University of California
Los Angeles

On the Learning Behavior of Adaptive Networks

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy
in Electrical Engineering

by

Jianshu Chen

2014

Abstract of the Dissertation

On the Learning Behavior of Adaptive Networks

by

Jianshu Chen

Doctor of Philosophy in Electrical Engineering

University of California, Los Angeles, 2014

Professor Ali H. Sayed, Chair

This dissertation deals with the development of effective information processing strategies for distributed optimization and learning over graphs. The work considers initially global cost functions that can be expressed as the aggregate sum of individual costs ("sum-of-costs") and proceeds to develop diffusion adaptation algorithms that enable distributed optimization through localized coordination among neighboring agents. The diffusion strategies allow the nodes to cooperate and diffuse information in real-time and they help alleviate the effects of stochastic approximations and gradient noise through a continuous learning process. Among other applications, the resulting strategies can be applied to large-scale machine learning problems, where a network of agents is used to learn a common model from big data sets that are distributed over the network.

The work also develops diffusion strategies for the solution of another class of problems where the global cost functions are expressed as regularized "cost-of-sum" forms. This situation arises when a large-scale model is stored and learned over a network of agents, with each agent being in charge of a portion of the model and it is not feasible to aggregate the entire model in one location due to communication and privacy considerations. It is shown that the "cost-

of-sum" problem can be transformed into a "sum-of-costs" problem by using dual decompositions and the concept of conjugate functions. The collaborative inference step in the dual domain is shown to generate dual variables that can be used by the agents to update their model without the need to share these model parameters or the training data with the other agents. This is a powerful property that leads to an efficient distributed procedure for learning large-scale models over networks.

The dissertation carries out a detailed transient and steady-state analysis of the learning behavior of multi-agent networks, and reveals interesting results about the learning abilities of distributed strategies. Among other results, the analysis reveals how combination policies influence the learning process of networked agents, and how these policies can steer the convergence point towards any of many possible Pareto optimal solutions. The results also establish that the learning process of an adaptive network undergoes three well-defined stages of evolution with distinctive convergence rates during the first two stages, while attaining a finite mean-square-error (MSE) level in the last stage. The analysis reveals what aspects of the network topology influence performance directly and suggests design procedures that can optimize performance by adjusting the relevant topology parameters. Interestingly, it is further shown that, in the adaptation regime, each agent in a sparsely connected network is able to achieve the same performance level as that of a centralized stochastic approximation strategy even for left-stochastic combination strategies. These results lead to a deeper understanding and useful insights on the convergence behavior of coupled distributed learners. The results also lead to effective design mechanisms to help diffuse information more thoroughly over networks.

The dissertation of Jianshu Chen is approved.

Adnan Darwiche

Suhas N. Diggavi

Lieven Vandenberghe

Ali H. Sayed, Committee Chair

University of California, Los Angeles

2014

Table of Contents

List of Figures

List of Tables

Acknowledgments

First, I would like to express my sincere gratitude to my advisor, Professor Ali H. Sayed, for his support, guidance and encouragement through my five years of Ph.D. study. His high standards in research, careful reviews of each of my papers and numerous discussions greatly enhanced the quality and presentation of my work. Working towards a PhD is always challenging, and without his kind support and help, this would have been even more demanding.

I would like to thank Professor Suhas Diggavi for insightful discussions on information-theoretic problems. I would also like to thank Dr. Li Deng for offering me the opportunity to do an internship at Microsoft Research, Redmond. I also appreciate Professors Lieven Vandenberghe, Adnan Darwiche and Suhas Diggavi for serving on my PhD committee.

I am glad to have met many good friends at the Adaptive Systems Laboratory (ASL) at UCLA, with whom I spent lots of time together during the past five years: Zaid J. Towfic, Shang-Kee Ting, Xiaochuan Zhao, Sheng-Yuan Tu, and Chung-Kai Yu. I also appreciate the opportunity to have met many friends who came to visit ASL: Oyvind L. Rortveit from Norway, Paolo Di Lorenzo from Italy, Jae-Woo Lee from Korea, Alexander Bertrand from Belgium, Victor Lora from France, Ricardo Merched and Cassio G. Lopes from Brazil, Reza Abdolee and Milad A. Toutounchian from Canada, Mohammad-Reza from Sweden, and Sergio Valcarcel Macua and Jesus F. Bes from Spain. I will always remember the magic whiteboard where we discussed so many interesting ideas, and I will also remember the great lunches and dinners we had together in Los Angeles.

Furthermore, I want to thank my parents for their support of my study over

the past years. From Yongjia to Harbin, Beijing and Los Angeles, my study is such a long journey that has taken more than two decades and involved traveling almost half of our planet. None of this would have been possible without your love and support over the years.

Finally, the work of this dissertation was supported in part by NSF grants ECS-0725441, CCF-1011918, and CCF-0942936, and by a Dissertation Year Fellowship from the UCLA Graduate Division. The support of the funding agencies is hereby acknowledged.

Vita

2005	B.S. in Electronic and Information Engineering Harbin Institute of Technology (HIT), Harbin, China.
2009	M.S. in Information and Communication Engineering Tsinghua University, Beijing, China.
2009–2014	Research Assistant Department of Electrical Engineering University of California, Los Angeles.
2011–2012	Teaching Assistant Department of Electrical Engineering University of California, Los Angeles.
2012–2013	Teaching Associate Department of Electrical Engineering University of California, Los Angeles.
2013–2014	Dissertation Year Fellowship University of California, Los Angeles.

CHAPTER 1

Introduction

In this dissertation, we study the learning behavior of multi-agent adaptive networks consisting of N connected agents. Each agent k receives a local data stream $\{x_{k,i}\}$ at time i and is able to communicate with its local neighbors — see Fig. 1.1. The objective of the network is for the agents to collaboratively solve a global optimization problem by using information that is collected/stored locally at the different agents. Such networks of interacting agents are useful in solving distributed estimation, learning, and decision making problems [7,48,100,115,117,130]. They are also useful in modeling biological networks [29,46,140], collective rational behavior [51,52], and in developing biologically-inspired designs [7,67]. The learning and adaptation processes of multi-agent systems typically consist of two components: self-learning from local data streams and social-learning from neighbors. During self-learning, each agent updates its state using its local data. During social learning, each agent aggregates information from its neighbors. We briefly discuss in this chapter the special features of these two components and then state contributions of the dissertation.

1.1 Single-Agent Adaptation

The self-learning procedure at each agent k extracts information from its local data streams $\{x_{k,i}\}$ in order to best represent, predict, interpret the local data,

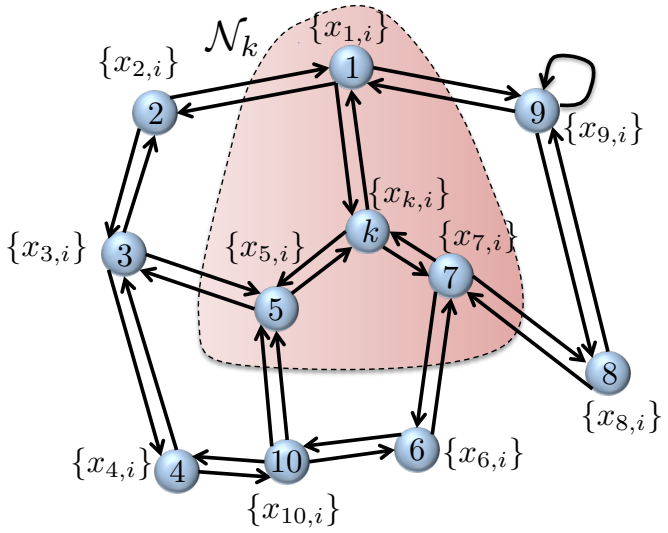


Figure 1.1: A network representing a multi-agent system. The set of all agents that can communicate with node k is denoted by $\mathcal{N}_k$. The edge linking any two agents is represented by two directed arrows to emphasize that information can flow in both directions.

infer parameters of interest, or to make the best decision. Typically, the statistics of the data streams are unknown to the agents, and this requires the networked agents to rely on data realizations to learn from streaming data.

To introduce the self-learning process, we first review adaptation and learning for stand-alone agents. Thus, consider a single agent k and introduce the problem of minimizing an expected loss function with respect to an unknown vector w:

$$\min_{w} \mathbb{E}Q_k(w; \boldsymbol{x}_{k,i}) \tag{1.1}$$

where $Q_k(w; \boldsymbol{x}_{k,i})$ is the loss function at agent k for data sample $\boldsymbol{x}_{k,i}$ (we use the boldface notation, $\boldsymbol{x}_{k,i}$, to highlight the random nature of the data and use regular font, $x_{k,i}$, to denote its realizations), and the expected loss $J_k(w) = \mathbb{E}Q_k(w; \boldsymbol{x}_{k,i})$ is called the risk function. A typical algorithm to solve (1.1) is the stochastic

gradient descent (SGD) algorithm, which updates estimates for w according to the following recursion [117]:

$$w_{k,i} = w_{k,i-1} - \mu_k(i) \cdot \nabla_w Q_k(w_{k,i-1}; x_{k,i}) \tag{1.2}$$

where $\mu_k(i)$ is a positive step-size parameter, which can be time-dependent or be set to a constant value. Note that we are using the gradient of the loss function $Q_k(w; x_{k,i})$ in (1.2) instead of the risk function $J_k(w)$ to update from $w_{k,i-1}$ to $w_{k,i}$. This is because we may not be able to evaluate $\nabla_w J_k(w)$ in closed-form since the statistics of the data are usually unavailable. The learning algorithm can also take other forms beyond gradient descent. For example, in order to improve the convergence behavior, we may multiply a gain matrix $D_{k,i}$ to the left of $\nabla_w Q_k(w; x_{k,i})$ so that the recursion becomes:

$$w_{k,i} = w_{k,i-1} - \mu_k(i) \cdot D_{k,i} \cdot \nabla_w Q_k(w_{k,i-1}; x_{k,i}) \tag{1.3}$$

This variation is popular in optimization problems to improve the convergence rate of deterministic optimization algorithms [10]. More generally, we can write

$$w_{k,i} = w_{k,i-1} - \mu_k(i) \cdot \hat{s}_{k,i}(w_{k,i-1}) \tag{1.4}$$

where $\hat{s}_{k,i}(w)$ denotes the update vector used by agent k at time i. The SGD algorithm (1.2) is a special case of (1.4) when the update vector $\hat{s}_{k,i}(w)$ is chosen as $\nabla_w Q_k(w; x_{k,i})$. The more general form (1.4) is referred to as a stochastic approximation algorithm [81, 99, 105, 116]. This type of algorithms is widely used in online learning and prediction [1, 6, 30, 44, 60, 65, 74, 101, 121, 122, 148]. It is also popular in large-scale machine learning problems, where the dataset contains a finite but a large number of data samples [6, 15–17]. In this context, it is common

to formulate deterministic cost measures and to require each agent to minimize an empirical cost defined in the form of a running sum:

$$J_k(w) = \frac{1}{T}\sum_{i=1}^{T} Q_k(w; x_{k,i}) \tag{1.5}$$

In these scenarios, the gradient descent algorithm assumes the following form:

$$w_{k,i} = w_{k,i-1} - \frac{\mu_k(i)}{T}\sum_{t=1}^{T} \nabla_w Q_k(w_{k,i-1}; x_{k,t}) \tag{1.6}$$

Note that each iteration now requires computing a total of T gradients and then averaging them together. When the dataset is large, computing the gradients can be very expensive. Instead, we can randomly fetch one sample at a time from the dataset and update $w_{k,i}$ according to the SGD recursion (1.2). By doing so, the computation complexity can be greatly reduced without much degradation in performance.

1.2 Multi-Agent Adaptation

1.2.1 Sum-of-Costs Formulation

The single-agent adaptation procedure serves as a building block for the multi-agent case. In multi-agent adaptation, agents collaborate with each other to solve a problem that combines the individual cost functions $\{J_k(w)\}$ in a certain way. One important case is through a "sum-of-costs" formulation, which we will study in some detail in Chapter 2, where the networked agents aim to minimize an

aggregate cost of the form:

$$J^{\text{glob}}(w) = \sum_{k=1}^{N} J_k(w) \tag{1.7}$$

This situation arises when data streams arrive at different agents in a distributed manner, and the agents want to work together to extract information from all the data streams, i.e., learning from "distributed data". Another example arises in large-scale machine learning problems, where the training dataset is finite but too large to be fitted into a single machine. Then, the entire dataset is partitioned into several subsets, which are allocated to different machines to perform parallel training. The individual cost functions in (1.7) can be either a risk function of the form

$$J_k(w) = \mathbb{E}Q_k(w; \boldsymbol{x}_{k,i}) \tag{1.8}$$

or an empirical cost of the form:

$$J_k(w) = \frac{1}{T}\sum_{i=1}^{T} Q_k(w; x_{k,i}) \tag{1.9}$$

As we will show in Chapter 2, problem (1.7) can be solved by the SGD iteration (1.2) interleaved with one combination step over neighborhoods at each agent. Specifically, one may employ consensus [48, 75–77, 97, 98, 137] or diffusion (ATC or CTA) strategies [26, 34, 36, 42, 89, 91, 115, 117, 146] of the following form:

$$\text{Consensus}: \begin{cases} \boldsymbol{\phi}_{k,i-1} = \displaystyle\sum_{l\in\mathcal{N}_k} a_{lk}\boldsymbol{w}_{l,i-1} \\ \boldsymbol{w}_{k,i} = \boldsymbol{\phi}_{k,i-1} - \mu_k(i)\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}_{k,i-1}) \end{cases} \tag{1.10}$$

$$\text{CTA diffusion}: \begin{cases} \boldsymbol{\phi}_{k,i-1} = \sum_{l \in \mathcal{N}_k} a_{lk} \boldsymbol{w}_{l,i-1} \\ \boldsymbol{w}_{k,i} = \boldsymbol{\phi}_{k,i-1} - \mu_k(i) \hat{\boldsymbol{s}}_{k,i}(\boldsymbol{\phi}_{k,i-1}) \end{cases} \tag{1.11}$$

$$\text{ATC diffusion}: \begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu_k(i) \hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{w}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{lk} \boldsymbol{\psi}_{l,i} \end{cases} \tag{1.12}$$

where $\boldsymbol{w}_{k,i} \in \mathbb{R}^M$ is the iterate of agent k at time i, usually an estimate for the minimizer of (1.7), $\boldsymbol{\phi}_{k,i-1} \in \mathbb{R}^M$ and $\boldsymbol{\psi}_{k,i} \in \mathbb{R}^M$ are intermediate variables generated at node k before updating to $\boldsymbol{w}_{k,i}$, $\mu_k(i)$ is a non-negative (time-dependent or constant) step-size parameter used by node k, and $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ is an $M \times 1$ update vector function at node k. In deterministic optimization problems, the update vectors $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ can be the gradient or Newton steps associated with the cost functions [97]. On the other hand, in stocastic approximation problems, such as adaptation, learning and estimation problems [26, 34, 36, 42, 48, 49, 58, 75, 77, 89, 91, 109, 115, 125, 130, 137, 146], the update vectors are usually computed from realizations of data samples that arrive sequentially at the nodes, with a typical choice being the stochastic gradient:

$$\hat{\boldsymbol{s}}_{k,i}(w) = \nabla_w Q_k(w; \boldsymbol{x}_{k,i}) \tag{1.13}$$

In the stochastic setting, the quantities appearing in (1.10)–(1.12) become random and we therefore use boldface letters to highlight their stochastic nature. The combination coefficients $\{a_{lk}\}$ in (1.10)–(1.12) are nonnegative weights that each node k assigns to the information arriving from node l; these coefficients are

required to satisfy:

$$\sum_{l=1}^{N} a_{lk} = 1 \tag{1.14}$$

$$a_{lk} \geq 0 \tag{1.15}$$

$$a_{lk} = 0, \quad \text{if } l \notin \mathcal{N}_k \tag{1.16}$$

Observe from (1.16) that the combination coefficients are zero if $l \notin \mathcal{N}_k$, where $\mathcal{N}_k$ denotes the set of neighbors of node k. Therefore, each summation in (1.10)–(1.12) is actually confined to the neighborhood of node k. We let A denote the $N \times N$ matrix that collects the coefficients $\{a_{lk}\}$. Then, condition (1.14) is equivalent to

$$A^T \mathbb{1} = \mathbb{1} \tag{1.17}$$

where $\mathbb{1}$ is the $N \times 1$ vector with all its entries equal to one. Condition (1.17) means that the matrix A is left-stochastic (i.e., the entries on each of its columns add up to one).

Observe from (1.10)–(1.12) that the convex combination steps appear in different locations in the consensus and diffusion implementations. Moreover, observe that the consensus strategy (1.10) evaluates the update direction $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ at $\boldsymbol{w}_{k,i-1}$, which is the estimator *prior* to the aggregation, while the diffusion strategy (1.11) evaluates the update direction at $\boldsymbol{\phi}_{k,i-1}$, which is the estimator *after* the aggregation. It was shown in [117, 139] that this asymmetry in the consensus update is a source of instability. For this reason, we shall focus mainly on diffusion strategies in later chapters.

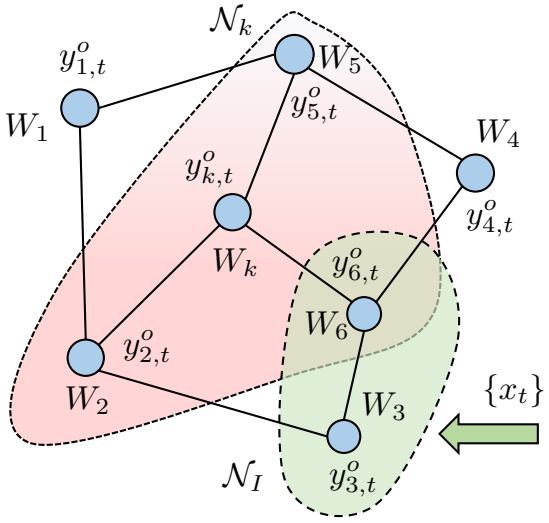


Figure 1.2: The data sample x_t at each time t is available to a subset $\mathcal{N}_I$ of agents in the network (e.g., agents 3 and 6 in the figure), and each agent k is in charge of one sub-dictionary, W_k, and the corresponding optimal sub-vector of coefficients estimated at time t, $y^o_{k,t}$. Each agent k can only exchange information with its immediate neighbors (e.g., agents 5, 2 and 6 in the figure and k itself). We use $\mathcal{N}_k$ to denote the set of neighbors of agent k.

1.2.2 Cost-of-Sum Formulations

Another important situation that leads to the "sum-of-costs" form is the "model-distributed" case, which we will study in Chapter 3. In this case, each agent in the network is only in charge of a portion of the model, and the agents work together to represent the data streams that arrive at a subset of the agents. This setup is important in large-scale machine learning applications, since the increasing amount of data samples allows us to use more sophisticated models to interpret the data. However, when the model is too large, it may not be possible to fit it into a single machine. One example is dictionary learning over large-scale models, as illustrated in Fig. 1.2. As we will reveal in Chapter 3, the problem involves a "cost-of-sum" form, which is not directly amenable to

distributed implementations. Specifically, the networked agents aim to learn a dictionary $W = [W_1, \ldots, W_N]$ by solving the following optimization problem:

$$\min_{\{W_1,\ldots,W_N\}} \quad \mathbb{E}\left[f\Big(\boldsymbol{x}_t - \sum_{k=1}^{N} W_k \boldsymbol{y}_{k,t}^o\Big) + \sum_{k=1}^{N} h_{y_k}(\boldsymbol{y}_{k,t}^o)\right] + \sum_{k=1}^{N} h_{W_k}(W_k) \tag{1.18a}$$

$$\text{s.t.} \quad W_k \in \mathcal{W}_k, \qquad k = 1, \ldots, N \tag{1.18b}$$

where W_k denotes the sub-dictionary at each agent k, $\boldsymbol{x}_t$ is the $M \times 1$ input data vector at time t, $f(u)$ in (1.18a) denotes a differentiable convex loss function for the residual error, $h_{y_k}(y_k)$ and $h_{W_k}(W_k)$ are convex (but not necessarily differentiable) regularization terms on y_k and W_k, respectively, and $\mathcal{W}_k$ denotes the convex constraint set on W_k. Moreover, for each given realization x_t, $y_{k,t}^o$ is a sub-coding vector at each agent k. $\{y_{1,t}^o, \ldots, y_{N,t}^o\}$ are defined as the solution to the following inference (sparse coding) problem:

$$\{y_{1,t}^o, \ldots, y_{N,t}^o\} = \arg \min_{\{y_1,\ldots,y_N\}} \left[f\Big(x_t - \sum_{k=1}^{N} W_k y_k\Big) + \sum_{k=1}^{N} h_{y_k}(y_k)\right] \tag{1.19}$$

Observe that the inference problem (1.19) is a regularized "cost-of-sum" problem. By using a dual decomposition technique and the concept of conjugate function, we will be able to convert the problem into the "sum-of-costs" form (1.7), which can be solved efficiently over networked agents. More specifically, we will show that problem (1.19) can be transformed into the following dual problem that assumes the "sum-of-costs" form:

$$\min_{\nu} \quad f^\star(\nu) - \nu^T x_t + \sum_{k=1}^{N} h_{y_k}^\star(W_k^T \nu) \tag{1.20a}$$

$$\text{s.t.} \quad \nu \in \mathcal{V}_f \tag{1.20b}$$

where $f^\star(\cdot)$ and $h^\star_{y_k}(\cdot)$ are the conjugate functions of $f(u)$ and $h_{y_k}(y_k)$, respectively, and $\mathcal{V}_f$ is the domain of $f^\star(\nu)$. Furthermore, for the dual "sum-of-costs" problem (1.20a)–(1.20b), the optimal solution ν_t^o can be used to update the dictionary components:

$$W_{k,t} = \Pi_{\mathcal{W}_k}\left\{\text{prox}_{\mu_w \cdot h_{W_k}}\left(W_{k,t-1} + \mu_w \nu_t^o (y_{k,t}^o)^T\right)\right\} \tag{1.21}$$

where $\Pi_{\mathcal{W}_k}(\cdot)$ denotes the projection operator onto $\mathcal{W}_k$, and $\text{prox}_{\mu_w \cdot h_{W_k}}(\cdot)$ denotes the proximal operator of $\mu_w \cdot h_{W_k}$. Note that the sub-dictionary update recursion at each agent k does not require further exchange of information among agents.

The distributed dictionary learning problem we are solving here is different from the useful work [31, 32], where it is assumed that the *entire* dictionary W [31, 32] is maintained at each agent in the network, whereas individual data samples generated by the same distribution, denoted by $\boldsymbol{x}_{k,t}$, are observed by the agents at each time t. That is, these previous works study *data distributed* formulations. What we are going to study in Chapter 3 is to find a distributed solution where each agent is only in charge of a *portion* of the dictionary (W_k for each agent k) while the incoming data, $\boldsymbol{x}_t$, is observed by only a subset of the agents. This scenario corresponds to a *model distributed* (or dictionary-distributed) formulation. A related albeit different model was considered in [43] in the context of distributed deep neural network (DNN) models over computer networks. In these models, each computer is in charge of a portion of neurons in the DNN, which exchange their private activation signals with neurons over the network to perform the classification task. As we will see in Chapter 3, our distributed model does not require exchanging either the private combination coefficients $\{y_k\}$ or the sub-dictionaries $\{W_k\}$ while still being able to model the data using the collective "wisdom" over the network. Another related but

different work is [142], where the authors study a special form of a distributed sparse basis pursuit problem under *fixed* sub-dictionaries at each agent. We instead allow the sub-dictionaries to be updated dynamically over time (rather than staying fixed) and this is accomplished without exchanging any further information after the distributed inference step — see (1.21) and also future Sec. 3.3.5.

The distributed model setting is important in practice because agents tend to be limited in their memory and computing power and they may not be able to store large dictionaries locally. Even if the agents were powerful enough, different agents may have access to different databases and different sources of information. Rather than aggregate the information in the form of large dictionaries at every single location, it is more advantageous to keep the information distributed due to potential excessive costs in exchanging large data sets, and also due to privacy considerations where different agents may not be in favor of sharing their data and dictionary. Therefore, by having distributed sub-dictionaries, and by having many agents cooperate with each other, a large model that is beyond the ability or reach of any single agent can be trained by the network in a distributed manner.

1.2.3 Social Learning

Note that in multi-agent systems, either data arrive at different agents in a distributed manner or the entire model is distributed over different agents. Therefore, in order to solve the global problem, agents need social-learning to consult with each other so that information extracted from local data streams can be propagated over the entire network. An important feature of social-learning is that it relies on limited interactions for at least two reasons. First, because the network is possibly sparsely connected, each agent in the network can only in-

teract with a limited number of intermediate one-hop neighbors. Second, due to privacy or security considerations, agents may be reluctant to share their raw or processed data more fully.

1.3 Objectives

In future chapters, we will develop algorithms that solve the aforementioned "sum-of-costs" and "cost-of-sum" problems in a distributed manner. To ensure continuous learning and adaptation to streaming data, we focus on multi-agent systems with *constant* step-sizes. As we already indicated, there are two types of learning processes involved in the dynamics of each agent k: (i) self-learning from locally sensed data and (ii) social learning from neighbors. All nodes implement the same self- and social learning structure. As a result, the learning dynamics of all nodes in the network are coupled; knowledge exploited from local data at node k will be propagated to its neighbors and from there to their neighbors in a diffusive learning process. It is expected that some global performance pattern will emerge from these localized interactions in the multi-agent system. In this dissertation, we address the following questions:

- Limit point: where does each state $\boldsymbol{w}_{k,i}$ converge to?
- Stability: under which conditions does convergence occur?
- Learning rate: how fast does convergence occur?
- Performance: how close does $\boldsymbol{w}_{k,i}$ get to the limit point?

We address the four questions by characterizing analytically the learning dynamics of the network to reveal the global behavior that emerges in the small step-size regime. A critical question to ask is whether it is possible that, under certain

conditions, the distributed strategy can achieve the same performance as the centralized strategy? The centralized strategy is the one that collects all data and has the entire model available at a powerful fusion center. This question will be addressed in later chapters.

In comparison with the existing literature [13, 21, 48, 71, 75–77, 84, 97, 109, 125, 126, 137], it is worth noting that most prior studies on distributed optimization algorithms focus on studying their performance and convergence under *diminishing* step-size conditions and for *doubly-stochastic* combination policies (i.e., matrices for which the entries on each of their columns *and* on each of their rows add up to one). These are of course useful conditions, especially when the emphasis is on solving *static* optimization problems. We focus instead on the case of *constant* step-sizes because, as explained earlier, they enable continuous adaptation and learning under drifting conditions; in contrast, diminishing step-sizes turn off learning once they approach zero. By using constant step-sizes, the resulting algorithms are able to track *dynamic* solutions that may slowly drift as the underlying problem conditions change. Moreover, we do not limit the combination policies to be doubly-stochastic; we only require condition (1.17). It turns out that left-stochastic matrices lead to superior mean-square error performance (see, e.g., expression (6.66) and also [26, 146]). Furthermore, constant step-sizes and left-stochastic combination policies enrich the learning dynamics of the network in interesting ways, as we are going to discover. In particular, under these conditions, we shall derive an interesting result that reveals how the topology of the network determines the limit point of the distributed algorithm. We will show that the combination weights steer the convergence point away from the expected solution and towards any of many possible Pareto optimal solutions. This is in contrast to commonly-used doubly-stochastic combination policies where the limit point of the network is fixed and cannot be changed regardless of the

topology. We will show that the limit point is determined by the right eigenvector that is associated with the eigenvalue at one for the matrix A. Therefore, left-stochastic policies enable the networks to converge to any of infinitely many Pareto optimal solutions. Moreover, the value of the limit point can be controlled through the selection of the Perron eigenvector.

