mu$. The above theorem implies the sensor iterates resulting from (3.16) converge weakly to a Markov switched ODE. This is in contrast to the standard analysis of stochastic approximation algorithms [68] where the limiting process is a deterministic ODE. The proof of the theorem uses the martingale problem formulation of Stroock and Varadhan, see also [49]. If a consensus is obtained by all sensors, that is $\hat{s}_i(t) = \hat{s}_j(t) \forall i, j \in V$, then this consensus is a stochastic process dictated by $\theta$. Again this is in contrast to works concerned with a static consensus formation (i.e. [16, 18, 19, 24, 25]), as well as others wherein the average-consensus estimate is a linear combination of time-varying signals unanimously observed by all sensors [8] or with observed rates of change [7]. In comparison, the consensus estimate we obtain is an average of piece-wise fixed finite-state Markov chain stationary distributions, where it may be assumed each Markov chain is observed by only one sensor.

**Long-Time Horizon.** Theorem 3.4.1 states a convergence result for small $\mu$ and large $k$ such that $\mu k$ remains bounded. However, if the consensus-tracking algorithm (3.16) is in operation for a long time, we would like to establish its behavior for a large-time horizon. We thus next consider the case when $\mu$ is small and $k$ is large such that $\mu k \to \infty$. This is essentially a stability result corresponding to the limit of the switching ODE (3.18) as $t \to \infty$. For this result we assume that $Q$ is irreducible or equivalently, the associated continuous-time Markov chain $\theta(t)$ is irreducible.

Recall that the irreducibility means that the system of equations,

\begin{align*}
\nu'Q &= 0 \\
1_m'\nu &= 1, \quad (3.19)
\end{align*}

has a unique solution satisfying $\nu_j > 0$ for all $j = 1, \ldots, m$. The vector $\nu = (\nu_1, \ldots, \nu_m)' \in \mathbb{R}^m$ is then the stationary distribution of $\theta$. Since $\theta$ is a finite-state Markov chain, we note that if $0 < H$ then

$$\sum_{i=1}^m \int_0^T \exp(-H(T-u))W^\pi(\theta^i)I_{\theta(u) = \theta^i} du$$

converges w.p.1 as $T \to \infty$ due to the exponential dominance of the term $\exp(-H(t-u))$. A
closer scrutiny permits an expression of this limit in closed form. Denote

\[ s_* = H^{-1} \sum_{i=1}^{m} \mathcal{W}^o \pi(\theta^i) \nu_i. \]

**Theorem 3.4.2** Assume conditions of Thm.3.4.1 with the modifications that \( Q \) is irreducible and \(-H\) is Hurwitz. Then for any sequence \( \{t_\mu\} \) satisfying \( t_\mu \to \infty \) as \( \mu \to 0 \), \( s^\mu(\cdot + t_\mu) \) converges weakly to \( s_* \). Moreover, for any \( 0 < T < \infty \), \( \sup_{|t| \leq T} |s^\mu(t + t_\mu) - s_*| \to 0 \) in probability.

In Theorem 3.4.2, the generator \( Q \) is assumed irreducible. We may also consider the case when \( Q \) has an absorbing state and thus the remaining \( m - 1 \) states are transient. This implies there is a zero row in \( Q \). In this case, although the Markov chain is not irreducible, the limit probability distribution still exists. This distribution is a vector with one component equal to unity (corresponding to the absorbing state), and the remaining components equal to zero. As a result, the techniques used in Thm.3.4.2 can still be applied. We state the following result, but omit its proof for brevity.

**Theorem 3.4.3** Assume the conditions of Thm.3.4.2 with the modification that the Markov chain \( \theta \) has an absorbing state. Without loss of generality, denote this state by \( \theta^1 \). Then for any sequence \( \{t_\mu\} \) satisfying \( t_\mu \to \infty \) as \( \mu \to 0 \), \( s^\mu(\cdot + t_\mu) \) converges weakly to \( s_a \), where

\[ s_a = H^{-1} \mathcal{W}^o \pi(\theta^1). \]

Moreover, for any \( 0 < T < \infty \), \( \sup_{|t| \leq T} |s^\mu(t + t_\mu) - s_a| \to 0 \) in probability.

**Set-Valued Consensus Formation.** In Theorem 3.4.1, it is assumed that for each \( \theta \), the observed Markov chains \( \{X_k^i : i \in \mathcal{V}\} \) will each have a respective transition matrix \( A^i(\theta) \) that is irreducible and aperiodic. This is an ergodicity condition that may be generalized to the case when the observed stationary distributions are not unique, but instead each belong to a set \( \mathcal{A}^i(\theta) \) of transition matrices with corresponding set of stationary distributions \( \Gamma^i_\pi(\theta) \). By the same DLA algorithm (3.16) then each sensor can reach a set-valued average-consensus regarding the collection of sets \( \Gamma(\theta) = [\Gamma^1_\pi(\theta), \ldots, \Gamma^n_\pi(\theta)] \). As each observed Markov chain \( X_k^i \) will have a stationary distribution belonging to exactly one element in the set \( \Gamma^i_\pi(\theta) \), then under (3.16) the collection of values that the estimate \( s^i \) will converge weakly to is given by,

\[
\mathcal{P}^i(\theta) = \bigcup_{\pi^i \in \Gamma^i_\pi(\theta)} \{\sum_{j=1}^{n} \psi_{ij} \pi^i(\theta)\}
\]  

(3.20)
3.4. ACP1: Algorithmic Solutions

where $\Lambda = (\psi_{ij})$ is a constant matrix. The set-valued average-consensus $\bar{P}(\theta)$ may likewise be expressed,

$$
\bar{P}(\theta) = \bigcup_{\pi^i \in \Gamma^+_i(\theta)} \left\{ \frac{1}{n} \sum_{i=1}^{n} \pi^i(\theta) \right\}.
$$

(3.21)

Due to the linearity of (3.16), the assumption of set-valued transition matrices $A^i(\theta)$ is in fact equivalent to considering set-valued exchange graphs $\{G^v, G^o\}$, since different exchange graphs result in each sensors estimation of a different linear combination of the time-varying singletons, $\{\pi^1(\theta_k), \ldots, \pi^n(\theta_k)\}$. Although some of these combinations may imply the formation of a consensus (i.e. $\lim_{t \to \infty} s^i(t) = s^j(t)$ for all $i, j \in V$ and each $\theta \in M$) this is certainly not in general true, and thus again leads to consideration of consensus formation not to a fixed point, but rather to a set such as (3.20) or (3.21). We next consider the case of set-valued transition matrices $A^i(\theta)$ and demonstrate that, in the limit as $\mu$ vanishes, the switching ODE (3.18) is replaced by a switching differential inclusion.

**Theorem 3.4.4** Assume the conditions of Thm.3.4.1 with the following modification. In lieu of the irreducibility and aperiodicity assumption of $A(\theta)$ in condition (B), assume for each $\theta \in M$ there is an invariant measure $\pi \in \Gamma_\pi(\theta)$ such that

$$
\operatorname{dist}\left(\frac{1}{n} \sum_{k=\ell}^{n+\ell-1} E_\ell X_k(\theta), \Gamma_\pi(\theta)\right) \to 0 \quad \text{in probability as } n \to \infty,
$$

where $\operatorname{dist}(x, B) = \inf_{y \in B} |x - y|$ is the usual distance function. The conclusion of Thm.3.4.1 is then modified as $(s^\mu(\cdot), \theta^\mu(\cdot))$ converges weakly to $(s(\cdot), \theta(\cdot))$ such that

$$
\frac{ds(t)}{dt} \in -Hs(t) + W^\sigma \Gamma_\pi(\theta(t)).
$$

(3.22)

We note the conditional set of estimates reached by each sensor may be viewed as the result of uncertainty regarding the observed Markov transition matrices, or in other words uncertainty in the signal dynamics. Equivalently, the conditional set of estimates can be seen as the result of uncertainty in the network exchange graphs, and hence the use of set-valued parameters rather than singletons.

3.4.3 Scaled Tracking Error and Cumulative Distribution Function (CDF) Estimation

Since the network consensus converges to a stochastic process, it is important to examine the asymptotic convergence rate of the algorithm (3.16). We proceed here with the study of the
scaled tracking errors of the sensor state-values \( s_k \). Then we consider consensus formation when sensors estimate the empirical CDF instead of the probability mass function.

We have shown that (3.18) implies the sensor estimates \( s(t) \) will converge weakly to a stochastically switching steady-state \( \Lambda \pi(\theta(t)) \) for some fixed equilibrium matrix \( \Lambda \in \mathbb{R}^{S_n \times S_n} \).

We now define the scaled tracking error

\[
v_k = \frac{s_k - \Lambda E(\pi(\theta_k))}{\sqrt{\mu}} \tag{3.23}
\]

where \( s_k \) is the vector of state-value estimates obtained from (3.16).

Next, assume that there exists a constant \( k_\mu > 0 \) sufficiently large such that \( \{v_k : k \geq k_\mu\} \) is tight, which implies that the normalized error process scaled by \( 1/\sqrt{\mu} \) does not blow up as \( k \to \infty \) and \( \mu \to 0 \). Let us explain this assumption. The requirement \( k \geq k_\mu \) is owing to the effect of initial conditions; the sequence of scaled errors needs to have some time to settle down. Recall that \( \{v_k : k \geq k_\mu\} \) is tight, if for any \( \delta > 0 \), there is a \( K_\delta \) such that for all \( n \geq \max(k_\mu, K_\delta) \), \( P(|v_k| \geq K_\delta) \leq \delta \). To verify this condition, by virtue of Tchebeshev’s inequality, it suffices to have a suitable Liapunov function \( \tilde{U}(\cdot) \) such that \( E\tilde{U}(v_k) \) is bounded. If the random noise is uncorrelated, this boundedness can be obtained easily. If correlated noise is involved, tightness is established using the perturbed Liapunov function approach [49] (see also [68, Chapter 10] for detailed discussion and references). In this approach, small perturbations are added to the Liapunov function to deal with the correlation, and the perturbations are constructed so as to result in the needed cancellations. We omit the verbatim proof and refer the reader to the aforementioned references for further details.

**Theorem 3.4.5** Assume the conditions of Theorem 3.4.1. Then the interpolated sequence of iterates \( v^\mu(\cdot) \) defined by \( v^\mu(t) = v_k \), for \( t \in [(k - k_\mu)\mu, ((k + 1) - k_\mu)\mu) \) converges weakly to a solution \( v(\cdot) \) of the switching diffusion

\[
dv(t) = -Hv(t)dt + (W^\mu\Sigma(\theta(t))W^\mu)^{1/2}dw, \tag{3.24}
\]

where \( w(\cdot) \) is a standard Brownian motion and for a fixed \( \theta \), \( \Sigma(\theta) \) is the covariance given by

\[
\Sigma(\theta) = \lim_{n \to \infty} \frac{1}{n} E \sum_{k=\ell}^{n+\ell-1} \sum_{j=\ell}^{n+\ell-1} (X_k(\theta) - EX_k(\theta))(X_j(\theta) - EX_j(\theta))'. \tag{3.25}
\]

From (3.24) it is clear the sensor averaging weights \( W^\mu \) have a direct effect on the sensors tracking error, in particular we see that any scaling of the averaging weights \( W^\mu \) implies the same scaling of the sensor diffusion process.
Average-Consensus on the Cumulative Distribution Function (CDF). So far we have assumed each sensor \( j \in V \) observes the state of a fast Markov chain \( X^j \) and tracks by (3.6) the associated stationary probability mass function \( \pi^j(\theta(t)) \). We now consider, for a given \( j \in V \), the approximation of the CDF of \( X^j_k \) by means of empirical measures. For each \( \theta \in M \), the CDF associated with \( X^j \) is denoted by \( \Pi^j(\theta, x) \) for any \( x \in \mathbb{R}^S \). For any \( j \in V \), \( 0 < T < \infty \), and any \( x \in \mathbb{R}^S \), define the empirical measure

\[
\eta_k = \frac{1}{k} \sum_{k_1=0}^{k-1} I\{X^j_{k_1} \leq x\}, \quad 0 \leq k \leq T/\varepsilon.
\]

Note that \( y \leq x \) with \( x = (x^i) \in \mathbb{R}^S \) and \( y = (y^i) \in \mathbb{R}^S \) is understood to hold component-wise (i.e., \( y^i \leq x^i \) for \( i = 1, \ldots, S \)). The sequence \( \eta_k \) may be written recursively as

\[
\eta_{k+1} = \eta_k - \frac{1}{k+1} \eta_k + \frac{1}{k+1} I\{X^j_{k} \leq x\}.
\]

So for sufficiently large \( k \), the empirical CDF can be estimated as

\[
\eta_{k+1} = \eta_k + \mu I\{X^j_{k} \leq x\}, \quad (3.26)
\]

for arbitrarily small \( \mu > 0 \). Define the continuous-time interpolated process \( \eta^\mu(t) = \eta_k \) for \( t \in [\mu k, \mu k + \mu) \). In the above, for simplicity, we have suppressed the \( j \)-dependence in both \( \eta_k \) and \( \eta^\mu \).

The following theorem is analogous to Theorem 3.4.1 but deals with average-consensus of the empirical CDF.

**Theorem 3.4.6** Under condition (B), \((\eta^\mu(\cdot), \theta^\mu(\cdot))\) converges weakly to \((\eta(\cdot), \theta(\cdot))\) such that \( \eta(\cdot) \) satisfies the switching ordinary differential equation

\[
\dot{\eta} = \pi^j(\theta(t))\Pi^j(\theta(t), x). \quad (3.27)
\]

**Remark 3.4.7** In analogy to Theorem 3.4.5 (which dealt with the probability mass function), we now comment on the scaled tracking error for average-consensus on the CDF when using (3.26). To proceed, define

\[
\xi^{\mu,j}(t, x) = \sqrt{\mu} \sum_{k=0}^{\lfloor t/\mu \rfloor - 1} \left[ I\{X^j_k \leq x\} - \Pi^j(\theta_k, x) \right].
\]

Using a combination of the techniques in the proof, Markov averaging and standard results in centered and scaled errors for empirical measures, we can show that \( \xi^{\mu,j}(t, x) \) converges weakly
3.4. ACP1: Algorithmic Solutions

to \(\xi(t,x)\), a switching Brownian bridge process. However, unlike the proof of Theorem 3.4.1, we can no longer use the martingale problem formulation. Nevertheless, we can use the basic techniques of weak convergence; see [69] for the related study on Brownian bridge processes. We illustrate in Sec.3.5 a sample path of the Brownian bridge resulting from the sensor scaled tracking error when a consensus regarding the average \(\Pi(\theta(t)) = \frac{1}{n} \sum_{i=1}^{n} \Pi^i(\theta(t))\) is obtained via (3.16).

Compared to \(\pi(\theta(t))\), estimation of \(\Pi(\theta(t))\) implies one less state-value of \(X^i\) for each \(i \in \mathcal{V}\) need be transmitted between sensors, thus reducing by a factor of \((1 - 1/S)\) any number of physical constraints on the sensor operations per communication link. If the number of states \(S\) is near unity, then estimation of \(\Pi(\theta(t))\) may result in a significant reduction of the resources required for sensor communication and thus consensus. The results of estimating \(\Pi(\theta(t))\) as compared to \(\pi(\theta(t))\) are illustrated in Sec.3.5, where it is clear estimation of the distribution function also results in greater scaled tracking error among the mid-states of \(S\) than the tracking error among extremum states.

3.4.4 Average-Consensus Exchange Graph Conditions

The continuous-time sensor estimates \(s(t)\) resulting from (3.16) was stated in Sec.3.4.2 to converge weakly to the solutions of (3.18) under the conditions \((B) - (C)\). In particular, it was assumed that \(\{G^v, G^o\}\) are such that all eigenvalues of \(H = (\mathcal{L} + \mathcal{D}^o)\) have either positive real parts or non-negative real parts. These basic constraints on the network exchange graphs are sufficient for the asymptotic convergence of (3.16), but are not sufficient to ensure each sensor \(i \in \mathcal{V}\) asymptotically obtains the average-consensus estimate \(\bar{\pi}(\theta(t))\). To proceed in this direction we will assume the following property on \(G^v\) holds in addition to the conditions \((B) - (C)\) stated in Sec.3.4.2.

(D) The graph \(G^v\) is non-time-respecting strongly connected \((\text{ntrcs})\), thus for any two nodes \(i, j \in \mathcal{V}\), there exists a path (a sequence of directed edges in \(E^v\)) that starts at node \(i\) and ends at \(j\).

The reason for condition \((D)\) is to simplify the general weight conditions required for average-consensus. In fact, with regard to consensus formation, only when assuming this connectivity condition on \(G^v\) may we assume a decentralized structure in \(G^o\). To see this, and as well motivate condition \((D)\), we note that if \(E^v\) were not \(\text{ntrcs}\) then there would exist a non-empty set of ordered pairs \(\mathcal{V}_{\text{SC}} = \{(i_1, j_1), \ldots, (i_h, j_h)\}\) for each element of which \(E^v\) contains no path. To then reach consensus by DLA would require \((i_l, j_l) \in E^o\) for each \((i_l, j_l) \in \mathcal{V}_{\text{SC}}\), thus implying a non-scalable and “centralized” structure in \(G^o\). Similarly, to ensure a network agreement each
node contained in an element of $V_{SC}$ would require an individual treatment. Although this is possible, we assume $G^v$ is \texttt{ntrcs} to avoid any centralization assumptions on $G^o$ and as well avoid a more detailed scrutiny that if necessary may be left as future research.

It is emphasized that for generality we seek to reach the average-consensus $\bar{\pi}(\theta(t))$ under the condition that in each regime the observed stationary distributions $\{\pi^1(\theta(t)), \ldots, \pi^n(\theta(t))\}$ are \textit{distinct}, that is

$$\pi^i(\theta(t)) \neq \pi^j(\theta(t)) \text{ for all } i, j \in V \text{ and any } \theta(t) \in M.$$  \hspace{1cm} (3.28)

If it were specifically known that two sensors may observe the \textit{same} stationary distribution, i.e. $\pi^i(\theta^*) = \pi^j(\theta^*)$ for some $i, j \in V$ and $\theta^* \in M$, then the average-consensus requirements stated in the lemma below could be relaxed conditional on the fact $\theta(t) = \theta^*$. We consider the most general case and do not make any assumptions regarding equality among the $\pi^i(\theta)$, in other words we assume (3.28) holds.

\textbf{Lemma 3.4.8} Assume conditions $(A) - (D)$, then each sensor estimate based on the discrete-time algorithm (3.16) will in the limit as $\mu \to 0$ possess the steady-state $\bar{\pi}(\theta(t))$ conditional on $\theta(t)$ if and only if one of the following two conditions hold,

- \textit{Condition (1)} $: 0 \preceq (L + D^o)$, $L$ is balanced, and $W^o = \alpha(L + D^o)$, where $\alpha \in \mathbb{R}$ .
- \textit{Condition (2)} $: 0 \prec (L + D^o)$, and the following equation holds,

$$ (L + D^o)^{-1}W^o = \Lambda $$ \hspace{1cm} (3.29)

for any matrix $\Lambda$ with the form,

$$ \Lambda = I + \beta(11' - nI), \ \beta = \text{diag}(\beta_1I_{S \times S}, \ldots, \beta_nI_{S \times S}) \in \mathbb{R}^{S \times S} \hspace{1cm} (3.30) $$

where $\beta$ is non-singular.

We here clarify that under condition $(C)$, if $0 \preceq (L + D^o)$ then we have $D^o = 0$ and the matrix $H = L + D^o$ will have only one eigenvalue with a real part equal to zero, with the corresponding eigenvector equal to $1$. To see this requires only noting that $D^o$ is diagonal and $Lv = 0$ if and only if $v = 1$ when $G^v$ is \texttt{ntrcs} [18], the result that $Hv = 0$ if and only if $v = 1$ follows immediately, as does the requirement $D^o = 0$. Thus under conditions $(A) - (C)$ we have both $W^o = \alpha L$ and $E^o = E^v$. Despite this simplification, we may still write $W^o = \alpha(L + D^o)$ since it technically holds and also maintains a clear notational consistency between conditions (1) and (2).
The consensus-tracking algorithm (3.16) requires either \( 0 \preceq (\mathcal{L} + \mathcal{D}^o) \) or \( 0 \prec (\mathcal{L} + \mathcal{D}^o) \) in order for the sensor estimates \( s_k \) to remain bounded in the limit as \( \mu \) vanishes and \( t \) increases. This is in parallel with the constraints \( 0 \preceq \mathcal{L} \) and \( 0 \prec \mathcal{L} \) that are required for the static Laplacian consensus algorithm (3.7) to remain bounded under the same asymptotic limits. Thus, unlike the works of [17, 18, 24], which consider the static algorithm (3.7), we may permit \( \mathcal{L} \) have negative eigenvalues. Denoting the minimum of these \( \lambda \), by then taking \( D^o = |\lambda|I + V \) for any diagonal matrix \( V \geq 0 \), we have \( (\mathcal{L} + D^o) = U(\mathcal{J} + |\lambda|I)U^{-1} + V = UJ^*U^{-1} + V = \tilde{\mathcal{L}} + \tilde{D}^o \), where now \( 0 \preceq \tilde{\mathcal{L}} = UJ^*U^{-1} \) and \( \tilde{D}^o = V \).

### Necessary Exchange Graph Edge Sets

The necessary and sufficient condition for average-consensus to be obtained by sensors under (3.16) and its sub-algorithms is that either one of the conditions stated in Lemma (3.4.8) holds. Each of these conditions specifically assume either \( 0 \prec H \) or \( 0 \preceq H \) (recall \( H = \mathcal{L} + \mathcal{D}^o \)). In both cases, the constraints posed on the exchange graphs \( \{\mathcal{G}^o, \mathcal{G}^v\} \) are based on the requirement that for average-consensus \( \Lambda \) must have, in each row, identical non-diagonal terms that do not equal zero. By then supplementing the estimates \( \hat{s}(t) \) with the local SA tracking estimates \( \tilde{s}(t) \) and normalizing by \( (n\beta)^{-1} \), the average \( \bar{\pi}(\theta(t)) \) is obtained under Condition (2). Under Condition (1) a similar linear procedure results in the estimate \( \bar{\pi}(\theta(t)) \) at all nodes.

We now characterize the exchange graph edge sets \( \{\mathcal{E}^o, \mathcal{E}^v\} \) for which either of the conditions stated in Lemma 3.4.8 are feasible. Without loss of generality we take \( S = 1 \), we also define the neighborhood of sensor \( i \) as \( \mathcal{E}^v_i = \{j : (i, j) \in \mathcal{E}^v\} \) and the complementary edge set of \( \mathcal{E} \) as \( \tilde{\mathcal{E}} \), that is \( (i, j) \in \tilde{\mathcal{E}} \) if and only if \( (i, j) \notin \mathcal{E} \). We assume that no sensor receives information directly from all other sensors, since otherwise we would have a centralized data fusion problem that would render a distributed algorithm such as (3.16) unnecessary. As a consequence of this assumption each row of either matrix \( \{\mathcal{W}^o, \mathcal{W}^v\} \) has at least one non-diagonal element equal to zero, thus there exists for each row \( i \) an element \( j^o_i \) such that \( (i, j^o_i) \notin \mathcal{E}^o \) and an element \( j^v_i \) such that \( (i, j^v_i) \notin \mathcal{E}^v \).

Below we state 3 results related to the necessary and sufficient communication graph edge sets for an average-consensus formation.

**Lemma 3.4.9** If \( \mathcal{E}^o = \mathcal{E}^v \) then average-consensus is possible only if Condition (1) of Lemma (3.4.8) holds.

**Lemma 3.4.10** If \( \mathcal{E}^o \neq \mathcal{E}^v \) then average-consensus is possible only if Condition (2) of Lemma (3.4.8) holds and \( \mathcal{E}^o_i \supseteq \mathcal{E}^v_i \) for at least one sensor \( i \in \mathcal{V} \).

**Corollary 3.4.11** If \( \mathcal{E}^o \subset \mathcal{E}^v \) then average-consensus is not possible.
3.5. ACP1: Numerical Examples

The statement $\mathcal{E}_i^o \supseteq \mathcal{E}_i^v$ implies that every sensor that does not send sensor $i$ state-value data must send sensor $i$ observation data. For sparse networks we might presume the cardinality of $\mathcal{E}_i^v$ is $|\mathcal{E}_i^v| \ll n$, and thus when $\mathcal{E}_i^o \neq \mathcal{E}_i^v$ by Lemma (3.4.10) an average-consensus requires $|\mathcal{E}_i^o| \approx n$ for some $i^* \in \mathcal{V}$. We note this implies that either a large group of sensors $h \subset \mathcal{V}$ send observation data to sensor $i^*$, or equivalently that sensor $i^*$ can itself observe the Markov chains that are observed by a large group of other sensors in $\mathcal{V}$.

3.5 ACP1: Numerical Examples

In this section we will provide numerical examples to illustrate how the theorems in Sec.3.4.2 - 3.4.3 result in the weak-convergence of each sensors estimate to the average-consensus distribution $\bar{\pi}(\theta(t))$. We consider two communication networks that each satisfy one of the following conditions,

- (Model 1) $\mathcal{W}_o = -\mathcal{L}$, $\mathcal{W}_v$ is balanced, and $\{\mathcal{G}_v, \mathcal{G}_o\}$ are such that $\mathcal{E}_i^v = \mathcal{E}_i^o$,
- (Model 2) $\{\mathcal{W}_o, \mathcal{W}_v\}$ satisfy $\{\mathcal{G}_v, \mathcal{G}_o\}$ and are such that $\mathcal{E}_i^v \neq \mathcal{E}_i^o$.

In the first Model $H \ll 0$, whereas in Model 2 $H < 0$. In both cases we set the network size at $n = 24$ sensors, and consider $S = 40$ observed states. Thus each sensor $i \in \mathcal{V}$ will observe $X^i$ in one of 40 states upon each iteration. In our simulation $\theta$ takes on 6 different states. For each of these regimes, the stationary distribution $\pi^i(\theta)$ of each Markov chain $X^i$ was randomly generated from the uniform distribution, thus validating an a priori knowledge set assumption.

The total number of communication links, that is the total number of elements in $\mathcal{E}_i^o$ and $\mathcal{E}_i^v$, is fixed at $2n(n-1) = 1104$. We also fix the sum of absolute averaging weights at approximately the constant 2755,

$$|\mathcal{E}_i^v| + |\mathcal{E}_i^o| = 2n(n-1), \quad 1'(|\mathcal{L}| + |\mathcal{W}_o|)1 \approx 2755.$$ 

Neither of these constraints are necessary, they are set identical for either model to better compare the two different average-consensus graph conditions. We note that under condition (1) the minimum number of edges required for an average-consensus can be proven $2n$, whereas under condition (2) this number is always strictly greater than $2n$.

Setting $\mu = 10^{-9}$ for both models, the sample path of the sensor iterates $s(t)$ under (3.16) is plotted as the unadjusted sensor estimates; see Fig.3.1 (each sensor estimate is plotted in a different color).

The estimates $s(t)_{(1)}$ and $s(t)_{(2)}$ are plotted as the adjusted sensor estimates under Model 1 and 2 respectively. We find that, in accordance with the adaptation rate $\lambda_2$, Model 1 converges
3.5. ACP1: Numerical Examples

Figure 3.1: Sensor estimation of the average-consensus $\bar{\pi}(\theta(t))$ under both Models 1 and 2. For illustrative purposes we have, without loss of generality and for this figure only, set $X(0) = 0$ and magnified the sensor observations of $X^i$ by a factor proportional to the number of previous states $\theta$ has occupied in the simulation.

Table 3.1 is that our simulations imply the ratio holds in approximation (we note that the two values under $U_{\text{samp}}$ refer to sensor estimation of $\pi(\theta)$ and $\Pi(\theta)$ respectively).

As a trade-off to the improved adaptation rate, there is an increase in the scaled error under Model 1 when averaged across sensors. This is shown in Fig.3.2, where we have plotted for both models the sample scaled tracking error when estimating $\pi(\theta)$, as well as estimation of $\Pi(\theta)$ by the empirical measure described in Sec.3.4.3. In Fig.3.2 the scaled error associated with each observed state (S=40) is plotted in a different color.

We see that Model 1 results in an increased average scaled error as compared to Model 2. The exact relation between the scaled error and adaptation rate $r$ is not explored here in detail. The analytic and sample variance of the scaled tracking error, as measured relative to the sensor trajectories, are plotted in Fig.3.3.

Our purpose in Fig.3.3 is to illustrate how the coefficients of the Brownian motion $dw$ vary between models (each sensor and observed state ($Sn = 24 \cdot 40 = 960$) are plotted in a different color). For state $\ell$ in $X^i$ and sensor $i \in \mathcal{V}$ under (3.16), the variance of the Brownian motion
3.5. ACP1: Numerical Examples

Figure 3.2: The absolute scaled error $U_{samp}$, plotted consecutively over the entire simulation time $[T_0, T_1]$ for each state in $S$. The results indicate a Brownian bridge across the state estimates ($S = 40$) of the average cumulative distribution $\overline{\Pi}$. The absolute scaled error $U_{samp}$, plotted consecutively over the entire simulation time $[T_0, T_1]$ for each state in $S$. The results indicate a Brownian bridge across the state estimates ($S = 40$) of the average cumulative distribution $\overline{\Pi}$. The absolute scaled error $U_{samp}$, plotted consecutively over the entire simulation time $[T_0, T_1]$ for each state in $S$. The results indicate a Brownian bridge across the state estimates ($S = 40$) of the average cumulative distribution $\overline{\Pi}$. The absolute scaled error $U_{samp}$, plotted consecutively over the entire simulation time $[T_0, T_1]$ for each state in $S$. The results indicate a Brownian bridge across the state estimates ($S = 40$) of the average cumulative distribution $\overline{\Pi}$.

in (3.24) was computed analytically as a function of the averaging weights,

$$
\sigma^2_{id} = \sum_{j=1}^{S_n} (W^o : \Sigma)_{(i-1)S+1+\ell} j, \quad (W^o : \Sigma) = (W^o \Sigma W^o)^{1/2}.
$$

We have plotted in Fig.3.3 the sample values of $\sigma_{id}$ in ascending order, the corresponding analytic values have been plotted in the same order to illustrate the similarity between the analytic and simulated results, thus verifying (3.24) as the correct expression for the scaled sensor tracking error. We note that

$$
\sigma^2 = \sum_{\ell=1}^{S} \sum_{i=1}^{n} \sigma^2_{id}.
$$

Finally, in Fig.3.4, we have plotted various features of our simulations per sensor (each sensor and observed state ($S_n = 24 \cdot 40 = 960$) are plotted in a different color). It is apparent that the ratios between these models regarding analytic variance $\sigma^2_{sensor} i = \sum_{\ell=1}^{S} \sigma^2_{id}$, Frobenius norm $||W^o||^2$, either summed squared weight values $1^t|W^o|^21$ or $1^t|H|^21$, and the absolute sampled error $U_{samp}$, hold in approximation.