We will also be able to characterize how close each agent in the network gets to this limit point. As a by-product of studying the transient behavior of the algorithms, we will be able to derive closed-form performance expressions for the steady-state mean-square-error (MSE) for a fairly general class of distributed strategies under broader (weaker) conditions than normally considered in the literature.

Other useful and related works in the literature appear in [13, 75–77]. These works, however, study the distribution of the error vector in steady-state under *diminishing* step-size conditions and using central limit theorem (CLT) arguments. They showed a Gaussian distribution for the error quantities in steady-state and derived an expression for the error variance but their expression naturally tends to zero as $i \to \infty$ since, under the conditions assumed in these works, the error vector $\tilde{\boldsymbol{w}}_{k,i}$ approaches zero almost surely. Such results are possible because, in the diminishing step-size case, the influence of gradient noise is annihilated by the decaying step-size. However, in the *constant* step-size regime, the influence of gradient noise is always present and seeps into the operation of the algorithm. In this case, the error vector does *not* approach zero any longer and its variance approaches instead a steady-state *positive-definite* value. Our objective is to characterize this steady-state value and to examine how it is influenced by the network topology, by the persistent gradient noise conditions, and by the data characteristics and utility functions. In the constant step-size regime, CLT argu-

ments cannot be employed anymore because the Gaussianity result does not hold any longer. Indeed, reference [145] illustrates this situation clearly; it derived an expression for the characteristic function of the limiting error distribution in the case of mean-square-error estimation and it was shown that the distribution is not Gaussian. For these reasons, the analysis in this dissertation is based on alternative techniques that do not assume any specific form for the steady-state distribution and that rely instead on the use of energy conservation arguments [34, 115, 116].

1.4 Overview of Main Results

Before we proceed to the formal analysis, we first give a brief overview of the main results that we are going to establish in this dissertation on the learning behavior of the distributed strategies (1.10)–(1.12) for sufficiently small step-sizes. Let θ denote the right eigenvector of the matrix $A = [a_{lk}]$ corresponding to the eigenvalue at one and whose entries are normalized to add up to one, i.e.,

$$A\theta = \theta, \quad \mathbb{1}^T\theta = \mathbb{1} \tag{1.22}$$

The first major conclusion is that for general *left-stochastic* primitive combination matrices A, the agents in the network will have their estimators $\boldsymbol{w}_{k,i}$ converge, in the mean-square-error sense, to the *same* vector w^o that corresponds to the unique solution of the following algebraic equation:

$$\sum_{k=1}^{N} p_k s_k(w) = 0 \tag{1.23}$$

where

$$s_k(w) \triangleq \mathbb{E}\left[\hat{\boldsymbol{s}}_{k,i}(w)|\mathcal{F}_{i-1}\right] \tag{1.24}$$

$$p_k \triangleq \theta_k \cdot \frac{\mu_k}{\mu_{\max}} \tag{1.25}$$

$$\mu_{\max} \triangleq \max_k \mu_k \tag{1.26}$$

$\mathcal{F}_{i-1}$ denotes the history of iterates up to time $i-1$, and θ_k is the kth entry of the right-eigenvector θ. For example, in the context of distributed optimization problems of the form (1.7), this result implies that for left-stochastic matrices A, the distributed strategies (1.10)–(1.12) will *not* converge to the global minimizer of the original aggregate cost (1.7), which is the unique solution to the alternative algebraic equation

$$\sum_{k=1}^{N} \nabla_w J_k(w) = 0 \tag{1.27}$$

Instead, these distributed solutions will converge to the global minimizer of the *weighted* aggregate cost $J^{\text{glob},\star}(w)$ defined in terms of the entries p_k:

$$J^{\text{glob},\star}(w) \triangleq \sum_{k=1}^{N} p_k J_k(w) \tag{1.28}$$

That is, the algorithms will converge to the unique solution of

$$\sum_{k=1}^{N} p_k \nabla_w J_k(w) = 0 \tag{1.29}$$

Result (1.29) means that the distributed strategies (1.10)–(1.12) converge to a Pareto optimal solution of the multi-objective problem

$$\min_{w} \{J_1(w), \ldots, J_N(w)\} \tag{1.30}$$

with one Pareto solution obtained for each selection of the topology parameters $\{p_k\}$. The distinction between the aggregate costs $J^{\text{glob}}(w)$ and $J^{\text{glob},\star}(w)$ does not appear in earlier studies on distributed optimization [75, 77, 97, 109, 125, 137] mainly because these studies focus on *doubly-stochastic* combination matrices, for which the entries $\{p_k\}$ will all become equal to each other for uniform step-sizes $\mu_k \equiv \mu$. In that case, the minimizations of (1.7) and (1.27) become equivalent and the solutions of (1.27) and (1.29) would then coincide. In other words, regardless of the choice of the doubly stochastic combination weights, when the $\{p_k\}$ are identical, the limit point will be unique and correspond to the solution of

$$\sum_{k=1}^{N} s_k(w) = 0 \tag{1.31}$$

In contrast, result (1.23) shows that left-stochastic combination policies add more flexibility into the behavior of the network. By selecting different combination weights, or even different topologies, the entries $\{p_k\}$ can be made to change and the limit point can be steered towards other desired Pareto optimal solutions.

The second major conclusion of the dissertation is that we will show in (5.145) further ahead that there always exist sufficiently small step-sizes such that the learning process over the network is mean-square stable. This means that the weight error vectors relative to w^o will satisfy

$$\limsup_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = O(\mu_{\max}) \tag{1.32}$$

so that the steady-state mean-square-error at each agent will be of the order of $O(\mu_{\max})$.

The third major conclusion of our analysis is that we will show that, during the convergence process towards the limit point w^o, the learning curve at each agent exhibits *three* distinct phases: Transient Phase I, Transient Phase II, and Steady-State Phase. These phases are illustrated in Fig. 1.3 and they are interpreted as follows. Let us first introduce a *reference* (centralized) procedure that is described by the following centralized-type recursion:

$$\bar{w}_{c,i} = \bar{w}_{c,i-1} - \mu_{\max} \sum_{k=1}^{N} p_k s_k(\bar{w}_{c,i-1}) \tag{1.33}$$

which is initialized at

$$\bar{w}_{c,0} = \sum_{k=1}^{N} \theta_k w_{k,0} \tag{1.34}$$

where θ_k is the kth entry of the eigenvector θ, $w_{k,0}$ is the initial value of the distributed strategy at agent k, and $\bar{w}_{c,i}$ is an $M \times 1$ vector generated by the reference recursion (1.33). The three phases of the learning curve will be shown to have the following features:

- **Transient Phase I:**

  If agents are initialized at different values, then the estimates of the various agents will initially evolve in such a way to make each $\boldsymbol{w}_{k,i}$ get closer to the reference recursion $\bar{w}_{c,i}$. The rate at which the agents approach $\bar{w}_{c,i}$ will be determined by $|\lambda_2(A)|$, the second largest eigenvalue of A in magnitude. If the agents are initialized at the same value, say, e.g., $\boldsymbol{w}_{k,0} = 0$, then the learning curves start at Transient Phase II directly.

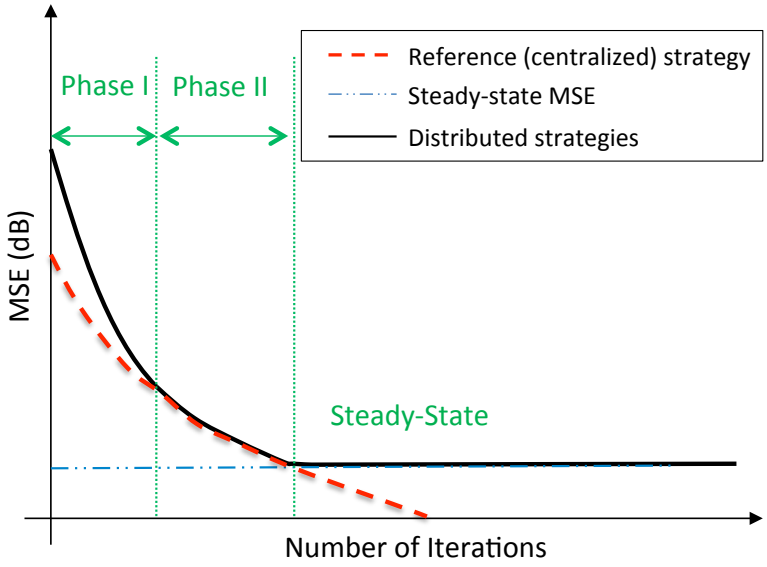


Figure 1.3: A typical mean-square-error (MSE) learning curve includes a transient stage that consists of two phases and a steady-state phase. The plot shows how the learning curve of a network of agents compares to the learning curve of a centralized reference solution. The analysis in this dissertation characterizes in detail the parameters that determine the behavior of the network (rate, stability, and performance) during each phase of the learning process.

- **Transient Phase II:**

  In this phase, the trajectories of all agents are uniformly close to the trajectory of the reference recursion; they converge in a coordinated manner to steady-state. The learning curves at this phase are well modeled by the same reference recursion (1.33) since we will show in (6.10) that:

$$\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \|\tilde{w}_{c,i}\|^2 + O(\mu_{\max}^{1/2}) \cdot \gamma_c^i + O(\mu_{\max}) \tag{1.35}$$

  Furthermore, for small step-sizes and during the later stages of this phase, $\bar{w}_{c,i}$ will be close enough to w^o and the convergence rate r will be shown to

satisfy:

$$r = \left[\rho(I_M - \mu_{\max} H_c)\right]^2 + O\big((\mu_{\max}\epsilon)^{\frac{1}{2(M-1)}}\big) \tag{1.36}$$

where $\rho(\cdot)$ denotes the spectral radius of its matrix argument, ϵ is an arbitrarily small positive number, and H_c is defined as

$$H_c \triangleq \sum_{k=1}^{N} p_k \nabla_{w^T} s_k(w^o) \tag{1.37}$$

- **Steady-State Phase:**

  The reference recursion (1.33) continues converging towards w^o so that $\|\tilde{w}_{c,i}\|^2 = \|w^o - \bar{w}_{c,i}\|^2$ will converge to zero ($-\infty$ dB in Fig. 1.3). However, for the distributed strategy (1.10)–(1.12), the mean-square-error $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \mathbb{E}\|w^o - \boldsymbol{w}_{k,i}\|^2$ at each agent k will converge to a *finite* steady-state value. We will be able to characterize this value in terms of the vector $p \triangleq \mathrm{col}\{p_1, \ldots, p_N\}$ as follows:

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \mu_{\max} \cdot \mathrm{Tr}\left\{X(p^T \otimes I_M)\mathcal{R}_v(p \otimes I_M)\right\} + o(\mu_{\max}) \tag{1.38}$$

where X is the solution to the Lyapunov equation described later in (6.42) (when $\Sigma = I$), and $o(\mu_{\max})$ denotes a strictly higher order term of $\mu_{\max}$. Expression (1.38) is a revealing result. It is a non-trivial extension of a classical result pertaining to the mean-square-error performance of stand-alone adaptive filters [54, 57, 72, 141] to the more demanding context when a multitude of adaptive agents are coupled together in a cooperative manner through a network topology. Expression (1.38) also extends the results that were developed for least-mean-square (LMS) adaptive networks (with

quadratic costs) [146] to the more general case where agents in the network are associated with general cost functions. This result has an important ramification, which we pursue later in Chapter 6. We will show there that no matter how the agents are connected to each other, there is always a way to select the combination weights such that the performance of the network is invariant to the topology. This will also imply that, for any connected topology, there is always a way to select the combination weights such that the performance of the network matches that of the centralized solution.

Finally, we will show that the convergence rate in Transient Phase II and the mean-square-error of Steady-State Phase match those of a centralized strategy described by the following recursion:

$$\boldsymbol{w}_{\text{cent},i} = \boldsymbol{w}_{\text{cent},i-1} - \mu_{\max} \sum_{k=1}^{N} p_k \hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}_{\text{cent},i-1}) \tag{1.39}$$

where the parameters $\mu_{\max}$ and $\{p_k\}$ are the same as those in the distributed strategies. That is, in the small constant step-size regime, the performance of the distributed strategies can approach a centralized strategy that collects all the data into a central agent. From the design perspective, the centralized strategy (1.39) could serve as a frame of reference for the distributed strategies. By designing the combination coefficients $\{a_{lk}\}$, we could steer the vector p in order to make the distributed strategies approach the performance of the centralized strategy.

1.5 Organization

The organization of the dissertation is summarized as follows.

- **Chapter 2:** In this chapter, we propose an adaptive diffusion mechanism to optimize a global cost function in a distributed manner over a network of agents. The cost function is assumed to be the sum of a collection of individual components, i.e., in the "sum-of-costs" form. Diffusion adaptation allows the nodes to cooperate and diffuse information in real-time; it also helps alleviate the effects of stochastic gradient noise and measurement noise through a continuous learning process.

- **Chapter 3:** Here, we examine problems involving an alternative form of global cost functions that can be expressed as regularized "cost-of-sum" forms. This formulation arises, for example, in dictionary learning over large-scale distributed models, where each agent is in charge of a portion of the dictionary and the agents collaborate to learn a best representation for the incoming data. We will show that "cost-of-sum" problems can be transformed to "sum-of-costs" problems of the form studied in Chapter 2 using the technique of dual decomposition and the concept of conjugate functions. For this reason, the problem can be solved in the dual domain using the methods developed in Chapter 2. Furthermore, besides its close connection to the "sum-of-costs" problem, the "cost-of-sum" problem has another special structure: its dual solution can provide a global gradient information. As we will explain in Chapter 3, this property is especially useful for learning large-scale distributed models.

- **Chapter 4:** From Chapters 2–3, we will conclude that both the "sum-of-costs" and "cost-of-sum" problems can be effectively solved by diffusion strategies. In this chapter, we analyze the stability and performance of the diffusion algorithm under the special case where *each individual cost function is strongly convex*. This assumption will be relaxed in later chap-

ters to only require *the aggregate cost to be strongly convex*. Analyzing the performance of diffusion strategies under the stronger assumption that each cost function is strongly convex is important since this assumption typically holds in practical applications. This is because quadratic regularization can always be added to convert each individual cost into a strongly convex function.

- **Chapter 5:** This chapter carries out a detailed transient analysis of the learning behavior of multi-agent networks, and reveals interesting results about the learning abilities of distributed strategies. Among other results, the analysis reveals how combination policies influence the learning process of networked agents, and how these policies can steer the convergence point towards any of many possible Pareto optimal solutions. The results also establish that the learning process of an adaptive network undergoes three (rather than two) well-defined stages of evolution with distinctive convergence rates during the first two stages, while attaining a finite mean-square-error (MSE) level in the last stage. The analysis reveals what aspects of the network topology influence performance directly and suggests design procedures that can optimize performance by adjusting the relevant topology parameters. Interestingly, it is further shown that, in the adaptation regime, each agent in a sparsely connected network is able to achieve the same performance level as that of a centralized stochastic-gradient strategy even for left- stochastic combination strategies. These results lead to a deeper understanding and useful insights on the convergence behavior of coupled distributed learners. The results also lead to effective design mechanisms to help diffuse information more thoroughly over networks.

- **Chapter 6:** This chapter examines the steady-state phase of distributed learning by networked agents. Apart from characterizing the performance of the individual agents, it is shown that the network induces a useful equalization effect across all agents. In this way, the performance of noisier agents is enhanced to the same level as the performance of agents with less noisy data. It is further shown that in the small step-size regime, each agent in the network is able to achieve the same performance level as that of a centralized strategy corresponding to a fully connected network. The results in this part reveal explicitly which aspects of the network topology and operation influence performance and provide important insights into the design of effective mechanisms for the processing and diffusion of information over networks.

1.6 Notation

All vectors are column vectors. We use boldface letters to denote random quantities (such as $\boldsymbol{u}_{k,i}$) and regular font to denote their realizations or deterministic variables (such as $u_{k,i}$). We use $\mathrm{diag}\{x_1, \ldots, x_N\}$ to denote a (block) diagonal matrix consisting of diagonal entries (blocks) $x_1, \ldots, x_N$, and use $\mathrm{col}\{x_1, \ldots, x_N\}$ to denote a column vector formed by stacking $x_1, \ldots, x_N$ on top of each other. The notation $x \preceq y$ means each entry of the vector x is less than or equal to the corresponding entry of the vector y, and the notation $X \preceq Y$ means each entry of the matrix X is less than or equal to the corresponding entry of the matrix Y. The notation $x = \mathrm{vec}(X)$ denotes the vectorization operation that stacks the columns of a matrix X on top of each other to form a vector x, and $X = \mathrm{vec}^{-1}(x)$ is the inverse operation. The operators ∇_w and ∇_{w^T} denote the column and row gradient vectors with respect to w. When ∇_{w^T} is applied to a column vector s,

it generates a matrix. The notation $a(\mu) = O(b(\mu))$ means that there exists a constant $C > 0$ such that $a(\mu) \leq C \cdot b(\mu)$. The notation $a(\mu) = o(b(\mu))$ means that $\lim_{\mu \to 0} a(\mu)/b(\mu) = 0$

CHAPTER 2

Sum-of-Costs Formulation

In this chapter, we propose an adaptive diffusion mechanism to optimize a global cost function in a distributed manner over a network of nodes. The cost function is assumed to be the sum of a collection of individual components, i.e., in the "sum-of-costs" form. Diffusion adaptation allows the nodes to cooperate and diffuse information in real-time; it also helps alleviate the effects of stochastic gradient noise and measurement noise through a continuous learning process. We apply the resulting distributed strategy to two applications: distributed estimation with sparse parameters and distributed localization. Compared to well-studied incremental methods, diffusion methods do not require the use of a cyclic path over the nodes and are robust to node and link failure. Diffusion methods also endow networks with adaptation abilities that enable the individual nodes to continue learning even when the cost function changes with time. The following presentation in this chapter is based on [34].

2.1 Problem Formulation

The objective is to determine, in a collaborative and distributed manner, the $M \times 1$ column vector w^o that minimizes a global cost of the form:

$$\boxed{J^{\text{glob}}(w) = \sum_{l=1}^{N} J_l(w)} \tag{2.1}$$

where $J_l(w)$, $l = 1, 2, \ldots, N$, are individual real-valued functions, defined over $w \in \mathbb{R}^M$ and assumed to be differentiable and strongly convex. Then, $J^{\text{glob}}(w)$ in (2.1) is also strongly convex so that the minimizer w^o is unique [105]. In this chapter we study the important case where the component functions $\{J_l(w)\}$ are minimized at the *same* w^o. This case is common in practice; situations abound where nodes in a network need to work cooperatively to attain a common objective (such as tracking a target, locating the source of chemical leak, estimating a physical model, or identifying a statistical distribution). This scenario is also frequent in the context of biological networks. For example, during the foraging behavior of an animal group, each agent in the group is interested in determining the *same* vector w^o that corresponds to the location of the food source or the location of the predator [138]. This scenario is equally common in online distributed machine learning problems, where data samples are often generated from the same underlying distribution and they are processed in a distributed manner by different nodes (e.g., [44, 133]). Later in Chapters 4–6, we will show that diffusion strategies are also applicable to the case when the $\{J_l(w)\}$ have different individual minimizers and nodes would converge instead to a Pareto-optimal solution.

Our strategy to optimize the global cost $J^{\text{glob}}(w)$ in a distributed manner is based on three steps. First, using a second-order Taylor series expansion, we

argue that $J^{\text{glob}}(w)$ can be approximated by an alternative localized cost that is amenable to distributed optimization — see (2.11). Second, each individual node optimizes this alternative cost via a steepest-descent procedure that relies solely on interactions within the neighborhood of the node. Finally, the local estimates for w^o are spatially combined by each node and the procedure repeats itself in real-time. The approach in this chapter extends the derivation from [26, 115], which focused on diffusion strategies for mean-square-error estimation problems (i.e., quadratic costs).

To motivate the approach, we start by introducing a set of nonnegative coefficients $\{c_{l,k}\}$ that satisfy:

$$\sum_{k=1}^{N} c_{l,k} = 1, \quad c_{l,k} = 0 \text{ if } l \notin \mathcal{N}_k, \quad l = 1, 2, \ldots, N \tag{2.2}$$

where $\mathcal{N}_k$ denotes the neighborhood of node k (including node k itself); the neighbors of node k consist of all nodes with which node k can share information. Each $c_{l,k}$ represents a weight value that node k assigns to information arriving from its neighbor l. Condition (2.2) states that the sum of all weights leaving each node l should be one. Using the coefficients $\{c_{l,k}\}$, we can express $J^{\text{glob}}(w)$ from (2.1) as

$$J^{\text{glob}}(w) = J_k^{\text{loc}}(w) + \sum_{l \neq k}^{N} J_l^{\text{loc}}(w) \tag{2.3}$$

where

$$J_k^{\text{loc}}(w) \triangleq \sum_{l \in \mathcal{N}_k} c_{l,k} J_l(w) \tag{2.4}$$

In other words, for each node k, we are introducing a new local cost function,

$J_k^{\rm loc}(w)$, which corresponds to a weighted combination of the costs of its neighbors. Since the $\{c_{l,k}\}$ are all nonnegative and each $J_l(w)$ is strongly convex, then $J_k^{\rm loc}(w)$ is also a strongly convex function.

Now, each $J_l^{\rm loc}(w)$ in the second term of (2.3) can be approximated via a second-order Taylor series expansion as:

$$J_l^{\rm loc}(w) \approx J_l^{\rm loc}(w^o) + \|w - w^o\|_{\Gamma_l}^2 \tag{2.5}$$

where $\Gamma_l = \frac{1}{2}\nabla_w^2 J_l^{\rm loc}(w^o)$ is the (scaled) Hessian matrix relative to w and evaluated at $w = w^o$, and the notation $\|a\|_\Sigma^2$ denotes $a^T\Sigma a$ for any weighting matrix Σ. The analysis in the subsequent sections will show that the second-order approximation (2.5) is sufficient to ensure mean-square convergence of the resulting diffusion algorithm. Now, substituting (2.5) into the right-hand side of (2.3) gives:

$$J^{\rm glob}(w) \approx J_k^{\rm loc}(w) + \sum_{l\neq k} \|w - w^o\|_{\Gamma_l}^2 + \sum_{l\neq k} J_l^{\rm loc}(w^o) \tag{2.6}$$

The last term in the above expression does not depend on the unknown w. Therefore, we can ignore it so that optimizing $J^{\rm glob}(w)$ is approximately equivalent to optimizing the following alternative cost:

$$J^{\rm glob'}(w) \triangleq J_k^{\rm loc}(w) + \sum_{l\neq k} \|w - w^o\|_{\Gamma_l}^2 \tag{2.7}$$

2.2 Diffusion Adaptation Strategies

2.2.1 Iterative Diffusion Solution

Expression (2.7) relates the original global cost (2.1) to the newly-defined local cost function $J_k^{\rm loc}(w)$. The relation is through the second term on the right-

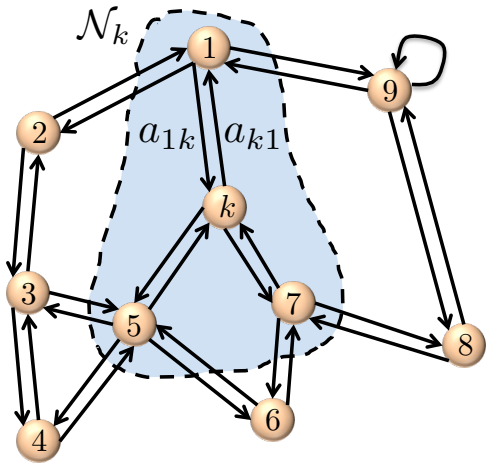


Figure 2.1: A network with N nodes; a cost function $J_k(w)$ is associated with each node k. The set of neighbors of node k is denoted by $\mathcal{N}_k$; this set consists of all nodes with which node k can share information.

hand side of (2.7), which corresponds to a sum of quadratic terms involving the minimizer w^o. Obviously, w^o is not available at node k since the nodes wish to estimate w^o. Likewise, not all Hessian matrices Γ_l are available to node k. Nevertheless, expression (2.7) suggests a useful approximation that leads to a powerful distributed solution, as we proceed to explain.

Our first step is to replace the global cost $J^{\text{glob}'}(w)$ by a reasonable *localized* approximation for it at every node k. Thus, initially we limit the summation on the right-hand side of (2.7) to the neighbors of node k and introduce the cost function:

$$J_k^{\text{glob}'}(w) \triangleq J_k^{\text{loc}}(w) + \sum_{l \in \mathcal{N}_k \backslash \{k\}} \|w - w^o\|_{\Gamma_l}^2 \tag{2.8}$$

Compared with (2.7), the last term in (2.8) involves only quantities that are available in the neighborhood of node k. The argument involving steps (2.5)–

(2.8) therefore shows us one way by which we can adjust the earlier local cost function $J_k^{\rm loc}(w)$ defined in (2.4) by adding to it the last term that appears in (2.8). Doing so, we end up replacing $J_k^{\rm loc}(w)$ by $J_k^{\rm glob'}(w)$, and this new localized cost function preserves the second term in (2.3) up to a second-order approximation. This correction will help lead to a diffusion step (see (2.14)–(2.15)).

Now, observe that the cost in (2.8) includes the quantities $\{\Gamma_l\}$, which belong to the neighbors of node k. These quantities may or may not be available. If they are known, then we can proceed with (2.8) and rely on the use of the Hessian matrices Γ_l in the subsequent development. Nevertheless, the more interesting situation in practice is when these Hessian matrices are not known beforehand (especially since they depend on the unknown w^o). For this reason, we approximate each Γ_l in (2.8) by a multiple of the identity matrix, say,

$$\Gamma_l \approx b_{l,k} I_M \tag{2.9}$$

for some nonnegative coefficients $\{b_{l,k}\}$; observe that we are allowing the coefficient $b_{l,k}$ to vary with the node index k. Such approximations are common in stochastic approximation theory and help reduce the complexity of the resulting algorithms — see [105, pp.20–28] and [116, pp.142–147]. Approximation (2.9) is reasonable since, in view of the Rayleigh-Ritz characterization of eigenvalues [59], we can always bound the weighted squared norm $\|w - w^o\|_{\Gamma_l}^2$ by the unweighted squared norm as follows

$$\lambda_{\min}(\Gamma_l) \cdot \|w - w^o\|^2 \leq \|w - w^o\|_{\Gamma_l}^2 \leq \lambda_{\max}(\Gamma_l) \cdot \|w - w^o\|^2$$

Thus, we replace (2.8) by

$$J_k^{\mathrm{glob}''}(w) \triangleq J_k^{\mathrm{loc}}(w) \; + \sum_{l \in \mathcal{N}_k \backslash \{k\}} b_{l,k} \|w - w^o\|^2 \tag{2.10}$$

As the derivation will show, we do not need to worry at this stage about how the scalars $\{b_{l,k}\}$ are selected; they will be embedded into other combination weights that the designer selects. If we replace $J_k^{\mathrm{loc}}(w)$ by its definition (2.4), we can rewrite (2.10) as

$$J_k^{\mathrm{glob}''}(w) = \sum_{l \in \mathcal{N}_k} c_{l,k} J_l(w) + \sum_{l \in \mathcal{N}_k \backslash \{k\}} b_{l,k} \|w - w^o\|^2 \tag{2.11}$$

Observe that cost (2.11) is different for different nodes; this is because the choices of the weighting scalars $\{c_{l,k}, b_{l,k}\}$ vary across nodes k; moreover, the neighborhoods vary with k. Nevertheless, these localized cost functions now constitute the important starting point for the development of diffusion strategies for the online and distributed optimization of (2.1).

Each node k can apply a steepest-descent iteration to minimize $J_k^{\mathrm{glob}''}(w)$ by moving along the negative direction of the gradient (column) vector of the cost function, namely,

$$w_{k,i} = w_{k,i-1} - \mu_k \sum_{l \in \mathcal{N}_k} c_{l,k} \nabla_w J_l(w_{k,i-1}) - \mu_k \sum_{l \in \mathcal{N}_k \backslash \{k\}} 2 b_{l,k} (w_{k,i-1} - w^o), \quad i \geq 0 \tag{2.12}$$

where $w_{k,i}$ denotes the estimate for w^o at node k at time i, and μ_k denotes a small *constant* positive step-size parameter. While vanishing step-sizes, such as $\mu_k(i) = 1/i$, can be used in (2.12), we consider the case of constant step-sizes. This is because we are interested in distributed strategies that are able to

continue adapting and learning. An important question to address therefore is how close each of the $w_{k,i}$ gets to the optimal solution w^o; we answer this question later in Chapters 4–6 under general conditions by means of a mean-square-error convergence analysis. It will be seen then that the mean-square-error (MSE) of the algorithm will be of the order of the step-size; hence, sufficiently small step-sizes will lead to sufficiently small MSEs.

Expression (2.12) adds two correction terms to the previous estimate, $w_{k,i-1}$, in order to update it to $w_{k,i}$. The correction terms can be added one at a time in a succession of two steps, for example, as:

$$\psi_{k,i} = w_{k,i-1} - \mu_k \sum_{l \in \mathcal{N}_k} c_{l,k} \nabla_w J_l(w_{k,i-1}) \tag{2.13}$$

$$w_{k,i} = \psi_{k,i} - \mu_k \sum_{l \in \mathcal{N}_k \backslash \{k\}} 2b_{l,k}(w_{k,i-1} - w^o) \tag{2.14}$$

Step (2.13) updates $w_{k,i-1}$ to an intermediate value $\psi_{k,i}$ by using a *combination* of local gradient vectors. Step (2.14) further updates $\psi_{k,i}$ to $w_{k,i}$ by using a *combination* of local estimates. However, two issues arise while examining (2.14):

(a) First, iteration (2.14) requires knowledge of the optimizer w^o. However, all nodes are running similar updates to estimate the w^o. By the time node k wishes to apply (2.14), each of its neighbors would have performed its own update similar to (2.13) and would have available their intermediate estimates, $\{\psi_{l,i}\}$. Therefore, we replace w^o in (2.14) by $\psi_{l,i}$. This step helps diffuse information over the network and brings into node k information that exists beyond its immediate neighborhood; this is because each $\psi_{l,i}$ is influenced by data from the neighbors of node l. We observe that this diffusive term arises from the quadratic approximation (2.5) we have made to the second term in (2.3).

(b) Second, the intermediate value $\psi_{k,i}$ in (2.13) is generally a better estimate for w^o than $w_{k,i-1}$ since it is obtained by incorporating information from the neighbors through (2.13). Therefore, we further replace $w_{k,i-1}$ in (2.14) by $\psi_{k,i}$. This step is reminiscent of incremental-type approaches to optimization, which have been widely studied in the literature [9, 88, 96, 108].

Performing the substitutions described in items (a) and (b) into (2.14), we obtain:

$$w_{k,i} = \psi_{k,i} - \mu_k \sum_{l \in \mathcal{N}_k \backslash \{k\}} 2b_{l,k}(\psi_{k,i} - \psi_{l,i}) \tag{2.15}$$

Now introduce the coefficients

$$a_{l,k} \triangleq 2\mu_k b_{l,k} \quad (l \neq k), \quad a_{k,k} \triangleq 1 - \mu_k \sum_{l \in \mathcal{N}_k \backslash \{k\}} 2b_{l,k} \tag{2.16}$$

Note that the $\{a_{l,k}\}$ are nonnegative for $l \neq k$ and $a_{k,k} \geq 0$ for sufficiently small step-sizes. Moreover, the coefficients $\{a_{l,k}\}$ satisfy

$$\sum_{l=1}^{N} a_{l,k} = 1, \quad a_{l,k} = 0 \text{ if } l \notin \mathcal{N}_k \tag{2.17}$$

Using (2.16) in (2.15), we arrive at the following Adapt-then-Combine (ATC) diffusion strategy (whose structure is the same as the ATC algorithm originally proposed in [25, 26, 89] for mean-square-error estimation):

$$\text{(ATC)} \quad \boxed{\begin{aligned} \psi_{k,i} &= w_{k,i-1} - \mu_k \sum_{l \in \mathcal{N}_k} c_{l,k} \nabla_w J_l(w_{k,i-1}) \\ w_{k,i} &= \sum_{l \in \mathcal{N}_k} a_{l,k} \psi_{l,i} \end{aligned}} \tag{2.18}$$

To run algorithm (2.18), we only need to select combination coefficients $\{a_{l,k}, c_{l,k}\}$

satisfying (2.2) and (2.17), respectively; there is no need to worry about the intermediate coefficients $\{b_{l,k}\}$ any more, since they have been blended into the $\{a_{l,k}\}$. The ATC algorithm (2.18) involves two steps. In the first step, node k receives gradient vector information from its neighbors and uses it to update its estimate $w_{k,i-1}$ to an intermediate value $\psi_{k,i}$. All other nodes in the network are performing a similar step and generating their intermediate estimate $\psi_{l,i}$. In the second step, node k aggregates the estimates $\{\psi_{l,i}\}$ of its neighbors and generates $w_{k,i}$. Again, all other nodes are performing a similar step. Similarly, if we reverse the order of steps (2.13) and (2.14) to implement (2.12), we can motivate the following alternative Combine-then-Adapt (CTA) diffusion strategy (whose structure is similar to the CTA algorithm originally proposed in [23, 25, 26, 87, 89, 90, 118] for mean-square-error estimation):

$$\text{(CTA)}\quad \boxed{\begin{aligned} \psi_{k,i-1} &= \sum_{l\in\mathcal{N}_k} a_{l,k} w_{l,i-1} \\ w_{k,i} &= \psi_{k,i-1} - \mu_k \sum_{l\in\mathcal{N}_k} c_{l,k} \nabla_w J_l(\psi_{k,i-1}) \end{aligned}} \tag{2.19}$$

Adaptive diffusion strategies of the above ATC and CTA types were first proposed and extended in [23–27, 87, 89, 90, 118] for the solution of distributed mean-square-error, least-squares, and state-space estimation problems over networks. The special form of ATC strategy (2.18) for minimum-mean-square-error estimation is listed further ahead as Eq. (2.30) in Example 2.3; the same strategy as (2.30) was used [126] albeit with a vanishing step-size sequence to ensure convergence towards consensus. A special case of the diffusion strategy (2.19) (corresponding to choosing $c_{l,k} = 0$ for $l \neq k$ and $c_{k,k} = 1$, i.e., without sharing gradient information) was used in the works [14, 109, 125] to solve distributed optimization problems that require all nodes to reach agreement about w^o by

relying on step-sizes that decay to zero with time. Diffusion recursions of the forms (2.18) and (2.19) are more general than these earlier investigations in a couple of respects. First, they do not only diffuse the local estimates, but they can also diffuse the local gradient vectors. In other words, two sets of combination coefficients $\{a_{l,k}, c_{l,k}\}$ are used. Second, the combination weights $\{a_{l,k}\}$ are not required to be doubly stochastic (which would require both the rows and columns of the weighting matrix $A = [a_{l,k}]$ to add up to one; as seen from (2.17), we only require the entries on the columns of A to add up to one). Finally, and most importantly, the step-size parameters $\{\mu_k\}$ in (2.18) and (2.19) are not required to depend on the time index i and are not required to vanish as $i \to \infty$. Instead, they can assume constant values, which is critical to endow the network with *continuous* adaptation and learning abilities (otherwise, when step-sizes die out, the network stops learning). Constant step-sizes also endow networks with tracking abilities, in which case the algorithms can track time changes in the optimal w^o.

We note that these strategies differ in important ways from traditional consensus-based distributed solutions, which are of the following form [12, 75, 97, 98]:

$$w_{k,i} = \sum_{l \in \mathcal{N}_k} a_{l,k} w_{k,i-1} - \mu_k(i) \cdot \nabla_w J_l(w_{k,i-1}) \tag{2.20}$$

usually with a time-variant step-size sequence, $\mu_k(i)$, that decays to zero. For example, if we set $C \triangleq [c_{l,k}] = I$ in the CTA algorithm (2.19) and substitute the combination step into the adaptation step, we obtain:

$$w_{k,i} = \sum_{l \in \mathcal{N}_k} a_{l,k} w_{k,i-1} - \mu_k \nabla_w J_l\Big(\sum_{l \in \mathcal{N}_k} a_{l,k} w_{k,i-1} \Big) \tag{2.21}$$

Thus, note that the gradient vector in (2.21) is evaluated at $\psi_{k,i-1}$, while in

(2.20) it is evaluated at $w_{k,i-1}$. Since $\psi_{k,i-1}$ already incorporates information from neighbors, we would expect the diffusion algorithm to perform better. Actually, it is shown in [117, 139] that, for mean-square-error estimation problems, diffusion strategies achieve higher convergence rate and lower mean-square-error than consensus strategies due to these differences in the dynamics of the algorithms.

2.2.2 Adaptive Diffusion Solution

The diffusion algorithms (2.18) and (2.19) depend on sharing local gradient vectors $\nabla_w J_l(\cdot)$. In many cases of practical relevance, the exact gradient vectors are not available and approximations are instead used. We model the inaccuracy in the gradient vectors as some *random* additive noise component, say, of the form:

$$\widehat{\nabla_w J_l}(w) = \nabla_w J_l(w) + \boldsymbol{v}_{l,i}(w) \tag{2.22}$$

where $\boldsymbol{v}_{l,i}(\cdot)$ denotes the perturbation and is often referred to as gradient noise. Note that we are using a boldface symbol $\boldsymbol{v}$ to refer to the gradient noise since it is generally stochastic in nature. Using the perturbed gradient vectors (4.8), the diffusion algorithms (2.18)–(2.19) become the following:

$$\text{(ATC)} \quad \boxed{\begin{aligned} \boldsymbol{\psi}_{k,i} &= \boldsymbol{w}_{k,i-1} - \mu_k \sum_{l\in\mathcal{N}_k} c_{l,k} \widehat{\nabla_w J_l}(\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{w}_{k,i} &= \sum_{l\in\mathcal{N}_k} a_{l,k} \boldsymbol{\psi}_{l,i} \end{aligned}} \tag{2.23}$$

$$\text{(CTA)}\quad \boxed{\begin{aligned} \boldsymbol{\psi}_{k,i-1} &= \sum_{l\in\mathcal{N}_k} a_{l,k}\boldsymbol{w}_{l,i-1} \\ \boldsymbol{w}_{k,i} &= \boldsymbol{\psi}_{k,i-1} - \mu_k \sum_{l\in\mathcal{N}_k} c_{l,k}\widehat{\nabla_w J_l}(\boldsymbol{\psi}_{k,i-1}) \end{aligned}} \tag{2.24}$$

Example 2.1. Assume the individual cost $J_l(w)$ at node l can be expressed as the expected value of a certain loss function $Q_l(\cdot,\cdot)$, i.e., $J_l(w) = \mathbb{E}\{Q_l(w, \boldsymbol{x}_{l,i})\}$, where the expectation is with respect to the randomness in the data samples $\{\boldsymbol{x}_{l,i}\}$ that are collected at node l at time i. Then, if we replace the true gradient $\nabla_w J_l(w)$ with its stochastic gradient approximation $\widehat{\nabla_w J_l}(w) = \nabla_w Q_l(w, \boldsymbol{x}_{l,i})$, we find that the gradient noise in this case can be expressed as

$$\boldsymbol{v}_{l,i}(w) = \nabla_w Q_l(w, \boldsymbol{x}_{l,i}) - \nabla_w \mathbb{E}\{Q_l(w, \boldsymbol{x}_{l,i})\} \tag{2.25}$$

□

Example 2.2. Consider an example in which the loss function at node l is chosen to be of the following quadratic form:

$$Q_l(w, \{\boldsymbol{u}_{l,i}, \boldsymbol{d}_l(i)\}) = |\boldsymbol{d}_l(i) - \boldsymbol{u}_{l,i}w|^2$$

for some scalars $\{\boldsymbol{d}_l(i)\}$ and $1 \times M$ regression vectors $\{\boldsymbol{u}_{l,i}\}$. The corresponding cost function is then:

$$J_l(w) = \mathbb{E}|\boldsymbol{d}_l(i) - \boldsymbol{u}_{l,i}w|^2 \tag{2.26}$$

Assume further that the data $\{\boldsymbol{u}_{l,i}, \boldsymbol{d}_l(i)\}$ satisfy the linear regression model

$$\boldsymbol{d}_l(i) = \boldsymbol{u}_{l,i}w^o + \boldsymbol{z}_l(i) \tag{2.27}$$

where the regressors $\{\boldsymbol{u}_{l,i}\}$ are zero mean and independent over time with covariance matrix $R_{u,l} = \mathbb{E}\{\boldsymbol{u}_{l,i}^T \boldsymbol{u}_{l,i}\}$, and the noise sequence $\{\boldsymbol{z}_k(j)\}$ is also zero mean, white, with variance $\sigma_{z,k}^2$, and independent of the regressors $\{\boldsymbol{u}_{l,i}\}$ for all l, k, i, j. Then, using (2.27) and (2.25), the gradient noise in this case can be expressed as:

$$\boldsymbol{v}_{l,i}(\boldsymbol{w}) = 2(R_{u,l} - \boldsymbol{u}_{l,i}^T \boldsymbol{u}_{l,i})(w^o - \boldsymbol{w}) - 2\boldsymbol{u}_{l,i}^T \boldsymbol{z}_l(i) \tag{2.28}$$

□

Example 2.3. Quadratic costs of the form (2.26) are common in mean-square-error estimation for linear regression models of the type (2.27). If we use instantaneous approximations, as is common in the context of stochastic approximation and adaptive filtering [64, 105, 116], then the actual gradient $\nabla_w J_l(w)$ can be approximated by

$$\begin{aligned} \widehat{\nabla_w J_l}(w) &= \nabla_w Q_l(w, \{\boldsymbol{u}_{l,i}, \boldsymbol{d}_l(i)\}) \\ &= -2\boldsymbol{u}_{l,i}^T [\boldsymbol{d}_l(i) - \boldsymbol{u}_{l,i} w] \end{aligned} \tag{2.29}$$

Substituting into (2.23)–(2.24), and assuming $C = I$ for illustration purposes, we arrive at the following ATC and CTA diffusion strategies originally proposed and extended in [25,26,87,89,90,118] for the solution of distributed mean-square-error estimation problems:

$$\text{(ATC)} \quad \boxed{\begin{aligned} \boldsymbol{\psi}_{k,i} &= \boldsymbol{w}_{k,i-1} + 2\mu_k \boldsymbol{u}_{k,i}^T [\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} \boldsymbol{w}_{k,i-1}] \\ \boldsymbol{w}_{k,i} &= \sum_{l \in \mathcal{N}_k} a_{l,k} \boldsymbol{\psi}_{l,i} \end{aligned}} \tag{2.30}$$

$$\text{(CTA)}\quad \boxed{\begin{aligned}&\boldsymbol{\psi}_{k,i-1} = \sum_{l\in\mathcal{N}_k} a_{l,k}\boldsymbol{w}_{l,i-1}\\ &\boldsymbol{w}_{k,i} = \boldsymbol{\psi}_{k,i-1} + 2\mu_k \boldsymbol{u}_{k,i}^T[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{\psi}_{k,i-1}]\end{aligned}} \tag{2.31}$$

□

2.3 Simulation Results

In this section we illustrate the performance of the diffusion strategies (2.23)–(2.24) by considering two applications. We consider a randomly generated connected network topology with a cyclic path so that the incremental strategy could also be implemented and compared. There are a total of $N = 10$ nodes in the network, and nodes are assumed to be connected when they are close enough geographically. In the simulations, we consider two applications: a regularized least-mean-squares estimation problem with sparse data, and a collaborative localization problem.

2.3.1 Distributed Estimation with Sparse Data

Assume each node k has access to data $\{\boldsymbol{U}_{k,i}, \boldsymbol{d}_{k,i}\}$, generated according to the following model:

$$\boldsymbol{d}_{k,i} = \boldsymbol{U}_{k,i} w^o + \boldsymbol{v}_{k,i} \tag{2.32}$$

where $\{\boldsymbol{U}_{k,i}\}$ is a sequence of $K \times M$ i.i.d. Gaussian random matrices, the entries of each $\boldsymbol{U}_{k,i}$ are i.i.d. Gaussian random variables with zero mean and unit variance, and $\boldsymbol{v}_{k,i} \sim \mathcal{N}(0, \sigma_v^2 I_K)$ is the measurement noise that is temporally and spatially white and is independent of $\boldsymbol{U}_{l,j}$ for all k, l, i, j. Our objective is

to estimate w^o from the data set $\{\boldsymbol{U}_{k,i}, \boldsymbol{d}_{k,i}\}$ in a distributed manner. In many applications, the vector w^o is sparse such as $w^o = [1\ 0\ \ldots\ 0\ 1]^T$. One way to search for sparse solutions is to consider a global cost function of the following form:

$$J^{\text{glob}}(w) = \sum_{l=1}^{N} \mathbb{E}\|\boldsymbol{d}_{l,i} - \boldsymbol{U}_{l,i}w\|_2^2 + \gamma R(w) \tag{2.33}$$

where $R(w)$ and γ are the regularization function and regularization factor, respectively. A popular choice is $R(w) = \|w\|_1$, which helps enforce sparsity and is convex. However, this choice is non-differentiable, and we would need to apply sub-gradient methods [105, pp.138–144] for a proper implementation. Instead, we use the following twice-differentiable approximation for $\|w\|_1$:

$$R(w) = \sum_{m=1}^{M} \sqrt{[w]_m^2 + \epsilon^2} \tag{2.34}$$

where $[w]_m$ denotes the m-th entry of w, and ϵ is a small number. We see that, as ϵ goes to zero, $R(w) \approx \|w\|_1$. Obviously, $R(w)$ is convex, and we can apply the diffusion algorithms to minimize (2.33) in a distributed manner. To do this, we decompose the global cost as a sum of N individual costs:

$$J_l(w) = \mathbb{E}\|\boldsymbol{d}_{l,i} - \boldsymbol{U}_{l,i}w\|_2^2 + \frac{\gamma}{N}R(w), \quad l = 1, \ldots, N \tag{2.35}$$

Then, by algorithms (2.18) and (2.19), each node k would update its estimate of w^o by using the gradient vectors of $\{J_l(w)\}_{l\in\mathcal{N}_k}$, which are given by:

$$\nabla_w J_l(w) = 2\mathbb{E}\left(\boldsymbol{U}_{l,i}^T\boldsymbol{U}_{l,i}\right)w - 2\mathbb{E}\left(\boldsymbol{U}_{l,i}^T\boldsymbol{d}_{l,i}\right) + \frac{\gamma}{N}\nabla_w R(w) \tag{2.36}$$

However, the nodes are assumed to have access to measurements $\{U_{l,i}, d_{l,k}\}$ and not to the second-order moments $\mathbb{E}\left(\boldsymbol{U}_{l,i}^T \boldsymbol{U}_{l,i}\right)$ and $\mathbb{E}\left(\boldsymbol{U}_{l,i}^T \boldsymbol{d}_{l,i}\right)$. In this case, nodes can use the available measurements to approximate the gradient vectors in (2.23) and (2.24) as:

$$\widehat{\nabla}_w J_l(w) = 2U_{l,i}^T \left[U_{l,i} w - d_{l,i}\right] + \frac{\gamma}{N} \nabla_w R(w) \tag{2.37}$$

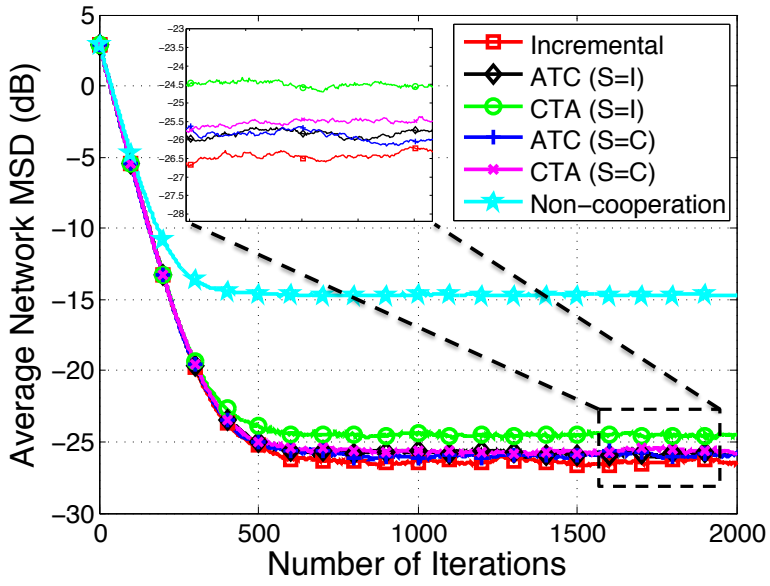
where

$$\nabla_w R(w) = \left[\frac{[w]_1}{\sqrt{[w]_1^2 + \epsilon^2}} \quad \cdots \quad \frac{[w]_N}{\sqrt{[w]_N^2 + \epsilon^2}}\right]^T \tag{2.38}$$

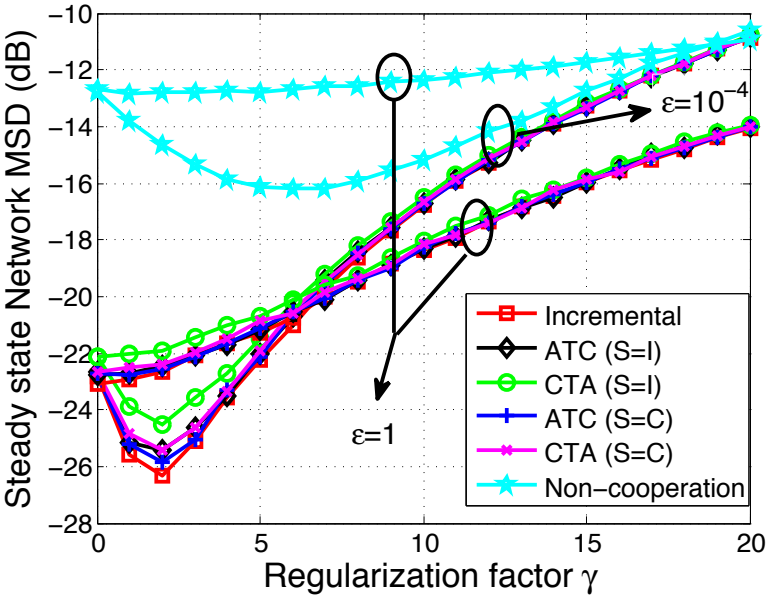
In the simulation, we set $M = 50$, $K = 5$, $\sigma_v^2 = 1$, and $w^o = [1\ 0\ \ldots\ 0\ 1]^T$. We apply both diffusion and incremental methods to solve the distributed learning problem, where the incremental approach [9, 88, 96, 108, 117] uses the following construction to determine $\boldsymbol{w}_i$:

$$\begin{cases} \text{Start with } \boldsymbol{\psi}_{0,i} = \boldsymbol{w}_{i-1} \text{ at the node at the beginning of the incremental cycle} \\ \text{Cycle through the nodes :} \\ \qquad \boldsymbol{\psi}_{k,i} = \boldsymbol{\psi}_{k-1,i} - \mu_k \widehat{\nabla_w J_k}(\boldsymbol{\psi}_{k-1,i}), \quad k = 1, \ldots, N \\ \text{Set } \boldsymbol{w}_i \leftarrow \boldsymbol{\psi}_{N,i} \\ \text{Repeat} \end{cases} \tag{2.39}$$

The results are averaged over 100 trials. The step-sizes for ATC and CTA are set to $\mu = 10^{-3}$, and the step-size for the incremental algorithm is set to $\mu = 10^{-3}/N$. This is because the incremental algorithm cycles through all N nodes every iteration. We therefore need to ensure the same convergence rate for both algorithms for a fair comparison [127, 144]. For ATC and CTA strategies, we use simple averaging weights for the combination step, and for ATC and CTA with gradient



(a) Learning curves ($\gamma = 2$ and $\epsilon = 10^{-3}$).



(b) Steady-state MSD ($\mu = 10^{-3}$).

Figure 2.2: Transient and steady-state performance of distributed estimation with sparse data.

exchange, we use Metropolis weights for $\{c_{l,k}\}$ to combine the gradients (see Table III in [26]). Fig. 2.2(a) shows the learning curves for different algorithms for $\gamma = 2$

and $\epsilon = 10^{-3}$. We see that the diffusion and incremental schemes have similar performance, and both of them have about 10 dB gain over the non-cooperation case. To examine the impact of the parameter ϵ and the regularization factor γ, we show the steady-state MSD for different values of γ and ϵ in Fig. 2.2(b). When ϵ is small ($\epsilon = 10^{-4}$), adding a reasonable regularization ($\gamma = 1 \sim 4$) decreases the steady-state MSD (even for the individual case). However, when ϵ is large ($\epsilon = 1$), expression (2.34) is no longer a good approximation for $\|w\|_1$, and regularization does not improve the MSD.

2.3.2 Distributed Collaborative Localization

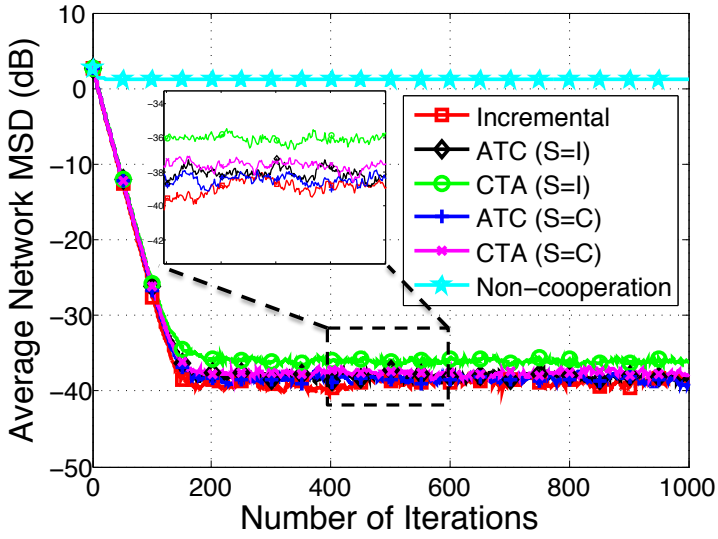
The previous example deals with a convex cost (2.33). Now, we consider a localization problem that has a non-convex cost function and apply the same diffusion strategies to its solution. Assume each node is interested in locating a common target located at $w^o = [0\ 0]^T$. Each node k knows its position x_k and has a noisy measurement of the squared distance to the target:

$$\boldsymbol{d}_k(i) = \|w^o - x_k\|^2 + \boldsymbol{v}_k(i), \quad k = 1, 2, \ldots, N$$

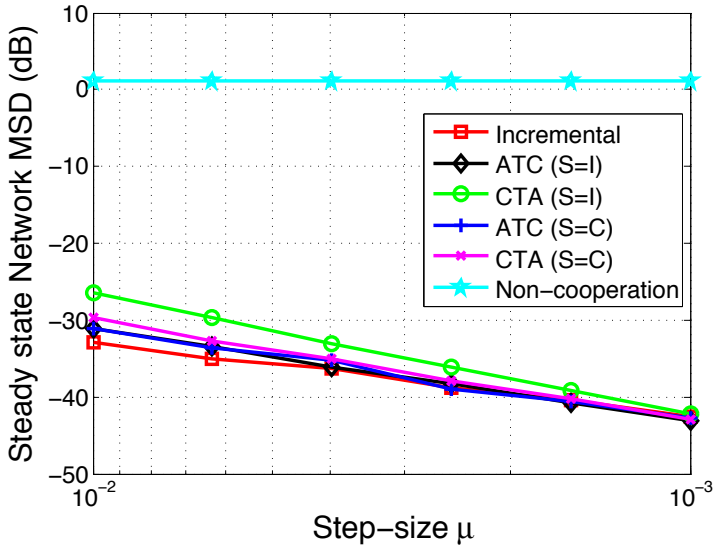
where $\boldsymbol{v}_k(i) \sim \mathcal{N}(0, \sigma_{v,k}^2)$ is the measurement noise of node k at time i. The component cost function $J_k(w)$ at node k is chosen as

$$J_k(w) = \mathbb{E}\left|\boldsymbol{d}_k(i) - \|w - x_k\|^2\right|^2 \tag{2.40}$$

If each node k minimizes $J_k(w)$ individually, it is not possible to solve for w^o. Therefore, we should use information from other nodes, and instead seek to min-



(a) Learning curves for stationary target ($\mu = 0.0025$).