In these numerical examples we employed the adaptation rate $r$ and $\sigma^2$ to explain our empirical results. We leave open the question of which model is superior in terms of maximizing $r$ while minimizing $\sigma^2$ and as well the weights and edges, or complexity thereof, required
Figure 3.3: The analytic and sample variance $\sigma_{i\ell}$ of the scaled tracking error under both Models 1 and 2, as given by (3.31). A point exists for each sensor $i \in \mathcal{V}$ and observed state $\ell \in \{1, \ldots, S\}$. The results have been ordered according to the sample variances to better show the similarity between the analytic and simulated variances.

for $\{G^o, G^v\}$ to ensure average-consensus. This question is of importance since the features of $\{G^o, G^v\}$ can be naturally associated with the communication costs assumed of the sensor network [18].

### 3.6 ACP1: Conclusions and Future Work

We have proven that under the linear distributed averaging protocol and a local stochastic approximation algorithm, the sensor state-values within a network communication graph may
Figure 3.4: Unordered Sensor Diffusion Coefficients, Absolute Error, and Functions of Weights.

The similarities between each quantity is apparent by the localization of larger values under Model 2, as compared to the uniform values obtained from Model 1. The approximate ratio is also seen here to generally hold per sensor, although only with $W^o$ replaced by $H$ as seen in Table 3.1.

under suitable averaging weights converge weakly to an average-consensus regarding the state of a two-time-scale Markov system. In our framework this consensus is based on the slowly time-varying stationary distribution of a Markov chain observed by each sensor. The result has been extended to the case of multiple ergodic classes in the observed Markov chains, or equivalently to network exchange graphs $\{G^o, G^v\}$ that belong to a set rather than a fixed parametrization $G = \{V, E, W\}$. The limiting switched ODE is in this case replaced by a switched differential inclusion, thus motivating the idea of set-valued consensus. In addition to this, the sensor scaled tracking error was proven to satisfy a switching diffusion equation and, in the case of CDF estimation by an empirical measure, a switching Brownian bridge.

Necessary and sufficient conditions were derived in regard to the network communication graph edge set and weights required to ensure average-consensus formation. As a result, computation of the correct consensus estimate requires (3.16) in conjunction with its two component sub-algorithms. Thus, obtaining the average-consensus does not require any greater complexity in sensor computing ability or network communication than (3.16) itself assumes. Lastly, we considered the adaptation rate and magnitude of diffusion present in the sampled sensor
3.6. ACP1: Conclusions and Future Work

trajectories, and proposed an optimization problem as well as an approximate ratio that relate both to various features of the sensor averaging weights. Future work may consider the edge set and averaging weights required for fast convergence to the average-consensus estimate without requiring strong connectivity assumptions or increased sum of the absolute averaging weights.
Chapter 4

ACP2: The Static Consensus Problem with \textit{a priori} Unknown Communication

In this chapter we address the consensus problem ACP2 (cf. Def.2.2.2), that is obtaining at each node the average initial input when inter-node communication is completely \textit{a priori} unknown. In particular we assume the communication sequence is asynchronous and possesses arbitrary time-delays. Our main contribution is Thm.4.4.9, which shows that average-consensus can be obtained for general communication patterns if a vector of $O(n)$ is transmitted along with the consensus estimates, where $n$ is the network size. For consensus estimates of $O(n)$ this result implies an order of magnitude less data to be communicated than an “ideal” protocol that floods the initial data through-out the network. In Sec.4.4.4 we provide 3 practical examples from the past literature that assume consensus estimates of $O(n)$.

There exist several results in previous literature that have proven the convergence of distributed algorithms to an asymptotic average-consensus under asynchronous communication and/or time-delayed signals. We note that asynchronous communication \textit{without} time-delays constitute a special case of the framework considered in this thesis. In particular [44,70–73] emphasize the asynchronous nature of their results, but (i) do not allow time-delayed signals, and (ii) assume a \textit{balanced} communication graph at each iteration. We require neither of these assumptions, and thus our consensus protocols achieve average-consensus under more general communication conditions than the cited works, albeit with the cost of an extra $O(n)$ memory and signal storage (recall however that this extra cost may be negligible in various practical applications of the static average-consensus problem ACP2).

Assuming the initial node states remain fixed, a distributed linear averaging protocol will imply that the local consensus estimate at each node can be expressed as a \textit{linear combination} of the set of initial node states. A novel feature of the protocols we propose in this chapter is that each node transmits not only their local consensus estimate, but also a local “normal” vector representing the \textit{linear dependence} their local estimate has on the set of all initial consensus vectors. Under an appropriate least-squares update rule it can be shown that when all elements
of the local normal vector are equal, the node will know that its local consensus estimate equals the average-consensus value. It is by maintaining the local normal vector that each node can obtain average-consensus in the presence of time-varying and asymmetrical time-delays.

To proceed we first define the local consensus input dynamics (Sec.4.1) as well as the inter-node communication assumptions under ACP2 (Sec.4.2). We then re-state ACP2 in these specific terms. A review of the literature with respect to the past algorithms from research in this area is provided in Sec.4.3. Subsequently we present the details of our proposed protocols (Sec.4.4). Numerical examples of the protocols are presented (Sec.4.5) and their performance is compared with those of previously proposed algorithms in the literature. Lastly, in Sec.4.6 we provide some concluding comments and potential directions for future work.

4.1 ACP2: Local Consensus Input Dynamics

Both ACPs described in Chapter 2 assume each node $i \in V$ is initially associated with a local consensus vector $s_i(0) \in \mathbb{R}^{d \times 1}$. Unlike ACP1, the consensus problem ACP2 assumes that the initial local consensus vector is static and does not vary over time. Thus for ACP2, the current average-consensus value $\bar{s}(t)$ defined in (2.1) remains equal to $\bar{s}(0) = \frac{1}{n} \sum_{i \in V} s_i(0)$ for all $t \in \mathbb{N}$. This is why we refer to ACP2 as a “static” ACP.

4.2 ACP2: Inter-node Communication Pattern and Problem Statement

The ACP2 requires a consensus protocol (cf. (4.1) – (4.2)) that operates under an “a priori unknown communication pattern” (UCP). This is the most general form of inter-node communication that can be assumed, and we must further define this type of communication before proceeding with any of the consensus protocols’ convergence results. To do so, we now state 3 assumptions that all consensus protocols must abide by under a UCP.

Definition 4.2.1 (UCP) A consensus protocol operating under an “a priori unknown communication pattern” (UCP) satisfies the following conditions,

- (A5): for each signal $S^{ij}(t_0, t_1)$, node $j$ does not have the ability to know the value of $i$ or $t_1$.
- (A6): for each signal $S^{ij}(t_0, t_1)$, node $i$ does not have the ability to control the value of $j$, $t_0$, or $t_1$. 
4.2. ACP2: Inter-node Communication Pattern and Problem Statement

- (A7): when a signal \(S^{ij}(t_0, t_1)\) is received, the knowledge set \(K_i(t_1)\) of the receiving node is updated by the time instant \(t_1 + 1\). (e.g. the “processing time” of each signal is one unit of time).

Assumptions (A5)-(A6) imply the communication process is a priori unknown at every node, furthermore there are no constraints on the time-delay between the transmission and reception time of any signal. We emphasize that together (A5)-(A6) allow for more general communication patterns than those in [18, 62, 72, 74–76]. The assumption (A7), together with (4.2) below, imply that a signal \(S^{ij}(t_0, t_1)\) may contain information that has been received at node \(j\) preceding time \(t_0\). This fact will be utilized later in Def. A.4.3, formalizing our notion of a “time-respecting communication path” between two nodes. Assumption (A7) is realistic (as well as natural in our discrete-time framework) since the protocols we propose will require only a few arithmetic operations in the update process.

We now re-state assumptions (A1)-(A4) as well as the definition of a consensus protocol for the readers convenience. We will assume,

- (A1): (knowledge set assumption) At any time \(t \in \mathbb{N}\), each node \(i \in V\) is equipped with a device that can update and store a “knowledge set” \(K_i(t)\). For each \(i \in V\), the knowledge set \(K_i(t)\) may have a time-varying cardinality.

- (A2): \(K_i(t) \supseteq \{\hat{s}_i(t)\}, \forall i \in V, \forall t \in \mathbb{N}\), where \(\hat{s}_i(t) \in \mathbb{R}^{d \times 1}\) represents the local estimate of \(s(t)\) at node \(i\) and time \(t\).

- (A3): (signal set assumption) At any time \(t_0 \in \mathbb{N}\), each node \(j \in V\) has the ability to transmit a “signal set” \(S^{ij}(t_0, t_1) \subseteq K_j(t_0)\) that will be received at some node \(i \in V\) at time \(t_1 \in \mathbb{N}\), where \(t_1 \geq t_0\).

- (A4): At any time \(t \in \mathbb{N}\), the knowledge set \(K_i(t)\) of each node \(i \in V\) satisfies \(K_i(t) \supseteq \{i, n, \hat{s}_i(t), s_i(t)\}\).

Given (A1)-(A4), we define a “consensus protocol” (P) in terms of its “knowledge set updating rule” \(f^P_K\{\cdot\}\) together with a “signal specification rule” \(f^P_S\{\cdot\}\). The knowledge set updating rule \(f^P_K\{\cdot\}\) defines the effect that a set of signals has on the knowledge set \(K_i(t_1)\) at the receiving node \(i\). The signal specification rule \(f^P_S\{\cdot\}\) specifies the elements contained in the signal \(S^{ij}(t_0, t_1)\) given as a function of the knowledge set \(K_j(t_0)\) at the transmitting node \(j\).

<table>
<thead>
<tr>
<th>The Consensus Protocol (P) under (A1)-(A4):</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge Set Updating Rule: (f^P_K : K_i(t_1) \bigcup S^{ij}(t_0, t_1) \rightarrow K_i(t_1 + 1)) (4.1)</td>
</tr>
</tbody>
</table>

44
4.3. ACP2: Relation to Past Literature

The consensus problem ACP2 is now re-stated in terms (A1)-(A7) and (4.1) – (4.2).

**Definition 4.2.2 (ACP2)** Under (A1)-(A7), assume that the input consensus parameters \( \{s_i(t) : i \in \mathcal{V}\} \) remain fixed at their initial values \( \{s_i(0) : i \in \mathcal{V}\} \) for all \( t \in \mathbb{N} \). A consensus protocol \((P)\) (cf. (4.1) – (4.2)) then solves the average-consensus problem (ACP2) at time \( t^* \) if as \( t \) approaches \( t^* \), the consensus estimate \( \hat{s}_i(t) \) of each node \( i \in \mathcal{V} \) converges (in the Euclidean norm \( ||\cdot||\)) to the average initial input parameter,

\[
\lim_{t \to t^*} ||\hat{s}_i(t) - \bar{s}(0)|| = 0.
\]

To solve ACP2 (cf. Def.4.2.2) we will propose 4 consensus protocols that each require a specific condition on the UCP \( C_{0,\infty} \) sufficient for their convergence. We devote Sec.4.4.1 to defining the 4 consensus protocols, Sec.4.4.2 to describing the respective UCP conditions sufficient for their convergence, and Sec.4.4.3 to formally state these results. Before proceeding however, we briefly provide an overview of past literature related to the results presented in this chapter.

### 4.3 ACP2: Relation to Past Literature

This chapter considers average-consensus in the presence of asynchronous and time-delayed communication. Several past papers have proposed distributed algorithms for asymptotic average-consensus under asynchronous communication and/or time-delayed signals, but they all also require communication constraints extraneous to ours. For instance, the works \([44,70–73]\) emphasize the asynchronous nature of their results, but they do not allow time-delayed signals and furthermore assume a *balanced* communication graph at each iteration. Conversely, \([18,62,74,76,77]\) show that in the presence of time-delays each local consensus estimate will converge (in some norm) to a common limit, yet they require extra communication constraints that afford a global conservation property. To obtain the conservation property the cited works require some form of *symmetry* in the time-delays,

- (R1): undirected communication links, with a common time-delay in both directions of each link, e.g. \([18,74,76,77]\)
4.3. ACP2: Relation to Past Literature

- (R2): a “response” signal for any initiating signal [62].

Both [74, 77] claim that it is often impossible to obtain average-consensus in the presence of asymmetrical time-delays, despite the previous results of [62]. Furthermore, the algorithm proposed in [36] obtains average-consensus in the presence of asymmetrical and time-invariant time-delays. However, in [36] each node must have a priori knowledge (obtained in a distributive fashion during a “start up phase”) of the underlying (time-invariant) communication digraph. Due to the “normal” vectors, the algorithms we present achieve average-consensus in the presence of (i) time-varying and asymmetrical time-delays, and (ii) time-varying communication digraphs. We do not require (R1) or (R2), nor do we require that the sequence of time-delays or communication digraphs adhere to any fixed pattern.

Similar to most consensus protocols (e.g. [16, 18, 43, 62]), the 4 protocols proposed here require each node to update a local estimate of the true average-consensus value via a weighted average of their current local estimate and the local estimate received from a neighboring node. Linear consensus protocols of this type imply that each local consensus estimate can always be expressed as a linear combination of the set of initial node states. A novel feature of the protocols we propose is that each node transmits not only their local consensus estimate, but also a local “normal” vector representing the linear dependence their local estimate has on the set of all initial consensus variables. When all elements of the local normal vector are equal, the node knows their local consensus estimate equals the average-consensus value. It is by maintaining a local normal vector that each node can obtain average-consensus in the presence of time-varying and asymmetrical time-delays.

Similar to our proofs, the works [43, 78, 79] do not require strict conservation properties or a priori knowledge of the communication pattern. However, these works can only prove asymptotic consensus on some unpredictable value, which is not useful in many practical applications [16, 36, 62, 80]. In contrast, the weighted gossip algorithm proposed in [75] guarantees asymptotic average-consensus without a conservation property of the local consensus estimates. However, [75] employs additional local “sum” and “weight” scalar variables that each obey a global conservation property. The algorithm in [75] obtains average-consensus under asynchronous communication digraphs, yet they do not allow time-delays, and furthermore they require the transmitting node knows how many nodes will receive their signal. Similarly, the algorithms proposed in [21, 81–84] converge relatively fast to average-consensus, yet they require well-behaved communication patterns and/or detailed knowledge at each node of the underlying network topology. In summary, the algorithms proposed in this chapter use local “normal” vectors to obtain average-consensus in the presence of asynchronous directed communication with asymmetrical time-varying time-delays. The drawback is that the local normal vectors require node communication and storage costs to scale linearly with network size. The draw-
back is nullified when an element-wise average-consensus is required on a set of vectors whose
dimension is on the same order of magnitude as the network size. Three practical examples of
this are discussed in Sec.4.4.4.

4.4 ACP2: Algorithmic Solutions

In this section we describe 4 consensus protocols that solve ACP2 under specific UCP conditions.
The first section Sec.4.4.1 defines the 4 consensus protocols, Sec.4.4.2 describes the respective
UCP conditions sufficient for their convergence, and Sec.4.4.3 formally states these results. We
then go on to discuss the implications of these results in relation to the literature on consensus
formation (Sec.4.4.4), present numerical examples (Sec.4.5), and provide directions of future
work (Sec.4.6).

4.4.1 ACP2: Consensus Protocols

In this section we define the 4 consensus protocols that will be analyzed in this chapter. We
first will discuss the common features among all 4 protocols, we then move on to specifically
define each respective protocol in terms of the general consensus protocol (4.1) – (4.2).

All 4 protocols have the common feature of assuming that the knowledge set of each node
contains a “normal” consensus estimate \( \hat{v}_i(t) \) \( \in \mathbb{R}^{n \times 1} \) representing the linear dependence the
consensus estimate \( \hat{s}_i(t) \) has on the matrix \( S = [s_1(0); s_2(0); \cdots ; s_n(0)] \in \mathbb{R}^{d \times n} \).

• (A8): \( K_i(t) \supseteq \{ \hat{v}_i(t) \} \), \( \forall i \in V \), \( \forall t \in \mathbb{N} \), where \( \hat{v}_i(t) \in \mathbb{R}^{n \times 1} \) satisfies \( \hat{s}_i(t) = S\hat{v}_i(t) \), and
we denote \( S = [s_1(0); s_2(0); \cdots ; s_n(0)] \in \mathbb{R}^{d \times n} \).

Combining (A4) and (A8) yields the minimum knowledge set for each of the 4 protocols,

• (A4'): \( \mathcal{K}_i(t) \supseteq \{ i, n, \hat{s}_i(t), \hat{v}_i(t), s_i(0) \} \), \( \forall i \in V \).

Assumption (A8) implies that \( \hat{s}_i(t) = \bar{s}(0) \) if \( \hat{v}_i(t) = \frac{1}{n} 1_n \), where \( 1_n \in \mathbb{R}^{n \times 1} \) denotes a vector
that consists only of unit-valued elements. Thus, in terms of the consensus problem ACP2,
assumption (A4') imply the average-consensus problem ACP2 is solved at time \( t \) if \( \hat{v}_i(t) = \frac{1}{n} 1_n \)
for all nodes \( i \in V \). Motivated by this fact, consider the following update scheme for each
normal consensus estimate \( \hat{v}_i(t) \). At any time \( t \), let \( V_i \) denote a matrix with each column
equal a unique member of the set \( \mathcal{H}_i = \{ v \in \mathcal{K}_i(t(+)) : s = Sv \ , \ s \in \mathcal{K}_i(t(+)) \} \), where
\( \mathcal{K}_i(t(+)) = \mathcal{K}_i(t) \cup S^{ij}(t_0, t) \). Note that (A4') implies \( \mathcal{H}_i \) has cardinality \( |\mathcal{H}_i| \geq 1 \). At any
time instant \( t \in \mathbb{N} \), the proposed protocols will update each normal consensus estimate \( \hat{v}_i(t) \)
4.4. ACP2: Algorithmic Solutions

according to the following optimization problem,

\[
\hat{v}_i(t + 1) = V_i \hat{a}, \\
\hat{a} = \arg \min_{a \in \mathbb{R}^{\lvert V_i \rvert}} \| V_i a - \frac{1}{n} 1_n \|_2^2,
\]

(4.4)

where \( \| \cdot \| \) denotes the Euclidean norm (the exception to this is the DDA protocol, which further requires each element of \( \hat{v}_i(t) \) to be either 0 or \((1/n))\). In order to maintain \((A8)\), the update (4.4) implies that the consensus estimate \( \hat{s}_i(t) \) must be updated as follows,

\[
\hat{s}_i(t + 1) = S_i \hat{a},
\]

(4.5)

where \( S_i \) is a matrix with each column equal to the vector \( s \in K_i(t(+)) \) that satisfies \( s = Sv \) with \( v \) equal to the corresponding column in \( V_i \). In other words, \( S_i = SV_i \). Note that if node \( i \) does not receive a signal at time \( t \), then \( K_i(t(+)) = K_i(t) \) and hence (4.4) – (4.5) imply \((A8)\) holds. In the remaining sections we show that by utilizing \((A4'), (4.4)\) and (4.5), a suitable consensus protocol \((4.1) – (4.2)\) can achieve average-consensus under relatively weak communication conditions when compared to past algorithms in the literature.

Protocol 1: Idealized Algorithm (IA)

The first protocol we define is the Idealized Algorithm (IA). The IA protocol presented in this section obtains average-consensus trivially. We propose it formally because the 3 other protocols are specific reductions of it in the sense that any data transmitted by the latter will also be transmitted by the IA algorithm, but not vice-versa. Furthermore, the IA protocol requires communication conditions that will be used in the non-trivial result Thm.4.4.9. The IA protocol implies that all of the initial consensus vectors \( \{s_i(0) : i \in V\} \) are flooded through-out the network and stored at each node. Thus IA can be seen as a trivial solution to the “multi-piece dissemination” problem [73]. The general methodology of the IA protocol has been discussed previously in the literature (e.g. [16, 62]) and it is often referred to as “flooding”. For completeness of our results, we show in Theorems 4.4.7 – 4.4.8 that regardless of the communication pattern between nodes, there exists no consensus protocol \((4.1) – (4.2)\) that can obtain average-consensus before the IA algorithm. This is why we have named it the “Idealized” algorithm.

Let \( \delta[\cdot] \) denote the Kronecker delta function applied element-wise. The IA signal specification (4.2) and knowledge set update (4.1) are respectively defined as (4.6) and (4.7), under which the updates (4.4) – (4.5) imply (4.8) – (4.10). Notice that in (4.7) – (4.9) below we will for notational convenience utilize a vector \( \mathbf{w} \in \mathbb{R}^{n \times 1} \) with \( \ell^{th} \) element denoted \( \mathbf{w}_\ell \).
Protocol 1: Idealized Algorithm (IA)

Signal Specification: \( S_{ij}(t_0, t_1) = \begin{cases} \mathcal{K}_j(t_0) \setminus \{j, n, \hat{s}_j(t_0)\} & \text{if } \hat{v}_j(t_0) \neq \frac{1}{n}1_n, (i, j) \in \mathcal{V}^2 \\ \mathcal{K}_j(t_0) & \text{if } \hat{v}_j(t_0) = \frac{1}{n}1_n \end{cases} \)

Knowledge Set Update:
\[
\mathcal{K}_i(t_1 + 1) = \begin{cases} \{i, n, \hat{v}_i(t_1 + 1), \hat{s}_i(t_1 + 1), s_\ell(0) : \forall \ell \text{ s.t. } \mathbf{w}_\ell = 1\} & \text{if } \mathbf{w} \neq 1_n \\ \{\hat{v}_i(t_1 + 1), \hat{s}_i(t_1 + 1)\} & \text{if } \mathbf{w} = 1_n \end{cases}
\]

Normal Consensus Estimate Update: \( \hat{v}_i(t_1 + 1) = \mathbf{w} \)

Consensus Estimate Update: \( \hat{s}_i(t_1 + 1) = \sum_{\ell=1}^n \mathbf{w}_\ell s_\ell(0) / \bar{s}(0) \)

Estimate Initialization: \( \hat{v}_i(0) = \frac{1}{n}e_i, \hat{s}_i(0) = \frac{1}{n}s_i(0) \).

Besides flooding the initial consensus vectors, the IA protocol (4.6)–(4.10) has an additional feature that is not necessary but is practical: once a node \( j \) obtains all initial vectors \( \{s_i(0) : i \in \mathcal{V}\} \), any signal thereafter transmitted from node \( j \) will contain only the consensus estimate \( \hat{s}_j(t) = \bar{s}(0) \) and normal consensus estimate \( \hat{v}_j(t) = \frac{1}{n}1_n \). In this way the average-consensus value \( \bar{s}(0) \) can be propagated throughout the network without requiring all \( n \) initial consensus vectors to be contained in every signal (this feature, however, requires that the network size \( n \) is known a priori). With or without this feature, IA implies communication and storage of a vector with a potentially very large dimension in the range \( O(nd) \) and \( \Omega(d) \).

Protocol 2: Distributed-Averaging (DA)

We now define the non-trivial Distributed-Averaging (DA) consensus protocol. To define the DA update procedures, let \( V^+ \) denote the pseudo-inverse of an arbitrary matrix \( V \), and \( e_i \) denote the \( i^{th} \) standard unit vector in \( \mathbb{R}^{n \times 1} \). The DA signal specification (4.2) and knowledge set update (4.1) are defined as (4.11) and (4.12), under which the updates (4.4)–(4.5) imply (4.13)–(4.15).
4.4. ACP2: Algorithmic Solutions

Protocol 2: Distributed Averaging (DA)

Signal Specification: \[ S^{ij}(t_0, t_1) = K_j(t_0) \setminus \{ j, n, s_j(0) \} \text{, } (i, j) \in V^2 \] (4.11)

Knowledge Set Update: \[ K_i(t_1 + 1) = \{ i, n, \hat{v}_i(t_1 + 1), \hat{s}_i(t_1 + 1), s_i(0) \} \] (4.12)

Normal Consensus Estimate Update: \[ \hat{v}_i(t_1 + 1) = V_{(DA)}^+ \frac{1}{n} 1_n, V_{(DA)} = [\hat{v}_i(t_1), \hat{v}_j(t_0), \frac{e_i}{n}] \] (4.13)

Consensus Estimate Update: \[ \hat{s}_i(t_1 + 1) = V_{s}^+ \frac{1}{n} 1_n, V_{s} = [\hat{s}_i(t_1), \hat{s}_j(t_0), \frac{1}{n}s_i(0)] \] (4.14)

Estimate Initialization: \[ \hat{v}_i(0) = \frac{1}{n} e_i, \hat{s}_i(0) = \frac{1}{n}s_i(0) . \] (4.15)

Notice that the DA knowledge set update (4.12) implies the initial consensus vector \( s_i(0) \) remains stored in the respective knowledge set of each node \( i \in V \). Without this property the DA protocol does not achieve average-consensus for the general class of communication patterns assumed in Thm. 4.4.9. Next observe that \( V_{(DA)} \) is a \( n \times 3 \) matrix. It is shown in [27] that the pseudo-inverse \( V_{(DA)}^+ \) in (4.13) – (4.14) can be computed using only a few arithmetic operations. Lastly, note that the DA signal specification (4.11) implies every signal must contain only the consensus estimate \( \hat{s}_j(t) \in \mathbb{R}^d_{\times 1} \) and normal consensus estimate \( \hat{v}_j(t) \in \mathbb{R}^n_{\times 1} \), thus the DA protocol requires communication and storage of a vector with dimensions in the range \( O(n + d) \) and \( \Omega(d) \). Most consensus protocols based only on distributed linear averaging (e.g. [43, 72, 75, 76]) have \( \Theta(d) \) communication and storage costs in our setting. Similarly, the class of “gossip” algorithms considered in [73] are required to have \( O(\text{poly}(\log(n))) + d \) storage costs, however [73] only considers bi-directional instantaneous communication patterns (such patterns comprise a very small subset of the communication sequences considered in this chapter).

Protocol 3: Discretized Distributed-Averaging (DDA)

Next we define the non-trivial Discretized Distributed-Averaging (DDA) consensus protocol. Let \( e_i \) denote the \( i^{th} \) standard unit vector in \( \mathbb{R}^{n\times 1} \), \( \hat{v}_i(t) \) denote the \( \ell^{th} \) element of \( \hat{v}_i(t) \), and \( \hat{v}_i^{-\ell}(t) \in \mathbb{R}^{(n-1)\times 1} \) denote the vector \( \hat{v}_i(t) \) with element \( v_{i\ell}(t) \) deleted. The DDA signal

\[ \hat{v}_i(t) = \frac{1}{n} e_i, \hat{s}_i(t) = \frac{1}{n}s_i(0) . \] 50

\[ \text{The lower bound can be achieved by a mapping to reduce the signal dimensionality. For example, the initialization (4.15) implies the } i^{th} \text{ element of } \hat{v}_i(0) \text{ equals } (1/n) \text{ and all other elements equal zero. The } n\text{-dimensional vector } \hat{v}_i(0) \text{ can thus be mapped to the } 2\text{-dimensional vector } [i, (1/n)]. \]
4.4. ACP2: Algorithmic Solutions

specification (4.2) and knowledge set update (4.1) are defined as (4.16) and (4.17), under which the updates (4.4) – (4.5) imply (4.18) – (4.21).

<table>
<thead>
<tr>
<th>Protocol 3: Discretized Distributed-Averaging (DDA)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Signal Specification:</strong> ( S^{ij}(t_0, t_1) = \mathcal{K}_j(t_0) \setminus { j, n, s_j(0) } ), ((i, j) \in \mathcal{V}^2) \hspace{1cm} (4.16)</td>
</tr>
<tr>
<td><strong>Knowledge Set Update:</strong> ( \mathcal{K}_i(t_1 + 1) = { i, n, \hat{v}_i(t_1 + 1), \hat{s}_i(t_1 + 1), s_i(0) } ) \hspace{1cm} (4.17)</td>
</tr>
<tr>
<td><strong>Normal Consensus Estimate Update:</strong> ( \hat{v}_i(t_1 + 1) = \hat{a} \hat{v}_i(t_1) + \hat{b} \hat{v}_j(t_0) + \hat{c} e_i ) \hspace{1cm} (4.18)</td>
</tr>
<tr>
<td>( (\hat{a}, \hat{b}, \hat{c}) = (1, 1, -\hat{v}_j(t_0)) ) if ( \hat{v}_i^{-i}(t_1) \hat{v}_j^{-i}(t_0) = 0 )</td>
</tr>
<tr>
<td>( (\hat{a}, \hat{b}, \hat{c}) = (0, 1, \frac{1}{n} - \hat{v}_j(t_0)) ) if ( \hat{v}_i^{-i}(t_1) \hat{v}_j^{-i}(t_0) &gt; 0 ), (</td>
</tr>
<tr>
<td>( (\hat{a}, \hat{b}, \hat{c}) = (1, 0, 0) ) if ( \hat{v}_i^{-i}(t_1) \hat{v}_j^{-i}(t_0) &gt; 0 ), (</td>
</tr>
<tr>
<td><strong>Consensus Estimate Update:</strong> ( \hat{s}_i(t_1 + 1) = \hat{a} \hat{s}_i(t_1) + \hat{b} \hat{s}_j(t_0) + \hat{c} s_i(0) ) \hspace{1cm} (4.20)</td>
</tr>
<tr>
<td><strong>Estimate Initialization:</strong> ( \hat{v}_i(0) = \frac{1}{n} e_i ), ( \hat{s}_i(0) = \frac{1}{n} s_i(0) ) \hspace{1cm} (4.21)</td>
</tr>
</tbody>
</table>

The only difference between DDA and the DA consensus protocols, is that DDA requires \( \hat{v}_i(t) \in \{0, \frac{1}{n}\}^n \) whereas DA requires \( \hat{v}_i(t) \in \mathbb{R}^{n \times 1} \), this is why we use the name “Discretized Distributed-Averaging”. It follows that, similar to the DA algorithm, the DDA requires communication and storage of a vector with dimension size in the range \( O(n + d) \) and \( \Omega(d) \). However, in Sec.4.4.4 we discuss how quantization issues imply that in practice the DDA may have significantly lower resource costs than DA. Note that most consensus protocols based only on distributed linear averaging (e.g. [43,62,72,75,76,85,86]) have \( \Theta(d) \) communication and storage costs in our setting.