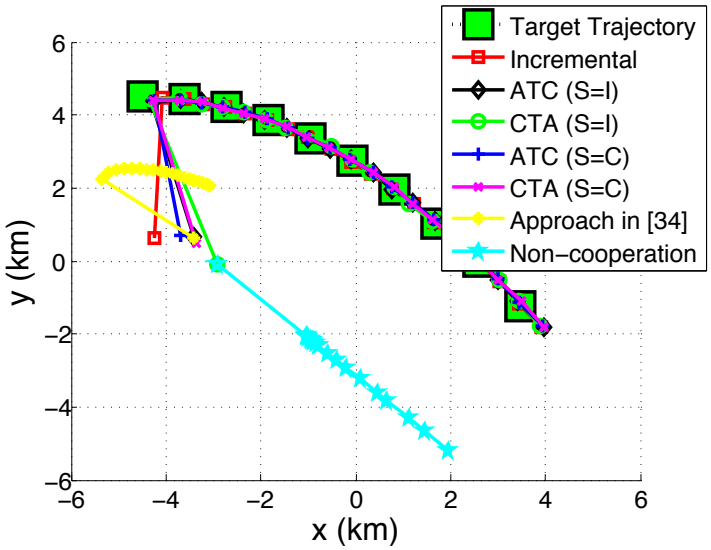


(b) Steady-state performance for stationary target.

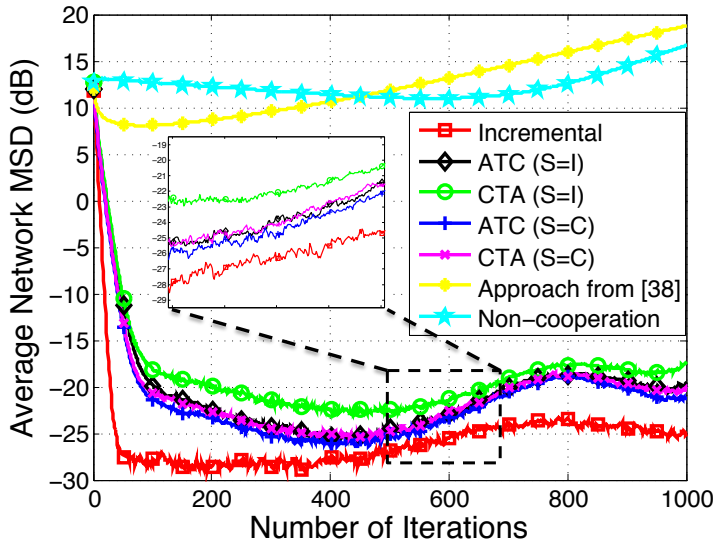
Figure 2.3: Performance of distributed localization for a stationary target.

imize the following global cost:

$$J^{\text{glob}}(w) = \sum_{k=1}^{N} \mathbb{E}\left|\boldsymbol{d}_k(i) - \|w - x_k\|^2\right|^2 \tag{2.41}$$



(a) Tracking a moving-target by node 1 ($\mu = 0.01$).



(b) Learning curves for moving target ($\mu = 0.01$).

Figure 2.4: Performance of distributed localization for a target. Diffusion strategies employ constant step-sizes, which enable continuous adaptation and learning even when the target moves (which corresponds to a changing cost function).

This problem arises, for example, in cellular communication systems, where multiple base-stations are interested in locating users using the measured distances between themselves and the user [119]. Diffusion algorithms (2.18) and (2.19)

can be applied to solve the problem in a distributed manner. Each node k would update its estimate of w^o by using the gradient vectors of $\{J_l(w)\}_{l\in\mathcal{N}_k}$, which are given by:

$$\nabla_w J_l(w) = -4\,\mathbb{E}\boldsymbol{d}_l(i)\,(w - x_l) + 4\|w - x_l\|^2(w - x_l) \tag{2.42}$$

However, the nodes are assumed to have access to measurements $\{d_l(i), x_l\}$ and not to $\mathbb{E}\boldsymbol{d}_l(i)$. In this case, nodes can use the available measurements to approximate the gradient vectors in (2.23) and (2.24) as:

$$\widehat{\nabla_w J_l}(w) = -4d_l(i)(w - x_l) + 4\|w - x_l\|^2(w - x_l) \tag{2.43}$$

If we do not exchange the local gradients with neighbors, i.e., if we set $S = C = I$, then the base-stations only share the local estimates of the target position w^o with their neighbors (no exchange of $\{x_l\}_{l\in\mathcal{N}_k}$).

We first simulate the stationary case, where the target stays at w^o. In Fig. 2.3(a), we show the MSD curves for non-cooperative, ATC, CTA, and incremental algorithms. The noise variance is set to $\sigma^2_{v,k} = 1$. We set the step-sizes to $\mu = 0.0025$ for ATC and CTA, and $\mu = 0.0025/N$ for the incremental algorithm. For ATC and CTA strategies, we use simple averaging for the combination step $\{a_{l,k}\}$, and for ATC and CTA with gradient exchange, we use Metropolis weights for $\{c_{l,k}\}$ to combine the gradients. The performance of CTA and ATC algorithms are close to each other, and both of them are close to the incremental scheme. In Fig. 2.3(b), we show the steady state MSD with respect to different values of μ. As the step-size becomes small, the performances of diffusion and incremental algorithms are close, and the MSD decreases as μ decreases. Furthermore, we see that exchanging only local estimates ($S = I$) is enough for localization, compared

to the case of exchanging both local estimates and gradients ($S = C$).

Next, we apply the algorithms to a non-stationary case, where the target moves along a trajectory, as shown in Fig. 2.4(a). The step-size is set to $\mu = 0.01$ for diffusion algorithms, and to $\mu = 0.01/N$ for the incremental approach. To see the advantage of using a constant step-size for continuous tracking, we also simulate the vanishing step-size version of the algorithm from [109] ($\mu_{k,i} = 0.01/i$). The diffusion algorithms track well the target but not the non-cooperative algorithm and the algorithm from [109], because a decaying step-size is not helpful for tracking. The tracking performance is shown in Fig. 2.4(b).

2.4 Conclusion

This chapter proposed diffusion adaptation strategies to optimize global cost functions over a network of nodes, where the cost is the sum of several components, i.e., in the "sum-of-costs" form. Diffusion adaptation allows the nodes to solve the distributed optimization problem via local interaction and online learning. We employ gradient approximations and constant step-sizes to endow the networks with continuous learning and tracking abilities. We applied the scheme to two examples: distributed sparse parameter estimation and distributed localization. Compared to incremental methods, diffusion strategies do not require a cyclic path over the nodes, which makes them more robust to node and link failure.

CHAPTER 3

Cost-of-Sum Formulations

In this chapter, we consider an alternative form of global cost function to be optimized over a network of agents, namely, regularized "cost-of-sum" forms such as

$$J^{\text{glob}}(y) = J_0\left(\sum_{k=1}^{N} W_k y_k\right) + \sum_{k=1}^{N} h_{y_k}(y_k) \tag{3.1}$$

where each agent k is in charge of finding a sub-vector y_k of $y \triangleq \text{col}\{y_1, \ldots, y_N\}$. Furthermore, the cost functions $h_{y_k}(y_k)$ and the matrices W_k are only known to agent k, and the form of the cost $J_0(\cdot)$ is known to all agents. We will motivate and solve this problem in the context of a typical application, namely, dictionary learning over large-scale distributed models. We will show that cost (3.1) can be transformed into a "sum-of-costs" problem of the form studied in Chapter 2 using the technique of dual decomposition and the concept of conjugate functions. Accordingly, the problem can be solved in the dual domain by using the methods developed in Chapter 2. Besides its close connection to the "sum-of-costs" problem, the cost (3.1) has another special structure: its dual solution will be shown to provide a global gradient information for $J_0(\cdot)$ evaluated at $\sum_{k=1}^{N} W_k y_k$. This property will prove to be especially useful for learning large-scale distributed models. The following presentation in this chapter is based on [39].

3.1 Motivation

Dictionary learning is a useful procedure by which dependencies among input features can be represented in terms of suitable bases [2, 41, 53, 78, 83, 93, 94, 123, 131, 149]. It has found applications in many machine learning and inference tasks including image denoising [53, 93], dimensionality-reduction [123, 149], bi-clustering [83], feature-extraction and classification [94], and novel document detection [78]. Dictionary learning usually alternates between two steps: (i) an inference (sparse coding) step and (ii) a dictionary update step. The first step finds a sparse representation for the input data using the existing dictionary by solving, for example, an ℓ_1-regularized regression problem, and the second step usually employs gradient descent approximation to update the dictionary entries.

With the increasing complexity of various learning tasks, it is natural that the size of the learning dictionaries is becoming demanding in terms of memory and computing requirements. It is therefore important to study scenarios where the dictionary need not be available in a single central location but is possibly spread out over multiple locations. This is particularly true in big data scenarios where multiple large dictionary models may already be available at separate locations and it is not feasible to aggregate all dictionaries in one location due to communication and privacy considerations. This observation motivates us to examine how to learn a dictionary model that is stored over a network of agents, where each agent is in charge of only a portion of the dictionary elements. Compared with other works, the problem we solve in this chapter and also in [39, 40] is how to learn a distributed dictionary model, which is, for example, different from the useful work in [31, 32] where it is assumed instead that each agent maintains the *entire* dictionary model.

3.2 Problem Formulation

3.2.1 General Dictionary Learning Problem

We seek to solve the following *global* general dictionary learning problem over a network of N agents connected by a topology:

$$\min_{W} \quad \mathbb{E}\Big[f(\boldsymbol{x}_t - W\boldsymbol{y}_t^o) + h_y(\boldsymbol{y}_t^o)\Big] + h_W(W) \tag{3.2}$$

$$\text{s.t.} \quad W \in \mathcal{W} \tag{3.3}$$

where $\mathbb{E}[\cdot]$ denotes the expectation operator, $\boldsymbol{x}_t$ is the $M \times 1$ input data vector at time t (we use boldface letters to represent random quantities), $\boldsymbol{y}_t^o$ is a $K \times 1$ coding vector defined further ahead as the solution to (3.8), and W is an $M \times K$ dictionary matrix. Moreover, the q-th column of W, denoted by $[W]_{:,q}$, is called the q-th dictionary element (or *atom*), $f(u)$ in (3.2) denotes a differentiable convex loss function for the residual error, $h_y(y)$ and $h_W(W)$ are convex (but not necessarily differentiable) regularization terms on y and W, respectively, and $\mathcal{W}$ denotes the convex constraint set on W. Depending on the application problem of interest, there are different choices for $f(u)$, $h_y(y)$ and $h_W(W)$. Table 3.1 lists some typical tasks and the corresponding choices for these functions. In regular dictionary learning [93], the constraint set $\mathcal{W}$ can be

$$\mathcal{W} = \{W : \ \|[W]_{:,q}\|_2 \leq 1\} \tag{3.4}$$

and in applications of nonnegative matrix factorization [93] and novel document detection (topic modeling) [78], it can be

$$\mathcal{W} = \{W : \ \|[W]_{:,q}\|_2 \leq 1, \ W \succeq 0\} \tag{3.5}$$

where the notation $[W]_{:,q}$ denotes the q-th column of the matrix W, the notation $W \succeq 0$ means each entry of the matrix W is nonnegative. We note that if there is a constraint on y, it can be absorbed into the regularization factor $h_y(y)$, by including an indicator function of the constraint into this regularization term. For example, if y is required to satisfy $y \in \mathcal{Y} = \{y : 0 \preceq y \preceq \mathbb{1}\}$, where $\mathbb{1}$ denotes an all-one vector, we can modify the original regularization $h_y(y)$ by adding an additional indicator function:

$$h_y(y) \leftarrow h_y(y) + I_{\mathcal{Y}}(y) \tag{3.6}$$

where the indicator function $I_{\mathcal{Y}}(y)$ for $\mathcal{Y}$ is defined as

$$I_{\mathcal{Y}}(y) \triangleq \begin{cases} 0 & \text{if } 0 \preceq y \preceq \mathbb{1} \\ +\infty & \text{otherwise} \end{cases} \tag{3.7}$$

The vector $\boldsymbol{y}_t^o$ in (3.2) is the solution to the following general inference problem for each input data sample x_t at time t for a given W (the regular font for x_t and y_t^o denotes realizations for the random quantities $\boldsymbol{x}_t$ and $\boldsymbol{y}_t^o$):

$$y_t^o \triangleq \arg\min_y \left[f(x_t - Wy) + h_y(y) \right] \tag{3.8}$$

Note that dictionary learning consists of two steps: the inference step (e.g., sparse coding) for the realization x_t at each time t in (3.8), and the dictionary update step (learning) in (3.2)–(3.3).

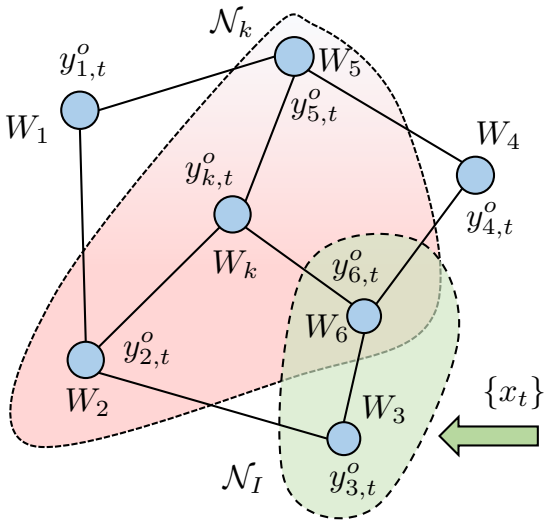


Figure 3.1: The data sample x_t at each time t is available to a subset $\mathcal{N}_I$ of agents in the network (e.g., agents 3 and 6 in the figure), and each agent k is in charge of one sub-dictionary, W_k, and the corresponding optimal sub-vector of coefficients estimated at time t, $y^o_{k,t}$. Each agent k can only exchange information with its immediate neighbors (e.g., agents 5, 2 and 6 in the figure and k itself). We use $\mathcal{N}_k$ to denote the set of neighbors of agent k.

Table 3.1: Examples of tasks solved by the general formulation (3.2)–(3.3). The loss functions $f(u)$ are illustrated in Fig. 3.2.

Tasks	$f(u)$	$h_y(y)$	$h_W(W)$	$h_{W_k}(W_k)$	$\mathcal{W}_k$
Sparse SVD	$\frac{1}{2}\|u\|_2^2$	$\gamma\|y\|_1 + \frac{\delta}{2}\|y\|_2^2$	0	0	$\{W_k : \|[W_k]_{:,q}\|_2 \leq 1\}$
Bi-Clustering	$\frac{1}{2}\|u\|_2^2$	$\gamma\|y\|_1 + \frac{\delta}{2}\|y\|_2^2$	$\beta \cdot \|\|\|W\|\|\|_1$ [a]	$\beta \cdot \|\|\|W_k\|\|\|_1$	$\{W_k : \|[W_k]_{:,q}\|_2 \leq 1\}$
Nonnegative Matrix	$\frac{1}{2}\|u\|_2^2$	$\gamma\|y\|_{1,+} + \frac{\delta}{2}\|y\|_2^2$ [b]	0	0	$\{W_k : \|[W_k]_{:,q}\|_2 \leq 1,\ W_k \succeq 0\}$
Factorization	$\sum_{m=1}^{M} L(u_m)$ [c]	$\gamma\|y\|_{1,+} + \frac{\delta}{2}\|y\|_2^2$	0	0	$\{W_k : \|[W_k]_{:,q}\|_2 \leq 1,\ W_k \succeq 0\}$

[a] The notation $|||W|||_1$ is used to denote the absolute sum of all the entries in the matrix W: $|||W|||_1 = \sum_{m=1}^{M}\sum_{q=1}^{K}|W_{mq}|$, which is different from the conventional matrix 1−norm defined as the maximum absolute column sum: $\|W\|_1 = \max_{1\leq q\leq K}\sum_{m=1}^{M}|W_{mq}|$.

[b] The notation $\|y\|_{1,+}$ is defined as $\|y\|_{1,+} = \|y\|_1$ if $y \succeq 0$ and $\|y\|_{1,+} = +\infty$ otherwise. It imposes infinite penalty on any negative entry appearing in the vector y. Since negative entries are already penalized in $\|y\|_{1,+}$, there is no need to penalize it again in the $\frac{\delta}{2}\|y\|_2^2$ term.

[c] The scalar Huber loss function is defined as $L(u_m) \triangleq \begin{cases} \frac{1}{2\eta}u_m^2, & |u_m| < \eta \\ |u_m| - \frac{\eta}{2}, & \text{otherwise} \end{cases}$, where η is a positive parameter.

3.2.2 Dictionary Learning over Networked Agents

Let the matrix W and the vector y be partitioned in the following block forms:

$$W = \begin{bmatrix} W_1 & \cdots & W_N \end{bmatrix} \tag{3.9}$$

$$y = \mathrm{col}\{y_1, \ \ldots, \ y_N\} \tag{3.10}$$

where W_k is an $M \times N_k$ *sub-dictionary* matrix and y_k is an $N_k \times 1$ sub-vector. Furthermore, we assume the regularization terms $h_y(y)$ and $h_W(W)$ admit the following decompositions:

$$h_y(y) = \sum_{k=1}^{N} h_{y_k}(y_k) \tag{3.11}$$

$$h_W(W) = \sum_{k=1}^{N} h_{W_k}(W_k) \tag{3.12}$$

Then, the objective function of the inference step (3.8) can be written as

$$Q(W, y; x_t) \triangleq f\Big(x_t - \sum_{k=1}^{N} W_k y_k\Big) + \sum_{k=1}^{N} h_{y_k}(y_k) \tag{3.13}$$

We observe from (3.13) that the sub-dictionary matrices $\{W_k\}$ are linearly combined to represent the input data x_t. By minimizing $Q(W, y; x_t)$ over y, the first term in (3.13) helps ensure that the representation error for x_t is small. The second term in (3.13), which usually involves a combination of ℓ_1 and ℓ_2 measures, as indicated in Table 3.1, helps ensure that each of the resulting combination coefficients $\{y_k\}$ is sparse and small. We require the regularization terms $h_{y_k}(y_k)$ to be strongly convex, which will allow us to develop a fully distributed strategy that enables the sub-dictionaries $\{W_k\}$ and the corresponding coefficients $\{y_k\}$

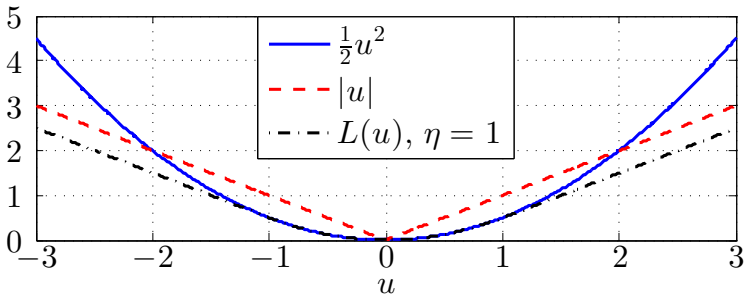


Figure 3.2: Illustration of the functions $\frac{1}{2}u^2$, $|u|$, and $L(u)$.

to be stored and learned in a distributed manner over the network; each agent k will infer its own y_k and update its own sub-dictionary W_k with limited interaction with its neighboring agents. Requiring $\{h_{y_k}(y_k)\}$ to be strongly convex is not restrictive since we can always add a small ℓ_2 regularization term to make it strongly convex. For example, in Table 3.1, we add an ℓ_2 term to ℓ_1 regularization so that the resulting $h_{y_k}(y_k)$ ends up amounting to elastic net regularization [149].

Figure 3.1 shows the configuration of the knowledge and data distribution over the network. The sub-dictionaries $\{W_k\}$ can be interpreted as the "wisdom" that is distributed over the network, and which we wish to combine in a distributed manner to form a greater "intelligence" for interpreting the data $\boldsymbol{x}_t$. Observe that we are allowing $\boldsymbol{x}_t$ to be observed by only a subset of the agents. By having the dictionary distributed over the agents, we would then like to develop a procedure that enables these networked agents to find the *global* solutions to both the inference problem (3.8) and the learning problem (3.2)–(3.3) with interactions that are limited to their neighborhoods.

3.2.3 Relation to Prior Work

The problem we are solving in this chapter is different from the useful work [31, 32] on distributed dictionary learning and from the traditional distributed learning setting [26, 36, 42, 120], where it is assumed that the *entire* dictionary W [31, 32] or the entire data model [26, 36, 42, 47, 120] is maintained at each agent in the network, whereas individual data samples generated by the same distribution, denoted by $\boldsymbol{x}_{k,t}$, are observed by the agents at each time t. That is, these previous works study *data distributed* formulations. What we are studying in this chapter is to find a distributed solution where each agent is only in charge of a *portion* of the dictionary (W_k for each agent k) while the incoming data, $\boldsymbol{x}_t$, is common and is observed by only a subset of the agents. This scenario corresponds to a *model distributed* (or dictionary-distributed) formulation. A related albeit different model was considered in [43] in the context of distributed deep neural network (DNN) models over computer networks. In these models, each computer is in charge of a portion of neurons in the DNN, which exchange their private activation signals with neurons over the network to perform the classification task. As we will see further ahead, our distributed model does not require exchanging either the private combination coefficients $\{y_k\}$ or the sub-dictionaries $\{W_k\}$ while still being able to model the data using the collective "wisdom" over the network. Another related but different work is [142], where the authors study a special form of a distributed sparse basis pursuit problem under *fixed* sub-dictionaries at each agent. In this chapter, we allow the sub-dictionaries to be updated dynamically over time (rather than staying fixed) and this is accomplished without exchanging any further information after the distributed inference step — see Sec. 3.3.5 further ahead.

The distributed model setting is important in practice because agents tend to

be limited in their memory and computing power and they may not be able to store large dictionaries locally. Even if the agents were powerful enough, different agents may have access to different databases and different sources of information. Rather than aggregate the information in the form of large dictionaries at every single location, it is more advantageous to keep the information distributed due to potential excessive costs in exchanging large data sets, and also due to privacy considerations where different agents may not be in favor of sharing their data and dictionary. Therefore, by having distributed sub-dictionaries, and by having many agents cooperate with each other, a large model that is beyond the ability or reach of any single agent can be analyzed by the network in a distributed manner.

3.3 Learning over Distributed Models

3.3.1 "Cost-of-Sum" vs. "Sum-of-Costs"

Observe that the cost function (3.13) is a regularized "*cost-of-sum*"; it consists of two terms: the first term has a sum of quantities associated to different agents inside a cost function $f(\cdot)$ and the second term is a collection of separable regularization terms $\{h_{y_k}(y_k)\}$. This is different from the classical "*sum-of-costs*" problem, where the global cost function $J^{\text{glob}}(w)$ is an aggregation of individual costs $\{J_k(w)\}$:

$$J^{\text{glob}}(w) = \sum_{k=1}^{N} J_k(w) \tag{3.14}$$

The "sum-of-costs" problem (3.14) is amenable to distributed implementations — see Chapter 2 and [12, 26, 35, 36, 77, 84, 120, 137]. However, minimizing the

regularized "cost-of-sum" problem in (3.13) directly for inference (sparse coding) at any agent would require knowledge of all sub-dictionaries $\{W_k\}$ and coefficients $\{y_k\}$ from the other agents due to the sum that runs from $k = 1$ up to N. Therefore, this formulation is not directly amenable to the distributed techniques from [12,26,35,36,77,84,120,137]. In [33], the authors propose a useful consensus-based primal-dual perturbation method to solve a similar constrained "cost-of-sum" problem for smart grid control, where an averaging consensus step is used to compute the sum inside the cost. However, different from [33], we arrive at a more efficient distributed strategy by transforming the original optimization problem into a dual problem that has the same form as (3.14) — see (3.31a)–(3.31b) further ahead. More importantly, we will reveal in Sec. 3.3.5 that the dual solution provides critical information for fully distributed dictionary updates. In particular, for each new input data sample x_t, after the dual inference problem is solved, there will be no need to exchange any further information among agents or use a consensus step to evaluate the sum inside the cost in order to update their own sub-dictionaries.

3.3.2 Inference over Distributed Models

To begin with, we first transform the minimization of (3.13) into the following equivalent constrained optimization problem:

$$\min_{\{y_k\},z} \quad f(x_t - z) + \sum_{k=1}^{N} h_{y_k}(y_k) \tag{3.15a}$$

$$\text{s.t.} \quad z = \sum_{k=1}^{N} W_k y_k \tag{3.15b}$$

Note that the above problem is convex over both $\{y_k\}$ and z since the objective is convex and the equality constraint is linear. Problem (3.15a)–(3.15b) is a convex optimization problem with linear constraints so that strong duality holds [10, p.514], meaning that the optimal solution to (3.15a)–(3.15b) can be found by solving its corresponding dual problem and then recovering the optimal $\{y_k\}$ and z. To arrive at the dual problem, we write the Lagrangian of (3.15a)–(3.15b) for each input realization x_t as

$$\begin{aligned} &L(\{y_k\}, z, \nu; x_t) \\ &\quad = f(x_t - z) + \sum_{k=1}^{N} h_{y_k}(y_k) + \nu^T \Big(z - \sum_{k=1}^{N} W_k y_k \Big) \\ &\quad = f(x_t - z) + \nu^T z + \sum_{k=1}^{N} \Big[h_{y_k}(y_k) - \nu^T W_k y_k \Big] \end{aligned} \tag{3.16}$$

where $\{y_k\}$ and z are the primal variables and ν is the Lagrange multiplier (also known as dual variable) of size $M \times 1$. The dual function $g(\nu; x_t)$ is defined as the infinum of the Lagrangian $L(\{y_k\}, z, \nu; x_t)$ over the primal variables $\{y_k\}$ and z for each given ν:

$$\begin{aligned} g(\nu; x_t) &\triangleq \inf_{\{y_k\}, z} L(\{y_k\}, z, \nu; x_t) \\ &= \inf_{z} \big[f(x_t - z) + \nu^T z \big] + \sum_{k=1}^{N} \inf_{y_k} \Big[h_{y_k}(y_k) - \nu^T W_k y_k \Big] \end{aligned} \tag{3.17}$$

With strong duality [20, p. 226] (a brief overview of duality can be found in Appendix 3.B), it is known that the minimum value of the cost function obtained from the original optimization problem (3.15a)–(3.15b) is equal to the maximum

value of $g(\nu; x_t)$ obtained from the following dual problem:

$$\max_{\nu} g(\nu; x_t) \tag{3.18}$$

Furthermore, if $f(u)$ and $\{h_{y_k}(y_k)\}$ are strongly convex, the infimum in (3.17) can be attained and the infimums become minimizations [105, p.15]. As a result, the optimal solution of (3.15a)–(3.15b) can be found by solving the above dual problem (3.18) to obtain:

$$\nu_t^o = \arg\max_{\nu} g(\nu; x_t) \tag{3.19}$$

and then uniquely recovering the optimal primal variables z^o and y_k^o via

$$z_t^o = \arg\min_{z} \left\{ f(x_t - z) + (\nu_t^o)^T z \right\} \tag{3.20}$$

$$y_{k,t}^o = \arg\min_{y_k} \left\{ h_{y_k}(y_k) - (\nu_t^o)^T W_k y_k \right\} \tag{3.21}$$

The strong convexity of $f(u)$ and $\{h_{y_k}(y_k)\}$ is needed if we want to uniquely recover z_t^o and $\{y_{k,t}^o\}$ from the dual problem (3.19). As we will show further ahead in (3.52), the quantities $\{y_{k,t}^o\}$ are always needed in the dictionary update. Therefore, we shall assume that the $\{h_{y_k}(y_k)\}$ are strongly convex throughout our presentation, which can always be satisfied by means of elastic net regularization as explained earlier. On the other hand, depending on the application, the recovery of z_t^o is not always needed and neither is the strong convexity of $f(u)$ (in these cases, it is sufficient to assume that $f(u)$) is convex). For example, as we will show in Sec. 3.4, the image denoising application requires recovery of z_t^o as the final reconstructed image. On the other hand, the novel document detection application in the same section does not require recovery of z_t^o. In-

stead, in this application, it suffices to recover the maximum value of the dual function, $g(\nu; x_t)$, which, by strong duality, is equal to the minimum value of the cost function (3.15a).