**Protocol 4: One-Hop (OH)**

Under the OH consensus protocol each signal \( S^{ij}(t_0, t_1) \) will either contain the local initial consensus vector \( s_j(0) \) and transmitting node identity \( j \), or (if \( n \) is known a priori at each node) the average-consensus value \( \bar{s}(0) \) and a scalar 0 to indicate that the transmitted vector is the true average-consensus value. For this reason the OH protocol requires communication of a vector with dimension size \( \Theta(d) \), but similar to the DA and DDA protocols OH requires storage of a vector with dimension size in the range \( O(n + d) \) and \( \Omega(d) \). Let \( S_1^{ij}(t_0, t_1) \) denote the first element in the signal set \( S^{ij}(t_0, t_1) \), and \( \delta(\cdot) \) denote the Kronecker delta function applied
element-wise. The OH signal specification (4.1) and knowledge set update (4.2) are defined as (4.22) and (4.23), under which the updates (4.4) – (4.5) imply (4.24) – (4.26). Note that in (4.23) – (4.24) below we will for notational convenience utilize a vector $w \in \mathbb{R}^{n \times 1}$ with $\ell^{th}$ element denoted $w_{\ell}$.

**Protocol 4: One-Hop (OH)**

*Signal Specification:*

$$S^{ij}(t_0, t_1) = \begin{cases} \{j, s_j(0)\} & \text{if } \hat{v}_j(t_0) \neq \frac{1}{n}1_n \\ \{0, \hat{s}_j(t_0)\} & \text{if } \hat{v}_j(t_0) = \frac{1}{n}1_n \end{cases}, \quad (i, j) \in V^2$$  

(4.22)

*Knowledge Set Update:*

$$K_i(t_1 + 1) = \begin{cases} \{i, n, \hat{v}_i(t_1 + 1), \hat{s}_i(t_1 + 1), s_i(0)\} & \text{if } w \neq \frac{1}{n}1_n \\ \{\hat{v}_i(t_1 + 1), \hat{s}_i(t_1 + 1)\} & \text{if } w = \frac{1}{n}1_n \end{cases}$$  

(4.23)

*Normal Consensus Estimate Update:*

$$\hat{v}_i(t_1 + 1) = \begin{cases} \frac{1}{n}w & \text{if } 0 \neq S^{ij}_1(t_0, t_1) \\ \frac{1}{n}1_n & \text{if } 0 = S^{ij}_1(t_0, t_1) \end{cases}$$  

(4.24)

*Consensus Estimate Update:*

$$\hat{s}_i(t_1 + 1) = \begin{cases} \hat{s}_i(t_1) + \left(\frac{1}{n} - \hat{v}_ij(t_1)\right)s_j(0) & \text{if } 0 \neq S^{ij}_1(t_0, t_1) \\ \hat{s}_j(t_0) & \text{if } 0 = S^{ij}_1(t_0, t_1) \end{cases}$$  

(4.25)

*Estimate Initialization:*

$$\hat{v}_i(0) = \frac{1}{n}e_i, \quad \hat{s}_i(0) = \frac{1}{n}s_i(0).$$  

(4.26)

The condition $0 = S^{ij}_1(t_0, t_1)$ requires $n$ is a priori known at each node. If $n$ is initially unknown at each node, then the condition $0 = S^{ij}_1(t_0, t_1)$ must be removed from the above OH protocol, but otherwise the protocol remains the same.

**Summary:** In this section we have defined 4 consensus protocols that in Sec.4.4.3 will be shown to solve ACP2. The algorithms can be viewed as special cases of the consensus protocol (4.1) – (4.2). For initial consensus vectors with dimension $d \geq n$, the communication costs of the DA and DDA protocols become $O(d)$, thereby making these protocols comparable to many of the other average-consensus algorithms in the literature that have $\Theta(d)$ costs. In Sec.4.4.4 we provide three practical examples from the literature to justify the assumption $d \geq n$.
4.4.2 ACP2: UCP Conditions

In this section we define 5 conditions on the UCP (cf. (A5)-(A7)) that will be shown necessary and/or sufficient for convergence of the 4 consensus protocols defined in Sec.4.4.1.

**Singly V Strongly Connected UCP**

A UCP communication sequence \( C_{t_0,t_1} \) is “Singly V Strongly Connected” (SVSC) if each node has “time-respecting communication path” to every other node in the network. We define a “time-respecting communication path” as follows.

**Definition 4.4.1** A communication sequence \( C_{t_0,t_1} \) contains a “time-respecting communication path” (trcp) from node \( j \) to node \( i \) if \( C_{t_0,t_1} \) contains a sub-sequence \( C_{t_0,t_1}^{(ij)} \) with the following connectivity property,

\[
C_{t_0,t_1}^{(ij)} = \{ S_{t_{\ell_1}}, S_{t_{\ell_1}+1}, S_{t_{\ell_2}}, \ldots, S_{t_{k(ij)}}, S_{t_{k(ij)}+1} \} \tag{4.27}
\]

where we have omitted the time indices but it is understood that the transmission time \( t_{\ell_1} \) of each signal \( S_{t_{\ell_1}} \) occurs after the reception time \( t_{\ell_1} \) of the preceding signal \( S_{t_{\ell_1}+1} \).

Note that the sub-sequence \( C_{t_0,t_1}^{(ij)} \) defined in (4.27) has a finite cardinality \( |C_{t_0,t_1}^{(ij)}| = k(ij) + 1 \geq 1 \).

With Def.4.4.1 we can now define a “Singly V Strongly Connected” (SVSC) communication sequence.

**Definition 4.4.2 (SVSC)** A communication sequence \( C_{t_0,t_1} \) is “singly V strongly connected” (SVSC) if \( C_{t_0,t_1} \) contains trcp from each node \( i \in V \) to every node \( j \in V_{-i} \).

For any finite \( t_0 \in \mathbb{N} \), we will let \( \ldots, C_{t_0,t_1}^{(ij)} \in \text{SVSC} \) denote that the sequence \( C_{t_0,t_1} \) is SVSC.

**Infinitely V Strongly Connected UCP**

A UCP communication sequence \( C_{0,\infty} \) is “Infinitely V Strongly Connected” (IVSC) if it can be partitioned into an infinite number of disjoint SVSC communication sequences.

**Definition 4.4.3 (IVSC)** A communication sequence \( C_{0,\infty} \) is “infinitely V strongly connected” (IVSC) if for each finite time instant \( t \in \mathbb{N} \) there exists a finite time \( T(t) \in \mathbb{N} \) such that \( C_{t,T(t)} \in \text{SVSC} \).

We will let \( C_{0,\infty} \in \text{IVSC} \) denote that the sequence \( C_{0,\infty} \) is IVSC.
4.4. ACP2: Algorithmic Solutions

Singly V Centrally Connected UCP

A UCP communication sequence \( C_{t^0, t^1} \) is “Singly V Centrally Connected” (SVCC) if all nodes have a communication path to some node \( \hat{i} \), and then the node \( \hat{i} \) transmits a signal to all other nodes. In this sense, the network is “centered” around at least one node \( \hat{i} \).

**Definition 4.4.4 (SVCC)** A communication sequence \( C_{t^0, t^1} \) is “singly V centrally connected” (SVCC) if there exists a time \( t^1/2 \) and node \( \hat{i} \in V \), such that

\[
C_{t^0, t^1} \subset C_{t^0, t^1/2-1}, \quad \forall j \in V_{-\hat{i}}.
\]

We let “\( C_{t^0, t^1} \in SVCC \)” denote that a sequence \( C_{t^0, t^1} \) is SVCC.

Infinitely V Centrally Connected UCP

A UCP communication sequence \( C_{0, \infty} \) is “Infinitely V Centrally Connected” (IVCC) if it can be partitioned into an infinite number of disjoint SVCC communication sequences.

**Definition 4.4.5 (IVCC)** A communication sequence \( C_{0, \infty} \) is “infinitely V centrally connected” (IVCC) if for any finite time instant \( t \in \mathbb{N} \) there exists a finite time \( T(t) \in \mathbb{N} \) such that \( C_{t, T(t)} \in SVCC \).

For an IVCC sequence the “central” node \( \hat{i} \) defined in Def.4.4.4 can vary between each SVCC sub-sequence, and furthermore we do not assume that any node knows the identity of the central node \( \hat{i} \). Let “\( C_{t^0, t^1} \in IVCC \)” denote that a sequence \( C_{t^0, t^1} \) is IVCC.

Dual of a Singly V Centrally Connected UCP

Next we define a UCP condition that will be shown sufficient for the OH protocol. To do so we can define the “dual” \( \tilde{C}_{t^0, t^1} \) of a communication sequence \( C_{t^0, t^1} \) as follows,

\[
S_{ij}(t_0, t_1) \in \tilde{C}_{t^0, t^1} \quad \text{if} \quad C_{i, t_1} \in C_{t^0, t_1}.
\]

We use the term “dual” because (4.28) implies that if \( C_{t^0, t^1} \) contains a signal (resp. communication path) from node \( j \) to \( i \), then so does \( \tilde{C}_{t^0, t^1} \).

A communication sequence \( C_{t^0, t^1} \) is “Singly V dual-Centrally Connected” (SV\( \tilde{C} \)C) if all nodes transmit a signal to some node \( \hat{i} \), and then the node \( \hat{i} \) has a communication path to

54
all other nodes. In other words, $C_{t^0,t^1}$ is “Singly V dual-Centrally Connected” (SVVC) if $	ilde{C}_{t^0,t^1} \in$ SVCC.

**Definition 4.4.6 (SVVC)** A communication sequence $C_{t^0,t^1}$ is “singly V dual-centrally connected” (SVVC) if $\tilde{C}_{t^0,t^1} \in$ SVCC.

Let “$C_{t^0,t^1} \in$ SVVC” denote that a sequence $C_{t^0,t^1}$ is SVVC.

### 4.4.3 ACP2: Convergence Results

Given the 5 UCP conditions defined in Sec.4.4.2 we can now present the necessary and sufficient communication conditions for the convergence of the 4 consensus protocols defined in Sec.4.4.1. These are stated in the following 5 theorems.

The theorem below states the necessary communication conditions for convergence of any consensus protocol (4.1) – (4.2). The theorem thus applies to each of the IA, DA, OH, and DDA protocols.

**Theorem 4.4.7 (Consensus Protocol (P) Necessary Communication Conditions)** A consensus protocol (cf. (4.1) – (4.2)) cannot solve ACP2 for any communication sequence $C_{0,t} \notin$ SVS (cf. Def.4.4.2).

The next theorem states the sufficient UCP conditions for convergence of the IA protocol. Note that for the IA protocol the SVSC condition is both necessary and sufficient.

**Theorem 4.4.8 (IA Sufficient UCP Conditions)** The IA protocol (4.6) – (4.10) solves ACP2 at time $(t^1 + 1)$ for any UCP $C_{0,t} \in$ SVSC (cf. Def.4.4.2).

Although the above two theorems (Thm.4.4.7, 4.4.8) can be proven trivially (see [87]), they will be referred to later and are presented for completeness of our results. The following theorem is non-trivial and states the sufficient UCP conditions for convergence of the DA protocol.

**Theorem 4.4.9 (DA Sufficient UCP Conditions)** The DA protocol (4.11) – (4.15) solves ACP2 as $t \to \infty$ for any UCP $C_{0,\infty} \in$ IVSC (cf. Def.4.4.3).

The next theorem is also non-trivial and states the sufficient UCP conditions for convergence of the DDA protocol.

**Theorem 4.4.10 (DDA Sufficient UCP Conditions)** The DDA protocol (4.16) – (4.21) solves ACP2 as $t \to \infty$ for any UCP $C_{0,t} \in$ IVCC (cf. Def.4.4.5).
4.4. ACP2: Algorithmic Solutions

The final theorem can be proven trivially (see [87]) and states the necessary and sufficient UCP conditions for convergence of the OH protocol.

**Theorem 4.4.11 (OH Necessary and Sufficient UCP Conditions)** The OH protocol (4.22) – (4.26) solves ACP2 at time \((t^1 + 1)\) for any UCP \(C_{0,t^1} \in \text{SV}^C\) (cf. Def.4.4.6). □

Note that Thm.4.4.11 together with Thm.4.4.7 imply that any SVSC communication sequence is also SVČC.

**Summary:** In this section we have stated necessary and sufficient UCP conditions for the convergence of the IA, DA, OH, and DDA protocols. The UCP conditions were defined in Sec.4.4.2 and termed “Singly \(\mathcal{V}\) Strongly Connected” (SVSC), “Infinitely \(\mathcal{V}\) Strongly Connected” (IVSC), “Singly \(\mathcal{V}\) Centrally Connected” (SVCC), “Infinitely \(\mathcal{V}\) Centrally Connected” (IVCC), and “Singly \(\mathcal{V}\) dual-Centrally Connected” (SVČC). In the subsequent section we relate our results to those in the literature, expand on the conclusions made in this section, and also describe an analogy in terms of the sufficient UCP conditions for convergence of the IA and DA, and OH and DDA protocols.

4.4.4 ACP2: Discussion of Results

The IVSC condition (4.4.3) assumes a recurring connectivity among nodes. It does not require any node to have a priori knowledge of the communication sequence, nor does it require any specific structure in the (directed) signal pattern and time-delays. The DA protocol solves ACP2 for any UCP satisfying the IVSC condition. The drawback to the DA protocol is that it requires \(O(n + d)\) communication and storage costs, whereas most past algorithms have only \(\Theta(d)\) costs (see e.g. [1, 32] for exceptions to this).

There has been much research considering algorithms (with only \(\Theta(d)\) costs) that can achieve consensus for arbitrary IVSC communication sequences (e.g. [43, 78, 79]). Typically, these “consensus algorithms” require only that there is a recurring “root” node (see [79] for definitions), whereas Theorem 4.4.9 requires that all nodes are a recurring “root” node. However, a major drawback of the cited algorithms is that they only guarantee a final consensus on some unknown linear combination of the initial consensus vectors, and such a consensus is not useful in many practical applications [16, 36, 62, 80].

Theorem 4.4.9 implies the DA algorithm will obtain average-consensus asymptotically under more general communication conditions than previous works such as [16, 18, 62, 74, 76, 77]. However, we clarify (i) there exist communication sequences for which the DA algorithm obtains average-consensus in finite time (see [27]), and (ii) Theorem 4.4.9 does not imply that the DA protocol always achieves average-consensus before all other known consensus protocols. In Sec.4.5, we apply the DA protocol to an example considered in [1]. Although the consensus
4.4. ACP2: Algorithmic Solutions

protocol of [1] obtains an exact average-consensus after a finite number of received signals, it is shown that for this example the DA protocol obtains a satisfactory approximation to the average-consensus at even an earlier time.

Analogy Between the IA and DA, and OH and DDA Protocols

The necessary and sufficient condition for average-consensus under the OH protocol requires that each node $j \in V$ will either (a) receive a signal from all other nodes, or (b) have a communication path from some node $i_j \in V_{-j}$ that has already received a signal from all other nodes $q \in V_{-i_j}$. Consider the latter possibility where $i_j = \hat{i}$ for all $j \in V_{-i}$, that is the $S\hat{V}\hat{C}C$ UCP condition holds. The $S\hat{V}\hat{C}C$ UCP condition is the weakest condition for average-consensus under the OH protocol in the sense that only one node must receive a signal from all other nodes (one is the minimum number of such nodes for convergence under the OH algorithm).

Utilizing $S\hat{V}\hat{C}C$ (cf. Def.4.4.6), we now describe how the (IA, DA) protocols are analogous to (OH, DDA) with respect to (i) the sufficient communication conditions, and (ii) the data required to be communicated.

In terms of communicated data, note that (i) the IA protocol requires communication of all currently known initial consensus vectors, whereas the OH protocol requires communication of only the initially known consensus vector, and (ii) the DDA protocol is a discretized version of the DA protocol (as discussed after (4.21)). Next observe that Theorems 4.4.7 – 4.4.8 imply the weakest sufficient condition for average-consensus under the IA protocol is the $S\hat{V}\hat{C}C$ UCP condition (cf. Def.4.4.2). The dual of any $S\hat{V}\hat{C}C$ sequence implies that every node $j \in V$ receives a signal from every other node $i \in V_{-j}$. It follows that if the dual of the weakest sufficient condition under the IA protocol occurs on an infinite number of contiguous time-intervals, then both the DA and DDA protocols obtain average-consensus. Likewise, if the dual of the $S\hat{V}\hat{C}C$ UCP condition occurs on an infinite number of contiguous time-intervals, both the DA and DDA protocols obtain average-consensus, since the dual of $S\hat{V}\hat{C}C$ is precisely the $S\hat{C}C$ condition.

In summary, if the dual of the minimum sufficient condition for convergence of the IA protocol occurs on an infinite number of disjoint time intervals then the OH, DA and DDA protocols converge. Likewise if the dual of the minimum sufficient condition for convergence of the OH protocol occurs on an infinite number of disjoint time intervals then the IA, DA and DDA protocols converge. We also find that the OH is minimized version of IA in terms of data communicated, and likewise is the DDA protocol a minimized version of the DA protocol.
4.4. ACP2: Algorithmic Solutions

Practical Examples for the DA and DDA Algorithms

The DA and DDA protocols require communication and storage of a vector with dimension of order $O(n + d)$, whereas the IA requires $O(nd)$ costs. Most average-consensus algorithms in the literature require only $\Theta(d)$ costs (e.g. [52, 71, 75, 79, 85, 86] and see [1, 32] for algorithms that potentially require larger costs). Our main results in this chapter imply the DA and DDA protocols can obtain average-consensus under general asynchronous and time-varying asymmetric time-delayed communication, which is a unique (modulo “flooding”) and desirable feature within the average-consensus literature. The issue of greater communication and storage costs is effectively nullified if $d \geq n$, in which case the resource costs of the DA and DDA belong to the same order of magnitude as that of most consensus algorithms and thereby motivate use of DA and DDA.

There are several broad examples in the literature where an average-consensus on $d \geq n$ elements may be needed. As our first example, suppose each node observes a $\sqrt{n}$ dimensional process in noise, then a distributed Kalman filter requires an average-consensus on $d = n + \sqrt{n}$ distinct scalars [8]. Similarly, if each node observes a $\sqrt{n}$ dimensional parameter in noise, then either a distributed maximum-likelihood estimate or best linear unbiased estimate requires average-consensus on $d = n + \sqrt{n}$ distinct scalars [16, 36]. Lastly, consider a sensor network where each node makes an observation on some $q$ dimensional parameter $z \in \mathbb{R}^{q \times 1}$, where for each node $i$ the observation $z$ is an independent realization of a random variable that obeys one of $p$ different known probability distributions. A distributed hypothesis test (DHT) on the most likely hypothesis (the maximum a posterior estimate) then requires an average-consensus on $pq$ scalar quantities [88]. It follows that whenever $p \geq n / q$ a DHT requires average-consensus on $d \geq n$ scalar values. For example, suppose $n$ nodes are dispersed in a region of space and seek to locate a fixed energy source ($q = 1$). A DHT with $p = n$ (each node’s coverage area) could then obtain the maximum likely source location by obtaining average-consensus on $d = n$ scalars. The above examples do not benefit the IA algorithm, since $O(nd) \gg O(d)$ for large $n$ and $d \geq n$.

Quantization of the Normal Consensus Estimate

For simplicity we have assumed that the communication and storage of a vector with dimension $q$ implies resource costs of order $O(q)$. The rationale for this is due to signal quantization. If each communicated scalar is quantized to the nearest $p$ digits, then for any fixed $p$ the number of digits in a signal with dimension $q$ is $O(q)$.

Signal quantization issues cannot be ignored in practice, and there has been growing interest in the effect of quantization has on the ability to obtain an accurate average-consensus [71, 72].
For the IA, OH, and DDA protocols each normal consensus estimate $\hat{v}_i(t)$ belongs to the discrete set $\{0, \frac{1}{n}\}^n$. For this reason the latter three protocols allow each element in $\hat{v}_i(t)$ to be expressed by the binary values $\{0, 1\}$, given that the integer value of $n$ is known. The DA protocol does not possess this feature. The update coefficients for the consensus estimate $\hat{s}_i(t)$ are identical to those of the normal consensus vector $\hat{v}_i(t)$, and thus an extensive quantization of the latter will imply a larger margin of error with respect to the optimal update coefficients for the former. It follows that in practice, signal quantization will in general affect the IA, OH, and DDA protocols far less than the DA protocol.

### 4.4.5 ACP2: Example SVĆC Sequence

In this section we provide an example communication sequence under which the DDA protocol does not obtain average-consensus. The communication sequence that will be defined contains an infinite number of disjoint SVĆC sub-sequences (cf. Def. 4.4.6). The purpose of this example is to illustrate that the SVĆC condition implies a fusion center (i.e. there exists a node to which all nodes send a signal directly) whereas the sufficient condition for convergence of the DDA protocol requires there exist a node from which all nodes receive a signal. The latter can be accomplished via a single broad-casting node, whereas the former will in general require all nodes (except one) to broad-cast their initial data (see Fig.4.1).

![Figure 4.1: Illustrations of the SVĆC and SVČČ communication conditions for $n = 5$ nodes.](image)

Consider the communication sequence for $n = 4$ nodes (we choose $n = 4$ for simplicity; the generalization to an arbitrary number of nodes can be obtained in an analogous manner by the following example).
4.5 ACP2: Numerical Examples

\[ C_{0,1} = \{S^{14}(0, 0), S^{24}(0, 0), S^{34}(0, 0), S^{41}(0, 0), S^{12}(0, 1), S^{23}(0, 1), S^{42}(0, 1), S^{31}(0, 1)\} \]
\[ C_{t,t} = \{S^{14}(t, t), S^{21}(t, t), S^{32}(t, t), S^{41}(t, t), S^{42}(t, t), S^{43}(t, t)\} \quad \forall \ t \geq 2 \quad (4.29) \]

Let \( \hat{V}(t) = [\hat{v}_1(t), \hat{v}_2(t), \hat{v}_3(t), \hat{v}_4(t)] \in \mathbb{R}^{4 \times 4} \). At time \( t = 2 \) the communication sequence (4.29) implies the following normal consensus node state values under both the OH and DDA protocol.

\[
\hat{V}(2) = \begin{bmatrix}
\frac{1}{n} & 0 & \frac{1}{n} & \frac{1}{n} \\
\frac{1}{n} & \frac{1}{n} & 0 & \frac{1}{n} \\
0 & \frac{1}{n} & \frac{1}{n} & 0 \\
\frac{1}{n} & \frac{1}{n} & \frac{1}{n} & \frac{1}{n}
\end{bmatrix} \quad (4.30)
\]

The communication sequence (4.29) implies that under the DDA protocol the normal consensus state values in (4.30) do not change for all \( t \geq 2 \), and thus the DDA protocol does not converge. On the other hand, the OH protocol will obtain average-consensus at time \( t = 4 \) under (4.29). Together with the fact the \( SV\tilde{C}C \) condition is necessary for convergence under the OH protocol, the example (4.29) illustrates why we need to define both the \( SV\tilde{C}C \) and \( SVCC \) communication sequences to characterize the convergence conditions of the OH and DDA protocols.

### 4.5 ACP2: Numerical Examples

This section presents numerical examples of the 4 protocols defined in Sec.4.4.1 (IA, DA, DDA, and OH). The first example is taken from Example 3 in [1], where 6 nodes possess synchronous bi-directional communication with no time-delays. Within this communication model, the protocol in [1] can obtain average-consensus in finite time under various assumptions: (i) given \textit{a priori} knowledge of the network topology [1] obtains average-consensus after 3 iterations (which is the same time as IA, and thus from Theorems 4.4.7 and 4.4.8 this is the earliest time any consensus protocol can obtain average-consensus), (ii) distributed computation of the network topology using \( n \)-dimensional vectors, which implies [1] obtains average-consensus after 8 iterations (5 to obtain the network topology at each node, and 3 to obtain the consensus) \(^3\), and (iii) distributed computation of the network topology using 1-dimensional vectors, which

\(^3\)In [1] this is the worst case scenario. If additional storage costs are permitted, then the algorithm in [1] can achieve consensus in as few as 5 iterations. However, the DA algorithm will obtain consensus after 4 iterations if multiple signals can be processed simultaneously, which we did not assume for simplicity of the convergence proof.
implies \([1]\) obtains average-consensus after 33 iterations (30 to obtain the network topology at each node, and 3 to obtain the consensus). The DA and DDA protocols are comparable to (ii) above, since DA and DDA also require communication of an \(n\)-dimensional vector. Fig.4.2 illustrates the convergence of (IA, DA, DDA, OH) along with the “Max-Degree algorithm” \([16]\), and an algorithm “Random” that assigns random weights (that sum to one) to each update. Vertical lines indicate the 3,8, and 33 time points corresponding to the assumptions (i)-(iii) mentioned above. Note that “Normal Consensus Error” at time \(t\) is defined as

\[
\lim_{t \to t_1} \sum_{i=1}^{n} ||\hat{v}_i(t) - \frac{1}{n}1_n||^2 = 0.
\]

Figure 4.2: Convergence Under Example 3 in [1]. Random Weights and OH do not achieve average-consensus, but all other protocols do. The DDA protocol obtains consensus after only 4 iterations, and at time \(t = 8\) the DA normal consensus error has been reduced by 94%, thus implying a close approximation to the true average-consensus.
Next, we illustrate the convergence of the (IA, DA, DDA, OH) protocols as compared to (i) a “Gossip” algorithm that averages each received estimate with the current local estimate, and (ii) a “Random” algorithm wherein each node assigns a random weight $w$ to each received estimate, and adds this to $(1 - w)$ of its current local estimate. Consider $n = 100$ nodes that communicate asynchronously with time-varying delays. Specifically, a digraph on 100 nodes was constructed by adding randomly chosen edges until the graph was strongly connected. At each iteration a random (directed) link is selected and used to communicate a signal transmitted at a random time point in the past (by random we mean uniformly random).

Figure 4.3: Convergence under Asynchronous Communication with Time-Varying Asymmetrically Time-Delays. Gossip, Random, and OH do not achieve consensus. The latter implies that no node has received a signal from all other nodes.

In Fig.4.3, it is apparent that node estimates under the “Random” and “Gossip” protocols hover around the same error and hence do not obtain average-consensus. Also in Fig.4.3 it is clear that the OH protocol does not converge, which by Thm.4.4.11 implies that no node
4.5. ACP2: Numerical Examples

![Graph showing convergence under asynchronous communication with time-varying asymmetric time-delays. The (IA, DA, DDA) protocols all obtain consensus.](image)

Figure 4.4: Convergence under Asynchronous Communication with Time-Varying Asymmetric Time-Delays. The (IA, DA, DDA) protocols all obtain consensus.
4.5. ACP2: Numerical Examples

Figure 4.5: Convergence under Asynchronous Communication with Time-Varying Asymmetrical Time-Delays and unknown network size. The initial (transient) normal consensus error is larger when $n$ is unknown, however the steady-state of the (IA,DA,DDA) protocols are not affected (the vertical line indicates the IA time of consensus).
has received a signal from all other nodes. In Fig.4.4 we plot the (IA,DA,DDA) protocols under the same communication sequence as that of Fig.4.3. In contrast to (Random, Gossip, OH), the (IA,DA,DDA) protocols all obtain average-consensus. Lastly, in Fig.4.5 we simulate the same communication sequence for the (IA,DA,DDA) protocols but now assume that $n$ is unknown initially at each node. By plotting the same normal consensus error as was shown in Fig.4.4, it is evident that if $n$ is unknown initially at each node, the at the time that IA (with $n$ known) obtains average-consensus, the normal consensus estimates of the (IA,DA,DDA) protocols converge exactly to the respective estimates of each protocol when $n$ is known (see [27] for a proof of this result). From Thm.4.4.7 it then follows that the (IA,DA,DDA) protocols will obtain average-consensus at the same time as they would when assuming $n$ is known. The OH protocol, on the other hand, does not transmit the normal consensus estimates, hence the normal consensus error under the OH protocol is largely affected by a priori knowledge of $n$ (the OH error is too large to plot in Fig.4.5, with a final normal consensus error reaching 850% of what it is when $n$ is known). We note that if $n$ is unknown, then to obtain average-consensus the OH algorithm requires a communication sequence that conforms only to a fully connected communication graph [86].

### 4.6 ACP2: Conclusions and Future Work

This chapter has described and analyzed 4 consensus protocols designed to solve the average-consensus problem under general unidirectional connectivity conditions with asymmetrical and time-varying time-delays. We derived necessary and sufficient conditions for the convergence to average-consensus under each respective protocol. The conditions for convergence were based on 5 types of UCP connectivity conditions, namely the (SVSC), (IVSC), (SVCC), (IVSC), and (SVCC) communication sequences. All 5 connectivity conditions allow arbitrary (finite) delays in the transmission time of each directed signal; the communication digraphs need not repeat any fixed pattern; and furthermore we did not assume that the sending node knows the identities (or cardinality) of the receiving nodes. A drawback of the non-trivial protocols (DA) and (DDA) is that they require communication and storage of a vector with dimension of order $O(n + d)$, whereas most consensus protocols based only on weighted linear updates require $\Theta(d)$ costs in the current setting. Numerical examples of these protocols were presented and compared with previous algorithms in the consensus literature, and practical examples of the non-trivial protocols were explained. A unique feature of each protocol is the requirement to store the initial consensus vector $s_i(0)$ in each node’s respective knowledge set $K_i(t)$. An interesting avenue of future work includes obtaining UCP conditions for which the same protocols achieve average-consensus when the initial consensus vector is not stored in each node’s respective
knowledge set. Further work also still need be done regarding the protocols application to dynamic consensus formation, when the initial consensus vectors change with time.

As a final note, all 4 protocols (IA, DA, OH, DDA) can obtain a consensus on any linear combination of the initial consensus vectors \( \{ s_i(0) \in \mathbb{R}^d : i \in V \} \) under the exact same communication conditions as stated in Thm.4.4.9–4.4.11. In other words, if we assume a vector \( w \in \mathbb{R}^{n \times 1} \) is initially known at each node, then if the (IA,DA,DDA,OH) protocols update the normal consensus estimate based on (4.4) with \( \frac{1}{n} 1_n \) replaced by the vector \( w \), then the same conditions stated in Sec.4.4.3 will imply \( \lim_{t \to t_1} \| \hat{s}_i(t) - \sum_{i \in V} s_i(0)w_i \| = 0 \) for all \( i \in V \). The proof of these results follow by identical arguments to those presented in [27] and [89], simply by replacing \( \frac{1}{n} 1_n \) by the vector \( w \).
Chapter 5

Conclusion

5.1 Summary of Contributions

This thesis has addressed two diametrically opposed aspects of the average-consensus problem. The first of these concerns average-consensus in a two-time-scale Markov system when inter-node communication is synchronous, instantaneous, and adheres to a fixed network graph. In this setting we showed that by using a local stochastic approximation algorithm together with the appropriate update weights and network communication graph, each node consensus estimate will weakly-converge to the average stationary distribution of all observed Markov chains. We also characterized the convergence rate of this algorithm in terms of a suitably scaled switched diffusion equation; showed how an empirical measure allows estimation of the cumulative distribution function of each stationary distribution; and also derived a set-valued switching ODE in the case when each node observes a set of Markov chains rather than a single chain. In addition to all of this, if the regime switching is slow enough we have shown each node consensus estimate is capable of tracking the slowly-switching average of the input distributions. This two-time-scale model constitutes a set of slowly-switching input parameters observed in Markovian noise.

The second aspect concerning the average-consensus problem that was addressed in this thesis was that of reaching an average-consensus on the initial local input variables when the inter-node communication is asynchronous, time-delayed, and adheres to no fixed network graph. In this setting we showed that by employing normal consensus estimates of order $O(n)$, average-consensus is still possible even when the inter-node communication is completely a priori unknown. For initial consensus inputs of $O(n)$, these results imply an order of magnitude reduction in the required transmitted data as compared to an ideal consensus protocol that simply floods the initial consensus inputs. Three practical examples of such instances in the consensus literature were discussed, thus motivating the practical implementation of our consensus protocols in this setting. These three examples in particular were hypothesis testing in Bayesian networks, Kalman filtering, and maximum likelihood estimation. Besides flooding, the only other algorithm known to achieve average-consensus under such general communication conditions is a randomized algorithm that generates exponential random variables.
of order $O(rd)$, where larger values of $r$ imply a more accurate approximation of the $O(d)$ average-consensus value. Further work is required to investigate how our $O(n + d)$ distributed averaging algorithm compares in terms of communication complexity and accuracy with the randomized algorithm.

In the appendix we proved that given an upper bound on the time until a communication sequence is non-time-respecting strongly-connected, there exists an $O(n!)$ upper bound on the completion time of the ideal protocol. We further conjectured in such a case that there is an $O(n)$ upper bound. These results allow for the consensus protocols described in Chapter 4 to weakly-converge to the average-consensus under the input dynamics of Chapter 3, thus combining the most difficult features of both average-consensus problems. In particular, when any node’s input changes in the two-time-scale Markov system, a general method to achieve average-consensus for \textit{a priori} unknown communication sequences would then consist of that node sending out a signal with a time counter (initialized at $(n - 1)$) along with either (i) a message to reset each node’s consensus estimates or (ii) containing the new input value at that node. The upper bound on the completion time of the ideal protocol also allows for a complete stopping of any distributed averaging algorithm once a sufficiently close approximation to the true average-consensus value has been reached. We note, however, that the only known consensus protocols that allow for such an approximation to be computed are the IA, DA, OH and DDA protocols, and that is by their common use of the normal consensus estimates.

## 5.2 Directions of Future Work

There exists a variety of directions for future work based on the results discussed in this thesis. We list these now,

1. worst case convergence rate of the DA (resp. DDA) consensus protocol for uniformly strongly (resp. centrally) connected communication sequences.

2. extension of the IA, DA, OH, and DDA consensus protocols to noisy and/or changing local input variables (i.e. the two-time-scale Markov system).

3. application of the DA consensus protocol to hypothesis testing in Bayesian networks.

4. extension of the DA and DDA consensus protocols to the distributed convex optimization problem.

We next will discuss each of these directions individually.

Note that in Chapter 4 we provided results on the convergence of the DA and DDA consensus protocols. Furthermore, in [27] we provide worst case error bounds on the consensus
estimates \( \hat{s}_i(t) \) given \textit{a priori} knowledge of upper and lower bounds on the element-wise entries of the local consensus inputs \( s_i(0) \). However, we do not know of any worst case or asymptotic convergence rate for either protocol under general strongly-connected or centrally-connected communication sequences. Theoretically such bounds could be computed by assuming either that (i) the communication sequence is \textit{uniformly} strongly (resp. centrally) connected or (ii) deriving a bound based on counting the number of previous disjoint time intervals over which the communication sequence was singly-strongly (resp. centrally) connected.

In Chapter 4 it was assumed that the network sought after an average-consensus on the \textit{initial} local input variables (i.e. a \textit{static} average-consensus). In contrast, Chapter 3 assumed the local input variables at each node were random variables (Markov chains) with stochastically switching stationary distributions. The latter input dynamics constitute a set of \textit{changing} inputs (because of stationary distributions are stochastically switching) that are also observed in noise (because the stationary distributions do not in general describe a deterministic value). It is possible that by modifying the IA, DA, OH, and/or DDA protocols the static average-consensus problem addressed in Chapter 4 can be extended to the dynamic average-consensus addressed in Chapter 3. This modification would then incorporate the most difficult features of both average-consensus problems, namely a consensus on the \textit{current} average stationary distribution using an \textit{a priori} unknown, asynchronous, and time-delayed communication sequence.

It was shown in [88] how linear distributed averaging can be used to conduct hypothesis tests in a Bayesian network when node communication is synchronous, fixed, and without time-delays. The DA consensus protocol generalizes these results to allow hypothesis testing for \textit{a priori} unknown communication sequences (asynchronous, with time-delays, and no fixed network topology). Furthermore, unlike the work in [88], by using the normal consensus estimates the DA protocol permits an absolute computation of the most likely hypothesis, that is to say each node will \textit{know} when it has computed the most likely hypothesis. Finally, as mentioned above, the distributed averaging can also come to a complete stop once the most likely hypothesis has been computed by any given node, simply if that node sends out a flag signal indicating the most likely hypothesis and a time counter initialized at \((n - 1)\).

Lastly, a more complicated distributed computation that uses the general problem of average-consensus as a primary is that of distributive convex optimization [35]. Specifically, if each node \( i \in \mathcal{V} \) has an associated convex function \( f_i(z_1, \ldots, z_d) \) and the network desires to obtain at each node the vector \( z \) that minimize \( \sum_{i \in \mathcal{V}} f_i(z_1, \ldots, z_d) \), then a sub-gradient push-sum protocol can be shown to achieve this result for time-varying directed communication graphs if each node knows its out-degree [35]. The DA and DDA protocols provide a possible extension of this work insofar as that these consensus protocols might be generalized to solve the aforementioned distributed convex optimization problem for \textit{a priori} unknown, asynchronous, and time-delayed
5.2. Directions of Future Work

Communication sequences.

All of these avenues constitute possible directions of future work based on the results described in this thesis.
REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


Appendix A

Proofs of Convergence Results

In this appendix we provide proof of the convergence results stated in this thesis. In Sec. A.1 we detail the proofs of Thm.3.4.1-Thm.3.4.6 as well as the Lem.3.4.8-Lem.3.4.10 that are contained in Chapter 3. In Sec. A.2 we prove Thm.4.4.9-4.4.10 that were stated in Chapter 4. The proofs of Thm.4.4.7-4.4.8 and Thm.4.4.11 are relatively trivial and can be found in [87]. Lastly, in Sec. A.3 we discuss 3 data dissemination results that can be utilized to combine the protocols in Chapter 4 with the input dynamics assumed in Chapter 3.

A.1 Proof of Results Contained in Chapter 3.

This section details the proofs of Thm.3.4.1-Thm.3.4.6 and Lem.3.4.8-Lem.3.4.10.

A.1.1 Proof of Theorem 3.4.1

We divide the proof into three distinct steps. Some of the steps are formulated as lemmas to improve clarity of the presentation.

Step 1: In this step, we show that a moment bound holds for the iterates under consideration. The assertion is stated as a lemma.

Lemma A.1.1 For each $0 < T < \infty$, under the conditions of Thm.3.4.1,

$$
\sup_{0 \leq k \leq T/\mu} E|s_k|^2 < \infty. \tag{A.1}
$$

Proof of Lem.A.1.1. Define $V(s) = s's/2$. The gradient and Hessian of $V(s)$ are given by $\nabla V(s) = s$ and $\nabla^2 V(s) = I$, respectively. Under the Condition (C) we assume $H$ has no eigenvalues with negative real parts, hence there is a $\lambda(0) > 0$ such that $s'Hs \geq \lambda(0)V(s)$. In addition, since $X_k$ is a conditional Markov chain with a finite state space, it is bounded uniformly w.p.1, which implies that

$$
\begin{align*}
    s'_k \mathcal{W}^o X_k &\leq \frac{1}{2}(|s_k|^2 + |\mathcal{W}^o|^2 |X_k|^2) \\
                     &\leq K(V(s_k) + 1)
\end{align*}
$$
A.1. Proof of Results Contained in Chapter 3.

for some \( K > 0 \). Here and hereafter, \( K \) is used to represent a generic constant, whose values may change for different appearances. Using (3.16) and the above estimates, we obtain

\[
V(s_{k+1}) = V(s_k) + \mu s'_k (-H s_k + W^o X_k) + O(\mu^2)(V(s_k) + 1)
\leq V(s_k) - \lambda(0)\mu V(s_k) + \mu s'_k W^o X_k + O(\mu^2)(V(s_k) + 1)
\leq (1 - \lambda(0)\mu)V(s_k) + O(\mu)(V(s_k) + 1).
\] (A.2)

Taking expectations and iterating on the resulting sequence, we obtain

\[
EV(s_{k+1}) \leq (1 - \lambda(0)\mu)^{k+1} EV(s(0)) + O(\mu^2) \sum_{j=0}^{k-1} (1 - \lambda(0)\mu)^{k-j} EV(s_j) + O(\mu^2) \sum_{j=0}^{k-1} (1 - \lambda(0)\mu)^{k-j}.
\]

An application of the Gronwall’s inequality yields that

\[
EV(s_{k+1}) \leq O(1) \exp(\mu k) \leq O(1) \exp(\mu(T/\mu)) \leq O(1).
\]

Taking \( \sup_{0 \leq k \leq T/\mu} \) on both sides above yields the desired result. ■

Note that by means of Tchebyshev’s inequality and Lem.A.1.1, for any \( \delta > 0 \) sufficiently small, there is a \( K_\delta = (1/\sqrt{\delta}) \) sufficiently large such that

\[
P(|s_k| \geq K_\delta) \leq E|s_k|^2 \leq \sup_{0 \leq k \leq T/\mu} E|s_k|^2 \leq \frac{K_\delta}{K^2} \leq K\delta.
\]

Thus \( \{s_k\} \) is tight. Since \( \theta_k \) takes values in a finite set, it is also tight. Thus, we obtain the following corollary.

**Corollary A.1.2** The pair of sequences \( \{s_k, \theta_k\} \) is tight.

Step 2: We now show that the process \( (s^\mu(\cdot), \theta^\mu(\cdot)) \) is tight in \( D([0,T] : \mathbb{R}^S \times M) \). Since the tightness of \( \theta^\mu(\cdot) \) has been demonstrated in [90], we focus on the tightness of \( s^\mu(\cdot) \). For any \( \delta > 0 \), \( t > 0 \), and \( 0 < u \leq \delta \), we have

\[
s^\mu(t+u) - s^\mu(t) = \mu \sum_{k=t/\mu}^{(t+u)/\mu} (-H s_k + W^o X_k).
\]

Thus by repeating the proof of Lem.A.1.1, we can show

\[
E_t^\mu |s^\mu(t+u) - s^\mu(t)|^2 \leq K\mu^2 \sum_{k=t/\mu}^{(t+u)/\mu-1} (E_t^\mu |s_k|^2 + 1)
\leq K\mu^2 \left(1 + \frac{u}{\mu} - \frac{t}{\mu}\right)^2 \leq K u^2 \leq K\delta,
\]

for \( u \) sufficiently small, where \( E_t^\mu \) denotes the conditional expectation with respect to the \( \sigma\)-
A.1. Proof of Results Contained in Chapter 3.

algebra \( \mathbb{F}_t = \{ s^\mu(u), \theta^\mu(u) : u \leq t \} \). Thus

\[
\lim_{\delta \to 0} \limsup_{\mu \to 0} EE_t^\mu |s^\mu(t + u) - s^\mu(t)|^2 = 0.
\]

The claim then follows from the well-known tightness criteria [p. 47] [91] (see also [Section 7.3] [68]).

Step 3: Characterization of the limit. Owing to Step 2, \((s^\mu(\cdot), \theta^\mu(\cdot))\) is tight. By Prohorov’s theorem, we can extract a weakly convergent subsequence. Without loss of generality, still index the selected sequence by \( \mu \) and denote the limit by \((s(\cdot), \theta(\cdot))\). By the Skorohod representation, we may assume (with slight abuse of notation), \((s^\mu(\cdot), \theta^\mu(\cdot))\) converges to \((s(\cdot), \theta(\cdot))\) w.p.1 and the convergence is uniform in any bounded \( t \) interval. For each \( t, u > 0 \), partition \([t, t + u]\) into small segments with the use of \( k_\mu \to \infty \) as \( \mu \to 0 \) but \( \delta_\mu := \mu k_\mu \to 0 \) as \( \mu \to 0 \). Then we have

\[
s^\mu(t + u) - s^\mu(t) = \sum_{t \leq \delta_\mu = t} \frac{1}{k_\mu} \sum_{k \leq k_\mu} (-H s_k + \mathcal{W}^\circ X_k),
\]

where \( \sum_{t \leq \delta_\mu = t} \) denotes the sum over \( l \) in the range \( t \leq l \delta_\mu < t + u \). For the following analysis, it is crucial to recognize the two-time-scale structure of the algorithm. In the segment \( l k_\mu \leq k \leq l k_\mu + k_\mu - 1 \), compared with \( s_k \) and \( \theta_k \), \( X_k \) varies much faster. Thus, in this segment, \( s_k \) and \( \theta_k \) can be viewed as fixed. As a result, although \( X_k \) depends on \( \theta \), the slowly varying \( \theta_k \) enables us to treat \( X_k \) as a “noise” with \( \theta_k \) fixed at a specific value \( \bar{\theta} \). In the end, \( X_k \) will be averaged out and replaced by its stationary measure. More precisely, let \( \delta_\mu \to \bar{u} \) as \( \mu \to 0 \). Then for all \( l k_\mu \leq k \leq l k_\mu + k_\mu - 1 \), \( \mu k \to \bar{u} \). To emphasize the \( \theta \) dependence in \( X_k \), we write it as \( X_k(\theta_k) \). It then follows that

\[
\frac{1}{k_\mu} \sum_{k = l k_\mu}^{l k_\mu + k_\mu - 1} \mathcal{W}^\circ X_k = \frac{1}{k_\mu} \sum_{k = l k_\mu}^{l k_\mu + k_\mu - 1} \mathcal{W}^\circ X_k(\theta_k) = \frac{1}{k_\mu} \sum_{k = l k_\mu}^{l k_\mu + k_\mu - 1} \mathcal{W}^\circ X_k(\theta_k) + o(1)
\]

\[
\to \mathcal{W}^\circ \pi(\theta(\bar{u})) \quad \text{in probability as} \quad \mu \to 0,
\]

where \( o(1) \to 0 \) in probability as \( \mu \to 0 \). The last line follows from the ergodicity of the \( \theta \) dependent Markov chain \( X_k(\theta) \). As a result

\[
\sum_{t \leq \delta_\mu = t} \delta_\mu \frac{1}{k_\mu} \sum_{k = l k_\mu}^{l k_\mu + k_\mu - 1} \mathcal{W}^\circ X_k \to \int_t^{t + u} \mathcal{W}^\circ \pi(\theta(\bar{u})) d\bar{u}. \tag{A.3}
\]

Likewise, we can work with the term involving \( s_k \) by using the continuity in \( s \) of the following
A.1. Proof of Results Contained in Chapter 3.

expression, which leads to

$$
\sum_{t \leq \delta \leq t} \delta \mu_{k_r} \sum_{k = tk_r}^{l_k + k_r - 1} Hs_k \to \int_t^{t+u} Hs(\tilde{u})d\tilde{u}.
$$

(A.4)

Now combining (A.3) and (A.4), following the argument as in the proof of Theorem 4.5 in [49], it can be shown that \((s(\cdot), \theta(\cdot))\) is a solution of the martingale problem associated with the operator given by

$$
\mathcal{L}f(s, \theta) = \nabla f'(s, \theta)[-Hs + \mathcal{W}^{\alpha}(\theta^i)] + Qf(s, \theta^i),
$$

(A.5)

where

$$
Qf(s, \theta^i) = \sum_{j=1}^m q_{ij} f(s, \theta^j).
$$

The desired results thus follow.

A.1.2 Proof of Theorem 3.4.2

The proof can be shown in three steps as follows.

(i) Define

$$
\tilde{s}^\mu(\cdot) = s^\mu(\cdot + t_\mu), \quad \tilde{\theta}^\mu(\cdot) = \theta^\mu(\cdot + t_\mu),
$$

where \(t_\mu\) is given in the statement of the theorem. Then \((\tilde{s}^\mu(\cdot), \tilde{\theta}^\mu(\cdot))\) is tight, which can be proven similar to that in Thm.3.4.1. For \(0 < T < \infty\), extract a convergent subsequence \(\{\tilde{s}^\mu(\cdot), \tilde{\theta}^\mu(\cdot - T)\}\) with a limit denoted by \((s(\cdot), s_T(\cdot))\). Note that \(s(0) = s_T(T)\). Also note that \(\{s_k\}\) is tight, which can be proven as in Cor.A.1.2. The tightness of \(\{s_T(0)\}\) then implies that \(\{s_T(0)\}\) is tight. By using the following representation of the solution of the switched ODE and noting \(T\) is arbitrary, we have

$$
s_T(T) = \exp(-HT)s_T(0) + \sum_{i=1}^m \int_0^T \exp(-H(T - u))\mathcal{W}^{\alpha}(\theta^i)I_{\{\theta(u) = i\}}du
$$

$$
= \exp(-HT)s_T(0) + \sum_{i=1}^m \int_0^T \exp(-H(T - u))\mathcal{W}^{\alpha}(\theta^i)\nu_i
$$

$$
+ \sum_{i=1}^m \int_0^T \exp(-H(T - u))\mathcal{W}^{\alpha}(\theta^i)[I_{\{\theta(u) = i\}} - \nu_i]du.
$$

(A.6)

(ii) We claim that as \(T \to \infty\), the last term above goes to 0 in probability. To show this, it suffices to work with a fixed \(i\). Define

$$
\xi(T) = E \left| \int_0^T \exp(-H(T - u))\mathcal{W}^{\alpha}(\theta^i)\nu_i du \right|^2.
$$

82
Then it is readily seen that
\[
|\xi(T)| \leq K \left| E \int_0^T (W^\theta \pi(\theta'))' \exp(-H'(T-t)) dt \right| \\
\times \left| \int_0^T \exp(-H(T-t)) W^\theta \pi(\theta') [I_{\{\theta(u)=i\}} - \nu_i] [I_{\{\theta(t)=i\}} - \nu_i] du \right| \tag{A.7}
\]

By using the Markov property, it can be shown that
\[
\left| E[I_{\{\theta(u)=i\}} - \nu_i][I_{\{\theta(t)=i\}} - \nu_i] \right| = \left| [P(\theta(u)=i) - \nu_i][P(\theta(t)=i|\theta(u)=i) - \nu_i] \right| \\
\leq K \exp(-\kappa_0 u) \exp(-\kappa_0(t-u)) \leq K \exp(-\kappa_0 t),
\]

where \(\kappa_0 > 0\) is a constant representing the spectrum gap in the Markov chain owing to the irreducibility of \(Q\) (see [92, p. 300]). In the above and hereafter in the proof, we use \(K\) as a generic positive constant whose value may change in different appearances. Since \(-H\) is Hurwitz,
\[
\int_0^T \left| \exp(-H(T-u)) \right| du \leq \int_0^T \exp(-\lambda_H(T-u)) du \leq K \exp(-\lambda_H(T-t)),
\]
where \(\lambda_H > 0\). Then we have
\[
\left| E \int_0^T W^\theta \pi(\theta') [I_{\{\theta(u)=i\}} - \nu_i] [I_{\{\theta(t)=i\}} - \nu_i] du \right| \\
\leq K \exp(-\lambda_H(T-t)) \exp(-\kappa_0 t).
\]

Then
\[
|\xi(T)| \leq K \int_0^T |W^\theta \pi(\theta')| \exp(-H'(T-t)) \exp(-\lambda_H(T-t)) \exp(-\kappa_0 t) dt.
\]

If \(\lambda_H \leq \kappa_0\), \(\exp((\lambda_H - \kappa_0)t) \leq 1\), so
\[
|\xi(T)| \leq K \int_0^T \exp(-\lambda_H(T-t)) \exp(-\lambda_H T) dt \leq K \exp(-\lambda_H T) \to 0 \text{ as } T \to \infty.
\]

If \(\lambda_H > \kappa_0\),
\[
|\xi(T)| \leq K \int_0^T \exp(-\lambda_H(T-t)) \exp(-(\lambda_H - \kappa_0)T) \exp(-\kappa_0 T) dt \\
\leq K \exp(-\kappa_0 T) \to 0 \text{ as } T \to \infty.
\]

Thus the claim in (ii) is verified.

(iii) The tightness of \(\{s_T(0)\}\) together with \(-H\) being Hurwitz implies the first term on the right-hand side of (A.6) tends 0 as \(T \to \infty\). The second term in (A.6) converges to \(s_*\) as
$T \to \infty$ by the hypothesis of this theorem. The desired result thus follows the above and (ii).

A.1.3 Proof of Theorem 3.4.4

The proof here is similar to Thm.3.4.1 with the modification of the inclusion. We omit the proof and refer the reader to [68] for related results on averaging leading to differential inclusions. Note that the main difference of the current result compared with Thm.3.4.1 is: In Thm.3.4.1, assumption (B) implies the existence of the unique invariant measure for each $\theta$. Here this condition is relaxed. For Markov chains with multiple ergodic classes, we refer the reader to [93, Chapter 1] for further details.

To proceed, we show that the sequence $(v_{\mu}(\cdot), \theta_{\mu}(\cdot))$ converges weakly to $(v(\cdot), \theta(\cdot))$ such that the limit is a solution of the martingale problem with the operator given by

\[
\mathbb{L}_v f(v, \theta^i) = -\nabla f(v, \theta^i) H v + \frac{1}{2} \text{tr} [\nabla^2 f(v, i) \Sigma(\theta^i) \Sigma^o ] + Q f(v, \cdot)(\theta^i), \; \theta^i \in M. \tag{A.8}
\]

The rest of the proof is similar to that of [49]. The details are omitted.

A.1.4 Proof of Theorem 3.4.6

The well-known Glivenko-Cantelli theorem (see [69, p. 103]) for mixing processes implies that

\[
\frac{1}{k_{\mu}} \sum_{k=lk_{\mu}}^{lk_{\mu}+k_{\mu}-1} I_{\{X_k(\theta_{j1}) \leq x\}} \to F(\theta_{j_1}, x) \quad \text{w.p.1 as } \mu \to 0 \quad \text{and hence } \; k_{\mu} \to \infty.
\]

The Markov structure implies that

\[(A(\theta_{j1}))^{k_{\mu}} \to 1 \pi(\theta_{j1}) \text{ as } \mu \to 0.\]

The limit is a matrix with identical rows containing the stationary distribution $\pi(\theta_{j1})$. Since $I_{\{\theta_{ik_{\mu}}=\theta_{j1}\}}$ can be written as $I_{\{\theta_{k\mu}(\delta_{\mu})=\theta_{j1}\}}$. As $\mu \to 0$ and $k\delta_{\mu} \to \tilde{u}$, we can show

\[
\frac{1}{k_{\mu}} \sum_{k=lk_{\mu}}^{lk_{\mu}+k_{\mu}-1} E_{lk_{\mu}} I_{\{X_k(\theta) \leq x\}} \to \sum_{j_1=1}^{m} \pi^j(\theta_{j1}) F(\theta_{j1}, x)I_{\{\theta(\tilde{u})=\theta_{j1}\}} = \pi^j(\theta(\tilde{u})) F(\theta(\tilde{u}), x).
\]

Thus we obtain that the limit $\eta(\cdot)$ satisfies

\[
\eta(t) - \eta(u) = \int_u^t \pi^j(\theta(\tilde{u})) F(\theta(\tilde{u}), x) d\tilde{u}.
\]

The desired result then follows immediately.
A.1. Proof of Results Contained in Chapter 3.

A.1.5 Proof of Lemma 3.4.8

Intuitively, we motivate Lemma 3.4.8 by the following rationale: assume all sensor computations possess the linear form (3.16). Then the average-consensus \( \bar{\pi}(\theta(t)) \) can be obtained under Condition (1) if each sensor maintains, in addition to (3.16), two distinct estimates based on the following sub-algorithms of (3.16):

1. The estimate \( \hat{s}_k \) based on only the local tracking sub-algorithm of (3.16),

\[
\hat{s}_{k+1}^i = \hat{s}_k^i + \mu(X_k^i - \tilde{s}_k^i), \quad \hat{s}(0) = X(0), \quad i \in \mathcal{V}
\]  

(A.9)

2. The estimate \( s_k^i \) based only on the (static) Laplacian consensus sub-algorithm of (3.16),

\[
s_{k+1}^i = (I - \mu \mathcal{L})s_k^i, \quad s(0) = X(0)
\]

As \( \mu \to 0 \), the local tracking estimates \( \hat{s}_k^i \) converge weakly to \( \pi^i(\theta(t)) \) for each \( i \in \mathcal{V} \). In comparison, it is well-known [18] that for any graph Laplacian \( 0 \preceq \mathcal{L} \) with left-right eigenvectors \( \omega_L, \omega_R \) corresponding to the zero eigenvalue and satisfying \( \omega_L^i \omega_R = 1, \omega_L > 0 \), the estimate \( s_k^i \) will then approach a convex-consensus \( s_\infty = \omega_L \omega_R^i s(0) = \omega_L \omega_R^i X(0) \) for all \( i \in \mathcal{V} \). Regardless, the linear combination denoted

\[
s^i(t)_{(1)} = \frac{1}{\alpha}(s^i(t) - \hat{s}^i(t)) + \hat{s}^i(t),
\]

(A.10)

has a steady-state \( \bar{\pi}(\theta(t)) \) for each \( i \in \mathcal{V} \). Under Condition (2) the linear combination,

\[
s^i(t)_{(2)} = \frac{1}{n\beta_i}(s^i(t) + (n\beta_i - 1)\hat{s}^i(t)),
\]

(A.11)

has a steady-state \( \bar{\pi}(\theta(t)) \) for each \( i \in \mathcal{V} \), as follows from (A.13).

**Exact Solutions to (3.18).** We first derive

\[
s(t) = e^{-Ht}(s(0) - \Lambda\pi(\theta(t))) + \Lambda\pi(\theta(t)),
\]

(A.12)

as an exact solution to (3.18) conditional on the state \( \theta(t) \) of \( \theta \in \mathcal{M} \). The equilibrium matrix \( \Lambda \) is defined by the following function of \( \{\mathcal{G}^v, \mathcal{G}^o\} \),

\[
\Lambda = \begin{cases} 
(L + \mathcal{D}^o)^{-1} \mathcal{W}^o & \text{if } 0 \preceq (L + \mathcal{D}^o), \\
(I - \omega_L \omega_R) & \text{if } 0 \preceq (L + \mathcal{D}^o), \text{ and } \mathcal{W}^o = L + \mathcal{D}^o.
\end{cases}
\]

(A.13)
A.1. Proof of Results Contained in Chapter 3.

If $0 \prec H$ then the above results follow immediately by standard techniques of solving systems of first order linear ODEs. On the other hand if $0 \preceq H$, then (A.13) is derived based on the conservation property,

$$
\frac{d}{dt} \omega'_\ell s(t) = \omega'_\ell ds(t) = -\omega'_\ell H s(t) + \omega'_\ell W^o \pi(\theta(t)) = 0,
$$

(A.14)

which follows under the assumption $W^o = H$, since $\omega'_\ell H = 0$ by definition. From (A.14) it is clear $\Lambda$ must satisfy $\omega'_\ell \Lambda = 0$, whereas by solving $\frac{ds(t)}{dt} = 0$ given (A.12) we have $H \Lambda = W^o$. These two constraints then uniquely determine the second steady-state in (A.13).