To continue, observe that the infimum in (3.17) over the variables $\{y_k\}$ and z for a given ν is decoupled, and the minimization[1] over each y_k is also decoupled for different k. Therefore, the infimum (minimization) over the primal variables can be done independently. However, we still need to determine the optimal dual variable ν_t^o by solving (3.19). This requires us to derive the closed-form expression for $g(\nu; x_t)$ by solving the infimum (minimizations) in (3.17). To do so, we shall explain first how the optimization over $\{z, y_k\}$ in (3.17) is related to the concept of conjugate functions in convex optimization [20, pp.90-95].

Thus, recall that for a function $r(x)$, its conjugate function, $r^\star(\nu)$, is defined as

$$r^\star(\nu) \triangleq \sup_x \left[\nu^T x - r(x)\right], \quad \nu \in \mathcal{V}_r \tag{3.22}$$

where the domain $\mathcal{V}_r$ is defined as the set of ν where the above supremun is finite. The conjugate function $r^\star(\nu)$ and its domain $\mathcal{V}_r$ are always convex regardless of whether $r(x)$ is convex or not [10, p.530] [20, p.91]. In particular, it holds that $\mathcal{V}_r = \mathbb{R}^M$ if $r(x)$ is strongly convex [66, p.240]. To see this, let x_1 denote a point where $r(x)$ is differentiable. Then, by strong convexity, we have

$$r(x) \geq r(x_1) + [\nabla_x r(x_1)]^T (x - x_1) + \frac{\lambda_r}{2}\|x - x_1\|_2^2 \tag{3.23}$$

[1]The infimum over y_k in (3.17) becomes minimization since we assume $h_{y_k}(y_k)$ is strongly convex so that the infimum can be attained.

where λ_r is some positive constant. Substituting (3.23) into (3.22), we obtain

$$\begin{aligned} r^\star(\nu) &\triangleq \sup_x \big[\nu^T x - r(x)\big] \\ &\le \sup_x \Big[\nu^T x - r(x_1) - [\nabla_x r(x_1)]^T (x - x_1) - \frac{\lambda_r}{2}\|x - x_1\|_2^2\Big] \\ &\overset{(a)}{=} \sup_x \Big\{ -\frac{\lambda_r}{2}\Big\|x - x_1 + \frac{1}{\lambda_r}\left(\nabla_x r(x_1) - \nu\right)\Big\|_2^2 \\ &\qquad + \frac{1}{2\lambda_r}\big\|\nabla_x r(x_1) - \nu\big\|_2^2 + \nu^T x_1 - r(x_1)\Big\} \\ &= \frac{1}{2\lambda_r}\big\|\nabla_x r(x_1) - \nu\big\|_2^2 + \nu^T x_1 - r(x_1) \end{aligned} \tag{3.24}$$

where in step (a) we completed the square. Therefore, $r^\star(\nu)$ is always upper bounded by a finite value for any given $\nu \in \mathbb{R}^M$, i.e., the domain $\mathcal{V}$ for $r^\star(\nu)$ is $\mathbb{R}^M$.

Applying the concept of conjugate functions to the first term in (3.17) we have:

$$\begin{aligned} \inf_z \big[f(x_t - z) + \nu^T z\big] &\overset{(a)}{=} \inf_u \big[f(u) - \nu^T u + \nu^T x_t\big] \\ &= -\sup_u \big[\nu^T u - f(u)\big] + \nu^T x_t \\ &= -f^\star(\nu) + \nu^T x_t, \quad \nu \in \mathcal{V}_f \end{aligned} \tag{3.25}$$

Likewise, applying the concept of conjugate functions again to the second term in (3.17) we get

$$\begin{aligned} \inf_{y_k} \big[h_{y_k}(y_k) - \nu^T W_k y_k\big] &= -\sup_{y_k} \big[(W_k^T \nu)^T y_k - h_{y_k}(y_k)\big] \\ &= -h_{y_k}^\star(W_k^T \nu), \quad \nu \in \mathcal{V}_{h_{y_k}} \end{aligned} \tag{3.26}$$

where in step (a) of (3.25) we introduced $u \triangleq x_t - z$, and $f^\star(\cdot)$ and $h_{y_k}^\star(\cdot)$ are the

conjugate functions of $f(\cdot)$ and $h_{y_k}(\cdot)$, respectively, with corresponding domains being $\mathcal{V}_f$ and $\mathcal{V}_{h_{y_k}}$, respectively. Now since $h_{y_k}(\cdot)$ is strongly convex, its domain $\mathcal{V}_{h_{y_k}}$ is the entire $\mathbb{R}^M$ [66, p.240]. If $f(u)$ happens to be strongly convex (rather than only convex, e.g., if $f(u) = \frac{1}{2}\|u\|_2^2$), then $\mathcal{V}_f$ would also be $\mathbb{R}^M$, otherwise it is a convex subset of $\mathbb{R}^M$. Therefore, the dual function defined by (3.17) becomes

$$g(\nu; x_t) = -f^\star(\nu) + \nu^T x_t - \sum_{k=1}^{N} h_{y_k}^\star(W_k^T \nu) \tag{3.27}$$

and the domain is $\nu \in \mathcal{V}_f$. The dual problem (3.19) can then be expressed as

$$\max_{\nu} \quad g(\nu; x_t) \tag{3.28a}$$

$$\text{s.t.} \quad \nu \in \mathcal{V}_f \tag{3.28b}$$

which is equivalent to

$$\min_{\nu} \quad f^\star(\nu) - \nu^T x_t + \sum_{k=1}^{N} h_{y_k}^\star(W_k^T \nu) \tag{3.29a}$$

$$\text{s.t.} \quad \nu \in \mathcal{V}_f \tag{3.29b}$$

Note that the objective function in the above optimization problem is an aggregation of many individual costs associated with sub-dictionaries at different agents (last term in (3.29a)), a component associated with data sample x_t (second term in (3.29a)), and a component that is the conjugate function of the residual cost (first term in (3.29a)). In contrast to (3.13), the cost function in (3.29a) is now in a form that is amenable to distributed processing. Specifically, diffusion strategies of the form described in Chapter 2 and [34, 120] can now be applied to obtain the optimal dual variable ν_t^o in a distributed manner at the various agents. Depending on how we assign $f^\star(\cdot)$ and $\nu^T x_t$, there can be many different config-

urations. For example, we can assign the term associated with data to a subset of agents. Then, only these agents will be required to know the data sample, and all other agents will learn and benefit from the cooperative process and attain the same variable ν_t^o as if they had seen the data x_t.

To arrive at the distributed solution, we proceed as follows. We denote the set of agents that observe the data sample x_t by $\mathcal{N}_I$. Motivated by (3.29a), with each agent k, we associate the local cost function:

$$J_k(\nu; x_t) \triangleq \begin{cases} -\dfrac{1}{|\mathcal{N}_I|}\nu^T x_t + \dfrac{1}{N} f^\star(\nu) + h_{y_k}^\star(W_k^T \nu), & k \in \mathcal{N}_I \\ \dfrac{1}{N} f^\star(\nu) + h_{y_k}^\star(W_k^T \nu), & k \notin \mathcal{N}_I \end{cases} \tag{3.30}$$

where $|\mathcal{N}_I|$ denotes the cardinality of $\mathcal{N}_I$. Then, the optimization problem (3.29a)–(3.29b) can be rewritten as

$$\min_{\nu} \quad \sum_{k=1}^{N} J_k(\nu; x_t) \tag{3.31a}$$

$$\text{s.t.} \quad \nu \in \mathcal{V}_f \tag{3.31b}$$

Note that the new equivalent form (3.31a) is an aggregation of individual costs associated with different agents; each cost $J_k(\nu; x_t)$ only requires knowledge of W_k. Consider first the case in which $f(u)$ is strongly convex. Then, it holds that $\mathcal{V}_f = \mathbb{R}^M$ and problem (3.31a)–(3.31b) becomes an unconstrained optimization problem and of the same general nature as problems studied in [35,36]. Therefore, we can directly apply the diffusion strategies developed in these works to solve (3.31a)–(3.31b) in a fully distributed manner. The algorithm takes the following

form:

$$\psi_{k,i} = \nu_{k,i-1} - \mu \cdot \nabla_{\nu} J_k(\nu_{k,i-1}; x_t) \tag{3.32a}$$

$$\nu_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \tag{3.32b}$$

where $\nu_{k,i}$ denotes the estimate of the optimal ν_t^o at agent k at iteration i (we will use i to denote the i-th iteration of the inference, and use t to denote the t-th data sample), $\psi_{k,i}$ is an intermediate variable, $\mathcal{N}_k$ denotes the neighborhood of agent k, μ is the step-size parameter chosen to be a small positive number, and $a_{\ell k}$ is the combination coefficient that agent k assigns to the information shared from agent ℓ and it satisfies

$$\sum_{\ell \in \mathcal{N}_k} a_{\ell k} = 1, \quad a_{\ell k} > 0 \text{ if } \ell \in \mathcal{N}_k, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k \tag{3.33}$$

Let A denote the $N \times N$ matrix that collects $a_{\ell k}$ as its (ℓ, k)-th entry. Then, it will be shown in Chapter 4 that there exists a small $\mu_0 > 0$ such that as long as the matrix A is doubly-stochastic and the step-size is sufficiently small satisfying $\mu < \mu_0$, then the algorithm (3.32a)–(3.32b) converges to a fixed point that is $O(\mu^2)$ away from the optimal solution of (3.31a) in squared Euclidean distance. We remark that a doubly-stochastic matrix is one that satisfies $A\mathbb{1} = A^T\mathbb{1} = \mathbb{1}$.

Consider now the case in which the constraint set $\mathcal{V}_f$ is *not* equal to $\mathbb{R}^M$ but is still known to all agents. In general, we need to solve the maximization in the second line of (3.25) to derive the expression for $f^\star(\nu)$ and determine the set $\mathcal{V}_f$ that makes the maximization in (3.25) finite. Fortunately, this can be done in closed-form for many typical choices of $f(u)$ that are of practical interest — see [20, pp.90-95]. Here we list in Table 3.2 the results that will be used in Sec. 3.4; part of results are derived in Appendix 3.A and the rest is from [20, pp.90-

Table 3.2: Conjugate functions used in this chapter for different tasks

Tasks	$f(u)$	$f^\star(\nu)$	$\mathcal{V}_f$	z_t^o	$h_{y_k}(y_k)$	$h_{y_k}^\star(W_k^T\nu)$	$\mathcal{V}_{h_{y_k}}$	$y_{k,t}^o$
Sparse SVD	$\frac{1}{2}\|u\|_2^2$	$\frac{1}{2}\|\nu\|_2^2$	$\mathbb{R}^M$	$x_t - \nu_t^o$	$\gamma\|y_k\|_1 + \frac{\delta}{2}\|y_k\|_2^2$	$\mathcal{S}_{\frac{\gamma}{\delta}}\left(\frac{W_k^T\nu}{\delta}\right)$ [b]	$\mathbb{R}^M$	$\mathcal{T}_{\frac{\gamma}{\delta}}\left(\frac{W_k^T\nu_t^o}{\delta}\right)$ [a]
Bi-Clustering	$\frac{1}{2}\|u\|_2^2$	$\frac{1}{2}\|\nu\|_2^2$	$\mathbb{R}^M$	$x_t - \nu_t^o$	$\gamma\|y_k\|_1 + \frac{\delta}{2}\|y_k\|_2^2$	$\mathcal{S}_{\frac{\gamma}{\delta}}\left(\frac{W_k^T\nu}{\delta}\right)$	$\mathbb{R}^M$	$\mathcal{T}_{\frac{\gamma}{\delta}}\left(\frac{W_k^T\nu_t^o}{\delta}\right)$
Nonnegative Matrix	$\frac{1}{2}\|u\|_2^2$	$\frac{1}{2}\|\nu\|_2^2$	$\mathbb{R}^M$	$x_t - \nu_t^o$	$\gamma\|y_k\|_{1,+} + \frac{\delta}{2}\|y_k\|_2^2$	$\mathcal{S}_{\frac{\gamma}{\delta}}^+\left(\frac{W_k^T\nu}{\delta}\right)$ [d]	$\mathbb{R}^M$	$\mathcal{T}_{\frac{\gamma}{\delta}}^+\left(\frac{W_k^T\nu_t^o}{\delta}\right)$ [c]
Factorization	$\sum_{m=1}^M L(u_m)$	$\frac{\eta}{2}\|\nu\|_2^2$	$\{\nu : \|\nu\|_\infty \le 1\}$	—	$\gamma\|y_k\|_{1,+} + \frac{\delta}{2}\|y_k\|_2^2$	$\mathcal{S}_{\frac{\gamma}{\delta}}^+\left(\frac{W_k^T\nu}{\delta}\right)$	$\mathbb{R}^M$	$\mathcal{T}_{\frac{\gamma}{\delta}}^+\left(\frac{W_k^T\nu_t^o}{\delta}\right)$

[a] $\mathcal{T}_\lambda(x)$ denotes the entry-wise soft-thresholding operator on the vector x: $[\mathcal{T}_\lambda(x)]_n \triangleq (|[x]_n| - \lambda)_+ \mathrm{sgn}([x]_n)$, where $(x)_+ = \max(x, 0)$.

[b] $\mathcal{S}_{\frac{\gamma}{\delta}}(x)$ is the function defined by $\mathcal{S}_{\frac{\gamma}{\delta}}(x) \triangleq -\frac{\delta}{2}\cdot\left\|\mathcal{T}_{\frac{\gamma}{\delta}}(x)\right\|_2^2 - \gamma\cdot\left\|\mathcal{T}_{\frac{\gamma}{\delta}}(x)\right\|_1 + \delta\cdot x^T\mathcal{T}_{\frac{\gamma}{\delta}}(x)$ for $x \in \mathbb{R}^M$.

[c] $\mathcal{T}_\lambda^+(x)$ denotes the entry-wise one-side soft-thresholding operator on the vector x: $[\mathcal{T}_\lambda^+(x)]_n \triangleq ([x]_n - \lambda)_+$.

[d] $\mathcal{S}_{\frac{\gamma}{\delta}}^+(x)$ is defined by $\mathcal{S}_{\frac{\gamma}{\delta}}^+(x) \triangleq -\frac{\delta}{2}\cdot\left\|\mathcal{T}_{\frac{\gamma}{\delta}}^+(x)\right\|_2^2 - \gamma\cdot\left\|\mathcal{T}_{\frac{\gamma}{\delta}}^+(x)\right\|_1 + \delta\cdot x^T\mathcal{T}_{\frac{\gamma}{\delta}}^+(x)$ for $x \in \mathbb{R}^M$.

[e] The functions $\mathcal{T}_\lambda(x)$, $\mathcal{T}_\lambda^+(x)$, $\mathcal{S}_{\frac{\gamma}{\delta}}(x)$, and $\mathcal{S}_{\frac{\gamma}{\delta}}^+(x)$ for the scalar x case are illustrated in Fig. 3.3.

95]. Usually, $\mathcal{V}_f$ for these typical choices of $f(u)$ are simple sets whose projection operators[2] can be found in closed-form — see [103]. For example, the projection operator onto the set

$$\mathcal{V}_f = \{\nu : \|\nu\|_\infty \leq 1\} = \{\nu : -\mathbb{1} \preceq \nu \preceq \mathbb{1}\} \tag{3.34}$$

that is listed in the third row of Table 3.2 is given by

$$[\Pi_{\mathcal{V}_f}(\nu)]_m = \begin{cases} 1 & \text{if } \nu_m > 1 \\ \nu_m & \text{if } -1 \leq \nu_m \leq 1 \\ -1 & \text{if } \nu_m < -1 \end{cases} \tag{3.35}$$

where $[x]_m$ denotes the m-th entry of the vector x and ν_m denotes the m-th entry of the vector ν. Now, the constraint set $\mathcal{V}_f$ can be enforced either by incorporating local projections onto $\mathcal{V}_f$ into the combination step (3.32b) at each agent [130] or by using the penalized diffusion method [134,135]. Specifically, the projection-based strategy is given by

$$\psi_{k,i} = \nu_{k,i-1} - \mu \cdot \nabla_\nu J_k(\nu_{k,i-1}; x_t) \tag{3.36a}$$

$$\nu_{k,i} = \Pi_{\mathcal{V}_f}\left[\sum_{\ell \in \mathcal{N}_k} a_{\ell k}\psi_{\ell,i}\right] \tag{3.36b}$$

where $\Pi_{\mathcal{V}_f}[\cdot]$ is a projection operator onto $\mathcal{V}_f$. On the other hand, the penalty-based approach is given by

$$\zeta_{k,i} = \nu_{k,i-1} - \mu \cdot \nabla_\nu J_k(\nu_{k,i-1}; x_t) \tag{3.37a}$$

$$\psi_{k,i} = \zeta_{k,i} - \mu \cdot \nabla_\nu J^{\text{pen}}_{\mathcal{V}_f}(\zeta_{k,i}) \tag{3.37b}$$

[2]The projection operator onto the set $\mathcal{V}_f$ is defined as $\Pi_{\mathcal{V}_f}(\nu) \triangleq \arg\min_{x \in \mathcal{V}_f} \|x - \nu\|_2$.

$$\nu_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \tag{3.37c}$$

where $J^{\text{pen}}_{\mathcal{V}_f}(\nu)$ is a penalty function that is equal to zero when $\nu \in \mathcal{V}_f$ and assumes a large value when $\nu \notin \mathcal{V}_f$. Examples of choices for the penalty function can be found in [134, 135].

3.3.3 Recovery of the Primal Variables

After the optimal dual variable ν_t^o has been estimated by the various agents, the optimal primal variable $y_{k,t}^o$ can be recovered from (3.21) since $h_k(y_k)$ is strongly convex; $h_k(y_k)$ being strongly convex makes the term $h_{y_k}(y_k) - (\nu_t^o)^T W_k y_k$ in (3.21) also strongly convex so that the minimum in (3.21) exists and is unique. Based on the argument in (3.26), expression (3.21) is equivalent to:

$$y_{k,t}^o = \arg\max_{y_k} \left[(W_k^T \nu_t^o)^T y_k - h_{y_k}(y_k) \right] \tag{3.38}$$

Fortunately, for many typical choices of $h_{y_k}(\cdot)$, the optimal $y_{k,t}^o$ can be expressed in closed form in terms of ν_t^o. In Table 3.2, we list the results that will be used later in Sec. 3.4 with the derivation given in Appendix 3.A. Now, with regards to z_t^o, we indicated earlier that depending on the application, we may need to recover z_t^o or not. For cases when z_t^o should be recovered, we need to assume $f(u)$ is strongly convex (e.g., $\frac{1}{2}\|u\|_2^2$). In these cases, z_t^o can be recovered from (3.20), which, according to the argument in (3.25) and the fact that $u = x_t - z$, is equivalent to following expression:

$$z_t^o = x_t - \arg\max_{u} \left[(\nu_t^o)^T u - f(u) \right] \tag{3.39}$$

When $f(u)$ is strongly convex, the term $(\nu_t^o)^T u - f(u)$ to be maximized in (3.39) will become strongly concave so that there is a unique maximizer in (3.39). Problem (3.39) can be solved in closed-form for many typical choices of $f(u)$ and we list in Table 3.2 the results that will be used in Sec. 3.4. For more examples, readers are referred to [20, pp.90-94].

3.3.4 Choice of Residual and Regularization Functions

In Tables 3.1–3.2, we list several typical choices for the residual function, $f(u)$, and the regularization functions, $\{h_{y_k}(y_k)\}$. In general, a careful choice of $f(u)$ and $\{h_{y_k}(y_k)\}$ can make the dual cost (3.29a) better conditioned than in the primal cost (3.15a). Recall that the primal cost (3.15a) may not be differentiable due to the choice of $h_{y_k}(y_k)$ (e.g., the elastic net). However, if $f(u)$ is chosen to be strictly convex with Lipschitz gradients and the $\{h_{y_k}(y_k)\}$ are chosen to be strongly convex (not necessarily differentiable), then the conjugate function $f^\star(\cdot)$ will be a differentiable strongly convex function with Lipschitz gradient and the $\{h_{y_k}^\star(\cdot)\}$ will be differentiable convex functions with Lipschitz gradients [66, pp.238–240]. Adding these two parts from $f^\star(\cdot)$ and $\{h_{y_k}^\star(\cdot)\}$ together in (3.29a) essentially transforms a non-differentiable primal cost (3.15a) into a differentiable strongly convex dual cost (3.29a) with Lipschitz gradients. As a result, the algorithms that optimize the dual problem (3.29a)–(3.29b) can generally enjoy a fast (geometric) convergence rate [36, 105].

3.3.5 Distributed Dictionary Updates

Now that we have shown how the inference task (3.8) can be solved in a distributed manner, we move on to explain how the local sub-dictionaries W_k can be updated through the solution of the stochastic optimization problem (3.2)–

(3.3), which is rewritten as:

$$\min_{W} \quad \mathbb{E}Q(W, \boldsymbol{y}_t^o; \boldsymbol{x}_t) + \sum_{k=1}^{N} h_{W_k}(W_k) \tag{3.40a}$$

$$\text{s.t.} \quad W_k \in \mathcal{W}_k, \quad k = 1, \ldots, N \tag{3.40b}$$

where the loss function $Q(W, \boldsymbol{y}_t^o; \boldsymbol{x}_t)$ is given in (3.13), $\boldsymbol{y}_t^o \triangleq \text{col}\{\boldsymbol{y}_{1,t}^o, \ldots, \boldsymbol{y}_{N,t}^o\}$, the decomposition for $h_W(W)$ from (3.12) is used, and we assume the constraint set $\mathcal{W}$ can be decomposed into a set of constraints $\{\mathcal{W}_k\}$ on the individual sub-dictionaries W_k, which usually holds for typical dictionary learning applications — see Table 3.1. Note that the cost function in (3.40a) consists of two parts, where the first term is differentiable with respect to W[3] while the second term, if it exists, is non-differentiable but usually has a simple form — see Table 3.1. A typical approach to optimizing cost functions of this type is the *proximal gradient* method [8, 55, 56, 103], which applies gradient descent to the first differentiable part followed by a proximal operator to the second non-differentiable part. This method is known to converge faster than applying the subgradient descent method to both parts. Therefore, our strategy is to apply the proximal gradient method to the cost function in (3.40a) and remove the expectation operator to obtain an instantaneous approximation to the true gradient; this is the approach typically used in adaptation [116] and stochastic approximation [81, 99]. Afterwards, we project the iterate onto the constraint set $\mathcal{W}_k$ to enforce the constraint (3.40b) [10, 105]:

$$W_{k,t} = \Pi_{\mathcal{W}_k}\Big\{\text{prox}_{\mu_w \cdot h_{W_k}}\big(W_{k,t-1} - \mu_w \nabla_{W_k} Q(W_{t-1}, y_t^o; x_t)\big)\Big\} \tag{3.41}$$

[3]Note from (3.13) that $Q(\cdot)$ depends on W via $f(\cdot)$, which is assumed to be differentiable.

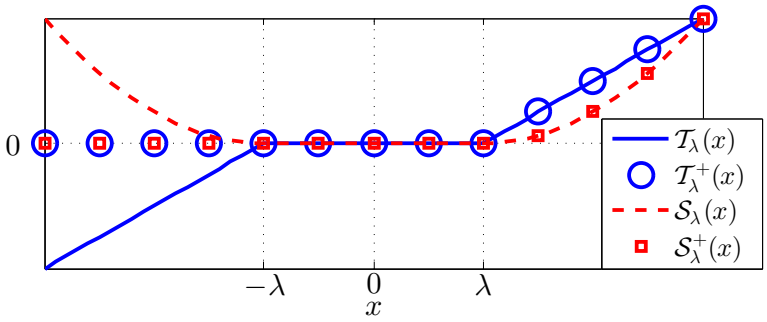


Figure 3.3: Illustration of the functions $\mathcal{T}_\lambda(x)$, $\mathcal{T}_\lambda^+(x)$, $\mathcal{S}_\lambda(x)$, and $\mathcal{S}_\lambda^+(x)$. Best viewed in color.

where $W_{t-1} \triangleq [W_{1,t-1}, \cdots, W_{N,t-1}]$, $\text{prox}_{\mu_w \cdot h_{W_k}}(\cdot)$ denotes the proximal operator of $\mu_w \cdot h_{W_k}(W_k)$, and $\Pi_{\mathcal{W}_k}(X)$ is the projection operator of the matrix X onto the set $\mathcal{W}_k$. The expression for the gradient $\mu_w \nabla_{W_k} Q(W_{t-1}, y_t^o; x_t)$ will be given further ahead in (3.49)–(3.52). The proximal operator of a vector function $h(u)$ is defined as [103, p.6]:

$$\text{prox}_h(x) \triangleq \arg\min_u \left(h(u) + \frac{1}{2}\|u - x\|_2^2 \right) \tag{3.42}$$

For a matrix function $h(U)$, the proximal operator assumes the same form as (3.42) except that the Euclidean norm in (3.42) is replaced by the Frobenius norm. The proximal operator for $\mu_w \cdot h_{W_k}(W_k) = \mu_w \beta \cdot |||W_k|||_1$ used in the bi-clustering task in Table 3.1 is the entry-wise soft-thresholding function [103, p.191]:

$$\text{prox}_{\mu_w \cdot h_{W_k}}(\cdot) = \text{prox}_{\mu_w \beta \cdot |||W_k|||}(\cdot) = \mathcal{T}_{\mu_w \cdot \beta}(\cdot) \tag{3.43}$$

and the proximal operator for $h_{W_k}(W_k) = 0$ for other cases in Table 3.1 is the identity mapping:

$$\text{prox}_0(x) = x \tag{3.44}$$

With regards to the projection operator used in (3.41), we provide some examples of interest for the current work. If the constraint set $\mathcal{W}_k$ is of the form:

$$\mathcal{W}_k = \{W_k : \ \|[W_k]_{:,q}\|_2 \leq 1\} \tag{3.45}$$

then the projection operator $\Pi_{\mathcal{W}_k}(\cdot)$ is given by [103, 130]

$$[\Pi_{\mathcal{W}_k}(X)]_{:,n} = \begin{cases} [X]_{:,n}, & \|[X]_{:,n}\|_2 \leq 1 \\ \frac{[X]_{:,n}}{\|[X]_{:,n}\|_2}, & \|[X]_{:,n}\|_2 > 1 \end{cases} \tag{3.46}$$

On the other hand, if the constraint set $\mathcal{W}_k$ is of the form:

$$\mathcal{W}_k = \{W_k : \ \|[W_k]_{:,q}\|_2 \leq 1, \ \mathcal{W} \succeq 0\} \tag{3.47}$$

then the projection operator $\Pi_{\mathcal{W}_k}(\cdot)$ becomes

$$[\Pi_{\mathcal{W}_k}(X)]_{:,n} = \begin{cases} \big([X]_{:,n}\big)_+, & \|\big([X]_{:,n}\big)_+\|_2 \leq 1 \\ \dfrac{\big([X]_{:,n}\big)_+}{\|\big([X]_{:,n}\big)_+\|_2}, & \|\big([X]_{:,n}\big)_+\|_2 > 1 \end{cases} \tag{3.48}$$

where $(x)_+ = \max(x, 0)$, i.e., it replaces all the negative entries of a vector x with zeros.

Now, we return to derive the expression for the gradient $\nabla_{W_k} Q(W_{t-1}, y_t^o; x_t)$ in (3.41). By (3.13), we have

$$\nabla_{W_k} Q(W_{t-1}, y_t^o; x_t) = -f_u'\Big(x_t - \sum_{k=1}^{N} W_{k,t-1} y_{k,t}^o\Big)(y_{k,t}^o)^T \tag{3.49}$$

where $f_u'(u)$ denotes the gradient of $f(u)$ with respect to the residual u. On the face of it, expression (3.49) requires global knowledge by agent k of all sub-

dictionaries $\{W_k\}$ across the network, which goes against the desired objective of arriving at a distributed implementation. However, we can develop a distributed algorithm by exploiting the structure of the problem as follows. Note from (3.16) that the optimal inference result should satisfy:

$$\begin{cases} 0 = \frac{\partial}{\partial z} L(\{y_{k,t}^o\}, z_t^o, \nu_t^o; x_t) \\ 0 = \frac{\partial}{\partial \nu} L(\{y_{k,t}^o\}, z_t^o, \nu_t^o; x_t) \end{cases} \quad \Leftrightarrow \quad \begin{cases} 0 = -f_u'(x_t - z_t^o) + \nu_t^o \\ z_t^o = \sum_{k=1}^{N} W_{k,t-1} y_{k,t}^o \end{cases} \tag{3.50}$$

which leads to

$$\begin{aligned} 0 &= -f_u'\Big(x_t - \sum_{k=1}^{N} W_{k,t-1} y_{k,t}^o\Big) + \nu_t^o \\ \Leftrightarrow \quad \nu_t^o &= f_u'\Big(x_t - \sum_{k=1}^{N} W_{k,t-1} y_{k,t}^o\Big) \end{aligned} \tag{3.51}$$

and, hence, the optimal dual variable ν_t^o will be equal to the gradient. Substituting (3.51) into (3.49), the dictionary learning update (3.41) becomes

$$W_{k,t} = \Pi_{\mathcal{W}_k} \Big\{ \mathrm{prox}_{\mu_w \cdot h_{W_k}} \big(W_{k,t-1} + \mu_w \nu_t^o (y_{k,t}^o)^T \big) \Big\} \tag{3.52}$$

which is now in a fully-distributed form. At each agent k, the above ν_t^o can be replaced by the estimate $\nu_{k,i}$ after a sufficient number of inference iterations (large enough i). We note that the dictionary learning update (3.52) has the following important interpretation. Let

$$u_t^o \triangleq x_t - \sum_{k=1}^{N} W_{k,t-1} y_{k,t}^o \tag{3.53}$$

which is the optimal prediction residual error using the entire existing dictionary

set $\{W_{k,t-1}\}_{k=1}^N$. Observe from (3.51) that ν_t^o is the gradient of the residual function $f(u)$ at the optimal u_t^o. The update term for dictionary element k in (3.52) is effectively the correlation between ν_t^o, the gradient of the residual function $f(u_t^o)$, and the coefficient $y_{k,t}^o$ (the activation) at agent k. In the special case of $f(u) = \frac{1}{2}\|u\|_2^2$, expression (3.51) implies that

$$\nu_t^o = u_t^o = x_t - \sum_{k=1}^N W_{k,t-1} y_{k,t}^o \tag{3.54}$$

In this case, ν_t^o has the interpretation of being equal to the optimal prediction residual error, u_t^o, using the entire existing dictionary set $\{W_{k,t-1}\}_{k=1}^N$. Then the update term for dictionary element k in (3.52) becomes the correlation between the optimal prediction error $\nu_t^o = u_t^o$ and the coefficient $y_{k,t}^o$ at agent k. Furthermore, recursion (3.52) reveals that, for each input data sample x_t, after the dual variable ν_t^o is obtained at each agent, there is no need to further exchange any information among agents in order to update their own sub-dictionaries. In other words, the dual variable ν_t^o already provides sufficient and critical information required for distributed dictionary updates. The fully distributed algorithm for dictionary learning is listed in Algorithm 3.1 and is also illustrated in Fig. 3.4.

3.4 Important Special Cases and Experiments

In this section, we specialize the general dictionary learning algorithm and apply it to two problems involving image denoising and novel document/topic detection.

Algorithm 3.1 Model-distributed diffusion strategy for dictionary learning (Main algorithm)

for each input data sample x_t **do**

Compute ν_t^o by iterating (3.32a)-(3.32b) until convergence: $\nu_t^o \approx \nu_{k,i}$. That is:

$$\begin{cases} \psi_{k,i} = \nu_{k,i-1} - \mu \cdot \nabla_\nu J_k(\nu_{k,i-1}; x_t) \\ \nu_{k,i} = \Pi_{\mathcal{V}_f} \left\{ \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \right\} \end{cases}$$

for each agent k **do**

Compute coefficient $y_{k,t}^o$ using Table 3.2 or (3.38):

$$y_{k,t}^o = \arg\max_{y_k} \left[(W_k^T \nu_t^o)^T y_k - h_{y_k}(y_k) \right]$$

Adjust dictionary element $W_{k,t}$ using (3.52):

$$W_{k,t} = \Pi_{\mathcal{W}_k} \left\{ \text{prox}_{\mu_w \cdot h_{W_k}} \left(W_{k,t-1} + \mu_w \nu_t^o (y_{k,t}^o)^T \right) \right\}$$

end for

end for

3.4.1 Tuning of the inference step-size

In the following experiments, it is necessary to select properly the step-size μ for the diffusion algorithm (3.32a)–(3.32b) to ensure that the estimate for ν_t^o converges sufficiently close to it after a reasonable number of iterations.

To choose μ, we first choose the number of diffusion iterations that can be afforded for the task of estimating ν_t^o, say, 1000. Second, we choose a data sample x from the training dataset. Using this x, we compute the optimal solution $y^o \triangleq \text{col}\{y_1^o, \ldots, y_N^o\}$ and its respective dual variable ν^o to the inference problem (3.15a)–(3.15b) using a non-distributed optimization package such as CVX [61]. We then adjust μ by plotting the signal-to-noise measures $\|y^o\|^2/\|y_i - y^o\|^2$ and $\|\nu^o\|^2/\|\nu_{k,i} - \nu^o\|^2$ against the iteration number i, as illustrated in Fig. 3.5. The value $\nu_{k,i}$ is obtained from the distributed algorithm (see (3.32b), (3.36b) or

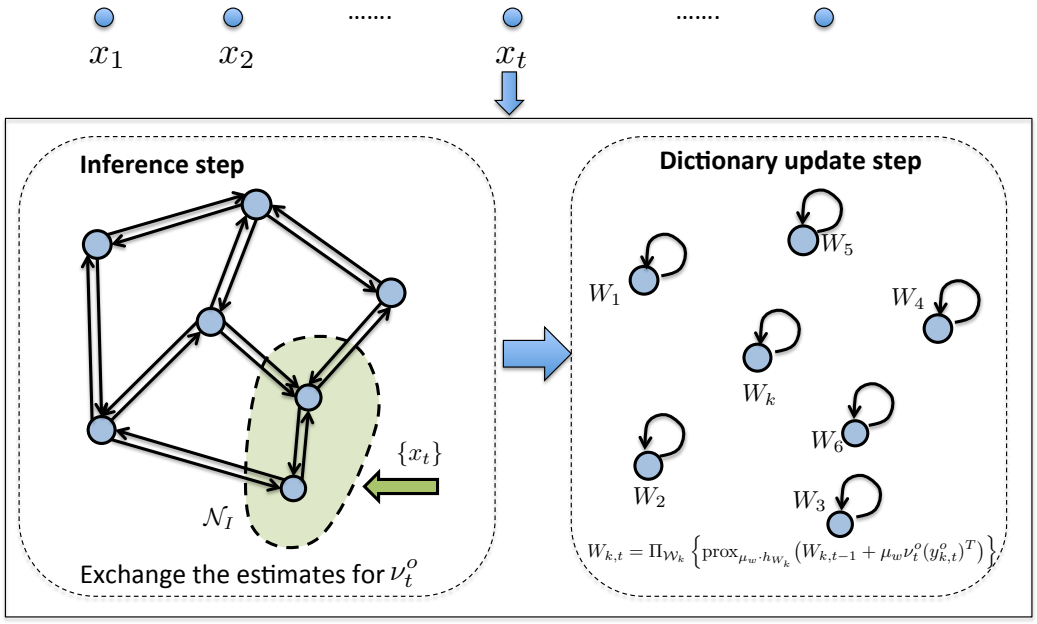


Figure 3.4: The distributed inference step and the dictionary update step over distributed models. In the inference step, after each data sample x_t arrives at a subset of the agents in the network, all the agents find the corresponding optimal dual variable ν_t^o by exchanging the estimates of ν_t^o with neighbors. In the dictionary update step, agents update their sub-dictionaries locally on their own using a step of proximal stochastic gradient descent as (3.52).

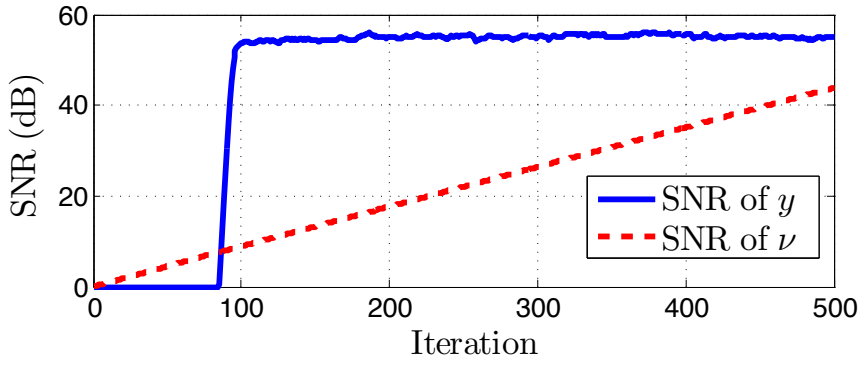


Figure 3.5: Learning curve for the Huber document detection example described by Alg. 3.4 with $\mu = 0.5$.

(3.37c)) at each iteration i and $y_i \triangleq \text{col}\{y_{1,i}, \ldots, y_{N,i}\}$ is calculated at each iteration according to:

$$y_{k,i} = \arg\max_{y_k} \left[(W_k^T \nu_{k,i})^T y_k - h_{y_k}(y_k) \right] \tag{3.55}$$

The chosen value of μ must guarantee that both curves reach an acceptable SNR value (around 40-50dB in this example) for the chosen number of diffusion iterations. Observe that the primal variable y generally reaches a high SNR value before the dual variable ν, but both are required to be found with reasonable accuracy for the dictionary update step (see (3.52)).

3.4.2 Image Denoising via Dictionary Learning

The image denoising application has been a staple of dictionary learning tasks [53, 86, 93, 95]. The task is to denoise an image corrupted by white Gaussian noise. In this section, we compare the performance of the proposed distributed algorithm to that of the centralized solution from [93]. We consider two simulation settings. In the first setting, a single agent has the image data. In the second setting, all agents in the network are assumed to have access to the image data. In the simulations, we choose $f(u)$, $h_{y_k}(y_k)$ and $h_{W_k}(W_k)$ according to the second row of Table 3.1.

The example we consider involves learning a 100×196 dictionary W over a network of $N = 196$ agents. The network is generated according to a random graph, where the probability that any agent is connected to another agent is 0.5. The network connectivity is checked by inspecting the algebraic connectivity of the graph Laplacian matrix, and we will repeat this random graph generation until we find a connected topology [115]. Each agent in the network is in charge

of one dictionary element. We extract a total of one million 10×10 patches from images 101-200 of the non-calibrated natural image dataset [63]. Each image is originally 1536×1024 pixels in size, but the outer most two-pixel border was discarded from each image. We then consider the top-left 1019×1019 pixels for patch extraction. With each data sample being a 10×10 patch, the dimension of the input data sample is $M = 100$ (vertically stacked columns). In each experiment, we randomly initialize each entry of the dictionary matrix W with a zero mean unit variance Gaussian random variable. The columns are then scaled to guarantee that the sub-unit-norm constraint (3.4) is satisfied. Furthermore, in the combination step (3.32b) of the distributed inference, we use the Metropolis rule [26,115,146], which is known to be doubly-stochastic. The patch extraction, preprocessing, and image reconstruction code (excluding dictionary learning and patch inference steps) is borrowed from [104].

We simulate the following two setups of the diffusion algorithm (see (3.30)):

1. Only node 1 has access to the image data, x_t ($\mathcal{N}_I = \{1\}$).
2. All nodes have access to the same image data, x_t ($\mathcal{N}_I = \{1, \ldots, N\}$).

In the first case, the other nodes in the network are unaware of the incoming data. In this way, they are only helping in the inference task despite the lack of information. To derive the algorithm, we note from (3.30) that $J_k(\nu; x)$ is given by:

$$J_k(\nu; x_t) \triangleq \begin{cases} -\dfrac{1}{|\mathcal{N}_I|}\nu^T x_t + \dfrac{1}{N} f^\star(\nu) + h^\star_{y_k}(w_k^T \nu), & k \in \mathcal{N}_I \\ \dfrac{1}{N} f^\star(\nu) + h^\star_{y_k}(w_k^T \nu), & k \notin \mathcal{N}_I \end{cases} \tag{3.56}$$

where we are using w_k instead of W_k because each agent k is in charge of one atom of the dictionary (i.e., the k-th column of W). Since, for this example, we are

setting $f(u) = \frac{1}{2}\|u\|^2$ and $h_{y_k}(y_k) = \gamma\|y\|_1 + \frac{\delta}{2}\|y\|_2^2$ (according to the second row of Table 3.1), we have that $f^\star(\nu) = \frac{1}{2}\|\nu\|_2^2$, $\mathcal{V}_f = \mathbb{R}^M$, and $h^\star_{y_k}(w_k^T\nu) = \mathcal{S}_{\frac{\gamma}{\delta}}\left(\frac{w_k^T\nu}{\delta}\right)$ according to Table 3.2. A straightforward calculation then shows that

$$\nabla_\nu f^\star(\nu) = \nu \tag{3.57}$$

$$\nabla_\nu h^\star_{y_k}(w_k^T\nu) = \frac{1}{\delta}\mathcal{T}_\gamma(w_k^T\nu)w_k \tag{3.58}$$

Substituting (3.57)–(3.58) into the gradient of (3.56), we obtain:

$$\nabla_\nu J_k(\nu; x_t) = \begin{cases} -\frac{x_t}{|\mathcal{N}_I|} + \frac{\nu}{N} + \frac{\mathcal{T}_\gamma(w_k^T\nu)w_k}{\delta}, & k \in \mathcal{N}_I \\ \frac{1}{N}\nu + \frac{1}{\delta}\mathcal{T}_\gamma(w_k^T\nu)w_k, & k \notin \mathcal{N}_I \end{cases} \tag{3.59}$$

By substituting (3.59) into the inference part of Alg. 3.1, we immediately obtain the inference part of Alg. 3.2. The learning portion of the algorithm (adaptation of w_k) is the same for both setups. First, we need to compute $y^o_{k,t}$ at node k once ν^o_t has been estimated. With our choices of $f(u)$ and $h(y_k)$, we observe from Table 3.2 that $y^o_{k,t}$ may be obtained as $y^o_{k,t} = \mathcal{T}_{\frac{\gamma}{\delta}}\left(\frac{w_k^T\nu_t^o}{\delta}\right) = \frac{1}{\delta}\mathcal{T}_\gamma\left(w_k^T\nu_t^o\right)$ (as listed in Alg. 3.2). Now, using the fact that $h_{w_k}(w_k) = 0$ (see Table 3.1), we have that the update rule for w_k from Alg. 3.1 becomes

$$w_{k,t} = \Pi_{\mathcal{W}_k}\left\{w_{k,t-1} + \mu_w\nu_t^o y^o_{k,t}\right\}$$

where $\mathcal{W}_k = \{w : \|w\|_2 \leq 1\}$ (see Table I).

For the dictionary learning, we utilize $\gamma = 45$, $\delta = 0.1$, and $\mu = 0.7$. Computer code from the SPAMS toolbox [92] was used to compare the algorithm from [93] using its default parameters except where otherwise stated. We used $\gamma = 45$ and

Algorithm 3.2 Model-distributed diffusion strategy for image denoising.

for each input data sample x_t, each node k **do**

Repeat until convergence:

$$\begin{cases}\psi_{k,i}=\nu_{k,i-1}-\mu\left(\frac{\nu_{k,i-1}}{N}-\frac{x_t}{|\mathcal{N}_I|}\theta_k\right)-\frac{\mu}{\delta}\mathcal{T}_\gamma(w_{k,t-1}^T\nu_{k,i-1})w_{k,t-1}\\ \nu_{k,i}=\sum_{\ell\in\mathcal{N}_k}a_{\ell k}\psi_{\ell,i}\end{cases}$$

where $\theta_k = 1$ when $k \in \mathcal{N}_I$ and is zero otherwise.

Set $\nu_t^o = \nu_{k,i}$. Compute $y_{k,t}^o = \frac{1}{\delta}\mathcal{T}_\gamma(w_{k,t-1}^T\nu_t^o)$.

Update the dictionary using:

$$w_{k,t} = \Pi_{\|w\|_2\leq 1}\left\{w_{k,t-1}+\mu_w\nu_t^o y_{k,t}^o\right\}$$

end for

$\delta = 0.1$ when training the dictionary with the algorithm from [93]. A step-size of $\mu_w = 5\times 10^{-5}$ was utilized for adapting the dictionary atoms in our Alg. 3.2. The number of iterations for the diffusion algorithm to optimize (3.8) was chosen to be 300 iterations. The data was presented in minibatches [45] of size four samples/minibatch and therefore the dictionary update gradients $\nu_t^o y_{k,t}^o$ were averaged over the four samples at each step[4]. The results are shown in Fig. 3.6. We observe that all dictionaries exhibit edge detection-like features. In denoising Fig. 3.6, the sparsity regularizer γ remained at $\gamma = 45$ for all algorithms and the step-size for our algorithm's inference was also increased to $\mu = 1$ to increase the quality of the inference result (ν). The number of iterations of the inference step increased to 500 iterations to ensure convergence and $\delta = 0.1$ remained constant for all algorithms. The PSNR[5] of the original corrupted image is 14.06dB, while the algorithm from [93] and our proposed distributed algorithm attain PSNR values of 21.77dB, 21.97dB (when the data is only available at agent 1), and

[4]We perform the inference for four samples at a time, for example, (x_1, x_2, x_3, x_4) to obtain $\{\nu_{k,1}^o, \nu_{k,2}^o, \nu_{k,3}^o, \nu_{k,4}^o\}$ (all using the same dictionary W). Then, we update W by averaging the gradient listed in (3.52) for those four samples.

[5]PSNR is the peak-signal-to-noise ratio defined as PSNR $\triangleq 10\log_{10}(I_{\max}^2/\text{MSE})$, where $I_{\max}$ is the maximum pixel intensity in the image and MSE is the mean-square-error over all image pixels.

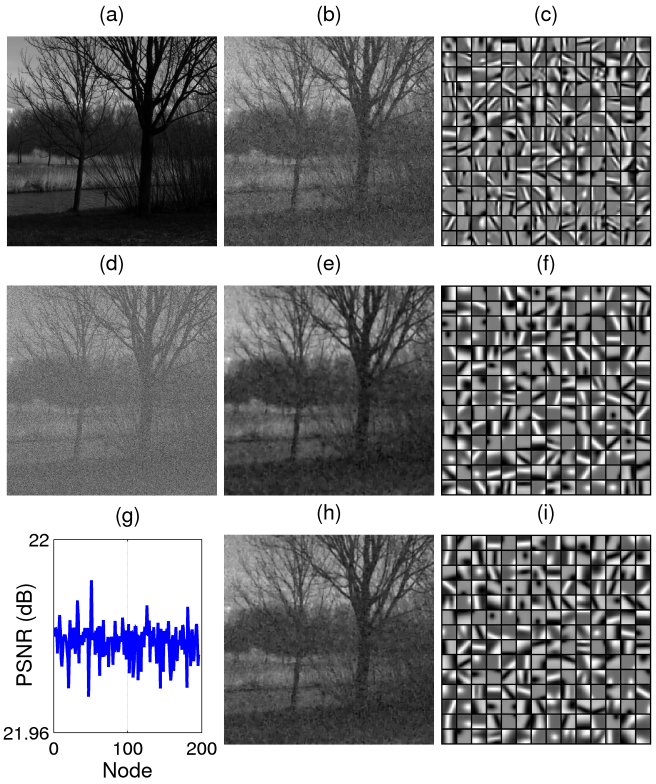


Figure 3.6: Application of dictionary learning to image denoising. (a) Original image; (b) denoised image by using the centralized method from [93]; (c) dictionary obtained by the centralized method from [93]; (d) image corrupted by additive white Gaussian noise; (e) denoised image by our proposed distributed method assuming only node 1 has access to the image; (f) dictionary obtained by our proposed distributed method obtained by only providing node 1 with the image data; (g) PSNR over the network if all nodes have access to the image data; (h) denoised image by our proposed distributed method at agent 1 assuming that all nodes have access to the image data, and (i) dictionary obtained by our proposed distributed method obtained by providing all nodes with the image data.

21.98dB (when the data is available to all nodes), respectively. Furthermore, we also show the PSNR of the recovered image at different agents in the network for the third case by using our distributed strategy. We can see that the performance is relatively uniform (around 21.97dB) across the network, meaning that while

no agent in the network had access to the entire dictionary, all agents were able to obtain a 7dB improvement in the PSNR of the corrupted image. In addition, even when only a single agent in the network has access to the data samples themselves, and does not have access to the entire dictionary, this one agent can still obtain the 7dB improvement in PSNR by cooperation.

3.4.3 Novel Document Detection via Dictionary Learning

In this section, we demonstrate our algorithm's performance on the novel document detection task [3,78,128]. In this application, a stream of documents arrives in blocks at the network, and the task is to detect which of the documents in the incoming batch are associated with topics that have not been observed previously, and to incorporate the new block of data into the knowledge database to detect new topics/documents in future incoming batches. We will simulate our dictionary learning algorithm on two different setups: 1) using the square-Euclidean norm as the residual metric $f(\cdot)$, and 2) using the Huber cost function as the residual metric. In the first setup, we compare our algorithm performance to that of the SPAMS toolbox [92,93] on the NIST Topic Detection and Tracking corpus (TDT2) dataset [22] where a test set from the corpus is separated out and the algorithm is repeatedly tested on it. In the second case, we use the same setup as in [78]. The TDT2 dataset contains news documents associated with their dominant topics collected over the first 27 weeks of 1998. The documents have been processed so that only the most frequent 30 topics (and documents associated with them) are preserved. In this experiment, we allow all agents in the network to observe the incoming data. The key observation is that if a document belongs to a topic that has been observed previously, then it is expected that the objective value of the optimization problem (3.15a)–(3.15b) will be "small" since

the document should be well modeled by the available dictionary. On the other hand, when the objective value is "large," then this is an indication that the document is not well modeled by the available dictionary, and hence the document is most likely associated with a topic that has not yet been observed.

The experiment setup is as follows. A collection of 1000 documents are presented to the algorithms in order to initialize the dictionary. The algorithm from [78] utilizes the data as a block, while the diffusion-based algorithms utilize the data incrementally. Once the dictionary is initialized, a new collection of 1000 documents are presented to the algorithms. The algorithms then process the data samples in order to determine if each of the new documents belong to a topic that has been previously observed, or not. This is done by determining if the value of the cost function is sufficiently large, in order to deem the data sample "novel." The detection result then produces a receiver operating characteristic (ROC) curve [79, p. 74], illustrated in Figs. 3.7–3.8. Following the production of the ROC curve, the previously new data set becomes the training dataset for the classifier in order to update the dictionary (the dictionary is also expanded at this point by adding nodes to the network). The process then repeats by testing the newly updated dictionary on a new set of documents that later become the training set, etc. We will call each generation of an ROC curve a "time-step" and we will designate it with the variable $1 \leq s \leq 8$ (since the TDT2 dataset only contains enough data for eight time-steps plus an initialization dataset). It is also important to observe that in some time-steps, no documents that are associated with novel classes are introduced to the algorithm, so an ROC curve is thus not generated.

3.4.3.1 Squared-Euclidean-norm Residual

We test our online algorithm on the top 30-category TDT2 dataset [22]. The data is compiled into a term frequency-inverse document frequency matrix $X \in \mathbb{R}^{M \times T}$, where $M = 19527$ and $T = 9394$, and normalized so that each column would possess a unit Euclidean norm. Out of the entire 9394 samples, we choose 1000 samples at random and set those aside as a test set. We verify that all 30 categories appear in this test set. The remaining data are ordered in the order of topics and used as the training set. For each algorithm, we create a non-negative random dictionary, initially of size $M \times 10$, but after each examination, the dictionary size is increased by 10 atoms. In the distributed algorithm implementation, each node in the network is responsible for a single atom (therefore, after each time step, 10 new nodes enter the network). When the dictionary size is increased, the previous atoms are preserved for all algorithms. All algorithms utilize $\gamma = 0.05$ and $\delta = 0.1$, and we do not utilize minibatches for any algorithm.

At each time step, each algorithm receives the same batch of 1000 document feature vectors. We test our algorithm in two cases: 1) fully connected, and 2) distributed. In the distributed case, a random topology is generated at each time step, where the probability of any two nodes being connected is 0.5. All algorithms are only allowed to observe each data sample once during the training of the dictionary (single epoch learning). We once again utilize the Metropolis rule to generate the combination matrix in a fully distributed manner. Both the fully connected and distributed algorithms utilize a learning step-size of $\mu_w(s) = 10/s$, where s is the current time-step for learning of the dictionary. For the inference, the fully connected algorithm utilizes $\mu^{\text{FC}} = 0.7$, while the distributed algorithm uses $\mu = 0.05$. The fully connected algorithm performs 100 iterations for the inference, while the distributed algorithm utilizes 1000 iterations for the inference

(the choice of the number of iterations and μ is discussed earlier in Sec. 3.4.1).