For any ntrcs graph $G^v$, if and only if $L$ is balanced will $\omega_r \omega'_\ell = \frac{1}{n} 11'$ [18]. On the other hand, if $0 \preceq H$ then $\omega_r = c1$ for any non-zero $c \in \mathbb{R}$. Since $\omega'_\ell \omega_r = 1$ it is clear that if $\omega'_\ell \omega_r \neq \frac{1}{n} 11'$ then the average-consensus cannot be obtained by any linear combination of $s(t)$, $\dot{s}(t)$ or $\ddot{s}(t)$. Furthermore, when $0 \preceq H$ and $W^o \neq H$, then (3.16), although stable in the limit as $\mu$ vanishes, is asymptotically unbounded as the number of $\lfloor 1/\mu \rfloor$ iterations increase. To see this assume $t \leq 0$ and $0 < \mu \ll 1$ are such that $(t/\mu) \in \mathbb{N}$. The $(t/\mu)$ iteration of (3.16) yields the sensor estimates,

$$
s_{t/\mu} = (I - \mu L - \mu D^o)^{t/\mu} s(0) + \sum_{l=0}^{t/\mu-1} (I - \mu L - \mu D^o)^{t/\mu} \mu W^o X_{t/\mu-1-l}.
$$

(A.15)

For arbitrary weight matrices $\{W^v, W^o\}$ the limit of (A.15) when $\mu$ approaches zero yields the same expression for the coefficient of $s(0)$ as the exact solution (A.12),

$$
\lim_{\mu \to 0} (I - \mu L - \mu D^o)^{t/\mu} = e^{-(L+D^o)t}.
$$

Similarly in the limit as $\mu$ vanishes, by (A.3) above we replace $X_{t/\mu-1-l}$ with the stationary measure $\pi(\theta(t))$, the summed coefficients of which yield the following limiting values,

$$
\lim_{\mu \to 0} \sum_{l=0}^{t/\mu-1} (I - \mu L - \mu D^o)^{t/\mu} \mu W^o = \begin{cases} H^{-1}(I - e^{-Ht}) W^o, & 0 \prec H \\ \alpha(I - e^{-Ht}) W^o, & 0 \preceq H, \ W^o = \alpha H, \ \alpha \in \mathbb{R}. \end{cases}
$$

(A.16)

Without loss of generality set $\alpha = 1$. As $t$ increases the above limits become those expressed in (A.13). If, however, $0 \preceq H$ and $W^o \neq H$, the above iteration results in an unbounded steady-state in the limit $t$ increases, as can be seen by re-expressing (A.16) in terms of the
Proof of Results Contained in Chapter 3.

A.1. Proof of Results Contained in Chapter 3.

A.1.1. Proof of Results Contained in Chapter 3.

A eigen-decompositions $H = UJU^{-1}$ and $W^o = ARA^{-1}$,

$$
\sum_{l=0}^{t/\mu-1} (I - \mu L - \mu D^o)^l \mu W^o = U \sum_{l=0}^{t/\mu-1} (I - \mu J)^l \mu U^{-1} ARA^{-1}. \tag{A.17}
$$

In the limit as $\mu$ vanishes this expression becomes $UJ^*U^{-1}ARA^{-1}$, where the zero eigenvalue of $J$ is replaced by $t$ in the matrix $J^*$ and all others replaced by $\frac{1-e^{-\lambda_i t}}{\lambda_i}$, for convenience we let the first element of $J$ denote this zero eigenvalue, $J_{11} = 0$. As $t$ increases it is clear $J^*$ approaches $J^{-1}$ except in its first element, which grows linearly with $t$. For (A.17) to remain bounded then as $t$ increases, the multiplications $UJ^*U^{-1}ARA^{-1}$ must eliminate the presence of $t$ in $J^*_{11}$. It is straightforward to see this occurs if and only if $W^o = H$, in which case $A = U$, $R = J$, and the zero eigenvalue in $R$ eliminates the presence of $t$ in $J^*$ regardless the value of $\mu$. In this case the denominator of $J^*$ is also eliminated by right multiplication of $ARA^{-1}$ and we have the second steady-state of (A.13).

Since we have considered all possible solutions to (3.18) with bounded steady-states, there exist no other methods by which to achieve average-consensus through (3.16) and its implied sub-algorithms. Our rationale has been to place conditions on $\{G^v, G^o\}$ such that the sensor steady-states under (3.16) can be adjusted by their local tracking estimates $\hat{s}(t)$ to obtain the average-consensus estimate $\bar{\pi}(\theta(t))$. This requires the equilibrium matrix $\Lambda$ have each of its rows $i = 1, \ldots, n$ comprised of a non-zero scalar $\beta_i$, with diagonals defined $(1 - (n-1)\beta_i)$, as we have expressed by (3.30). The diagonals are defined as such due to the property $\Lambda 1 = 1$ which holds only when $0 \prec H$, as can be seen by (A.13) and the fact

$$(L + D^o)^{-1} W^o 1 = 1 \Rightarrow D^o 1 = (L + D^o) 1 \tag{A.18}$$

which holds since $L 1 = 0$ by definition. Assuming $0 \prec H$, the necessity of $\Lambda 1 = 1$ is evident simply from the contradiction entailed by (A.18) when $\Lambda 1 \neq 1$.

The equality among non-diagonal elements within each row $i$ implies that the sensor estimate $s^i(t)$ approaches a linear average with uniform weights $(\beta_i)$ assigned to the currently observed stationary distributions of all other sensors, and a disproportionate weight $(1 - (n-1)\beta_i)$ given to its own locally observed stationary distribution. If the weights $\beta_i$ are known then by the correction with $\tilde{s}^i_k$ and re-scaling, as stated in (A.11), we obtain the steady-state $\bar{\pi}(\theta(t))$. ■

We make two remarks here.

1. The average-consensus estimate $s(t)_{(1)}$ in (A.10) assumes $L$ is balanced and that each sensor knows the scale factor $\alpha$ between $(L + D^o)$ and $W^o$. In Condition (2), the state-value exchange graph Laplacian $L$ need not be balanced, but each sensor $i \in \mathcal{V}$ must know the total network size $n$, as well as the $i^{th}$ diagonal element of the matrix $\beta$ in (3.30). Computation
of any diagonal element in $\beta$ will by (3.29) require near complete knowledge of the exchange graphs $\{G^o, G^v\}$, and so under Condition (2) we must presume each sensor has this knowledge.

2. Under condition (2), it follows by (A.11) that $s(t) = \tilde{s}(t)$ if and only if $\beta = \frac{1}{n}I$. In this case the extra local tracking estimate $\tilde{\beta}$ of any diagonal element in

$$A.1.6 \text{ Proof of Lemma 3.4.9}$$

The proof follows by showing (3.29) is infeasible when $0 \prec H$ and $E^o = E^v$. Since $0 \prec H$ we can rearrange (3.29) such that $W^v = HA$. Denote the $i^{th}$ row of $H$ by the row vector $w_i$. Since each row of $W^v$ has at least one zero non-diagonal element (since we are assuming a decentralized communication graph), thus there exists an element $j_i$ such that $w_{j_i} = 0$ for each row $i \in V$. Since $E^o = E^v$ we then have by (3.29) and (3.30) that $1' \beta w'_i = 0$ for each row $i$. This implies the rows of $H$ are linearly dependent and thus $H$ has an eigenvalue at 0, which contradicts our assumption $0 \prec H$. Thus (3.29) is infeasible when $E^o = E^v$ and $0 \prec H$. On the other hand, if $E^o = E^v$ then we may set $W^o = \alpha(L + D^o)$ and Condition (1) holds if $L$ is balanced.

$$A.1.7 \text{ Proof of Lemma 3.4.10}$$

By our assumption of a decentralized network, there exists for each row $i$ an element $j_i^o$ such that $(i, j_i^o) \notin E^o$ and an element $j_i^v$ such that $(i, j_i^v) \notin E^v$. For any row $i$, if $j_i^o = j_i^v$ then we have $1' \beta w'_i = 0$. By Lemma (3.4.9) if $1' \beta w'_i = 0$ holds for all $i \in V$ then average-consensus is infeasible when $E^o \neq E^v$, thus there must be at least one row $i^*$ such that $1' \beta w'_{i^*} \neq 0$. This implies $(i^*, j_{i^*}^o) \in E^o_{i^*}$ for at least one $j_{i^*}^o$ such that $(i^*, j_{i^*}^v) \notin E^v_{i^*}$, which implies $(i^*, j_{i^*}^v) \in E^o_{i^*}$ for all $j_{i^*}^o$ such that $(i^*, j_{i^*}^v) \notin E^v_{i^*}$. Thus $E^o_{i^*} \supseteq E^v_{i^*}$. On the other hand, Condition (1) of Lemma (3.4.8) cannot be satisfied if $E^o \neq E^v$, thus in this case we require Condition (2).

$$A.1.8 \text{ Proof of Corollary 3.4.11}$$

The result follows immediately from Lemma 3.4.10.

**A.2 Proof of Results Contained in Chapter 4.**

This section details the proof of Thm.4.4.9 and Thm.4.4.10. Before proceeding, we clarify the following notations. An arbitrary signal $S^{ij}(t_0, t_1)$ is now specified as $S^{ij}(t_0^{ij}, t_1^{ij})$, and an
A.2. Proof of Results Contained in Chapter 4.

arbitrary communication sequence $C_{[t_0,t_1]}$ is now denoted $C_{[t_0,t_1]}$. We define the squared “error” of the normal consensus estimate $\hat{v}_i(t)$ as,

$$E^2(\hat{v}_i(t)) = ||\hat{v}_i(t) - \frac{1}{n}1_n||^2. \quad (A.19)$$

The total reduction in normal consensus squared error resulting from the sequence $C_{[t_0,t_1]}$ is defined as,

$$E^2(C_{[t_0,t_1]}) = \lim_{t \to t_1} \sum_{S^{ij}(t_{ij}^0,t_{ij}^1) \in C_{[t_0,t_1]}} E^2(S^{ij}(t_{ij}^0,t_{ij}^1)), \quad (A.20)$$

where $E^2(S^{ij}(t_{ij}^0,t_{ij}^1))$ is defined using the normal consensus error (A.19) as follows,

$$E^2(S^{ij}(t_{ij}^0,t_{ij}^1)) = E^2(\hat{v}_i(t_{ij}^1)) - E^2(\hat{v}_i(t_{ij}^1 + 1)). \quad (A.21)$$

Let $\hat{v}_i(t)$ denote the $i^{th}$ element of the vector $\hat{v}_i(t)$. Lastly, it will be convenient to use the following definition.

Definition A.2.1 Under (A4'), the average-consensus problem (4.2.2) is solved at time $t_1 \in (0,\infty]$ if the “network normal consensus error” $\sum_{i=1}^n E^2(\hat{v}_i(t))$ approaches zero as $t$ approaches $t_1$, that is,

$$\lim_{t \to t_1} \sum_{i=1}^n E^2(\hat{v}_i(t)) = 0. \quad (A.22)$$

A.2.1 Proof of Theorem 4.4.9

Outline of Proof: The proof of Theorem 4.4.9 follows from four observations: (i) each normal consensus estimate satisfies the normalization property $||\hat{v}_i(t)||^2 = (1/n)\hat{v}_i(t)1_n$ (Lemma A.2.3), (ii) each normal consensus estimate satisfies a “zero local error” property $\hat{v}_i(t) = 1/n$ (Lemma A.2.7), (iii) the normal consensus reduction in error resulting from any signal will eventually vanish (Lemma A.2.11), (iv) any IVSC sequence contains an infinite number of contiguous SVSC sub-sequences, cf. (4.4.2). By combining these four features we can show that each $\hat{v}_i(t)$ must asymptotically approach $\frac{1}{n}1_n$ in the $L_2$-norm.

To help clarify our proof, we note that Lemmas A.2.2-A.2.13 deal with the properties of the DA normal consensus estimates, whereas Lemmas A.2.14-A.2.17 deal with the convergence of the DA normal consensus estimates in the $L_2$-norm. Lemma A.2.14 shows the receiving estimate (prior to update) converges to the transmitted estimate. Lemma A.2.15 shows the receiving estimate (prior to update) converges to its value after update. Lemma A.2.16 combines the previous two lemmas to show the receiving estimate (after update) converges to the transmitted estimate. By then combining properties (i),(ii), and (iv) above, we obtain the desired result.
Before each lemma we describe in words its intended purpose.

**Lemma A.2.2 (DA Consensus Estimate Updates)** Applying (4.11)–(4.12) to (4.4)–(4.5) yields the DA consensus estimate updates (4.13)–(4.14).

**Proof.** Under (4.11) – (4.12) we have for any knowledge set \( K_i(t_{ij}^1) \) and signal \( S^{ij}(t_{ij}^0, t_{ij}^1) \),
\[
K_i(t_{ij}^1) \cup S^{ij}(t_{ij}^0, t_{ij}^1) = \{ i, n, s_i(0), \dot{s}_i(t_{ij}^1), \dot{v}_i(t_{ij}^1), \ddot{s}_j(t_{ij}^0), \ddot{v}_j(t_{ij}^0) \},
\]
where \( s_i(0) = Sc_i \) , \( \dot{s}_i(t_{ij}^1) = S\dot{v}_i(t_{ij}^1) \) , \( \ddot{s}_j(t_{ij}^0) = S\ddot{v}_j(t_{ij}^0) \).
It follows that \( V_i = [e_i, \dot{v}_i(t_{ij}^1), \ddot{v}_j(t_{ij}^0)] \) and \( S_i = [s_i(0), \dot{s}_i(t_{ij}^1), \ddot{s}_j(t_{ij}^0)] \). We can thus re-write (4.4) as,
\[
\dot{v}_i(t_{ij}^1 + 1) = \arg \min_{\ddot{v} \in \text{span}\{\dot{v}_i(t_{ij}^1), \ddot{v}_j(t_{ij}^0), e_i\}} ||\ddot{v} - \frac{1}{n}1_n||^2.
\]
(A.23)
The update (4.13) follows from (A.87) by standard results in least-squares optimization, and the update (4.14) then follows from (4.5).

**Lemma A.2.3 (DA Consensus Estimate Normalization)** Every normal consensus estimate \( \dot{v}_i(t) \) satisfies
\[
||\dot{v}_i(t)||^2 = \frac{1}{n} \dot{v}_i(t)'1_n , \forall i \in V , \forall t \geq 0 .
\]
(A.24)

**Proof.** From (4.15) we have the initial condition \( \dot{v}_i(0) = \frac{1}{n}e_i \), which satisfies (A.80) for each \( i \in V \). Next observe that under (A7) the estimate \( \dot{v}_i(t) \) will not change unless a signal \( S^{ij}(t_{ij}^0, t_{ij}^1) \) is received at node \( i \). If a signal is received then by Lemma A.2.2 the estimate \( \dot{v}_i(t) \) is updated to the unique solution of (A.87). Thus to finish the proof it suffices to show that if a vector \( v \in \mathbb{R}^{n \times 1} \) does not satisfy (A.80) then the vector \( v \) is not the solution to (A.87). To prove this we show that if (A.80) does not hold then the vector \( w \) defined as,
\[
w = v(v'1_n)/(n||v||^2) ,
\]
will satisfy the inequality
\[
||w - \frac{1}{n}1_n||^2 < ||v - \frac{1}{n}1_n||^2 .
\]
(A.25)
Since \( w \) is contained in span(\( v \)), the inequality (A.25) implies that \( v \) is not the solution to (A.87). Next observe that if a vector \( v \) does not satisfy (A.80) then,
\[
( ||v||^2 - \frac{1}{n}v'1_n)^2 > 0 .
\]
(A.26)

\(^4\)For brevity we do not show that (A4’), (4.4), and (4.5) together imply the DA initialization (4.15). In lieu of a formal proof, note that under (A4’) the initialization (4.15) can simply be assumed, regardless of (4.4) – (4.5).
Expanding (A.26) yields,

$$
(||v||^2)^2 - 2\frac{1}{n} v'1_n ||v||^2 + \left(\frac{1}{n} v'1_n\right)^2 > 0.
$$

(A.27)

Re-arranging (A.27) then implies (A.25),

$$
(||v||^2)^2 - 2\frac{1}{n} v'1_n ||v||^2 > -\left(\frac{1}{n} v'1_n\right)^2,

||v||^2 - 2\frac{1}{n} v'1_n + \frac{1}{n} > \left(\frac{v'1_n}{n||v||^2}\right)^2 ||v||^2 - 2\frac{(v'1_n)^2}{n^2||v||^2} \frac{1}{n},

||v - \frac{1}{n} 1_n||^2 > ||v'1_n\||^2 - \frac{1}{n} ||1_n||^2 = ||w - \frac{1}{n} 1_n||^2.
$$

\[\square\]

**Lemma A.2.4 (DA Non-Decreasing Normal Consensus Estimate Magnitude)** Each magnitude $||\hat{v}_i(t)||^2$ is a non-decreasing function of $t \geq 0$ for all $i \in \mathcal{V}$. □

**Proof.** Under (A7) the estimate $\hat{v}_i(t)$ will not change unless a signal is received at node $i$. If a signal $S^{ij}(t_0^{ij}, t_1^{ij})$ is received then the DA update problem (A.87) implies the update $\hat{v}_i(t_1^{ij} + 1)$ must satisfy,

$$
||\hat{v}_i(t_1^{ij} + 1) - \frac{1}{n} 1_n||^2 \leq ||w - \frac{1}{n} 1_n||^2,

\forall \ w \in \text{span} \{\hat{v}_i(t_1^{ij}), \hat{v}_j(t_1^{ij}), e_i\}.
$$

(A.28)

Since $\{\hat{v}_i(t_1^{ij}), \hat{v}_j(t_1^{ij}), e_i\}$ is a non-decreasing function of $t \geq 0$ for all $i \in \mathcal{V}$. □

Next observe that if a vector $v \in \mathbb{R}^{n \times 1}$ satisfies (A.80) then,

$$
||v - \frac{1}{n} 1_n||^2 = ||v||^2 - 2\frac{1}{n} v'1_n + \frac{1}{n} = \frac{1}{n} - ||v||^2.
$$

(A.30)

Due to Lemma A.2.3, all normal consensus estimates satisfy (A.80), thus we can apply (A.30) to (A.29) and obtain,

$$
\frac{1}{n} - ||\hat{v}_i(t_1^{ij} + 1)||^2 \leq \min\{\frac{1}{n} - ||\hat{v}_i(t_1^{ij})||^2, \frac{1}{n} - ||\hat{v}_j(t_0^{ij})||^2\}.
$$

(A.31)

Subtracting both sides of (A.31) from $\frac{1}{n}$ then yields,

$$
||\hat{v}_i(t_1^{ij} + 1)||^2 \geq \max\{||\hat{v}_i(t_1^{ij})||^2, ||\hat{v}_j(t_0^{ij})||^2\} \geq ||\hat{v}_i(t_1^{ij})||^2,
$$

(A.32)
thus each magnitude $||\hat{v}_i(t)||^2$ is a non-decreasing function of $t \geq 0$ for all $i \in \mathcal{V}$.

**Lemma A.2.5 (Equality of Normalized Linearly Dependent Vectors)** Any two non-zero vectors that satisfy (A.80) are linearly dependent iff they are identical.

**Proof.** The proof is omitted for brevity. See [87] for details.

**Remark A.2.6** The initialization (4.15) combined with Lemma A.2.4 implies that every DA normal consensus estimate will be non-zero for all $t \geq 0$. Due then to Lemma A.2.3, the result stated in Lemma A.2.5 is applicable to all DA normal consensus estimates.

**Lemma A.2.7 (DA Zero Local Error Property)** Every normal consensus estimate $\hat{v}_i(t)$ satisfies

$$\hat{v}_i(t) = \frac{1}{n} \cdot \forall i \in \mathcal{V}, \forall t \geq 0.$$  

(A.33)

**Proof.** The initial condition (4.15) implies $\hat{v}_i(0) = \frac{1}{n}$ for each $i \in \mathcal{V}$. Next observe that under (A7) the estimate $\hat{v}_i(t)$ will not change unless a signal $S^{ij}(t_0^{ij}, t_1^{ij})$ is received at node $i$. If a signal is received then $\hat{v}_i(t)$ is updated to the solution of (A.87). To finish the proof, we now consider the three possibilities: (i) $\{\hat{v}_i(t_1^{ij}), \hat{v}_j(t_0^{ij}), e_i\}$ contain three linearly dependent vectors, (ii) $\{\hat{v}_i(t_1^{ij}), \hat{v}_j(t_0^{ij}), e_i\}$ contain two linearly dependent vectors, and (iii) $\{\hat{v}_i(t_1^{ij}), \hat{v}_j(t_0^{ij}), e_i\}$ contain no linearly dependent vectors.

**Part (i).** If the set of vectors $\{\hat{v}_i(t_1^{ij}), \hat{v}_j(t_0^{ij}), e_i\}$ are a linearly dependent set, then from (A.87) we have $\hat{v}_i(t_1^{ij} + 1) \in \text{span}\{e_i\}$ which implies from Lemma A.2.3 that $\hat{v}_i(t_1^{ij} + 1) = \frac{1}{n}e_i$, thus satisfying (A.83).

**Part (ii).** Next assume that only two vectors in the set $\{\hat{v}_i(t_1^{ij}), \hat{v}_j(t_0^{ij}), e_i\}$ are linearly dependent. In this case, without loss of generality, (A.87) reduces to

$$\hat{v}_i(t_1^{ij} + 1) = \text{arg min}_{\bar{v} \in \text{span}\{e_i, v\}} \|\bar{v} - \frac{1}{n}1_n\|^2,$$

$$= \text{arg min}_{\bar{v} \in \{a \frac{1}{n}e_i + bv : (a, b) \in \mathbb{R}^2\}} \|\bar{v} - \frac{1}{n}1_n\|^2,$$

(A.34)

where $v \in \{\hat{v}_i(t_1^{ij}), \hat{v}_j(t_0^{ij})\}$ is linearly independent of $e_i$. The objective function in (A.34) is

$$f(a, b) = \|a \frac{1}{n}e_i + bv - \frac{1}{n}1_n\|^2,$$

$$= a^2\|\frac{1}{n}e_i\|^2 + b^2\|v\|^2 + 2ab\frac{1}{n}e_i'v$$

$$- 2a\frac{1}{n}e_i'1_n - 2bv\frac{1}{n}1_n + \frac{1}{n}.$$  

(A.35)

The Lemma A.2.3 implies that $v$ satisfies (A.80). Note also that $\frac{1}{n}e_i$ satisfies (A.80), thus the objective function (A.35) can be simplified,

$$f(a, b) = (a^2 - 2a)\|\frac{1}{n}e_i\|^2 + (b^2 - 2b)\|v\|^2 + 2ab\frac{1}{n}e_i'v + \frac{1}{n}.$$  

92
The first-order partial derivatives of \( f(a, b) \) are,
\[
\begin{align*}
\frac{\partial f(a,b)}{\partial a} &= 2(a-1)||\frac{1}{n}e_i||^2 + 2b\frac{1}{n}e'_i v, \\
\frac{\partial f(a,b)}{\partial b} &= 2(b-1)||v||^2 + 2a\frac{1}{n}e'_i v.
\end{align*}
\]
(A.36)

The second-order partial derivatives of \( f(a, b) \) are,
\[
\begin{align*}
\frac{\partial^2 f(a,b)}{\partial a^2} &= 2||\frac{1}{n}e_i||^2 , \\
\frac{\partial^2 f(a,b)}{\partial a \partial b} &= 2||v||^2 , \\
\frac{\partial^2 f(a,b)}{\partial b^2} &= \frac{\partial^2 f(a,b)}{\partial b \partial a} = 2\frac{1}{n}e'_i v .
\end{align*}
\]

The determinant of the Hessian matrix of \( f(a, b) \) is thus,
\[
\det[H(f(a,b))] = 2(||\frac{1}{n}e_i||^2||v||^2 - (\frac{1}{n}e'_i v)^2) .
\]
(A.37)

Since \( \frac{1}{n}e_i \) and \( v \) are linearly independent, the determinant (A.37) is strictly positive by the Cauchy-Schwartz inequality. This implies the Hessian matrix \( H(f(a,b)) \) is positive-definite, thus setting the right-hand side (RHS) of (A.36) to zero and solving for \((a, b)\) yields the unique optimal values \((\hat{a}, \hat{b})\),
\[
\hat{a} = \frac{||v||^2\frac{1}{n}e'_i (\frac{1}{n}e_i - v)}{||\frac{1}{n}e_i||^2||v||^2 - (\frac{1}{n}e'_i v)^2} , \quad \hat{b} = \frac{||\frac{1}{n}e_i||^2v'(v - \frac{1}{n}e_i)}{||\frac{1}{n}e_i||^2||v||^2 - (\frac{1}{n}e'_i v)^2} .
\]
(A.38)

From (A.38) the unique solution to (A.34) is thus obtained,
\[
\hat{v}_i(t_{1}^{ij} + 1) = \frac{||v||^2\frac{1}{n}e'_i (\frac{1}{n}e_i - v)}{||\frac{1}{n}e_i||^2||v||^2 - (\frac{1}{n}e'_i v)^2} \frac{1}{n}e_i
\]
\[
+ \frac{||\frac{1}{n}e_i||^2v'(v - \frac{1}{n}e_i)}{||\frac{1}{n}e_i||^2||v||^2 - (\frac{1}{n}e'_i v)^2} v .
\]
(A.39)

Denoting the \( i^{th} \) element of \( v \) as \( v_i \), we can express the element \( v_i(t_{1}^{ij} + 1) \) based on (A.39) as follows,
\[
\hat{v}_{ii}(t_{1}^{ij} + 1) = \frac{||v||^2\frac{1}{n}e'_i (\frac{1}{n}e_i - v)}{||\frac{1}{n}e_i||^2||v||^2 - (\frac{1}{n}e'_i v)^2} \frac{1}{n}e_i
\]
\[
+ \frac{||\frac{1}{n}e_i||^2v'(v - \frac{1}{n}e_i)}{||\frac{1}{n}e_i||^2||v||^2 - (\frac{1}{n}e'_i v)^2} v_i
\]
\[
= \frac{||v||^2\frac{1}{n}e'_i (\frac{1}{n}e_i - v)}{||v||^2 - \frac{1}{n}v_i^2} \frac{1}{n}
\]
\[
+ \frac{\frac{1}{n}v_i^2}{||v||^2 - \frac{1}{n}v_i^2} \frac{1}{n}v_i
\]
\[
= \frac{\frac{1}{n}||v||^2 - v_i^2}{||v||^2 - \frac{1}{n}v_i^2} + \frac{v_i^2}{||v||^2 - \frac{1}{n}v_i^2} v_i
\]
\[
= \frac{\frac{1}{n}||v||^2 - v_i^2}{||v||^2 - \frac{1}{n}v_i^2} + \frac{\frac{1}{n}v^2}{||v||^2 - \frac{1}{n}v_i^2} v_i = \frac{1}{n} .
\]
(A.40)

The last equality in (A.40) follows since Lemma A.2.5 implies \( ||v||^2 \neq v_i^2 \) under the assumption that \( v \) satisfies (A.80) and is linearly independent from \( \frac{1}{n}e_i \).
A.2. Proof of Results Contained in Chapter 4.

Part (iii). Lastly, we assume that \{\hat{v}_i(t_1^j), \hat{v}_j(t_0^j), e_i\} is a linearly independent set of vectors. In this case the update (4.13) can be expressed,

\[ \hat{v}_i(t_1^j + 1) = V_{(DA)}(V_{(DA)}' V_{(DA)})^{-1} V_{(DA)}' \frac{1}{n} 1_n, \]

\[ V_{(DA)} = [\hat{v}_i(t_1^j), \hat{v}_j(t_0^j), \frac{1}{n} e_i]. \]

For notational convenience we denote \( \hat{v}_i = \hat{v}_i(t_1^j) \) and \( \hat{v}_j = \hat{v}_j(t_0^j) \). The product \( (V_{(DA)}') \) has the inverse \( (A.41) \),

\[ (V_{(DA)}')^{-1}(\hat{v}_i) \]

\[ = \frac{1}{\det(V_{(DA)}')}(||\hat{v}_i||^2||\hat{v}_j|| - (\hat{v}_i' \hat{v}_j)^2 + 2\hat{v}_i \hat{v}_j) \]

where the determinant \( \det(V_{(DA)}') \) can be computed as,

\[ \det(V_{(DA)}') = \frac{1}{n^2}(||\hat{v}_i||^2||\hat{v}_j|| - (\hat{v}_i' \hat{v}_j)^2 + 2\hat{v}_i \hat{v}_j) \]

Right-multiplying \( (A.41) \) by \( V_{(DA)}' \frac{1}{n} 1_n \) and then left-multiplying by the \( \ell \)th row of \( V_{(DA)} \) yields the following closed-form expression for \( \hat{v}_i(t_1^j + 1) \),

\[ \hat{v}_i(t_1^j + 1) = \frac{1}{\det(V_{(DA)}')}(||\hat{v}_i||^2||\hat{v}_j|| - (\hat{v}_i' \hat{v}_j)^2 + 2\hat{v}_i \hat{v}_j) \]

Taking \( \ell = i \) in (A.42) then yields (A.83), see [87] for details.

Lemma A.2.8 (DA Lower Bound On Signal Reduction in Error) Upon reception of any signal \( S^{ij}(t_0^j, t_1^j) \), the decrease in the updated normal consensus squared error has the following lower bound:

\[ E^2(\hat{v}_i(t_1^j)) - E^2(\hat{v}_i(t_1^j + 1)) \geq \max\left\{ \begin{array}{c} \|\hat{v}_j(t_0^j)\|^2 - \|\hat{v}_i(t_1^j)\|^2, \\ \|\hat{v}_j(t_0^j)\|^2 - \|\hat{v}_j(t_1^j)\|^2 \end{array} \right\} \]

(4.33)
A.2. Proof of Results Contained in Chapter 4.