To obtain the distributed algorithm, we have from (3.30) that

$$J_k(\nu; x_t) \triangleq \frac{1}{N}(f^\star(\nu) - \nu^T x_t) + h^\star_{y_k}(w_k^T \nu), \tag{3.60}$$

where we let $\mathcal{N}_I = \mathcal{N}$ and all the agents in the network have access to x_t and each agent is in charge of one atom of the dictionary, i.e., w_k. Since we now choose $f(u) = \frac{1}{2}\|u\|^2$ and $h_{y_k}(y_k) = \gamma\|y\|_{1,+} + \frac{\delta}{2}\|y\|_2^2$ (according to the fourth row of Table 3.1), we have that $f^\star(\nu) = \frac{1}{2}\|\nu\|_2^2$, $h^\star_{y_k}(w_k^T\nu) = \mathcal{S}^+_{\frac{\gamma}{\delta}}\left(\frac{w_k^T\nu}{\delta}\right)$, and $\mathcal{V}_f = \mathbb{R}^M$ according to Table 3.2. A straightforward calculation then gives

$$\nabla_\nu f^\star(\nu) = \nu \tag{3.61}$$

$$\nabla_\nu h^\star_{y_k}(w_k^T\nu) = \frac{1}{\delta}\mathcal{T}^+_\gamma(w_k^T\nu)w_k \tag{3.62}$$

Substituting (3.61)–(3.62) into the gradient of (3.60), we obtain:

$$\nabla_\nu J_k(\nu; x_t) = \frac{1}{N}(\nu - x_t) + \frac{1}{\delta}\mathcal{T}^+_\gamma(w_k^T\nu)w_k \tag{3.63}$$

By substituting (3.63) into the inference part of Alg. 3.1, we immediately obtain the inference part of Alg. 3.3. For the learning portion of the algorithm, we need to compute $y^o_{k,t}$ at node k once ν^o_t has been estimated. With our choices of $f(u)$ and $h_{y_k}(y_k)$, we observe from Table 3.2 that $y^o_{k,t}$ may be obtained as

$$y^o_{k,t} = \mathcal{T}^+_{\frac{\gamma}{\delta}}\left(\frac{w_k^T\nu_t^o}{\delta}\right) = \frac{1}{\delta}\mathcal{T}^+_\gamma\left(w_k^T\nu_t^o\right) \tag{3.64}$$

Now, using the fact that $h_{w_k}(w_k) = 0$ (see Table 3.1), we have that the update

rule for w_k from Alg. 3.1 becomes

$$w_{k,t} = \Pi_{\mathcal{W}_k} \left\{ w_{k,t-1} + \mu_w \nu_t^o y_{k,t}^o \right\} \tag{3.65}$$

where $\mathcal{W}_k = \{w : \|w\|_2 \leq 1, w \succeq 0\}$ (see Table 3.1).

The fully connected version of the algorithm may be attained by replacing μ with μ^{FC} and the combination matrix with $A = \frac{1}{N}\mathbb{1}\mathbb{1}^T$. In the algorithm, χ is the threshold used to distinguish between novel and non-novel documents; it is treated as a tunable parameter in order to generate the ROC curves. Interestingly, since strong duality holds for this example, we do not need to recover z_t^o in (3.39), but we only need to recover the cost value, $g(\nu_t^o, h_t)$ for a test data sample h_t; we use h_t to differentiate it from the training data sample x_t. This can be done in many ways, one of them being the diffusion strategy. In order to obtain a scaled multiple of $g(\nu_t^o, x_t) = -\sum_{k=1}^N J_k(\nu_t^o, h_t)$, we setup the following scalar optimization problem:

$$\min_g \sum_{k=1}^N V_k(g) \tag{3.66}$$

where

$$V_k(g) \triangleq \frac{1}{2} \left(J_k(\nu_t^o, h_t) + g \right)^2 \tag{3.67}$$

from which we can obtain the following scalar diffusion algorithm [36]:

$$\begin{cases} \phi_k(i) = g_k(i-1) - \mu_g (J_k(\nu_t^o, h_t) + g_k(i-1)) \\ g_k(i) = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \phi_\ell(i) \end{cases} \tag{3.68}$$

After sufficient iterations, recursion (3.68) approximates the following value:

$$g_t^o = -\frac{1}{N}\sum_{k=1}^{N} J_k(\nu_t^o, h_t) \tag{3.69}$$

This is sufficient since the positive scaling factor, $1/N$, may be absorbed into the threshold parameter, χ.

In Fig. 3.7, we show the result of the experiment for the three algorithms. We observe that for the first two time steps, the algorithm from [93] slightly outperforms our fully connected and distributed algorithms. Recall, however, that the algorithm from [93] fully optimizes y_k, while we stop at 100 and 1000 iterations for fully connected and distributed algorithms, respectively. After the first two time steps, the algorithm from [93] never again outperforms our algorithms.

We also list the area under the curves in Table 3.3. The cases where the distributed algorithm outperforms the fully connected algorithm (although by a small amount) can be explained by different random initialization of the dictionary atoms.

Table 3.3: Area under the curve measure for the three tested algorithms.

Time Step	[93]	Diffusion (Fully Connected)	Diffusion
1	**0.97**	0.93	0.94
2	**0.95**	0.91	0.90
3	0.75	0.89	**0.91**
4	0.78	0.91	**0.92**
5	0.78	**0.91**	**0.91**
6	0.72	**0.92**	**0.92**
7	0.66	**0.90**	0.85
8	0.55	**0.87**	0.78

Algorithm 3.3 Model-distributed diffusion strategy for distributed novel document detection (Square Euclidean Norm Residual).

for each time step $s = 1, 2, \ldots, 8$ **do**
Dictionary Learning:
for each training data sample x_t^s from time-step s, each node k **do**

Repeat until convergence:

$$\begin{cases} \psi_{k,i} = \nu_{k,i-1} - \frac{\mu}{N}(\nu_{k,i-1} - x_t^s) - \frac{\mu}{\delta}\mathcal{T}_\gamma^+(w_{k,t-1}^T \nu_{k,i-1}) w_{k,t-1} \\ \nu_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{cases}$$

Set $\nu_t^o = \nu_{k,i}$. Compute $y_{k,t}^o = \frac{1}{\delta}\mathcal{T}_\gamma^+(w_{k,t-1}^T \nu_t^o)$.
Update the dictionary using:

$$w_{k,t} = \Pi_{\|w\|_2 \leq 1}\left\{\Pi_{w \succeq 0}\left\{w_{k,t-1} + \mu_w(s)\nu_t^o y_{k,t}^o\right\}\right\}$$

end for
Novel Document Detection:
for each test data sample h_t, each node k **do**

Repeat until convergence:

$$\begin{cases} \psi_{k,i} = \nu_{k,i-1} - \frac{\mu}{N}(\nu_{k,i-1} - h_t) - \frac{\mu}{\delta}\mathcal{T}_\gamma^+(w_{k,t-1}^T \nu_{k,i-1}) w_{k,t-1} \\ \nu_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{cases}$$

Set $\nu_t^o = \nu_{k,i}$.
Perform diffusion strategy to optimize (3.66) until convergence:

$$\begin{cases} \phi_k(i) = g_k(i-1) - \mu_g(J_k(\nu_t^o, h_t) + g_k(i-1)) \\ g_k(i) = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \phi_\ell(i) \end{cases}$$

where $J_k(\nu, \cdot)$ is defined in (3.60).
Set $g_t^o = g_{k,i}$.
if $g_t^o > \chi$ **then**
declare document as novel.
else
declare document as not novel.
end if
end for
Add nodes to network (expand the dictionary)
end for

3.4.3.2 Huber Residual

We now test our algorithm on $f(u) = \sum_{m=1}^M L(u_m)$, where $L(u_m)$ is the scalar Huber function defined in Table 3.2. Interestingly, the conjugate function $f^\star(\cdot)$ of the Huber residual is strongly-convex. This allows our algorithm to guarantee

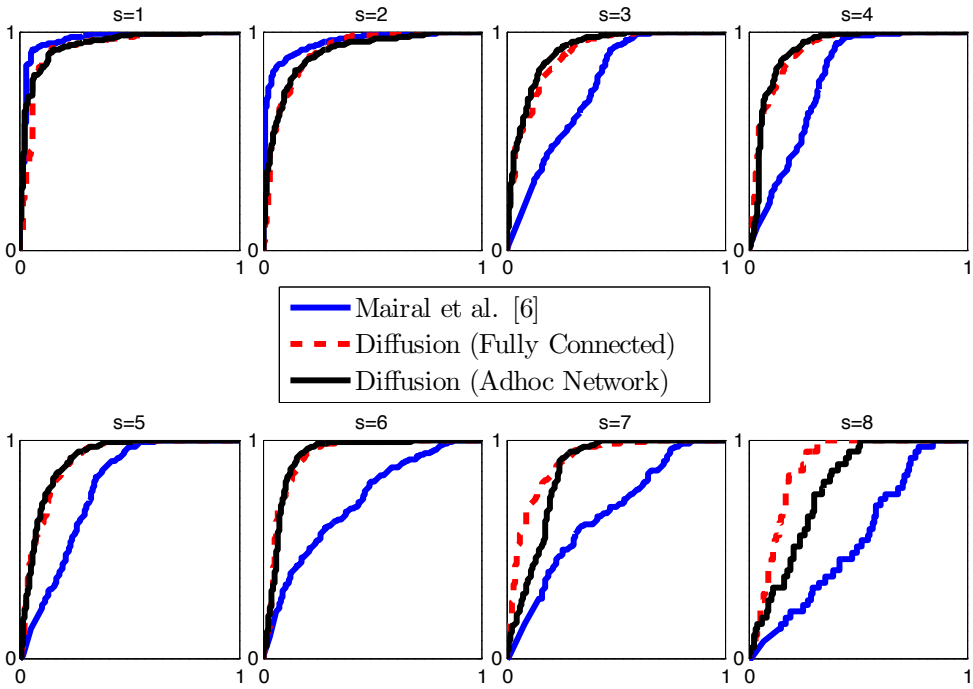


Figure 3.7: Application of dictionary learning to novel document/topic detection. At each time step, the algorithms receive 1000 documents. The task is to determine which documents are associated with topics that have already been observed, and which documents are associated with topics that have not yet been observed. These curves represent the ROC associated with each time step against a fixed test set. The x-axis represents probability of false alarm while the y-axis represents the probability of detection. The area under each curve is listed in Table 3.3.

relatively fast convergence. The setup for this section is the same as in [78][6], except that we start with only ten dictionary atoms, and add ten additional atoms after each time-step. We simulate the last line of the non-negative matrix factorization setup in Table 3.2. We compare our algorithm to the one proposed in [78], which simulates the setup where $f(u) = \|u\|_1$, $h_y(y) = \|y\|_1$, and $\mathcal{W}_k = \{w : \|w\|_1 \leq 1\}$.

[6]We would like to thank S. P. Kasiviswanathan for kindly sharing his MATLAB code though e-mail communication in order to reproduce the simulation in [78], including the ordered data.

Algorithm 3.4 Model-distributed diffusion strategy for distributed novel document detection (Huber Loss Residual).

for each time step $s = 1, 2, \ldots, 8$ **do**

Dictionary Learning:

for each training data sample x_t^s from time-step s, each node k **do**

Repeat until convergence:

$$\begin{cases} \psi_{k,i} = \nu_{k,i-1} - \frac{\mu}{N}(\eta\nu_{k,i-1} - x_t^s) - \frac{\mu}{\delta}\mathcal{T}_\gamma^+(w_{k,t-1}^T\nu_{k,i-1})w_{k,t-1} \\ \nu_{k,i} = \Pi_{\nu\in[-1,1]}\left\{\sum_{\ell\in\mathcal{N}_k} a_{\ell k}\psi_{\ell,i}\right\} \end{cases}$$

where the above projection is carried out according to (3.35).

Set $\nu_t^o = \nu_{k,i}$. Compute $y_{k,t}^o = \frac{1}{\delta}\mathcal{T}_\gamma^+(w_{k,t-1}^T\nu_t^o)$.

Update the dictionary using:

$$w_{k,t} = \Pi_{\|w\|_2\le 1}\left\{\Pi_{w\succeq 0}\left\{w_{k,t-1} + \mu_w(s)\nu_t^o y_{k,t}^o\right\}\right\}$$

end for

Novel Document Detection:

for each test data sample h_t, each node k **do**

Repeat until convergence:

$$\begin{cases} \psi_{k,i} = \nu_{k,i-1} - \frac{\mu}{N}(\eta\nu_{k,i-1} - h_t) - \frac{\mu}{\delta}\mathcal{T}_\gamma^+(w_{k,t-1}^T\nu_{k,i-1})w_{k,t-1} \\ \nu_{k,i} = \Pi_{\nu\in[-1,1]}\left\{\sum_{\ell\in\mathcal{N}_k} a_{\ell k}\psi_{\ell,i}\right\} \end{cases}$$

Set $\nu_t^o = \nu_{k,i}$.

Perform diffusion strategy to optimize (3.66) until convergence:

$$\begin{cases} \phi_k(i) = g_k(i-1) - \mu_g(J_k(\nu_t^o, h_t) + g_k(i-1)) \\ g_k(i) = \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\phi_\ell(i) \end{cases}$$

where $J_k(\nu, \cdot)$ is defined in (3.70).

Set $g_t^o = g_{k,i}$.

if $g_t^o > \chi$ **then**

declare document as novel.

else

declare document as not novel.

end if

end for

Add nodes to network (expand the dictionary)

end for

For the simulation of the diffusion algorithm, the data are normalized so that $\|x_t\|_2 = 1$. On the other hand, when testing on the centralized ADMM-based algorithm from [78], the data are normalized so that $\|x_t\|_1 = 1$, in keeping with the proposed simulation setup there. The constraint set for W for the diffusion-based algorithm is $\{W : \|[W]_{:,q}\|_2 \le 1,\ W \succeq 0\}$, while the constraint set for the ADMM-based algorithm from [78] is $\{W : \|[W]_{:,q}\|_1 \le 1,\ W \succeq 0\}$. We choose

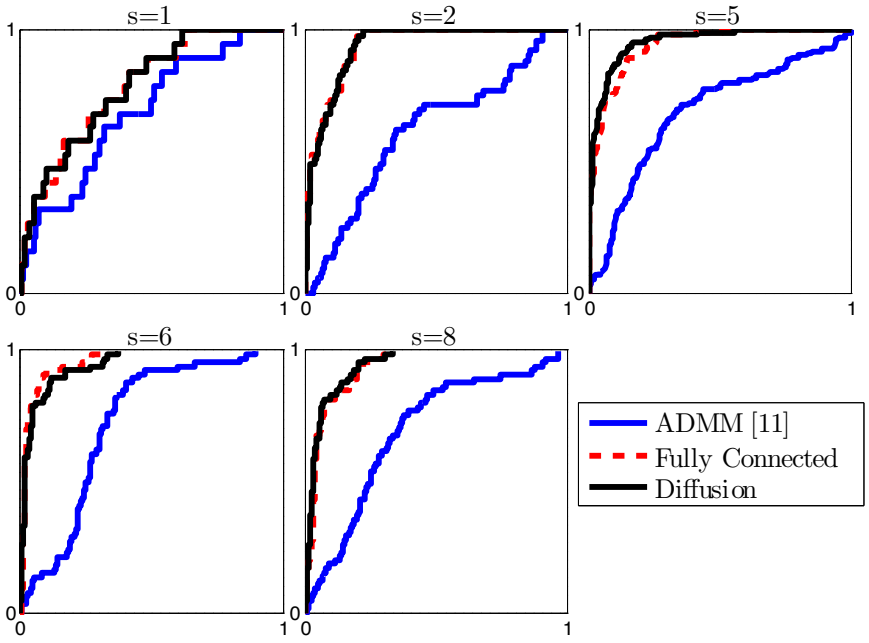


Figure 3.8: Application of dictionary learning to novel document/topic detection. At each time step, the algorithms receive 1000 documents. The task is to determine which documents are associated with topics that have already been observed, and which documents are associated with topics that have not yet been observed. These curves represent the ROC curve associated with each time step against a changing test set. The x-axis represents probability of false alarm while the y-axis represents probability of detection. The area under each cuve is listed in Table 3.4.

$\gamma = 1$ and $\delta = 0.1$. For the initialization of the dictionary for the ADMM algorithm from [78], we let the algorithm iterate between the sparse coding step and the dictionary learning step 35 times. The diffusion algorithm runs through the data once. We choose $\eta = 0.2$ for the connection point between the quadratic part and the linear part of the Huber loss function. We use the same step-size choices as in the square Euclidean norm simulation described in Sec.3.4.3.1. Samples 1-1000 are used for the initialization of the dictionary. In this simulation setup, since the data is ordered differently from the last section (although it is still the TDT2 corpus data), novel documents are only introduced at the first

(samples 1001-2000), second (2001-3000), fifth (5001-6000), sixth (6001-7000), and eighth (8001-9000) time-steps. For this reason, we only execute the novel document detection part of the algorithm at those time-steps, and present the ROC curves for those time-steps. We run our algorithm using the fully connected case, where $A = \frac{1}{N}\mathbb{1}\mathbb{1}^T$ and the distributed case where the probability that two nodes are connected is 0.5, and the combination matrix is the Metropolis rule.

To obtain the distributed algorithm, we note from (3.30) that

$$J_k(\nu; x_t) \triangleq \frac{1}{N}(f^\star(\nu) - \nu^T x_t) + h^\star_{y_k}(w_k^T \nu) \tag{3.70}$$

Since we now use $f(u) = \sum_{m=1}^{M} L(u_m)$ and $h_{y_k}(y_k) = \gamma\|y\|_{1,+} + \frac{\delta}{2}\|y\|_2^2$ (according to the last row of Table 3.1), we obtain that $f^\star(\nu) = \frac{\eta}{2}\|\nu\|_2^2$, $\mathcal{V}_f = \{\nu : \|\nu\|_\infty \leq 1\}$, and $h^\star_{y_k}(w_k^T\nu) = \mathcal{S}^+_{\frac{\gamma}{\delta}}\left(\frac{w_k^T\nu}{\delta}\right)$ according to Table 3.2. A straightforward calculation then shows that

$$\nabla_\nu f^\star(\nu) = \eta \cdot \nu \tag{3.71}$$

$$\nabla_\nu h^\star_{y_k}(w_k^T\nu) = \frac{1}{\delta}\mathcal{T}^+_\gamma(w_k^T\nu)w_k \tag{3.72}$$

Substituting (3.71)–(3.72) into the gradient of (3.70), we obtain:

$$\nabla_\nu J_k(\nu; x_t) = \frac{1}{N}(\eta \cdot \nu - x_t) + \frac{1}{\delta}\mathcal{T}^+_\gamma(w_k^T\nu)w_k \tag{3.73}$$

where we let $\mathcal{N}_I = \mathcal{N}$ and all the agents in the network have access to x_t. By substituting (3.73) into the inference part of Alg. 3.1, we immediately obtain the inference part of Alg. 3.4. For the learning portion of the algorithm, we need to compute $y^o_{k,t}$ at node k once ν^o_t has been estimated. With our choices of $f(u)$ and $h(y_k)$, we observe from Table 3.2 that $y^o_{k,t}$ may be obtained as $y^o_{k,t} = \mathcal{T}^+_{\frac{\gamma}{\delta}}\left(\frac{w_k^T\nu^o_t}{\delta}\right) =$

$\frac{1}{\delta}\mathcal{T}_{\gamma}^{+}\left(w_k^T\nu_t^o\right)$ (as listed in Alg. 3.4). Now, using the fact that $h_{w_k}(w_k) = 0$ (see Table 3.1), we have that the update rule for w_k from Alg. 3.1 becomes

$$w_{k,t} = \Pi_{\mathcal{W}_k}\left\{w_{k,t-1} + \mu_w \nu_t^o y_{k,t}^o\right\} \tag{3.74}$$

where $\mathcal{W}_k = \{w : \|w\|_2 \leq 1, w \succeq 0\}$ (see Table 3.1).

The final algorithm is listed in Alg. 3.4. Again, each node in the network is responsible for a single dictionary atom. The sparse coding stages of the centralized ADMM-based algorithm from [78] utilize 35 iterations, and the number of iterations of the dictionary update steps are capped at 10 for all iterations other than the initialization step. We observe that the performance of the centralized ADMM-based algorithm reproduced in this manuscript is competitive with that in [78], even though the initial dictionary size is chosen to be ten, as opposed to 200 atoms (as was done in the experiment in [78]).

The performance of the algorithms is illustrated in Fig. 3.8. We observe that the Huber loss function improves performance relative to the ℓ_1 function. The area under each ROC curve is listed in Table 3.4. Since the different algorithms were initialized with different dictionaries, it may be possible for the sparsely-connected diffusion strategy to slightly outperform the fully-connected diffusion strategy. We observe this effect in Table 3.4, where the sparsely-connected network outperforms the fully-connected network by 0.01 (area under the curve measure).

3.5 Conclusion

In this chapter, we studied the "cost-of-sum" problem in the context of dictionary learning problems over distributed models, where each agent in a connected

Table 3.4: Area under the curve measure for the three tested algorithms. No novel documents were presented in time-steps 3, 5, and 7.

Time Step	ADMM [78]	Diffusion (Fully Connected)	Diffusion
1	0.69	**0.79**	**0.79**
2	0.61	**0.94**	0.93
5	0.69	0.94	**0.95**
6	0.73	**0.96**	0.95
8	0.69	0.93	**0.94**

network is in charge of a portion of the dictionary atoms and the agents collaborate to represent the data. Using the concepts of conjugate function and dual decomposition, we transform the original global dictionary learning problem into a form that is amenable to distributed optimization, which is solved by diffusion strategy. The collaborative inference step generates the dual variables that can be used by the agents to update their dictionary atoms without the need to share these dictionaries or even the coefficient models for the training data. The proposed algorithm is tested over two typical tasks of dictionary learning, namely, image denoising and novel document detection. The results illustrate the superior performance of our proposed algorithm.

3.A Derivation of Some Typical Conjugate Functions

In this appendix, we derive the conjugate functions listed in Table 3.2. The conjugate functions for $\frac{1}{2}\|u\|_2^2$, and their corresponding domains can be found in [20, pp.90-94]. The conjugate function for the scalar Huber loss $L(u_m)$ can be found in [143] as

$$L^\star(\nu_m) = \frac{1}{2}\nu_m^2, \quad |\nu_m| \leq 1 \tag{3.75}$$

Therefore, by the "sums of independent functions" property[7] in [20, p.95], the conjugate function of $\sum_{m=1}^{M} L(u_m)$ is given by

$$\sum_{m=1}^{M} L^\star(\nu_m) = \sum_{m=1}^{M} \frac{1}{2}\nu_m^2 = \frac{1}{2}\|\nu\|_2^2, \tag{3.76}$$

where the domain is given by

$$|\nu_m| \le 1, \quad m = 1, \ldots, M \quad \Leftrightarrow \quad \|\nu\|_\infty \le 1 \tag{3.77}$$

Next, we derive the conjugate functions for the elastic net regularization term:

$$h_{y_k}(y_k) = \gamma\|y_k\|_1 + \frac{\delta}{2}\|y_k\|_2^2 \tag{3.78}$$

By the definition of conjugate functions in (3.22), we have

$$\begin{aligned}
h_{y_k}^\star(W_k^T\nu) &= \sup_{y_k}\left[(W_k^T\nu)^T y_k - h_{y_k}(y_k)\right] \\
&= -\inf_{y_k}\left[h_{y_k}(y_k) - (W_k^T\nu)^T y_k\right] \\
&= -\inf_{y_k}\left[\gamma\|y_k\|_1 + \frac{\delta}{2}\|y_k\|_2^2 - (W_k^T\nu)^T y_k\right] \qquad (3.79) \\
&= -\delta\cdot\inf_{y_k}\left[\frac{\gamma}{\delta}\|y_k\|_1 + \frac{1}{2}\left\|y_k - \frac{1}{\delta}W_k^T\nu\right\|_2^2\right] \\
&\quad + \frac{1}{2\delta}\|W_k^T\nu\|_2^2 \qquad (3.80)
\end{aligned}$$

where the last step completes the square. Note from (3.42) that the optimal y_k that minimizes the term inside the bracket of (3.80) can be expressed as the proximal operator of $(\gamma/\delta)\|y_k\|_1$, which is known to be given by the entry-wise

[7] If $f(x_1, \ldots, x_N) = f_1(x_1) + \cdots f_N(x_N)$, then the conjugate function for $f(x_1, \ldots, x_N)$ is given by $f^\star(y_1, \ldots, y_N) = f_1^\star(y_1) + \cdots + f_N^\star(y_N)$, where $f_1^\star(y_1), \ldots, f_N^\star(y_N)$ are the conjugate functions for $f_1(x_1), \ldots, f_N(x_N)$, respectively.

soft-thresholding operator [103, p.188] [50]:

$$\begin{aligned} y_{k,t}^{o} &= \arg\min_{y_k} \left[\frac{\gamma}{\delta}\|y_k\|_1 + \frac{1}{2}\left\| y_k - \frac{1}{\delta} W_k^T \nu \right\|_2^2 \right] \\ &= \text{prox}_{\frac{\gamma}{\delta}\|\cdot\|_1}\left(\frac{W_k^T \nu}{\delta} \right) \\ &= \mathcal{T}_{\frac{\gamma}{\delta}}\left(\frac{W_k^T \nu}{\delta} \right) \end{aligned} \tag{3.81}$$

where

$$[\mathcal{T}_\lambda(x)]_n \triangleq (|[x]_n| - \lambda)_+ \text{sgn}([x]_n) \tag{3.82}$$

and

$$(x)_+ = \max(x, 0) \tag{3.83}$$

Substituting (3.81) into (3.79), we obtain

$$h_{y_k}^{\star}(W_k^T \nu) = \mathcal{S}_{\frac{\gamma}{\delta}}\left(\frac{W_k^T \nu}{\delta} \right) \tag{3.84}$$

where

$$\mathcal{S}_{\frac{\gamma}{\delta}}(x) \triangleq -\gamma \cdot \left\| \mathcal{T}_{\frac{\gamma}{\delta}}(x) \right\|_1 - \frac{\delta}{2}\left\| \mathcal{T}_{\frac{\gamma}{\delta}}(x) \right\|_2^2 + \delta \cdot x^T \mathcal{T}_{\frac{\gamma}{\delta}}(x) \tag{3.85}$$

Finally, we derive the conjugate function for the nonnegative elastic net regularization function:

$$h_{y_k}(y_k) = \gamma \|y_k\|_{1,+} + \frac{\delta}{2}\|y_k\|_2^2 \tag{3.86}$$

Following the same line of argument from (3.79) and (3.80), we get

$$h^{\star}_{y_k}(W_k^T\nu) = -\inf_{y_k}\left[\gamma\|y_k\|_{1,+}+\frac{\delta}{2}\|y_k\|_2^2-(W_k^T\nu)^T y_k\right] \tag{3.87a}$$

$$= -\delta\cdot\inf_{y_k}\left[\frac{\gamma}{\delta}\|y_k\|_{1,+}+\frac{1}{2}\left\|y_k-\frac{1}{\delta}W_k^T\nu\right\|_2^2\right] + \frac{1}{2\delta}\|W_k^T\nu\|_2^2 \tag{3.87b}$$

By (3.42), the optimal $y^o_{k,t}$ that minimizes the term inside the bracket of (3.87b) is given by

$$y^o_{k,t} = \arg\min_{y_k}\left[\frac{\gamma}{\delta}\|y_k\|_{1,+} + \frac{1}{2}\left\|y_k-\frac{1}{\delta}W_k^T\nu\right\|_2^2\right] \tag{3.88}$$

Applying an argument similar to the one used in [8], we can express the optimal $y^o_{k,t}$ in (3.88) as

$$y^o_{k,t} = \mathcal{T}^+_{\frac{\gamma}{\delta}}\left(\frac{W_k^T\nu}{\delta}\right) \tag{3.89}$$

where $\mathcal{T}^+_{\lambda}(\cdot)$ is the one-sided soft-thresholding operator:

$$[\mathcal{T}^+_{\lambda}(x)]_n \triangleq ([x]_n-\lambda)_+ \tag{3.90}$$

Substituting (3.89) into (3.87a), we obtain

$$h^{\star}_{y_k}(W_k^T\nu) = \mathcal{S}^+_{\frac{\gamma}{\delta}}\left(\frac{W_k^T\nu}{\delta}\right) \tag{3.91}$$

where

$$\mathcal{S}^+_{\frac{\gamma}{\delta}}(x) \triangleq -\gamma\cdot\left\|\mathcal{T}^+_{\frac{\gamma}{\delta}}(x)\right\|_{1,+}-\frac{\delta}{2}\left\|\mathcal{T}^+_{\frac{\gamma}{\delta}}(x)\right\|_2^2+\delta\cdot x^T\mathcal{T}^+_{\frac{\gamma}{\delta}}(x)$$

$$= -\gamma \cdot \left\| \mathcal{T}^{+}_{\frac{\gamma}{\delta}}(x) \right\|_1 - \frac{\delta}{2} \left\| \mathcal{T}^{+}_{\frac{\gamma}{\delta}}(x) \right\|_2^2 + \delta \cdot x^T \mathcal{T}^{+}_{\frac{\gamma}{\delta}}(x) \tag{3.92}$$

where the last step uses the fact that the output of $\mathcal{T}^{+}_{\gamma}(\cdot)$ is always nonnegative so that $\| \mathcal{T}^{+}_{\frac{\gamma}{\delta}}(x) \|_{1,+} = \| \mathcal{T}^{+}_{\frac{\gamma}{\delta}}(x) \|_1$.