Proof. By Lemma A.2.3 we can apply (A.30) to the left-hand side of (A.43),

\[ E^2(\hat{v}_i(t_{i}^{ij})) - E^2(\hat{v}_i(t_{i}^{ij} + 1)) = ||\hat{v}_i(t_{i}^{ij} + 1)||^2 - ||\hat{v}_i(t_{i}^{ij})||^2. \]  
(A.44)

Applying the first line of (A.32) to the RHS of (A.44) then yields,

\[ E^2(\hat{v}_i(t_{i}^{ij})) - E^2(\hat{v}_i(t_{i}^{ij} + 1)) \]
\[ \geq \max\{||\hat{v}_i(t_{i}^{ij})||^2 , ||\hat{v}_j(t_{0}^{ij})||^2\} - ||\hat{v}_i(t_{i}^{ij})||^2 , \]
\[ = \max\{0 , ||\hat{v}_j(t_{0}^{ij})||^2 - ||\hat{v}_i(t_{i}^{ij})||^2\} , \]
\[ \geq ||\hat{v}_j(t_{0}^{ij})||^2 - ||\hat{v}_i(t_{i}^{ij})||^2 . \]  
(A.45)

Next observe that for any two vectors \( \hat{v}_i, \hat{v}_j \) in \( \mathbb{R}^{n \times 1} \),

\[ \text{span} \{ \hat{v}_i, \hat{v}_j \} \subseteq \text{span} \{ \hat{v}_i, \hat{v}_j, e_1 \} . \]

It thus follows that,

\[ E^2(\arg \min_{\tilde{v}} \in \text{span}\{\hat{v}_i, \hat{v}_j, e_1\} \ ||\tilde{v} - \frac{1}{n} \mathbf{1}_n||^2 ) \]
\[ \leq E^2(\arg \min_{\tilde{v}} \in \text{span} \{ \hat{v}_i, \hat{v}_j \} \ ||\tilde{v} - \frac{1}{n} \mathbf{1}_n||^2 ) . \]  
(A.46)

Let \( \hat{v}_i = \hat{v}_i(t_{i}^{ij}) \), \( \hat{v}_j = \hat{v}_j(t_{0}^{ij}) \) and \( \hat{v}_i(t_{i}^{ij} + 1) = \hat{v}_i(+) \) for notational convenience. From (A.46) we have,

\[ E^2(\hat{v}_i) - E^2(\hat{v}_i(+)) \]
\[ \geq ||\arg \min_{\tilde{v}} \in \text{span}\{\hat{v}_i, \hat{v}_j\} \ ||\tilde{v} - \frac{1}{n} \mathbf{1}_n||^2 || - ||\hat{v}_i||^2 , \]
\[ = ||\tilde{w}||^2 - ||\hat{v}_i||^2 = \frac{1}{n} (\tilde{w}' \mathbf{1}_n - \hat{v}_i' \mathbf{1}_n) , \]  
(A.47)

where the last equality holds due to Lemma A.2.3, and \( \tilde{w} \) is defined,

\[ \tilde{w} = \arg \min_{\tilde{v}} \in \text{span}\{\hat{v}_i, \hat{v}_j\} \ ||\tilde{v} - \frac{1}{n} \mathbf{1}_n||^2 , \]
\[ = \arg \min_{\tilde{v}} \in \{ a\hat{v}_i + b\hat{v}_j : (a, b) \in \mathbb{R}^2 \} \ ||\tilde{v} - \frac{1}{n} \mathbf{1}_n||^2 . \]  
(A.48)

Note that Lemma A.2.4 together with the initialization (4.15) implies \( \hat{v}_i \) and \( \hat{v}_j \) are non-zero. Next assume that \( \hat{v}_i \) and \( \hat{v}_j \) are linearly dependent. In this case (A.48) reduces to,

\[ \tilde{w} = \arg \min_{\tilde{v}} \in \text{span}\{\hat{v}_i\} \ ||\tilde{v} - \frac{1}{n} \mathbf{1}_n||^2 , \]
\[ = V(V'V)^{-1}V' \frac{1}{n} \mathbf{1}_n , \]
\[ = \frac{\hat{v}_i}{n ||\hat{v}_i||^2} = \hat{v}_i , \]  
(A.49)

95
A.2. Proof of Results Contained in Chapter 4.

where the last equality follows due to Lemma A.2.3. Applying (A.49) to (A.47) implies,

\[ E^2(\hat{v}_i) - E^2(\hat{v}_i(+)) \geq ||\hat{w}||^2 - ||\hat{v}_i||^2 \]
\[ = ||\hat{v}_i||^2 - ||\hat{v}_i||^2 = 0 = n(\hat{v}_j(0) - \hat{v}_j')^2 \]

where the last equality follows by Lemma A.2.5 since Lemma A.2.3 implies both \( \hat{v}_i \) and \( \hat{v}_j \) satisfy (A.80), and we are assuming \( \hat{v}_i, \hat{v}_j \) are linearly dependent.

Next assume \( \hat{v}_i \) and \( \hat{v}_j \) are linearly independent. In this case (A.48) can be solved analogous to (A.39) based on the optimization problem (A.34),

\[
\hat{w} = \arg \min_{\mathcal{V}} \{a\hat{v}_i + b\hat{v}_j : (a, b) \in \mathbb{R}^2\} ||\hat{v} - \frac{1}{n}\hat{v}_n||^2, \\
\]

Substituting the second line of (A.51) for \( \hat{w} \) in the third line of (A.47) then yields,

\[
\frac{1}{n}(\hat{w}'\hat{v}_n - \hat{v}'_n\hat{1}_n) = \frac{1}{n}(||\hat{v}_j||^2\hat{v}'_j(\hat{v}_i - \hat{v}_i)\hat{v}'_n + ||\hat{v}_j||^2\hat{v}'_j(\hat{v}_i - \hat{v}_i)\hat{v}'_n - \hat{v}'_n\hat{1}_n) \\
= ||\hat{v}_j||^2||\hat{v}_j||^2(\hat{v}_i - \hat{v}_i) + ||\hat{v}_j||^2||\hat{v}_j||^2(\hat{v}_i - \hat{v}_i - (||\hat{v}_j||^2||\hat{v}_j||^2(||\hat{v}_j||^2||\hat{v}_j||^2))||\hat{v}_j||^2 \\
= ||\hat{v}_j||^2||\hat{v}_j||^2(\hat{v}_i - \hat{v}_i) + ||\hat{v}_j||^2||\hat{v}_j||^2(\hat{v}_i - \hat{v}_i) \\
\geq (||\hat{v}_j||^2 - 2\hat{v}'_j\hat{v}_j||\hat{v}_j||^2 + (\hat{v}'_j\hat{v}_j)^2 ||\hat{v}_j||^2 \\
(\hat{v}_j - \hat{v}_i)^2)^2 \\
= n(||\hat{v}_j||^2 - (\hat{v}_j')^2), \\
\]

where the last inequality follows since \( E^2(\hat{v}_j) \geq 0 \) implies \( ||\hat{v}_j||^2 \leq \frac{1}{n} \). Combining (A.47) and (A.52) implies,

\[ E^2(\hat{v}_i) - E^2(\hat{v}_i(+)) \geq n(||\hat{v}_j||^2 - (\hat{v}_j')^2) \]

Together (A.45), (A.50) and (A.53) imply (A.43).

\[ \text{Lemma A.2.9} \ (DA \ Non-Decreasing \ Magnitude \ for \ any \ Communication \ Time-Respecting Path) \ Any \ time-respecting \ communication \ path \ \ C_{[t_0(ij), t_1(ij)]} \ implies, \]

\[ ||\hat{v}_i(t_1(ij) + 1)||^2 \geq ||\hat{v}_j(t_0(ijj))||^2. \]

\[ \text{Proof.} \ This \ result \ follows \ directly \ from (A.87), \ Lemma A.2.4, \ and \ the \ definition \ of \ a \ time-recting \ communication \ path \ (A.43). \ See [87] \ for \ details. \]

\[ \text{Lemma A.2.10} \ (Error \ Expression \ for \ C_{[0, \infty]} \ satisfying \ (A.43)) \ For \ any \ communication \ sequence \ C_{[0, \infty]} \ satisfying \ (A.43) \ the \ total \ reduction \ in \ normal \ consensus \ squared \ error}
A.2. Proof of Results Contained in Chapter 4.

\( \tilde{E}^2(C_{[0,\infty)}) \) defined in (A.20) is,

\[
\tilde{E}^2(C_{[0,\infty)}) = \lim_{t \to \infty} \sum_{i=1}^{n} (E^2(\hat{v}_i(0)) - E^2(\hat{v}_i(t))) \\
= \frac{n-1}{n} - \lim_{t \to \infty} \sum_{i=1}^{n} E^2(\hat{v}_i(t)) \\
= \sum_{\ell \in \mathbb{N}} \tilde{E}^2(C_{[t_0^\ell,t_1^\ell]}), \quad C_{[t_0^\ell,t_1^\ell]} \in \text{SVSC}, \quad t_1^\ell < t_0^{\ell+1} \\
\leq \frac{n-1}{n}. \tag{A.54}
\]

Proof. The first line follows from (A.20) – (A.21). The second line in (A.54) is due to the initialization (4.15) and (A.19). The third line in (A.54) follows since any communication sequence \( C_{[0,\infty)} \) satisfying (4.4.3) can be partitioned into an infinite number of disjoint sequences \( C_{[t_0^\ell,t_1^\ell]} \in \text{SVSC} \). The last line in (A.54) follows since the minimum error of any normal consensus estimate is 0.

Lemma A.2.11 (Vanishing Reduction in Error for \( C_{[0,\infty]} \) satisfying (4.4.3)) For any communication sequence \( C_{[0,\infty]} \) satisfying (4.4.3) there exists an integer \( \ell_\varepsilon \) such that,

\[
\tilde{E}^2(C_{[t_0^\ell,t_1^\ell]}) \leq \varepsilon, \quad \forall \quad \ell \geq \ell_\varepsilon, \tag{A.55}
\]

for any \( \varepsilon > 0 \), where \( \tilde{E}^2(C_{[t_0^\ell,t_1^\ell]}) \) is defined by (A.20).

Proof. The third line of (A.54) implies that for any \( C_{[0,\infty]} \) satisfying (4.4.3) the quantity \( \tilde{E}^2(C_{[0,\infty)}) \) is the sum of an infinite number of non-negative terms \( \tilde{E}^2(C_{[t_0^\ell,t_1^\ell]}) \), the fourth line in (A.54) implies \( \tilde{E}^2(C_{[0,\infty)}) \) is bounded above, thus (A.55) follows by the monotone convergence theorem for sequences of real numbers.

Lemma A.2.12 (DA Lower Bound on Reduction in Error for any \( C_{[t_0,t_1]} \) satisfying (4.4.2)) Any communication sequence \( C_{[t_0,t_1]} \) satisfying (4.4.2) implies,

\[
\tilde{E}^2(C_{[t_0,t_1]}) \geq n \left( \min_{i \in V} \{ ||\hat{v}_i(t_1 + 1)||^2 \} - \max_{i \in V} \{ ||\hat{v}_i(t_0)||^2 \} \right) \geq 0. \tag{A.56}
\]

Proof. The first inequality in (A.56) holds for any communication sequence \( C_{[t_0,t_1]} \),

\[
\tilde{E}^2(C_{[t_0,t_1]}) = \sum_{i=1}^{n} \left( E^2(\hat{v}_i(t_0)) - E^2(\hat{v}_i(t_1 + 1)) \right) \\
= \sum_{i=1}^{n} \left( ||\hat{v}_i(t_1 + 1)||^2 - ||\hat{v}_i(t_0)||^2 \right) \\
\geq n \left( \min_{i \in V} \{ ||\hat{v}_i(t_1 + 1)||^2 \} - \max_{i \in V} \{ ||\hat{v}_i(t_0)||^2 \} \right).
\]

97
To prove the second inequality in (A.56) it is required that \( C_{[t_0,t_1]} \) satisfies (4.4.2). Let us define,

\[
\ell = \arg \min_{i \in V} \{ ||\hat{v}_i(t_1 + 1) ||^2 \},
\]

\[
\bar{\ell} = \arg \max_{i \in V} \{ ||\hat{v}_i(t_0) ||^2 \}.
\]

Since \( C_{[t_0,t_1]} \) satisfies (4.4.2) there exists a time-respecting communication path between any two nodes in \( V \), in particular we have \( C_{[t_0(t_1)],[t_1(t_0)]} \) \( \in C_{[t_0,t_1]} \). The Lemma A.2.9 then implies,

\[
||\hat{v}_\ell(t_1(t_0) + 1)||^2 \geq ||\hat{v}_\ell(t_0(t_1))||^2.
\]

The second inequality in (A.56) then follows because,

\[
\min_{i \in V} \{ ||\hat{v}_i(t_1 + 1) ||^2 \} = ||\hat{v}_\ell(t_1 + 1)||^2 \\
\geq ||\hat{v}_\ell(t_1(t_0) + 1)||^2 \geq ||\hat{v}_\ell(t_0(t_1))||^2 \\
\geq ||\hat{v}_\ell(t_0)||^2 = \max_{i \in V} \{ ||\hat{v}_i(t_0) ||^2 \},
\]

where the first and third inequality hold due to Lemma A.2.4 because \( t_1 \geq t_1(t_0) \) and \( t_0(t_1) \geq t_0 \) respectively.

**Lemma A.2.13 (DA Properties of the Normal Consensus Update)** Upon reception of a signal \( S_{ij}(t_{ij}^0, t_{ij}^1) \), the normal consensus estimate \( \hat{v}_i(t_{ij}^1 + 1) \) that results from the update (A.87) will satisfy,

\[
\hat{v}_i(t_{ij}^1)'(\hat{v}_i(t_{ij}^1) - \hat{v}_i(t_{ij}^1 + 1)) \leq 0. \tag{A.57}
\]

**Proof.** Let us define \( \tilde{w} \),

\[
\tilde{w} = \arg \min_{\tilde{v} \in \text{span}\{\hat{v}_i(t_{ij}^1 + 1), \hat{v}_i(t_{ij}^1), e_i\}} ||\tilde{v} - \frac{1}{n} \mathbf{1}_n||^2, \tag{A.58}
\]

where \( \hat{v}_i(t_{ij}^1 + 1) \) is given by (A.87). Observe that \( \hat{v}_i(t_{ij}^1 + 1) \in \text{span}\{\hat{v}_i(t_{ij}^1), \hat{v}_j(t_{ij}^0), e_i\} \) implies,

\[
\text{span}\{\hat{v}_i(t_{ij}^1 + 1), \hat{v}_i(t_{ij}^1), e_i\} \subseteq \text{span}\{\hat{v}_i(t_{ij}^1), \hat{v}_j(t_{ij}^0), e_i\}. \tag{A.59}
\]

From (A.59) we have,

\[
E^2(\tilde{w}) \geq E^2(\hat{v}_i(t_{ij}^1 + 1)), \tag{A.60}
\]

and thus combining (A.60) and (A.30) implies,

\[
||\tilde{w}||^2 \leq ||\hat{v}_i(t_{ij}^1 + 1)||^2. \tag{A.61}
\]
Next observe that since \( \bar{w} \) is defined by (A.58), if (A.61) is applied to the hypothetical (and by (A.87) necessarily redundant) signal \( S^{ii}(t_{1}^{ij}, t_{1}^{ij} + 1) \), then,

\[
\hat{E}^2(S^{ii}(t_{1}^{ij}, t_{1}^{ij} + 1)) = E^2(\hat{v}_i(t_{1}^{ij} + 1)) - E^2(\bar{w}),
\]

\[= ||\bar{w}||^2 - ||\hat{v}_i(t_{1}^{ij} + 1)||^2 \leq 0. \tag{A.62} \]

Applying Lemma A.2.8 to the signal \( S^{ii}(t_{1}^{ij}, t_{1}^{ij} + 1) \) then implies,

\[
0 \geq \hat{E}^2(S^{ii}(t_{1}^{ij}, t_{1}^{ij} + 1)) \\
\geq \max\{||\hat{v}_i(t_{1}^{ij})||^2 - ||\hat{v}_i(t_{1}^{ij} + 1)||^2, \ n(\hat{v}_i(t_{1}^{ij}) - \hat{v}_i(t_{1}^{ij} + 1))^2 \} \tag{A.63} \]

where the first line follows from (A.62) and the last line implies (A.57).

**Lemma A.2.14 (DA Local Convergence of Normal Consensus Estimates)** For any communication sequence \( C_{[0,\infty]} \) satisfying (4.4.3) there exists an integer \( \ell_\chi \) such that for all \( \ell \geq \ell_\chi \),

\[
||\hat{v}_i(t_{1}^{ij}) - \hat{v}_j(t_{0}^{ij})||^2 \leq \chi, \ \forall S^{ij}(t_{0}^{ij}, t_{1}^{ij}) \in C_{[t_0^{ij}, t_1^{ij}]} \tag{A.64} \]

for any \( \chi > 0. \]

**Proof.** For any communication sequence \( C_{[0,\infty]} \) satisfying (4.4.3) the Lemma A.2.11 implies there exists an integer \( \ell_\varepsilon \) such that (A.55) holds for any \( \varepsilon > 0. \) For all \( \ell \geq \ell_\varepsilon \) we thus have for any signal \( S^{ij}(t_{0}^{ij}, t_{1}^{ij}) \in C_{[t_0^{ij}, t_1^{ij}]} \),

\[
||\hat{v}_i(t_{1}^{ij})||^2 - ||\hat{v}_j(t_{0}^{ij})||^2 \leq ||\hat{v}_i(t_{1}^{\ell}) + 1||^2 - ||\hat{v}_j(t_{0}^{\ell})||^2, \]

\[
\leq \sum_{r=1}^{n} (||\hat{v}_r(t_{1}^{\ell} + 1)||^2 - ||\hat{v}_r(t_{0}^{\ell})||^2), \tag{A.65} \]

\[
= \hat{E}^2(C_{[t_0^{ij}, t_1^{ij}]}) \leq \varepsilon. \]

The first inequality in (A.65) holds by Lemma A.2.4 since \( t_{0}^{ij} \geq t_{0}^{\ell} \) and \( t_{1}^{ij} \leq t_{1}^{\ell} \) for any \( S^{ij}(t_{0}^{ij}, t_{1}^{ij}) \in C_{[t_0^{ij}, t_1^{ij}]} \). The second inequality in (A.65) holds since,

\[
\sum_{r=1}^{n} (||\hat{v}_r(t_{1}^{\ell} + 1)||^2 - ||\hat{v}_r(t_{0}^{\ell})||^2) \\
= (||\hat{v}_i(t_{1}^{\ell} + 1)||^2 - ||\hat{v}_i(t_{0}^{\ell})||^2) + (||\hat{v}_j(t_{1}^{\ell} + 1)||^2 - ||\hat{v}_j(t_{0}^{\ell})||^2) \tag{A.66} \\
+ \sum_{r \in \mathcal{V} \setminus \{i,j\}} (||\hat{v}_r(t_{1}^{\ell} + 1)||^2 - ||\hat{v}_r(t_{0}^{\ell})||^2),
\]

where

\[
\sum_{r \in \mathcal{V} \setminus \{i,j\}} (||\hat{v}_r(t_{1}^{\ell} + 1)||^2 - ||\hat{v}_r(t_{0}^{\ell})||^2) \geq 0 \tag{A.67}
\]
holds due to Lemma A.2.4, and similarly \(||\hat{v}_j(t^i_1 + 1)||^2 - ||\hat{v}_i(t^i_0)||^2 \geq 0\) because,

\[
||\hat{v}_j(t^i_1 + 1)||^2 - ||\hat{v}_i(t^i_0)||^2 \geq \min_{r \in V} \{||\hat{v}_r(t^i_1 + 1)||^2\} - \max_{r \in V} \{||\hat{v}_r(t^i_0)||^2\} \geq 0 . \tag{A.68}
\]

The second inequality in (A.68) holds due to Lemma A.2.12 since \(C_{[t^i_0, t^i_1]} \in \text{SYSC}\). Together (A.66), (A.67) and (A.68) imply the second inequality in (A.65). Applying Lemma A.2.8 to Lemma A.2.11 implies that for all \(S^{ij}(t^i_0, t^i_0) \in C_{[t^i_0, t^i_1]}\) and \(\ell \geq \ell_\varepsilon\),

\[
\varepsilon \geq E^2(C_{[t^i_0, t^i_1]}) \geq E^2(S^{ij}(t^i_0, t^i_0)) ,
\]

\[
= E^2(\hat{v}_i(t^i_1)) - E^2(\hat{v}_i(t^i_1 + 1)) ,
\]

\[
\geq \max \{||\hat{v}_j(t^i_0)||^2 - ||\hat{v}_i(t^i_0)||^2 ,
\]

\[
\sum_{i,j} \{\hat{v}_j(t^i_0)'(\hat{v}_j(t^i_0) - \hat{v}_i(t^i_0))\}^2 ,
\]

for any \(\varepsilon > 0\). For notational convenience denote \(\hat{v}_i = \hat{v}_i(t^i_1)\) and \(\hat{v}_j = \hat{v}_j(t^i_0)\). Combining (A.65) and (A.69) implies that for any \(\varepsilon > 0\) there exist an integer \(\ell_\varepsilon\) such that,

\[
||\hat{v}_i||^2 - ||\hat{v}_j||^2 \leq \varepsilon , \quad \hat{v}_j(\hat{v}_j - \hat{v}_i) \leq \sqrt{\varepsilon/n} , \tag{A.70}
\]

for any \(S^{ij}(t^i_0, t^i_1) \in C_{[t^i_0, t^i_1]}\) and \(\ell \geq \ell_\varepsilon\). To obtain (A.64) we observe that (A.70) implies,

\[
||\hat{v}_i - \hat{v}_j||^2 = ||\hat{v}_i||^2 - 2\hat{v}_j \hat{v}_j + ||\hat{v}_j||^2 = ||\hat{v}_i||^2 - F||\hat{v}_j||^2 + 2\hat{v}_j(\hat{v}_j - \hat{v}_i) ,
\]

\[
\leq \varepsilon + 2\sqrt{\varepsilon/n} \leq \sqrt{\varepsilon(1 + 2\sqrt{m})} , \quad \forall \varepsilon \in (0, 1] . \tag{A.71}
\]

We thus define \(\varepsilon(\chi)\),

\[
\varepsilon(\chi) = \left(\frac{\chi}{1 + 2\sqrt{m}}\right)^2 . \tag{A.72}
\]

For any \(\chi \in (0, 1]\) and \(\varepsilon \in (0, \varepsilon(\chi)]\) the result (A.64) then follows from (A.71).

\[\square\]

**Lemma A.2.15 (DA Vanishing Change in Normal Consensus Update)** For any communication sequence \(C_{[0, \infty]}\) satisfying (4.4.3) there exists an integer \(\ell_\varepsilon\) such that for all \(\ell \geq \ell_\varepsilon\),

\[
||\hat{v}_i(t^i_1 + 1) - \hat{v}_i(t^i_1)||^2 \leq \varepsilon , \quad \forall S^{ij}(t^i_0, t^i_0) \in C_{[t^i_0, t^i_1]} , \tag{A.73}
\]

for any \(\varepsilon > 0\).

\[\square\]

**Proof.** Recall that Lemma A.2.11 implies that for any communication sequence \(C_{[0, \infty]}\) satisfying (4.4.3) there exists an integer \(\ell_\varepsilon\) such that (A.55) holds for any \(\varepsilon > 0\), we thus
observe for $\ell \geq \ell_\varepsilon$,
\[
\varepsilon \geq \tilde{E}^2(C_{t_0^\ell,t_1^\ell}) \\
\geq \tilde{E}^2(S^{ij}(t_0^{ij}, t_1^{ij})) , \quad \forall \ S^{ij}(t_0^{ij}, t_1^{ij}) \in C_{t_0^\ell,t_1^\ell} \\
= ||\hat{v}_i(t_1^{ij} + 1)||^2 - ||\hat{v}_i(t_1^{ij})||^2.
\]
For all $\ell \geq \ell_\varepsilon$ and signals $S^{ij}(t_0^{ij}, t_1^{ij}) \in C_{t_0^\ell,t_1^\ell}$ we then have,
\[
||\hat{v}_i(t_1^{ij} + 1) - \hat{v}_j(t_1^{ij})||^2 = ||\hat{v}_i(t_1^{ij} + 1)||^2 - ||\hat{v}_i(t_1^{ij})||^2 \\
+ 2\hat{v}_i(t_1^{ij})'(\hat{v}_i(t_1^{ij}) - \hat{v}_j(t_1^{ij} + 1)) \\
\leq \tilde{E}^2(S^{ij}(t_0^{ij}, t_1^{ij})) \leq \varepsilon,
\]
where the first inequality follows from Lemma A.2.13 and the second inequality from Lemma A.2.15.

**Lemma A.2.16 (DA Vanishing Change between Normal Consensus Update and Signal)** For any communication sequence $C_{0,\infty}$ satisfying (4.4.3), there exists an integer $\ell_\gamma$ such that,
\[
||\hat{v}_i(t_1^{ij} + 1) - \hat{v}_j(t_1^{ij})||^2 \leq \gamma , \quad \forall \ S^{ij}(t_0^{ij}, t_1^{ij}) \in C_{t_0^\ell,t_1^\ell},
\]
for all $\ell \geq \ell_\gamma$ and any $\gamma > 0$.

**Proof.** For any communication sequence $C_{0,\infty}$ satisfying (4.4.3), Lemma A.2.14 implies that (A.64) holds for any $\chi \in (0,1)$ and $\varepsilon \in (0,\bar{\varepsilon}(\chi)]$, where $\bar{\varepsilon}(\chi)$ is given by (A.72). Lemma A.2.15 implies that (A.73) holds for any $\varepsilon > 0$, thus for any $\ell \geq \ell_\varepsilon(\chi)$ and $S^{ij}(t_0^{ij}, t_1^{ij}) \in C_{t_0^\ell,t_1^\ell}$ the triangle inequality then implies,
\[
||\hat{v}_i(t_1^{ij} + 1) - \hat{v}_j(t_0^{ij})|| \leq ||\hat{v}_i(t_1^{ij} + 1) - \hat{v}_j(t_1^{ij})|| \\
+ ||\hat{v}_i(t_1^{ij}) - \hat{v}_j(t_0^{ij})|| \\
\leq \sqrt{\bar{\varepsilon}(\chi) + \sqrt{\chi}} \leq 2\sqrt{\chi} , \quad \forall \ \chi \in (0,1].
\]
Any $\chi \in (0,\gamma/4]$ and $\varepsilon \in (0,\bar{\varepsilon}(\chi)]$ thus yields (A.74) for any $\gamma \in (0,4]$.

**Lemma A.2.17 (DA Network Convergence to Average-Consensus)** For any communication sequence $C_{0,\infty}$ satisfying (4.4.3) there exists an integer $\ell_\xi$ such that for all $\ell \geq \ell_\xi$,
\[
\sum_{i=1}^{n} ||\hat{v}_i(t_1^{\ell} + 1) - \frac{1}{n}1_n||^2 \leq \xi , \quad \forall \ C_{t_0^\ell,t_1^\ell} \in C_{[0,\infty]},
\]
for any $\xi > 0$. 

101
A.2. Proof of Results Contained in Chapter 4.

Proof. Since $C_{[0,\infty]}$ satisfies (4.4.3) we have $C^{ij}_{[t_0^\ell,t_1^\ell]} \in \text{SVSC}$ for each $\ell \in \mathbb{N}$, thus there exists a time-respecting communication path $C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]} \in C^{ij}_{[t_0^\ell,t_1^\ell]}$ for any $i \in \mathcal{V}, j \in \mathcal{V}_-,$ and $\ell \in \mathbb{N}$. For any $i \in \mathcal{V}, j \in \mathcal{V}_-$ and $\ell \in \mathbb{N}$ the triangle inequality then implies,

$$
\|\hat{v}_{ij}(t_1^\ell + 1) - \hat{v}_{jj}(t_1^\ell)\| \\
\leq \sum_{S^{\gamma q}(t_0^\ell,t_1^\ell) \in C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}} \|\hat{v}_{jr}(t_1^\gamma + 1) - \hat{v}_{jr}(t_1^\gamma)\| \\
+ \sum_{k(ij)+1} \sum_{S^{\gamma q}(t_0^\ell,t_1^\ell) \in Q^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}} \|\hat{v}_{jr}(t_1^\gamma + 1) - \hat{v}_{jr}(t_1^\gamma)\|
$$

(A.76)

where we define $C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]} = C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]} \setminus C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}$ and,

$$
Q^{ij}_{q}(ij) = \{S^{\gamma q}(t_0^\ell,t_1^\ell) \in C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]} : \\
\ell_1^q(t_0^\ell \in (t_1^\ell,q_{k(ij)+1}^\ell) \},
$$

for $q = 1,\ldots,k(ij)$, where $\ell_0 = j, \ell_{k(ij)+1} = i$,

$$
Q^{ij}_{q}(ij) = \{S^{\gamma q}(t_0^\ell,t_1^\ell) \in C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]} : t_1^q \in (t_1^\ell, t_{k(ij)+1}^\ell) \}.
$$

(A.77)

We clarify that the RHS of (A.76) includes the differences between the received normal consensus estimate $\hat{v}_{q_{k(ij)+1}}(t_1^\ell,t_1^\ell)$ and the updated normal consensus estimate $\hat{v}_{q_{k(ij)+1}}(t_1^\ell,t_1^\ell + 1)$ that results from each signal contained in the time-respecting communication path $(\text{trc}) C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]} \in C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}$. Each set $Q^{ij}_{q}(ij)$ defined in (A.77) contains the signals received at each node after the respective signal in the $(\text{trc}) C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}$ was received, but before the respective signal in the $(\text{trc}) C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}$ was sent, as is required for an application of the triangle inequality. The set $Q^{ij}_{q}(ij)$ contains the signals received at node $i$ after the last signal in the $(\text{trc}) C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}$ was received, but no later then the end of the communication sequence $C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}$, again this is required for application of the triangle inequality.

For any communication sequence $C_{[0,\infty]}$ satisfying (4.4.3) the Lemma A.2.15 implies there exists an integer $\ell_\varepsilon$ such that (A.73) holds for any $\varepsilon > 0$. Likewise, for any communication sequence $C_{[0,\infty]}$ satisfying (4.4.3) the Lemma A.2.16 implies there exists an integer $\ell_\gamma$ such that (A.74) holds for any $\gamma > 0$. Thus for any $\varepsilon \in (0,\gamma/4]$ and $\varepsilon \in (0,\varepsilon(\chi))$ then for any $\ell \geq \ell_\varepsilon(\chi)$,

$$
\|\hat{v}_{ij}(t_1^\ell + 1) - \hat{v}_{jj}(t_1^\ell)\| \\
\leq \|C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}\|_\gamma \\
+ \sum_{k(ij)+1} \|C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}\|_\gamma \\
\leq \|C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}\|_\gamma \\
\leq \|C^{ij}_{[t_0^\ell(t_0^\ell,t_1^\ell)]}\|_\gamma,
$$

(A.78)

where the second inequality holds since $0 < \varepsilon(\chi) < \gamma$, and the last inequality holds since every
signal contained in $C_{[t_0^\ell, t_1^\ell]}$ is represented by at most one term on the RHS of (A.76), that is,

$$|C_{[t_0^\ell, t_1^\ell]}| \geq |C_{[t_0(ij), t_1(ij)]}| + \sum_{q=1}^{k(ij)+1} |Q_q(ij)| .$$

Due to (A.78), any $\xi > 0$ and $\gamma \in (0, \xi/(n|C_{[t_0^\ell, t_1^\ell]}|)^2]$ then implies,

$$(\hat{v}_{ij}(t_1^\ell + 1) - \hat{v}_{jj}(t_0^\ell(j)))^2 = (\hat{v}_{ij}(t_1^\ell + 1) - 1/n)^2 \leq \xi/n^2 , \quad (A.79)$$

for any $(i, j) \in V^2$, where the first equality holds due to Lemma A.2.7. The result (A.75) then follows from (A.79) since,

$$\sum_{i=1}^{n} ||\hat{v}_i(t_1^\ell + 1) - 1/n, 1_n||^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} (\hat{v}_{ij}(t_1^\ell + 1) - \hat{v}_{jj}(t_0^\ell(j)))^2 \leq n^2(\xi/n^2) ,$$

where the first equality holds again due to Lemma A.2.7. ■

### A.2.2 Proof of Theorem 4.4.10

**Outline of Proof:** The proof of Theorem 4.4.10 can be divided into the following three main areas: (A) properties of the DDA normal consensus estimates, (B) finite time until the reduction in DDA normal consensus error vanishes, and (C) DDA normal consensus estimate properties given no reduction in error.

The Lemmas A.2.18-A.2.24 cover area (A). The key results are:

- **Lemma A.2.19** proves that each DDA normal consensus estimate $\hat{v_i}(t)$ satisfies the normalization property $||\hat{v_i}(t)||^2 = (1/n)\hat{v_i}(t)'1_n$.
- **Lemma A.2.20** proves that each DDA normal consensus estimate $\hat{v_i}(t)$ must satisfy the “zero local error” property $\hat{v_{ii}}(t) = 1/n$.
- **Lemma A.2.22** shows that the error of each normal consensus estimate is non-decreasing with time.
- **Lemma A.2.23** proves that the normal consensus estimate will not change unless the change reduces its error.

The Lemmas A.2.25-A.2.28 cover area (B). The key result is Lemma A.2.28, which shows that any IVCC communication sequence implies there will exist a finite time after which there will be no further reduction in the total normal consensus error. The Lemmas A.2.29 – A.2.30 cover
A.2. Proof of Results Contained in Chapter 4.

area (C), after which we present the proof of Thm. 4.4.10. Preceding the statement of each lemma we describe in words its intended purpose.

Lemma A.2.18 (DDA Normal Consensus Estimate Discretization) Every normal consensus estimate \( \hat{v}_i(t) \) satisfies \( \hat{v}_i(t) \in \{0, \frac{1}{n}\}^n \) for all \( i \in \mathcal{V} \) and \( t \geq 0 \).

Proof. The initialization (4.21) implies \( \hat{v}_i(0) \in \{0, \frac{1}{n}\}^n \). The optimization problem (4.4) requires that any solution \( \hat{v}_i(t + 1) \) satisfies \( \hat{v}_i(t + 1) \in \{0, \frac{1}{n}\}^n \). Under the DDA algorithm, the assumption (A7) implies every normal consensus estimate remains fixed unless it is updated via (4.4), thus it follows that \( \hat{v}_i(t) \in \{0, \frac{1}{n}\}^n \) for all \( i \in \mathcal{V} \) and \( t \geq 0 \).

Lemma A.2.19 (DDA Consensus Estimate Normalization) Every normal consensus estimate \( \hat{v}_i(t) \) satisfies,

\[
||\hat{v}_i(t)||^2 = \frac{1}{n} \hat{v}_i(t)'1_n , \quad \forall \ i \in \mathcal{V} , \quad \forall \ t \geq 0 ,
\]

(A.80)

\[
||\hat{v}_i^{-i}(t)||^2 = \frac{1}{n} \hat{v}_i^{-i}(t)'1_{n-1} , \quad \forall \ i \in \mathcal{V} , \quad \forall \ t \geq 0 .
\]

(A.81)

Proof. The Lemma A.2.18 implies (A.80) as follows,

\[
\frac{1}{n} \hat{v}_i(t)'1_n = \frac{1}{n} \sum_{\ell=1}^n \hat{v}_{i\ell}(t) ,
\]

\[
= \frac{1}{n} \sum_{\ell \in \mathcal{V}} : \hat{v}_{i\ell}(t) = \frac{1}{n} \left( \frac{1}{n} \right) ,
\]

\[
= \frac{1}{n} \sum_{\ell \in \mathcal{V}} : \hat{v}_{i\ell}(t) = \frac{1}{n} \left( \frac{1}{n^n} \right) ,
\]

\[
= \frac{1}{n} \sum_{\ell=1}^n \hat{v}_{i\ell}(t)^2 = ||\hat{v}_i(t)||^2 .
\]

(A.82)

If \( \hat{v}_i(t) \in \{0, \frac{1}{n}\}^n \) then \( \hat{v}_i^{-i}(t) \in \{0, \frac{1}{n}\}^{n-1} \), thus a similar argument to (A.82) implies (A.81).

Lemma A.2.20 (DDA Local Zero Error Property) Every normal consensus estimate \( \hat{v}_i(t) \) satisfies

\[
\hat{v}_{ii}(t) = \frac{1}{n} , \quad \forall \ i \in \mathcal{V} , \quad \forall \ t \geq 0 .
\]

(A.83)

Proof. The initialization (4.21) implies \( \hat{v}_{ii}(0) = \frac{1}{n} \) for each \( i \in \mathcal{V} \). Next observe that under (A7) the estimate \( \hat{v}_i(t) \) will not change unless a signal \( S^{ij}(t_{t_0}^{ij}, t_{t}^{ij}) \) is received at node \( i \). If a

\[\text{For brevity we do not show that (A4'), (4.4), and (4.5) together imply the DDA initialization (4.21). In lieu of a formal proof, note that under (A4') the initialization (4.21) can simply be assumed, regardless of (4.4) - (4.5). Similarly, it is shown in [87] that (4.4) - (4.5) imply (4.18) under the DDA knowledge set update (4.17) and signal specification (4.16). Under (4.16) - (4.17) one can simply assume the update (4.18) regardless of (4.4) - (4.5), and in fact (4.18) is a non-unique solution to (4.4) - (4.5) given (4.16) - (4.17).}

104
A.2. Proof of Results Contained in Chapter 4.

For any two normal consensus estimates \( \hat{v}_i(t) \) and \( \hat{v}_j(t) \), this update can be re-written as,

\[
\hat{v}_i(t_{1}^{ij}+1) = \begin{cases} 
\hat{v}_i(t_{1}^{ij}) + \hat{v}_j(t_{0}^{ij}) - \hat{v}_{ji}(t_{0}^{ij})e_i & , \text{if } \hat{v}_i(t_{1}^{ij}) - \hat{v}_j(t_{0}^{ij}) = 0 \\
\hat{v}_j(t_{0}^{ij}) + e_i\left(\frac{1}{n} - \hat{v}_{ji}(t_{0}^{ij})\right) & , \text{if } \hat{v}_i(t_{1}^{ij}) - \hat{v}_j(t_{0}^{ij}) > 0 , \|\hat{v}_i(t_{1}^{ij})\|^2 < \|\hat{v}_j(t_{0}^{ij})\|^2 \\
\hat{v}_i(t_{1}^{ij}) & , \text{if } \hat{v}_i(t_{1}^{ij}) - \hat{v}_j(t_{0}^{ij}) > 0 , \|\hat{v}_i(t_{1}^{ij})\|^2 \geq \|\hat{v}_j(t_{0}^{ij})\|^2 .
\end{cases}
\]

(A.84)

To finish the proof, it suffices to show that the solution \( \hat{v}_i(t_{1}^{ij} + 1) \) specified by (A.84) implies (A.83) under the assumption that \( \hat{v}_i(t_{1}^{ij}) \) satisfies (A.83). If \( \hat{v}_i(t_{1}^{ij}) - \hat{v}_j(t_{0}^{ij}) = 0 \) then \( \hat{v}_i(t_{1}^{ij} + 1) = \hat{v}_i(t_{1}^{ij}) + \hat{v}_j(t_{0}^{ij}) - \hat{v}_{ji}(t_{0}^{ij})e_i \) and thus,

\[
\hat{v}_{ii}(t_{1}^{ij} + 1) = \hat{v}_{ii}(t_{1}^{ij}) + \hat{v}_{ji}(t_{0}^{ij}) - \hat{v}_{ji}(t_{0}^{ij}) = \frac{1}{n} .
\]

If \( \hat{v}_i(t_{1}^{ij}) - \hat{v}_j(t_{0}^{ij}) > 0 \) and \( \|\hat{v}_i(t_{1}^{ij})\|^2 < \|\hat{v}_j(t_{0}^{ij})\|^2 \) then \( \hat{v}_i(t_{1}^{ij} + 1) = \hat{v}_j(t_{0}^{ij}) + e_i\left(\frac{1}{n} - \hat{v}_{ji}(t_{0}^{ij})\right) \) and thus,

\[
\hat{v}_{ii}(t_{1}^{ij} + 1) = \hat{v}_{ji}(t_{0}^{ij}) + \frac{1}{n} - \hat{v}_{ji}(t_{0}^{ij}) = \frac{1}{n} .
\]

Finally, if \( \hat{v}_i(t_{1}^{ij}) - \hat{v}_j(t_{0}^{ij}) > 0 \) and \( \|\hat{v}_i(t_{1}^{ij})\|^2 \geq \|\hat{v}_j(t_{0}^{ij})\|^2 \) then \( \hat{v}_i(t_{1}^{ij} + 1) = \hat{v}_i(t_{1}^{ij}) \) and thus (A.83) follows by the hypothesis.

\[\text{Lemma A.2.21 (DDA Normal Consensus Estimate Magnitude Equivalence to Error)}\]

For any two normal consensus estimates \( \hat{v}_i(t) \) and \( \hat{v}_j(t) \),

\[
\|\hat{v}_i(t)\|^2 \geq \|\hat{v}_j(t)\|^2 \iff E^2(\hat{v}_i(t)) \leq E^2(\hat{v}_j(t)) .
\]  

(A.85)

Proof. For any normal consensus estimate \( \hat{v}_i(t) \),

\[
E^2(\hat{v}_i(t)) = \|\hat{v}_i(t) - \frac{1}{n}1_n\|^2 ,
\]

\[
= \|\hat{v}_i(t)\|^2 - 2\frac{1}{n}\hat{v}_i(t)^T1_n + \frac{1}{n} = \frac{1}{n} - \|\hat{v}_i(t)\|^2
\]  

(A.86)

where the last equality holds due to Lemma A.2.19. The equivalence (A.85) then follows directly from (A.86).

\[\text{Lemma A.2.22 (DDA Normal Consensus Estimate Non-Decreasing Error)}\]

The error of every normal consensus estimate \( \hat{v}_i(t) \) is a non-decreasing function of \( t \geq 0 \) for all \( i \in \mathcal{V} \).

Proof. Under (A7) the normal consensus estimate \( \hat{v}_i(t_{1}^{ij}) \) will only change if a signal is received at node \( i \) at time \( t_{1}^{ij} \). If a signal is received at node \( i \), then the DDA protocol updates the normal consensus estimate \( \hat{v}_i(t_{1}^{ij}) \) by the optimization problem (4.4). Under (4.16) – (4.17)
we have for any knowledge set $K_i(t^{ij}_1)$ and signal $S^{ij}(t^{ij}_0, t^{ij}_1)$,

$$K_i(t^{ij}_1) \cup S^{ij}(t^{ij}_0, t^{ij}_1) = \{i, n, s_i(0), \hat{s}_i(t^{ij}_1), \hat{v}_i(t^{ij}_1), \hat{s}_j(t^{ij}_0), \hat{v}_j(t^{ij}_0)\},$$

where $s_i(0) = S e_i$, $\hat{s}_i(t^{ij}_1) = S \hat{v}_i(t^{ij}_1)$, $\hat{s}_j(t^{ij}_0) = S \hat{v}_j(t^{ij}_0)$.

It follows that $V_i = [e_i, \hat{v}_i(t^{ij}_1), \hat{v}_j(t^{ij}_0)]$ and $S_i = [s_i(0), \hat{s}_i(t^{ij}_1), \hat{s}_j(t^{ij}_0)]$. We can thus re-write (4.4) as,

$$\hat{v}_i(t^{ij}_1 + 1) = \arg \min_{\hat{v} \in \text{span}\{\hat{v}_i(t^{ij}_1), \hat{v}_j(t^{ij}_0), e_i\} \cap \{0, \frac{1}{n}\}^n} ||\hat{v} - \frac{1}{n}1_n||^2. \quad (A.87)$$

The Lemma A.2.18 implies $\hat{v}_i(t^{ij}_1) \in \{0, \frac{1}{n}\}^n$, thus (A.87) implies that any candidate solution $\hat{v}(1)(t^{ij}_1 + 1)$ that does not satisfy $E^2(\hat{v}(1)(t^{ij}_1 + 1)) \leq E^2(\hat{v}(t^{ij}_1))$ cannot be a solution to (A.87). $\blacksquare$

**Lemma A.2.23 (DDA Fixed Normal Consensus Estimate Given No Reduction in Error)** Under the DDA algorithm, for any signal $S^{ij}(t^{ij}_0, t^{ij}_1)$ we have,

$$E^2(\hat{v}_i(t^{ij}_1 + 1)) = E^2(\hat{v}_i(t^{ij}_1)) \iff \hat{v}_i(t^{ij}_1 + 1) = \hat{v}_i(t^{ij}_1). \quad (A.88)$$

Proof. It is clear from (A.19) that $\hat{v}_i(t^{ij}_1 + 1) = \hat{v}_i(t^{ij}_1)$ implies $E^2(\hat{v}_i(t^{ij}_1 + 1)) = E^2(\hat{v}_i(t^{ij}_1))$. Next observe that (A.86) implies,

$$E^2(\hat{v}_i(t^{ij}_1 + 1)) = E^2(\hat{v}_i(t^{ij}_1))$$

$$\iff ||\hat{v}_i(t^{ij}_1 + 1)||^2 = ||\hat{v}_i(t^{ij}_1)||^2.$$

To finish the proof we thus have only to show,

$$\hat{v}_i(t^{ij}_1 + 1) \neq \hat{v}_i(t^{ij}_1) \Rightarrow ||\hat{v}_i(t^{ij}_1 + 1)||^2 \neq ||\hat{v}_i(t^{ij}_1)||^2. \quad (A.89)$$

Under the update (A.84), to prove (A.89) it suffices to show that either of the two cases,

$$\hat{v}_i(t^{ij}_1 + 1) = \hat{v}_i(t^{ij}_1) + \hat{v}_j(t^{ij}_0) - \hat{v}_j(t^{ij}_0)e_i,$$

if $\hat{v}_i^{-1}(t^{ij}_1)\hat{v}_j^{-1}(t^{ij}_0) = 0$,

$$\hat{v}_i(t^{ij}_1 + 1) = \hat{v}_j(t^{ij}_0) + e_i\left(\frac{1}{n} - \hat{v}_j(t^{ij}_0)\right),$$

if $\hat{v}_i^{-1}(t^{ij}_1)\hat{v}_j^{-1}(t^{ij}_0) > 0$, $||\hat{v}_i^{-1}(t^{ij}_1)||^2 < ||\hat{v}_j^{-1}(t^{ij}_0)||^2$

will imply $||\hat{v}_i(t^{ij}_1 + 1)||^2 > ||\hat{v}_i(t^{ij}_1)||^2$. If $\hat{v}_i^{-1}(t^{ij}_1)\hat{v}_j^{-1}(t^{ij}_0) = 0$, then (A.90) together with (A.80)
A.2. Proof of Results Contained in Chapter 4.

imply,

\[ ||\tilde{v}_i(t_{1}^{ij} + 1)||^2 = \frac{1}{n} \tilde{v}_i(t_{1}^{ij} + 1)'e_1 \]

where the last inequality holds because Lemma A.2.20 implies \( \hat{v}_j^{-i}(t) \mathbf{1}_{n-1} \geq \frac{1}{n} \) for all \( j \in \mathcal{V}_i \) and \( t \geq 0 \). If \( \hat{v}_i^{-i}(t_{1}^{ij})'\hat{v}_j^{-i}(t_{0}^{ij}) > 0 \) and \( ||\tilde{v}_i^{-i}(t_{1}^{ij})||^2 < ||\hat{v}_j^{-i}(t_{0}^{ij})||^2 \), then from (A.90) and (A.80) it follows that,

\[ ||\tilde{v}_i(t_{1}^{ij} + 1)||^2 = \frac{1}{n} \tilde{v}_i(t_{1}^{ij} + 1)'e_1 \]

Lemma A.2.24 (DDA Lower Bound on Increase in Normal Consensus Magnitude)

Under the DDA algorithm, for any signal \( S^{ij}(t_{0}^{ij},t_{1}^{ij}) \) we have,

\[ ||\tilde{v}_i^{-i}(t_{1}^{ij} + 1)||^2 \geq \max\{||\hat{v}_i^{-i}(t_{1}^{ij})||^2 , ||\hat{v}_j^{-i}(t_{0}^{ij})||^2 \} \]

Proof. If \( \tilde{v}_i^{-i}(t_{1}^{ij})'\tilde{v}_j^{-i}(t_{0}^{ij}) = 0 \) then \( \hat{v}_i(t_{1}^{ij} + 1) = \hat{v}_i(t_{1}^{ij}) + \hat{v}_j(t_{0}^{ij}) - \hat{v}_{ji}(t_{0}^{ij})e_i \). In this case we apply (A.81) to obtain,

\[ ||\tilde{v}_i^{-i}(t_{1}^{ij} + 1)||^2 = \frac{1}{n} \tilde{v}_i^{-i}(t_{1}^{ij} + 1)'e_1 \]

If \( \tilde{v}_i^{-i}(t_{1}^{ij})'\tilde{v}_j^{-i}(t_{0}^{ij}) > 0 \) and \( ||\tilde{v}_i^{-i}(t_{1}^{ij})||^2 < ||\hat{v}_j^{-i}(t_{0}^{ij})||^2 \), then \( \hat{v}_i(t_{1}^{ij} + 1) = \hat{v}_j(t_{0}^{ij}) + e_i(\frac{1}{n} - \hat{v}_{ji}(t_{0}^{ij})) \).

In this case we have,

\[ ||\tilde{v}_i^{-i}(t_{1}^{ij} + 1)||^2 = \frac{1}{n} \tilde{v}_i^{-i}(t_{0}^{ij})'e_1 \geq \max\{||\tilde{v}_i^{-i}(t_{0}^{ij})||^2 , ||\hat{v}_j^{-i}(t_{0}^{ij})||^2 \} \]

where the last inequality holds by the assumption \( ||\tilde{v}_i^{-i}(t_{1}^{ij})||^2 < ||\hat{v}_j^{-i}(t_{0}^{ij})||^2 \). Finally, if \( \tilde{v}_i^{-i}(t_{1}^{ij})'\tilde{v}_j^{-i}(t_{0}^{ij}) > 0 \) and
Lemma A.2.26 (DDA Vanishing Reduction in Error for \(C_{[0,\infty]} \in \mathcal{IVCC}\)) For any communication sequence \(C_{[0,\infty]} \in \mathcal{IVCC}\) there exists an integer \(\ell \leq \ell_c\) such that, \[
abla E^2(C_{[0,\infty]}) \leq \varepsilon , \forall \ell \geq \ell_c , \forall \varepsilon > 0 , \quad C_{[t_0, t_1]} \in \mathcal{SVCC}. \]

Proof. The third line of (91) implies that for any \(C_{[0,\infty]} \in \mathcal{IVCC}\) the quantity \(\nabla E^2(C_{[0,\infty]})\) is the sum of an infinite number of non-negative terms \(\nabla E^2(C_{[t_0, t_1]}),\) where \(C_{[t_0, t_1]} \in \mathcal{SVCC}\) for each \(\ell \in \mathbb{N}\). The fourth line in (91) implies \(\nabla E^2(C_{[0,\infty]})\) is bounded above, thus the result follows by the monotone convergence theorem for sequences of real numbers.

Lemma A.2.27 (DDA Lower Bound on Non-Zero Reduction in Error) For any communication sequence \(C_{[t_0, t_1]}\) we have, \[
abla E^2(C_{[t_0, t_1]}) > 0 \Rightarrow \nabla E^2(C_{[t_0, t_1]}) \geq \frac{1}{n^2} . \]  

Proof. Applying Lemma A.2.21 to the left-hand side of (92) implies there exists a signal \(S^{ij}(t_0^i, t_1^j) \in C_{[t_0, t_1]}\) such that, \[
abla ||\hat{v}_i(t_1^j + 1)||^2 > ||\hat{v}_i(t_1^j)||^2. \]
A.2. Proof of Results Contained in Chapter 4.

We now show that (A.93) implies \( ||\hat{v}_i(t_{1}^{ij} + 1)||^2 \geq ||\hat{v}_i(t_{1}^{ij})||^2 + \frac{1}{n^2} \), the result (A.92) then follows directly from Lemma A.2.21 and Lemma A.2.22. The Lemma A.2.18 implies \( \hat{v}_i(t) \in \{0, \frac{1}{n}\}^n \) for all \( i \in \mathcal{V} \) and \( t \geq 0 \), thus under (A.93) it follows that,

\[
||\hat{v}_i(t_{1}^{ij} + 1)||^2 = \frac{1}{n} \hat{v}_i(t_{1}^{ij} + 1)'1_n ,
= \frac{1}{n^2}||\{\ell : \hat{v}_\ell(t_{1}^{ij} + 1) = \frac{1}{n}\}|| ,
> ||\hat{v}_i(t_{1}^{ij})||^2 = \frac{1}{n} \hat{v}_i(t_{1}^{ij})'1_n ,
= \frac{1}{n^2}||\{\ell : \hat{v}_\ell(t_{1}^{ij}) = \frac{1}{n}\}|| .
\]  

(A.94)

From the second and fourth lines in (A.94) we have,

\[
||\{\ell : \hat{v}_\ell(t_{1}^{ij} + 1) = \frac{1}{n}\}|| > ||\{\ell : \hat{v}_\ell(t_{1}^{ij}) = \frac{1}{n}\}|| \Rightarrow ||\{\ell : \hat{v}_\ell(t_{1}^{ij} + 1) = \frac{1}{n}\}|| - 1 \geq ||\{\ell : \hat{v}_\ell(t_{1}^{ij}) = \frac{1}{n}\}|| .
\]  

(A.95)

Under the constraint \( \hat{v}_i(t) \in \{0, \frac{1}{n}\}^n \), the last inequality in (A.95) implies \( ||\hat{v}_i(t_{1}^{ij} + 1)||^2 \geq ||\hat{v}_i(t_{1}^{ij})||^2 + \frac{1}{n^2} \).

\[ \begin{proof}

Lemma A.2.28 (DDA Existence of a Time for Zero Reduction in Error) For any communication sequence \( C_{[0,\infty]} \in IVCC \) there exists an integer \( \ell_{n-2} \) such that,

\[ E^2(C|_{[t_0,t_1]}) = 0 , \forall \ell \geq \ell_{n-2}, C|_{[t_0,t_1]} \in SVCC. \]  

(A.96)

Proof. The result follows immediately by applying Lemmas A.2.22 and A.2.27 to Lemma A.2.26.

\[ \]  

\[ \begin{proof}

Lemma A.2.29 (DDA Signal Update Property Given Zero Reduction in Error) Assume \( ||\hat{v}_i(t_{1}^{ij})||^2 = ||\hat{v}_j(t_{0}^{ij})||^2 \), then \( \hat{v}_j(t_{0}^{ij}) = \frac{1}{n} \) if the signal \( S^{ij}(t_{0}^{ij}, t_{1}^{ij}) \) results in no reduction in error.

Proof. Lemma A.2.20 together with \( ||\hat{v}_i(t_{1}^{ij})||^2 = ||\hat{v}_j(t_{0}^{ij})||^2 \) implies \( ||\hat{v}_i^{-i}(t_{1}^{ij})||^2 = ||\hat{v}_j^{-j}(t_{0}^{ij})||^2 \). From Lemma A.2.24 we thus have,

\[ ||\hat{v}_i^{-i}(t_{1}^{ij})||^2 \geq ||\hat{v}_j^{-i}(t_{0}^{ij})||^2 \geq ||\hat{v}_j^{-j}(t_{0}^{ij})||^2 = ||\hat{v}_i^{-i}(t_{0}^{ij})||^2 . \]  

(A.97)

From (A.97) we have \( ||\hat{v}_i^{-i}(t_{1}^{ij})||^2 = ||\hat{v}_i^{-i}(t_{0}^{ij})||^2 \). Note that Lemma A.2.20 together with \( ||\hat{v}_i(t_{1}^{ij})||^2 = ||\hat{v}_j(t_{0}^{ij})||^2 \) implies \( ||\hat{v}_i^{-i}(t_{1}^{ij})||^2 + \frac{1}{n^2} = ||\hat{v}_j^{-i}(t_{0}^{ij})||^2 + \hat{v}_{ji}(t_{0}^{ij})^2 \), and thus the result follows since \( ||\hat{v}_i^{-i}(t_{1}^{ij})||^2 = ||\hat{v}_j^{-i}(t_{0}^{ij})||^2 \).

\[ \]
A.3. Data Dissemination Results

**Lemma A.2.30 (DDA Local Average-Consensus Property)** If the magnitude of any normal consensus vector \( \hat{v}_i(t) \) is \( \frac{1}{n} \), then \( \hat{v}_i(t) = \frac{1}{n} \mathbf{1}_n \).

**Proof.** If \( ||\hat{v}_i(t)||^2 = \frac{1}{n} \), then from Lemma A.2.19 we have \( \frac{1}{n}\hat{v}_i(t)'\mathbf{1}_n = \frac{1}{n} \). From Lemma A.2.18 each element in \( \hat{v}_i(t) \) belongs to \( \{0, \frac{1}{n}\} \), thus if any element in \( \hat{v}_i(t) \) did not equal \( \frac{1}{n} \) then \( \hat{v}_i(t)'\mathbf{1}_n < 1 \), so the result follows by contradiction.

**Theorem 4.4.10: (DDA Network Convergence to Average-Consensus)** The DDA protocol (4.16) – (4.21) solves ACP2 as \( t \to \infty \) for any UCP \( C_0,t \in IVCC \) (cf. Def. 4.4.5). □

**Proof:** For any communication sequence \( C_{[0,t]} \in IVCC \) the Lemma A.2.28 implies there exists an integer \( \ell_{n-2} \) such that (A.96) holds. To finish the proof it suffices to show that the condition,

\[
E^2(C'_{[t_0,t_1]}) = 0, \quad C'_{[t_0,t_1]} \in SYCC
\] (A.98)

implies (A.22) at time \( t = t_0^f \), and hence by Def. A.2.1 an average-consensus is obtained at \( t_0^f \).

Applying Lemmas A.2.21 and A.2.22 to (A.98) implies that for all \( t \in [t_0^f,t_1^f] \),

\[
||\hat{v}_i(t)||^2 \geq ||\hat{v}_j(t)||^2, \quad \forall (i,j) \in \mathcal{V}^2
\] (A.99)

thus \( ||\hat{v}_i(t)||^2 = ||\hat{v}_j(t)||^2 \) for all \( t \in [t_0^f,t_1^f] \).

Since \( C'_{[t_0^f,t_1^f]} \in SYCC \), it follows from Lemmas A.2.20, A.2.23 and A.2.29 that there exists a node \( i \) satisfying \( \hat{v}_{ij}(t) = \frac{1}{n} \) for all \( j \in \mathcal{V} \), hence \( ||\hat{v}_i(t)||^2 = \frac{1}{n} \). From Lemma A.2.30 and (A.99) we then have \( \hat{v}_j(t) = \frac{1}{n} \mathbf{1}_n \) for all \( j \in \mathcal{V} \). Thus by Lemma A.2.23 and Def. A.2.1 an average-consensus is obtained at the finite time \( t_0^f \). □

### A.3 Data Dissemination Results

This section provides a proof of an upper bound on the time until a communication sequence \( C_{0,t} \) is “time-respecting strongly connected” (trecs), given an upper bound on the time until \( C_{0,t} \) is “non-time-respecting strongly connected” (ntrecs).

In Sec. A.4 we state the definitions regarding the communication sequence that will be required in the results subsequently proven in Sec. A.5 – A.8. In Sec. A.5 we prove an \( (n-1) \) upper bound on the completion time of the flooding protocol (cf. Def. A.4.12) given the relatively strict communication condition of \( C_{0,n-2} \in RGF_{SC} \) (cf. Def. A.4.6 and Def. A.4.8). The subsequent corollary, Cor. A.6.1, proves a super-exponentially greater upper bound \( ((n-2)(n!)+1) \), but allows for the weaker (albeit much lengthier) communication condition of \( C_{0,(n-2)(n!)} \in RGF_{SC} \) (cf. Def. A.4.7 and Def. A.4.6). We then combine the best features of the preceding theorem and corollary in our main result (Conj. A.7.1), which states an \( (n-1) \) upper bound given only
A.4 Preliminary Definitions

This section restates some of the communication assumptions made throughout this thesis. We then also define specific communication sequences that will be required for the results proven in Sec. A.5 – A.8.

Assume there exists a fixed set of $n \in \mathbb{N}$ nodes denoted $V = \{1, 2, \ldots, n\}$. We let $t \in \mathbb{N}$ denote discrete time and $V_i = V \setminus \{i\}$ for any node $i \in V$.

**Definition A.4.1 (Signal)** For any ordered pair $(t_0, t_1) \in \mathbb{N}^2$, a “signal” transmitted from node $j \in V$ at time $t_0$ and received at node $i \in V$ at time $t_1$ will be denoted $S_{ij}^{t_0, t_1}$.

Due to causality, we assume that any signal $S_{ij}^{t_0, t_1}$ cannot be received prior to its transmission time, that is $t_0 \leq t_1$.

**Definition A.4.2 (Communication Sequence)** For any $0 \leq t_0 \leq t_1$, a “communication sequence” $C_{t_0, t_1}$ is the set of all signals transmitted no earlier than time $t_0$ and received no later than time $t_1$, that is,

$$C_{t_0, t_1} = \{S_{i_1 j_1}, S_{i_2 j_2}, S_{i_3 j_3}, \ldots\}$$

where we have omitted the time indices but it is understood that the transmission time $t_0$ and reception time $t_1$ of each signal $S_{i \ell j \ell}$ belong to the interval $[t_0, t_1]$.

We next define the notion of a “time-respecting communication path”, which will be abbreviated as “trcp”.

**Definition A.4.3** A communication sequence $C_{t_0, t_1}$ contains a “time-respecting communication path” (trcp) from node $j$ to node $i$ if $C_{t_0, t_1}$ contains a sub-sequence $C_{t_0, t_1}^{ij}$ with the following connectivity property,

$$C_{t_0, t_1}^{ij} = \{S_{i_1 j_1}^{t_0, t_1}, S_{i_2 j_2}^{t_0, t_1}, S_{i_3 j_3}^{t_0, t_1}, \ldots\}$$

where we have omitted the time indices but it is understood that the transmission time $t_0$ of each signal $S_{i \ell j \ell}$ occurs after the reception time $t_1$ of the preceding signal $S_{i \ell-1 j \ell-1}$.

Note that the sub-sequence $C_{t_0, t_1}^{ij}$ in Def.A.4.3 has a finite cardinality $|C_{t_0, t_1}^{ij}| = k(ij) \geq 1$.

Similar to the above definition, we now define the notion of a “non-time-respecting communication path”, which will be abbreviated as “ntrcp”.

Finally, we generalize these results for a time-delayed and asynchronous communication sequence in Sec.A.8.
A.4. Preliminary Definitions

Definition A.4.4 A communication sequence $C_{\ell_0, t_1}$ contains a “non-time-respecting communication path” (ntrcp) from node $j$ to node $i$ if $C_{\ell_0, t_1}$ contains a sub-sequence $C_{i,j}^{\ell_0(t_{i,j}), t_1(t_{i,j})}$ with the following connectivity property,

$$C_{i,j}^{\ell_0(t_{i,j}), t_1(t_{i,j})} = \{ S_{t_0}, S_{t_1}, S_{t_2}, \ldots, S_{t_k}, S_{t_{k+1}} \}$$

where we have omitted the time indices but (in contrast to a trcp) it is not required that the transmission time $t_0$ of each signal $S_{t_0}$ occurs after the reception time $t_1$ of the preceding signal $S_{t_1}$. □

In full analogy to the above two definitions, we can define the notions of a “time-respecting” and “non-time-respecting” strongly-connected communication sequence.

Definition A.4.5 A communication sequence $C_{\ell_0, t_1}$ is “time-respecting strongly-connected” (trcs) if $C_{\ell_0, t_1}$ contains a trcp from each node $i \in V$ to every node $j \in V_{-i}$. □

The following definition is identical to the preceding definition, with the exception that an ascending time-order in signals is not required.

Definition A.4.6 A communication sequence $C_{\ell_0, t_1}$ is “non-time-respecting strongly-connected” (ntrcs) if $C_{\ell_0, t_1}$ contains a ntrcp from each node $i \in V$ to every node $j \in V_{-i}$. □

The following definition restricts the class of the communication sequences that will be assumed in our main conjecture (Conj.A.7.1).

Definition A.4.7 (Ring Graph) At any time instant $t \in \mathbb{N}$, a communication sequence $C_{t, t}$ is a “ring graph” (RG) if every node in $V$ sends and receives exactly one signal. □

Let $C_{t, t} \in RG$ denote that a communication sequence $C_{t, t}$ satisfies Def.A.4.7. Further let $C_{t_0, t_1} \in RG$ denote a sequence of ring graphs over the interval $[t_0, t_1]$, that is $C_{t_0, t_1} \in RG$ implies $C_{t_0, t_1} = \{ C_{t, t} \in RG : \forall \ell \in \{t_0, t_0+1, \ldots, t_1\} \}$.

The first theorem in this chapter will require the following class of communication sequences.

Definition A.4.8 A communication sequence $C_{t_0, t_1}$ is a “fixed ring graph” (RGF) over the time interval $[t_0, t_1]$ if $C_{t_0, t_1} = \{ C_{t, t} \in RG, C_{t_0, t_1} = C_{t, t} : \forall \ell \in \{t_0, t_0+1, \ldots, t_1\} \}$. □

Let $C_{t_0, t_1} \in RGF$ denote that the communication sequence $C_{t_0, t_1}$ is RGF (cf. Def.A.4.8). Also let $C_{t, t} \in RG_{SC}$ imply that the communication sequence $C_{t, t}$ is both RG (cf. Def.A.4.7) and trcs (cf. Def.A.4.6). Similarly, let $C_{t_0, t_1} \in RG_{SC}$ denote that $C_{t_0, t_1} = \{ C_{t, t} \in RG_{SC} : \forall \ell \in \{t_0, t_0+1, \ldots, t_1\} \}$, and $C_{t_0, t_1} \in RG_{FSC}$ denote that $C_{t_0, t_1} = \{ C_{t, t} \in RG_{SC}, C_{t_0, t_1} = C_{t, t} : \forall \ell \in \{t_0, t_0+1, \ldots, t_1\} \}$. 

112
A.4. Preliminary Definitions

**Definition A.4.9** A communication sequence \( C_{t_0,t_1} \) contains a “\( k \)-cycle” if there exists a subset \( \mathcal{V}(k) \subseteq \mathcal{V} \) such that \( C_{t_0,t_1} \) contains a \textit{ntrep} from each node \( i \in \mathcal{V}(k) \) to every node \( j \in \mathcal{V}(k)-i \), and furthermore \( |\mathcal{V}(k)| = k \).

We let \( (k)\textit{ntrcs} \subseteq C_{t_0,t_1} \) denote that the communication sequence \( C_{t_0,t_1} \) contains a “\( k \)-cycle”.

**Definition A.4.10** *(Signal Notation)* Assume \( C_{0,t} \in \mathcal{R} \) and let \( a_{0,t} = \{t_1, t_2, \ldots, t_k\} \in \mathcal{P}(0,t) \), where \( \mathcal{P}(0,t) \) denotes the power set of \( \{0,1,\ldots,t\} \). Next define \( i(\emptyset) = i \), and let \( i(t_k) \) denote the node which at time \( t_k \) sends a signal to node \( i \). Similarly, let \( i(t_{k-1},t_k) \) denote the node which at time \( t_{k-1} \) sends a signal to node \( i(t_k) \), that is \( i(t_k)(t_{k-1}) = i(t_{k-1},t_k) \). Proceeding in this way, let \( i(a_{0,t}) \) denote the node which at time \( t_1 \) sends a signal to the node which at time \( t_2 \) sends a signal to the node which at time \( t_3 \) (and so on) until node \( i \) receives at time \( t_k \) a signal from node \( i(t_k) \).

The Lemma A.6.2 in Sec.A.6 proves that for any \( C_{0,t} \in \mathcal{R} \) there exists a unique node \( i(a_{0,t}) \) for any \( a_{0,t} \in \mathcal{P}(0,t) \), thus the above definition is not ambiguous. The diagram Fig.A.1 illustrates the notation presented in Def.A.4.10. Note that by repeating the iteration \( i(t_{k})(t_{k-1}) = i(t_{k-1},t_k) \) as stated in the above definition, we obtain the formula \( i(\{t_{\ell},t_{\ell+1},\ldots,t_k\})(t_{\ell-1}) = i(\{t_{\ell-1},t_{\ell},t_{\ell+1},\ldots,t_k\}) \) for any \( t_{\ell-1} \) that renders \( \{t_{\ell-1},t_{\ell},\ldots,t_k\} \in \mathcal{P}(0,t) \).

![Figure A.1: A diagram of the Signal Notation in Def.A.4.10.](image-url)
A.4. Preliminary Definitions

**Definition A.4.11 (Input Set)** For each \( t \in \mathbb{N} \) and node \( i \in V \) the “input set” \( \tilde{V}_i(t) \) is a set of \( 2^t \) possibly repeated elements. Assume the initialization \( \tilde{V}_i(0) = \{ i(\emptyset) \} \) for all \( i \in V \), and assume that when a signal from node \( j = i(t) \) is received at time \( t \) at node \( i \), the input set \( \tilde{V}_i(t) \) is updated as follows,

\[
\tilde{V}_i(t + 1) = \tilde{V}_i(t) \cup \tilde{V}_i(t)(t) .
\]  

(A.100)

If we let \( \tilde{V}_i^\ell(t) \) denote the \( \ell \)th element in \( \tilde{V}_i(t) \), the update (A.100) then allows us to define the \( \ell \)th element in \( \tilde{V}_i(t + 1) \) as \( i(\mathbf{P}_\ell(0,t)) \), where \( \mathbf{P}_\ell(0,t) \) denotes the \( \ell \)th set in \( \mathbf{P}(0,t) \). To summarize, the update (A.100) implies \( \tilde{V}_i^\ell(t) = i(\mathbf{P}_\ell(0,t - 1)) \forall \ell \in \{1, \ldots, 2^t\} \).

We next define the time at which the update (A.100) is “complete”. Recall that \( \tilde{V}_i^\ell(t + 1) = i(\mathbf{P}_\ell(0,t)) \) and \( \mathbf{P}_\ell(0,t) \) denotes the \( \ell \)th set in \( \mathbf{P}(0,t) \).

**Definition A.4.12** The update (A.100) is “complete” at time \( t \in \mathbb{N} \) if,

\[
\bigcup_{\ell \in \{1, \ldots, 2^t\}} \tilde{V}_i^\ell(t) \supseteq V , \forall i \in V .
\]  

(A.101)

We note that for any communication sequence \( C_{0,t} \), the time of completion of (A.100) is identical to the time at which any node \( i \in V \) can flood the network \( V \) with the initial consensus value \( s_i(0) \). In other words, if \( i \in \tilde{V}_j(t) \) for all \( j \in V \setminus i \), then the data initially held at node \( i \) will be known at every node \( j \in V \setminus i \) and thus the “flooding algorithm” (as described in e.g. [16]) will no longer increase the knowledge set of any node after (A.100) is complete.

Lastly, we define the “reduced input set” \( \bar{V}_i(t) \) as follows.

**Definition A.4.13 (Reduced Input Set)** For each \( t \in \mathbb{N} \) and each node \( i \in V \), the “reduced input set” \( \bar{V}_i(t) \) is an unordered set that is obtained from the input set \( \tilde{V}_i(t) \) (cf. Def.A.4.11) as follows,

\[
i(\mathbf{P}_\ell(0,t - 1)) \in \bar{V}_i(t) \iff \exists \ell \text{ s.t. } i(\mathbf{P}_\ell(0,t - 1)) \in \tilde{V}_i(t) ,
\]

\[
|\bar{V}_i(t)| = |\bigcup_{a_{0,t-1} \in \mathbf{P}(0,t-1)} i(a_{0,t-1})| .
\]  

(A.102)

Unlike the input set \( \tilde{V}_i(t) \), the reduced input set \( \bar{V}_i(t) \) contains no repeated node values (in particular, \( \tilde{V}_i(t) \) contains no repeated node values within the set \( \{ i(a_{0,t-1}) \in V : a_{0,t-1} \in \mathbf{P}(0,t - 1) \} \)). Furthermore, by the signal notation (cf. Def.A.4.10) it follows that (A.102) implies \( j \in \tilde{V}_i(t) \) if and only if (iff) node \( j \) has a trcp (cf. Def.A.4.3) to node \( i \) over the time interval \([0, t - 1]\). Also, for any communication sequence it follows from (A.100) and (A.102) that the reduced input set \( \bar{V}_i(t) \) is updated as,

\[
\bar{V}_i(t + 1) = \bar{V}_i(t) \cup \bar{V}_i(t)(t) .
\]  

(A.103)
Lastly, the Def.A.4.12 states that (A.101) is satisfied and thus from (A.102) the update (A.100) is completed at time $t$ iff $\bar{V}_i(t) = \mathcal{V}$ for each node $i \in \mathcal{V}$. That is, the update (A.103) will be “completed” at the same time as the update (A.100).

With the above definitions, the first theorem in this chapter (Thm.A.5.1) will prove an $(n-1)$ upper bound on the time it takes the update (A.100) to be completed (cf. Def.A.4.12) given the relatively strict communication condition of $C_{0,n-2} \in RGF_{SC}$ (cf. Def.A.4.6 and Def.A.4.8). The subsequent corollary, Cor.A.6.1, proves a super-exponentially greater upper bound $((n-2)(n!) + 1)$, but allows for the weaker (albeit much lengthier) communication condition of $C_{0,(n-2)(n!)} \in RG_{SC}$ (cf. Def.A.4.7 and Def.A.4.6). We then combine the best features of both results in Conj.A.7.1, which proves an $(n-1)$ upper bound given only that $C_{0,n-2} \in RG_{SC}$.

### A.5 Fixed Ring Graph Theorem

In this section we prove that for any fixed ring graph $C_{0,n-2} \in RGF_{SC}$ the update (A.100) will be completed at time $(n-1)$.

#### A.5.1 Statement of Theorem A.5.1.

**Theorem A.5.1 (RGF-Thm.)** Assume that $C_{0,n-2} \in RGF$ (cf. Def.A.4.8). Given this parameterization of $C_{0,n-2}$, it is both necessary and sufficient that $C_{0,n-2} \in RGF_{SC}$ for the update (A.100) to be completed at time $(n-1)$. $\square$

#### A.5.2 Proof of Theorem A.5.1.

Proof of Thm.A.5.1. (Sufficiency)

We can show sufficiency follows by induction. Note that in the statement of Thm.A.5.1 it is assumed that $C_{0,n-2} \in RGF_{SC}$, thus by lemma A.5.3 we can WLG consider $C_{\ell,t}$ to satisfy (A.108) for each $\ell \in \{0,n-2\}$. This standardized form will be denoted as $C_{0,n-2} \in RGF_{SC}$ for convenience.

**Base Case:** At the initial time $t = 0$ the reduced input set $\bar{V}_i(t)$ (cf. Def.A.4.13) of each node $i \in \mathcal{V}$ satisfies,

$$
\bar{V}_i(t) = \{i, i+1, \ldots, \min\{n, t+i\}\} \cup \{1, 2, \ldots, t+i - \min\{n, t+i\}\}.
$$  \hspace{1cm} (A.104)

The base case holds because Def.(A.4.11) states the initial condition $\bar{V}_i(0) = \{i(\emptyset)\}$ for each
Proof of Thm. A.5.1: (Necessity) 

The proof of necessity follows by contradiction. Assume that \( C_{0,n-2} \in RGF \) and \( C_{0,n-2} \notin RGF_{SC} \). If \( C_{0,n-2} \notin RGF_{SC} \) then by Def.A.4.6 and Def.A.4.9 we have \((k)\mathbf{ntres} \subseteq C_{t,t} \) for
A.5. Fixed Ring Graph Theorem

each $\ell \in \{0, 1, \ldots, n - 2\}$ and some $k < n$. From the proof of sufficiency as given above it follows that $|\tilde{V}_i(n-1)| \leq k < n$ and thus the update (A.101) is not complete (cf. Def.A.4.12) at $t = (n-1)$.

A.5.3 Supporting Lemmas for Theorem A.5.1.

**Lemma A.5.2** For any communication sequence $C_{t,t} \in RGF_{SC}$ (cf. Def.A.4.6 – A.4.7), it follows that $C_{t,t}$ does not contain any $k$-cycles (cf. Def. A.4.9) for all $k \in \{1, \ldots, n-1\}$. □

Proof of Lemma A.5.2.

The result follows from a proof by contradiction. Assume $C_{t,t} \in RGF_{SC}$ and that $C_{t,t}$ contains a $k$-cycle (cf. Def. A.4.9) for some $k \in \{1, \ldots, n-1\}$. Next observe that by Def.A.4.6 that $C_{t,t} \in RGF_{SC}$ implies that $C_{t,t}$ contains a ntrcp from each node $i \in V$ to every node $j \in V_{-i}$. If there existed a subset $V(k) \subset V$ such that $C_{t,t}$ contains a ntrcp from each node $i \in V(k)$ to every node $j \in V(k)_{-i}$, then there exists a subset of nodes $V \setminus V(k)$ to which there could exist a ntrcp from some node $i \in V(k)$ only if node $i$ transmitted two (or more) signals (one signal being transmitted to the receiving node contained in $V(k)$, and one signal being transmitted to the node contained in $V \setminus V(k)$). The $C_{t,t} \in RGF_{SC}$ condition (cf. Def.A.4.7) requires that each node transmits one and only one signal, thus we find a contradiction for any $C_{t,t} \in RGF_{SC}$ that contains a $k$-cycle for any $k \in \{1, \ldots, n-1\}$. □

**Lemma A.5.3** For any communication sequence $C_{0,t} \in RGF_{SC}$ we can without loss of generality (WLG) assume that $C_{\ell,\ell}$ is expressed as follows for each $\ell \in \{0, 1, \ldots, t\}$,

$$C_{\ell,\ell} = \{S^1 i_2, S^2 i_3, S^3 i_4, \ldots, S^{(n-1)} i_n, S^n i_1\}$$  \hspace{1cm} (A.108)

where we have omitted the time indices but it is understood each signal is sent and received at time $\ell$. □

Proof of Lemma A.5.3.

For any $C_{0,t} \in RGF_{SC}$ each $C_{\ell,\ell} \in RGF_{SC}$ does not vary over the time interval $\ell \in \{0, 1, \ldots, t\}$. By Def.A.4.6 – A.4.7, the following condition is sufficient for $C_{\ell,\ell} \in RGF_{SC}$,

$$C_{\ell,\ell} = \{S^{i_1} i_2, S^{i_2} i_3, S^{i_3} i_4, \ldots, S^{i_{n-1}} i_n, S^{i_n} i_1\} , i_k \in V \setminus \{\cup_{r=1}^{k-1} i_r\} , \forall k \in V$$  \hspace{1cm} (A.109)
where we have omitted the time indices but it is understood each signal is sent and received at time \( \ell \). The condition (A.109) can also be proven necessary for any \( C_{\ell,\ell} \in RGF_{SC} \) by observing that if any signal is added or deleted then \( C_{\ell,\ell} \notin \mathcal{R} \) and thus Def.A.4.7 is not satisfied. Similarly, if the receiving node \( j(i) \) of any node \( i \) and transmitting node \( r(\ell) \) of any node \( \ell \in V_{-j(i)} \) are interchanged, then there exists a \( k \)-cycle (cf. Def.A.4.9) for some \( k < n \), and thus by lemma A.5.2 the communication sequence \( C_{\ell,\ell} \) is not \text{mtrcs} (cf. Def. A.4.6). Conversely, if the receiving node \( j(i) \) of any node \( i \) and transmitting node \( r(\ell) \) of the node \( \ell = j(i) \) are interchanged, then there is no change in the condition (A.109) because the actual values of each \( i_k \in V \) are arbitrary under the constraint \( i_k \in V \setminus \{ \cup_{r=1}^{k-1} i_r \} , \forall k \in V \), as is stated in (A.109).

We have shown that (A.109) is both necessary and sufficient for \( C_{\ell,\ell} \in RGF_{SC} \). Besides the constraint \( i_k \in V \setminus \{ \cup_{r=1}^{k-1} i_r \} , \forall k \in V \) as stated in (A.109), the values of each \( i_k \in V \) are arbitrary and thus we can assume WLG that (A.108) holds for all \( C_{\ell,\ell} \in RGF_{SC} \). □

A.6 Corollary A.6.1.

In this section we prove a corollary to Thm.A.5.1. That is for any sequence of ring graphs \( C_{0,(n-2)(n!)} \in RGF_{SC} \) the update (A.100) will be completed at time \( ((n-2)(n!)+1) \).


**Corollary A.6.1 (Extended RGF-Thm.)** For any \( C_{0,(n-2)(n!)} \in RGF_{SC} \) the update (A.100) will be completed at time \( ((n-2)(n!)+1) \). □


Proof of Cor.A.6.1.

The Thm.A.5.1 implies that for any \( C_{0,0} \in RGF_{SC} \) and \( k \in \mathbb{N} \) we have \( |\hat{V}_i(1)| \geq 2 \) for all \( i \in V \). Next observe that lemma A.6.3 implies any communication sequence \( C_{1,k(n!)} \in RGF_{SC} \) will necessarily contain at least \( k \) identical ring graphs \( C_{\ell,\ell} \in RGF_{SC} \) among the set \( \{ C_{\ell,\ell} \in C_{1,k(n!)} : \ell \in \{1, \ldots, k(n!)\} \} \), and thus from Thm.A.5.1 for any \( C_{0,k(n!)} \in RGF_{SC} \) we have \( |\hat{V}_i(k(n!) + 1)| \geq k + 2 \) for all \( i \in V \) and any \( k \in \{1, \ldots, n-2\} \). It then follows that for any \( C_{0,(n-2)(n!)} \in RGF_{SC} \) the reduced input set of each node \( i \in V \) will satisfy \( |\hat{V}_i((n-2)(n!)+1)| = n \) and thus the update (A.100) is completed at time \( ((n-2)(n!)+1) \). □


Lemma A.6.2 For any communication sequence $C_{0,t} \in RG$ (cf. Def.A.4.7) there exists one and only one node $i(a_{0,t})$ for any $i \in V$ and sequence $a_{0,t} \in P(0,t)$.

Proof of Lemma A.6.2.

By Def.A.4.7 each node receives exactly one signal at each time instant, thus if $i(x_k)$ exists then it must be unique. Likewise if $i(x_{k-1},x_k)$ exists then it must be unique. Iterating this argument we find that if $i(a_{0,t})$ exists then it must be unique. Next observe that by Def.A.4.7 each node in $V$ sends exactly one signal at each time instant, thus $i(a_{0,t})$ is contained in $V$ and hence exists.

Lemma A.6.3 For any communication sequence $C_{t,t} \in RG$ (cf. Def.A.4.7) there exists $(n!)$ possible communication sequences.

Proof of Lemma A.6.3.

For any $C_{t,t} \in RG$, the Lemma A.6.2 implies that at the time instant $t$ each node $i \in V$ will send exactly one signal, and this signal will be received by a unique node. For each node $i \in V$ let this unique node be denoted $j(i)$. Let us first consider node 1, for which there exists $n$ possible nodes $j(1) \in V$ that may receive the signal from node 1. Next consider node 2, for which (since $j(i)$ is unique) there exists $n-1$ possible nodes $j(2) \in V \setminus j(1)$ that may receive the signal from node 2. Proceeding in this way, we find that for node $k$ there exists $n-k+1$ possible nodes $j(k) \in V \setminus \{\cup_{r=1}^{k-1} j(r)\}$ that may receive the signal from node $k$. Besides the constraint $j(k) \in V \setminus \{\cup_{r=1}^{k-1} j(r)\}$, each choice of the receiving node $j(k)$ is independent from the previous choices of the receiving nodes, we can thus multiply the number of possible nodes that each node can send a signal to and thereby obtain $(n!)$ as the total number of possible communication sequences for any $C_{t,t} \in RG$.

A.7 Conjecture A.7.1.

In this section we state a conjecture that combines the best features of Thm.A.5.1 and Cor.A.6.1. That is for any sequence of ring graphs $C_{0,(n-2)} \in RG_{SC}$ the update (A.100) will be completed at time $(n - 1)$. 
A.8. Extensions to Time-Delayed and Asynchronous Communication Sequences

A.7.1 Statement of Conjecture A.7.1.

**Conjecture A.7.1** *(n-Thm.)* For any communication sequence \( C_{0,n-2} \in RG_{SC} \), the update (A.100) will be completed at time \((n-1)\). □

We note the above conjecture implies a super-exponential reduction in the completion time of (A.100) as compared to Cor.A.6.1.

A.8 Extensions to Time-Delayed and Asynchronous Communication Sequences

This section generalizes Cor.A.6.1 and Conj.A.7.1 to time-delayed and asynchronous communication sequences. The proof of both of these generalizations relies only on lemma A.8.3, which is proven in Sec.A.8.4, subsequent to the statement of the generalizations in Sec.A.8.1.

A.8.1 Statement of Extensions to Cor.A.6.1 and Conj.A.7.1

For the following two results we assume there exists an upper bound denoted \( SC_t \) on the time that any sub-communication sequence \( C_{t,t+SC_t} \subseteq C_{0,t} \) is \( ntrcs \) (thus implying a “uniformly” \( ntrcs \) communication sequence). Given such an upper bound, the Cor.A.6.1 and Conj.A.7.1 each imply an upper bound \( SC_t \) on time until any sub-communication sequence \( C_{t,t+SC_t} \subseteq C_{0,t} \) is \( trcs \) (thus implying a “uniformly” \( trcs \) communication sequence), as we now will state and then prove in Sec.A.8.4.

**Theorem A.8.1** Assume \( C_{t,t+SC_t} \in ntrcs \) for all \( t \in \mathbb{N} \) and that \( C_{0,(n-2)n!(SC_t+1)+SC_t} \in RG_{SC} \) the update (A.100) will be completed at time \(((n-2)n!+1)(SC_t+1))\). □.

The following conjecture implies a super-exponential reduction in the completion time of (A.100) as compared to Thm.A.8.1.

**Conjecture A.8.2** Assume \( C_{t,t+SC_t} \in ntrcs \) for all \( t \in \mathbb{N} \), then the update (A.100) will be completed at time \((n-2)(SC_t+1)+1)\). □

In Fig.A.2 below we illustrate the rationale behind Thm.A.8.1. The Fig.A.3 provides an illustration of the result stated in Conj.A.8.2.

A.8.2 Proof of Theorem A.8.1.

Proof of Thm.A.8.1.
A.8. Extensions to Time-Delayed and Asynchronous Communication Sequences

The updates (A.100) and (A.102) implies the reduced input set $\hat{V}(t)$ is a non-decreasing function of $t$. The lemma A.8.3 in Sec.A.8.4 proves that any trcp implies a ntrcp and thus assuming Cor.A.6.1, the Thm.A.8.1 follows immediately using lemma A.8.3.

A.8.3 Proof of Conjecture A.8.2.

Proof of Conj.A.8.2.

The updates (A.100) and (A.102) implies the reduced input set $\hat{V}(t)$ is a non-decreasing function of $t$. The lemma A.8.3 in Sec.A.8.4 proves that any trcp implies a ntrcp and thus assuming Conj.A.7.1, the Conj.A.8.2 follows immediately using lemma A.8.3.

A.8.4 Supporting Lemmas for Thm.A.8.1 and Conj.A.8.2

Lemma A.8.3 Consider any communication sequence $C_{0,t}$ within which each signal is received at the same time as its transmission, that is,

$$S^{ij}(t_0, t_1) \in C_{0,t} \iff t_0 = t_1 = \ell \in [0, t].$$

(A.110)
A.8. Extensions to Time-Delayed and Asynchronous Communication Sequences

Next consider the asynchronous and time-delayed communication sequence \( \hat{C}(C_{0,t}) \) which contains every signal in \( C_{0,t} \) yet with possibly time-delayed and asynchronous signals, that is,

\[
S_{ij}^{t_0,t_1} \in C_{0,t} \iff \left( S_{ij}^{t_0,t_1} \in \hat{C}(C_{0,t}) , \ 0 \leq t_0 \leq t_1 \leq t \right). \tag{A.111}
\]

If \( C_{0,t} \) contains a \textbf{ntrep} from node \( j \) to node \( i \) then so does \( \hat{C}(C_{0,t}) \). \hfill \Box

Proof of Lemma A.8.3.

We observe that any pair of asynchronous time-delayed signals \( S_{ji}^{t_1,t_2} \) and \( S_{qj}^{t_3,t_4} \) will imply a \textbf{trcs} from node \( j \) to node \( q \) iff \( t_2 \leq t_3 \). Since any \textbf{trcp} from node \( j \) to node \( i \) implies a \textbf{ntrep} from node \( j \) to node \( i \), it follows that any communication sequence \( \hat{C}(C_{0,t}) \) as defined in (A.111) will imply a \textbf{ntrep} from node \( j \) to node \( i \) if \( C_{0,t} \) does. \hfill ■