3.B Overview of Duality Theory

In this appendix, we give a brief overview of duality theory in convex optimization. For more thorough treatments of duality theory, the readers are referred to [10, 20, 105]. First, consider the following convex optimization problem:

$$\min_{x} \quad f_0(x) \tag{3.93a}$$

$$\text{s.t.} \quad f_k(x) \leq 0, \quad k = 1, \ldots, K \tag{3.93b}$$

$$Ax = b \tag{3.93c}$$

where $f_0(x), f_1(x), \ldots, f_K(x)$ are convex cost functions defined over $\mathbb{R}^M$, A is an $N \times M$ matrix, and b is an $N \times 1$ vector. Problem (3.93a)–(3.93c) is called the primal problem, and the variable x is called the primal variable. Then, the Lagrangian corresponding to the primal problem (3.93a)–(3.93c) is defined as

$$L(x, \nu, \lambda) = f_0(x) + \nu^T (Ax - b) + \sum_{k=1}^{K} \lambda_k f_k(x) \tag{3.94}$$

$$= f_0(x) + \nu^T (Ax - b) + \lambda^T f(x) \tag{3.95}$$

where $f(x) \triangleq \text{col}\{f_1(x), \ldots, f_K(x)\}$, $\nu \in \mathbb{R}^N$ is the Lagrange multiplier vector associated with the equality constraint (3.93c), $\lambda \triangleq \text{col}\{\lambda_1, \ldots, \lambda_K\}$ is the Lagrange multiplier vector associated with the inequality constraints (3.93b) and is

required to satisfy $\lambda \succeq 0$. The dual function is constructed as

$$g(\nu, \lambda) \triangleq \inf_{x} L(x, \nu, \lambda) \tag{3.96}$$

and the dual problem corresponding to (3.93a)–(3.93c) is defined as

$$\max_{\nu,\lambda} \quad g(\nu, \lambda) \tag{3.97a}$$

$$\text{s.t.} \quad \lambda \succeq 0 \tag{3.97b}$$

We now introduce the concept of *weak duality* [20, pp.225-226].

Theorem 3.1 (Weak duality)**.** *Let $p^\star$ denote the value of $f_0(x)$ at the optimal solution to the primal problem* (3.93a)–(3.93c)*, and let $d^\star$ denote the value of $g(\nu, \lambda)$ corresponding to the optimal solution to the dual problem* (3.97a)–(3.97b)*. Then, it always holds that*

$$p^\star \geq d^\star \tag{3.98}$$

□

In other words, the optimal value of the dual problem is always a lower bound for the optimal value of the primal problem. In fact, the weak duality inequality (3.98) holds even when the primal problem (3.93a)–(3.93c) is not convex (i.e., even when none of the functions $f_0(x), f_1(x), \ldots, f_K(x)$ is convex).

We say that *strong duality* [20, p.226] holds if, and only if

$$p^\star = d^\star \tag{3.99}$$

That is, the optimal value of the dual problem (3.97a)–(3.97b) is equal to the

optimal value of the primal problem (3.93a)–(3.93c). In general, strong duality does not hold for non-convex problems, and it may not even hold for some convex problems. Nevertheless, if certain additional conditions (called *constraint qualifications*) hold in convex optimization problems, then strong duality can be established. One simple and useful (sufficient) constraint qualification is *Slater's condition* [20, p.226–227], which requires that there should exist an $x \in \mathbb{R}^M$ such that

$$f_k(x) < 0, \quad k = 1, \ldots, K, \qquad Ax = b \tag{3.100}$$

That is, there exists a point x such that the equality constraint is satisfied and the inequality constraint (3.93b) is strictly satisfied. Such a point is called a strictly feasible point. If Slater's condition holds for a convex optimization problem, then strong duality (3.99) holds. Another sufficient condition occurs when the inequality constraints (3.93c) happen to be affine, such that

$$\min_x \quad f_0(x) \tag{3.101a}$$

$$\text{s.t.} \quad Bx \preceq 0 \tag{3.101b}$$

$$Ax = b \tag{3.101c}$$

Then, strong duality also holds [20, p.226], [10, p.514].

Once we get the optimal solutions (ν^o, λ^o) for the dual problem (3.97a)–(3.97b), then the optimal solution to the primal problem (3.93a)–(3.93c) can be obtained as

$$x^o = \arg\min_x L(x, \nu^o, \lambda^o) \tag{3.102}$$

if the minimizer of $L(x, \nu^o, \lambda^o)$ is unique. However, if there are multiple minimizers for $L(x, \nu^o, \lambda^o)$, then some of the minimizers might be primal-infeasible [10, p.603]. That is, some of the minimizers of $L(x, \nu^o, \lambda^o)$ may not satisfy the feasibility conditions (3.93b)–(3.93c). In this case, we need to select the x^o to be the ones that satisfy the constraint (3.93b)–(3.93c) from all the minimizers of $L(x, \nu^o, \lambda^o)$. In this chapter, we only use (3.102) because the cost functions are selected to ensure a unique minimizer of $L(x, \nu^o, \lambda^o)$ — see Sec. 3.3.3.

3.C Overview of Proximal Gradient Algorithms

In this appendix, we give a short overview of proximal gradient algorithms and the proximal operator. For a useful survey, the readers are referred to [103]. The proximal gradient algorithm is applicable to optimization problems that assume the following form:

$$\min_{x} \quad f(x) + h(x) \tag{3.103}$$

where $f(x)$ is a continuously differentiable convex function defined over $\mathbb{R}^M$, and $h(x)$ is a *non-differentiable* convex function over $\mathbb{R}^M$. That is, the cost function consists of the sum of a differentiable part and a non-differentiable part. One approach to minimizing non-differentiable cost functions is to employ the sub-gradient method [10, 105]. However, sub-gradient methods tend to converge slowly at the rate of $O(1/\sqrt{i})$, where i is the number of iterations. Nevertheless, if $h(x)$ in (3.103) assumes certain forms such that its proximal operator (defined by (3.104) below) can be evaluated easily, then the proximal gradient algorithm provides a more efficient technique to minimize (3.103) by exploiting the differentiable nature of $f(x)$ and the structure of $h(x)$.

Table 3.5: Examples of proximal operators

$h(x)$	$\text{prox}_{\beta h}(x), \quad \beta > 0$
0	x
$a^T x + b$	$x - \beta a$
$\frac{1}{2} x^T A x + b^T x + c,\ A$ is positive definite	$(I + \beta A)^{-1}(x - \beta b)$
$\frac{1}{2}\|x\|_2^2$	$\frac{1}{1+\beta} \cdot x$
$-\ln(x)$	$\frac{1}{2}\left(x + \sqrt{x^2 + 4\beta}\right)$
$\|x\|_1$	$\mathcal{T}_\beta(x)$
$\|x\|_1 + \frac{\gamma}{2}\|x\|_2^2$	$\frac{1}{1+\beta\gamma} \cdot \mathcal{T}_\beta(x)$

The proximal operator of a function $h(x)$ is defined as

$$\text{prox}_h(x) = \arg\min_u \left(h(u) + \frac{1}{2}\|u - x\|_2^2 \right) \tag{3.104}$$

Note that the above definition of the proximal operator is for both differentiable and non-differentiable functions. Closed-form expressions for some useful choices for $h(x)$ can be found in Table 3.5 and [103, pp.172–195]. For example, if $h(x) = \|x\|_1$, then the proximal operator for $\beta \cdot h(x)$ is the (entry-wise) soft-thresholding operator:

$$[\text{prox}_{\beta \cdot \|x\|_1}(x)]_m = [\mathcal{T}_\beta(x)]_m = \begin{cases} x_m - \beta & x_m \geq \beta \\ 0 & |x_m| \leq \beta \\ x_m + \beta & x_m \leq -\beta \end{cases} \tag{3.105}$$

where $[\cdot]_m$ denotes the mth entry of the vector argument, and x_m denotes the

mth entry of the vector x.

The proximal gradient method for solving the problem (3.103) is given by the following iteration:

$$x_i = \text{prox}_{\mu \cdot h}\big(x_{i-1} - \mu \nabla_x f(x_{i-1})\big) \tag{3.106}$$

where μ is a positive step-size parameter. That is, the algorithm applies gradient descent step to the differentiable part, $f(x)$, and then applies the proximal operator to the non-differentiable part. By doing so, it is shown in [8] that the convergence rate becomes $O(1/i)$ for any convex function $f(x)$ with Lipschitz gradients, which is faster than the $O(1/\sqrt{i})$ rate for sub-gradient methods.

A useful interpretation of the proximal gradient method (3.106) is the *majorization minimization* strategy [69, 103], which minimizes a function $g(x)$ by iteratively minimizing a particular upper bound defined using the previous iterate x_{i-1}. Assume $f(x)$ in (3.103) is continuously differentiable with Lipschitz gradients, i.e.,

$$\|\nabla f(x) - \nabla f(y)\| \leq L \cdot \|x - y\|, \quad \forall x, y \tag{3.107}$$

where L is the Lipschitz constant. Then, $f(x)$ can be upper bounded as

$$\begin{aligned} f(x) &= f(x_{i-1}) + \left[\int_0^1 \nabla f(x_{i-1} + t(x - x_{i-1}))dt\right]^T (x - x_{i-1}) \\ &= f(x_{i-1}) \\ &\quad + \left[\int_0^1 \big[\nabla f(x_{i-1}) + \nabla f(x_{i-1} + t(x - x_{i-1})) - \nabla f(x_{i-1})\big]dt\right]^T (x - x_{i-1}) \\ &= f(x_{i-1}) + [\nabla f(x_{i-1})]^T (x - x_{i-1}) \end{aligned}$$

$$
\begin{aligned}
&+\left(\int_0^1 \left[\nabla f(x_{i-1}+t(x-x_{i-1}))-\nabla f(x_{i-1})\right] dt\right)^T (x-x_{i-1}) \\
&\overset{(a)}{\leq} f(x_{i-1})+[\nabla f(x_{i-1})]^T(x-x_{i-1}) \\
&\quad+\int_0^1 \|\nabla f(x_{i-1}+t(x-x_{i-1}))-\nabla f(x_{i-1})\|\, dt\cdot\|x-x_{i-1}\| \\
&\overset{(b)}{\leq} f(x_{i-1})+[\nabla f(x_{i-1})]^T(x-x_{i-1})+\int_0^1 tdt\cdot L\cdot\|x-x_{i-1}\|^2 \\
&= f(x_{i-1})+[\nabla f(x_{i-1})]^T(x-x_{i-1})+\frac{L}{2}\cdot\|x-x_{i-1}\|^2 \\
&\overset{(c)}{\leq} \underbrace{f(x_{i-1})+[\nabla f(x_{i-1})]^T(x-x_{i-1})+\frac{1}{2\mu}\cdot\|x-x_{i-1}\|^2}_{\triangleq f_\mu(x,x_{i-1})}
\end{aligned}
\tag{3.108}
$$

where step (a) uses Cauchy-Schwartz inequality ($x^Ty \leq |x^Ty| \leq \|x\|\cdot\|y\|$), step (b) uses the Lipshitz condition (3.107) on the gradient, and step (c) holds for any μ that satisfies $0<\mu\leq 1/L$. That is, for any $\mu\in(0,1/L]$, we have the following *majorization* relation for $f(x)$:

$$
f(x)\leq f_\mu(x,x_{i-1}) \tag{3.109}
$$

which implies the following *majorzation* relation for $f(x)+h(x)$:

$$
\begin{aligned}
f(x)+h(x) &\leq f_\mu(x,x_{i-1})+h(x) \\
&= f(x_{i-1})+[\nabla f(x_{i-1})]^T(x-x_{i-1})+\frac{1}{2\mu}\cdot\|x-x_{i-1}\|^2+h(x)
\end{aligned}
\tag{3.110}
$$

The *majorization minimization* strategy minimizes $f(x)+h(x)$ by minimizing its upper bound $f_\mu(x,x_{i-1})+h(x)$ at each iteration i:

$$
x_i=\arg\min_x\left(f_\mu(x,x_{i-1})+h(x)\right)
$$

$$
\begin{aligned}
&= \arg\min_x \left(f(x_{i-1}) + [\nabla f(x_{i-1})]^T (x - x_{i-1}) + \frac{1}{2\mu} \cdot \|x - x_{i-1}\|^2 + h(x) \right) \\
&\overset{(a)}{=} \arg\min_x \left([\nabla f(x_{i-1})]^T x + \frac{1}{2\mu} \cdot \|x - x_{i-1}\|^2 + h(x) \right) \\
&= \arg\min_x \left(\mu \cdot h(x) + \mu [\nabla f(x_{i-1})]^T x + \frac{1}{2} \|x - x_{i-1}\|^2 \right) \\
&\overset{(b)}{=} \arg\min_x \Big(\mu \cdot h(x) + \frac{1}{2} \left\| x - [x_{i-1} - \mu \nabla f(x_{i-1})] \right\|^2 \\
&\qquad\qquad + \frac{1}{2} \|x_{i-1}\|^2 - \frac{1}{2} \|x_{i-1} - \mu \nabla f(x_{i-1})\|^2 \Big) \\
&\overset{(c)}{=} \arg\min_x \Big(\mu \cdot h(x) + \frac{1}{2} \left\| x - [x_{i-1} - \mu \nabla f(x_{i-1})] \right\|^2 \Big) \\
&= \text{prox}_{\mu h}\big(x_{i-1} - \mu \nabla f(x_{i-1}) \big)
\end{aligned} \tag{3.111}
$$

where steps (a) and (c) drop the terms that are independent of x, and step (b) completes the square to absorb the linear term of x into the quadratic term. From the above derivation (3.111), the majorization process is only performed on the differentiable part, $f(x)$, in (3.103). The non-differentiable part, $h(x)$, remains unchanged. The majorization is approximating $f(x)$ by a quadratic function using its local gradient at the previous iterate x_{i-1} and a sufficiently large Hessian I/μ, which provides local information about $f(x)$ around x_{i-1}. On the other hand, by keeping $h(x)$ unchanged during the majorization process, we retain its global information. Therefore, the proximal gradient algorithm (3.106) is exploiting local information for $f(x)$ and global information for $h(x)$ during the iteration, which is expected to achieve a faster convergence than the sub-gradient method, which uses only the local information for both $f(x)$ and $h(x)$.

CHAPTER 4

Mean-Square Analysis

From Chapters 2–3, we know that both the "sum-of-costs" and "cost-of-sum" problems can be effectively solved by diffusion strategies. Starting from this chapter, we proceed to analyze the performance of the diffusion strategies. In this chapter, we first analyze the stability and performance of the algorithm under the assumption that *each cost function $J_k(w)$ is strongly convex*. In later chapters, we will relax this assumption to require only *the aggregate cost $J^{\text{glob}}(w) = \sum_{k=1}^{N} J_k(w)$ to be strongly convex*. The analysis in the latter case is more involved. Nevertheless, studying the performance of diffusion strategies under the stronger assumptions ("where each cost function is strongly convex") is still important since this assumption typically holds in practical applications. This is because quadratic regularization can be added to convert each $J_k(w)$ into a strongly convex function. The following presentation in this chapter is based on [36].

4.1 General Diffusion Adaptation Strategies

In Chapter 2, we motivated and derived diffusion strategies for distributed optimization of the following aggregate cost

$$J^{\text{glob}}(w) = \sum_{k=1}^{N} J_k(w) \tag{4.1}$$

where we assumed that the cost functions $\{J_k(w)\}$ share a common minimizer. In this chapter, we are going to show that this assumption is not necessary and diffusion strategies still work even when the minimizers of $\{J_k(w)\}$ are not necessarily equal to each other. Recall that the diffusion strategies (ATC and CTA strategies) are captured by the following general description:

$$\phi_{k,i-1} = \sum_{l=1}^{N} a_{1,lk} w_{l,i-1} \tag{4.2}$$

$$\psi_{k,i} = \phi_{k,i-1} - \mu_k \sum_{l=1}^{N} c_{lk} \nabla_w J_l(\phi_{k,i-1}) \tag{4.3}$$

$$w_{k,i} = \sum_{l=1}^{N} a_{2,lk} \psi_{l,i} \tag{4.4}$$

where $w_{k,i}$ is the local estimate for w^o at node k and time i, μ_k is the step-size parameter used by node k, and $\{\phi_{k,i-1}, \psi_{k,i}\}$ are intermediate estimates for w^o. Moreover, $\nabla_w J_l(\cdot)$ is the (column) gradient vector of $J_l(\cdot)$ relative to w. The non-negative coefficients $\{a_{1,lk}\}$, $\{c_{lk}\}$, and $\{a_{2,lk}\}$ are the (l,k)-th entries of matrices A_1, C, and A_2, respectively, and they are required to satisfy:

$$\begin{cases} A_1^T \mathbb{1} = \mathbb{1},\ A_2^T \mathbb{1} = \mathbb{1},\ C\mathbb{1} = \mathbb{1}, \\ a_{1,lk} = 0,\ a_{2,lk} = 0,\ c_{lk} = 0 \text{ if } l \notin \mathcal{N}_k \end{cases} \tag{4.5}$$

where $\mathbb{1}$ denotes a vector with all entries equal to one, $\mathcal{N}_k$ denotes the neighborhood of node k (including node k itself); the neighbors of node k consist of all nodes with which node k can share information. Note from (4.5) that the combination coefficients $\{a_{1,lk}, a_{2,lk}, c_{lk}\}$ are nonzero only for those $l \in \mathcal{N}_k$. Therefore, the sums in (4.2)–(4.4) are confined within the neighborhood of node k. Condition (4.5) requires the combination matrices $\{A_1, A_2\}$ to be left-stochastic, while C is right-stochastic. We therefore note that each node k first aggregates the ex-

isting estimates from its neighbors through (4.2) and generates the intermediate estimate $\phi_{k,i-1}$. Then, node k aggregates gradient information from its neighborhood and updates $\phi_{k,i-1}$ to $\psi_{k,i}$ through (4.3). All other nodes in the network are performing these same steps simultaneously. Finally, node k aggregates the estimates $\{\phi_{l,i}\}$ through step (4.4) to update its weight estimate to $w_{k,i}$.

Algorithm (4.2)–(4.4) can be simplified to several special cases for different choices of the matrices $\{A_1, A_2, C\}$. For example, the choice $A_1 = I$, $A_2 = A$ and $C = I$ reduces to the adapt-then-combine (ATC) strategy that has no exchange of gradient information [26, 34, 89]:

$$\boxed{\begin{aligned}\psi_{k,i} &= w_{k,i-1} - \mu_k \nabla_w J_k(w_{k,i-1}) \\ w_{k,i} &= \sum_{l \in \mathcal{N}_k} a_{lk} \psi_{l,i}\end{aligned}} \quad (\text{ATC},\ C = I) \tag{4.6}$$

while the choice $A_1 = A$, $A_2 = I$ and $C = I$ reduces to the combine-then-adapt (CTA) strategy, where the order of the combination and adaptation steps are reversed relative to (4.6) [26, 89]:

$$\boxed{\begin{aligned}\psi_{k,i-1} &= \sum_{l \in \mathcal{N}_k} a_{lk} w_{l,i-1} \\ w_{k,i} &= \psi_{k,i-1} - \mu_k \nabla_w J_k(\psi_{k,i-1})\end{aligned}} \quad (\text{CTA},\ C = I) \tag{4.7}$$

Furthermore, if in the CTA implementation (4.7) we enforce A to be doubly stochastic, replace $\nabla_w J_k(\cdot)$ by a subgradient, and use a time-decaying step-size parameter ($\mu_k(i) \to 0$), then we obtain the unconstrained version used by [109]. In the sequel, we continue with the general recursions (4.2)–(4.4), which allow us to examine the convergence properties of several algorithms in a unified manner. The challenge we encounter now is to show that this class of algorithms can

optimize the cost (2.1) in a distributed manner when the individual costs $\{J_l(w)\}$ do not necessarily have the same minimizer. This is a demanding task, as the analysis in the coming sections reveals.

4.2 Modeling Assumptions

In situations of adaptation and learning, the true gradient vectors needed in (4.3) are not available. Instead, these gradients are replaced by approximate values, which we model as:

$$\widehat{\nabla_w J_l}(\boldsymbol{w}) = \nabla_w J_l(\boldsymbol{w}) + \boldsymbol{v}_{l,i}(\boldsymbol{w}) \tag{4.8}$$

where the random noise term, $\boldsymbol{v}_{l,i}(\boldsymbol{w})$, may depend on $\boldsymbol{w}$ and will be required to satisfy certain conditions given by (4.13)–(4.14). We refer to the perturbation in (4.8) as gradient noise. Using (4.8), the diffusion algorithm (4.2)–(4.4) becomes the following, where we are using boldface letters for various quantities to highlight the fact that they are now stochastic in nature due to the randomness in the gradient noise component:

$$\boldsymbol{\phi}_{k,i-1} = \sum_{l=1}^{N} a_{1,lk}\boldsymbol{w}_{l,i-1} \tag{4.9}$$

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{\phi}_{k,i-1} - \mu_k \sum_{l=1}^{N} c_{lk}\big[\nabla_w J_l(\boldsymbol{\phi}_{k,i-1}) + \boldsymbol{v}_{l,i}(\boldsymbol{\phi}_{k,i-1})\big] \tag{4.10}$$

$$\boldsymbol{w}_{k,i} = \sum_{l=1}^{N} a_{2,lk}\boldsymbol{\psi}_{l,i} \tag{4.11}$$

Using (4.9)–(4.11), we now proceed to examine the mean-square performance of the diffusion strategies. Specifically, in the sequel, we study: (i) how fast and (ii) how close the estimator $\boldsymbol{w}_{k,i}$ at each node k approaches the minimizer w^o of the

aggregate cost (4.1) in the mean-square-error sense. We establish the convergence of all nodes towards the same w^o within a small MSE bound. The approach we employ to examine the convergence properties of the diffusion strategy is a system-theoretic approach that examines the flow of energy through the network, and calls upon the fixed-point theorem for contractive mappings [80, pp.299–303].

To proceed with the analysis, we introduce the following assumptions on the cost functions and gradient noise.

Assumption 4.1 (Bounded Hessian)**.** *Each component cost function $J_l(w)$ has a Hessian matrix that is bounded from below and from above, i.e., there exist $\lambda_{l,\min} \geq 0$ and $\lambda_{l,\max} > 0$ such that, for each $k = 1, \ldots, N$:*

$$\lambda_{l,\min} I_M \leq \nabla_w^2 J_l(w) \leq \lambda_{l,\max} I_M \tag{4.12}$$

with $\sum_{l=1}^{N} c_{lk}\lambda_{l,\min} > 0$. Inequality (4.12) means that the eigenvalues of $\nabla_w^2 J_l(w)$ are upper and lower bounded by $\lambda_{l,\max}$ and $\lambda_{l,\min}$, respectively. □

Assumption 4.2 (Gradient noise)**.** *There exist $\alpha \geq 0$ and $\sigma_v^2 \geq 0$ such that, for all $\boldsymbol{w} \in \mathcal{F}_{i-1}$:*

$$\mathbb{E}\left\{\boldsymbol{v}_{l,i}(\boldsymbol{w}) \mid \mathcal{F}_{i-1}\right\} = 0 \tag{4.13}$$

$$\mathbb{E}\left\{\|\boldsymbol{v}_{l,i}(\boldsymbol{w})\|^2\right\} \leq \alpha \cdot \mathbb{E}\|\nabla_w J_l(\boldsymbol{w})\|^2 + \sigma_v^2 \tag{4.14}$$

for all i, l, where $\mathcal{F}_{i-1}$ denotes the past history of estimators $\{\boldsymbol{w}_{k,j}\}$ for $j \leq i-1$ and all k. □

Remark 4.1. Assumption 4.1 ensures the strong convexity of the aggregate cost $J^{\text{glob}}(w)$ defined by (4.1); condition (4.12) would require the individual costs to be strongly convex when $C = I$. This condition is applicable to many situations

of interest; one of its main benefits is that it ensures that the Hessian matrix is not close-to-singular or ill-conditioned. Strong convexity is prevalent in many other studies on optimization techniques as well. For example, the individual costs $J_l(w)$ are assumed to be stronlgy convex in [125, 132] in order to derive upper bounds on the limit superior ("lim sup") of the mean-square-error of their estimates $\boldsymbol{w}_{k,i}$ or the expected value of their cost function at $\boldsymbol{w}_{k,i}$ when constant step-sizes are used, i.e., to derive results of the form

$$\limsup_{i\to\infty} \mathbb{E}\|\boldsymbol{w}_{k,i} - w^o\|^2 \leq \eta \tag{4.15}$$

$$\limsup_{i\to\infty} \mathbb{E}J^{\text{glob}}(\boldsymbol{w}_{k,i}) \leq J^{\text{glob}}(w^o) + \eta \tag{4.16}$$

where η is the upper bound. For example, in Theorem 5 of [125], each individual cost function is assumed to be continuously differentiable and strongly convex. Likewise, in Proposition 2 of [132], each individual cost function is also assumed to be strongly convex. When the strong convexity assumption is relaxed, no upper bounds on the limit superiors similar to (4.15) or (4.16) are established in [97, 109, 132]. Instead, only upper bounds on the limit inferior ("lim inf") of $\mathbb{E}J^{\text{glob}}(\boldsymbol{w}_{k,i})$ are derived in the presence of noise [109, 132], or in the absence of noise [71, 97], such as the bound

$$\liminf_{i\to\infty} \mathbb{E}J^{\text{glob}}(\boldsymbol{w}_{k,i}) \leq J^{\text{glob}}(w^o) + \eta \tag{4.17}$$

By the definitions of lim sup and lim inf [4, p.353–355], inequality (4.17) means that $\mathbb{E}J^{\text{glob}}(\boldsymbol{w}_{k,i})$ can only be smaller than the upper bound on the right-hand side infinitely often as $i \to \infty$. However, it can also be arbitrarily far away from the upper bound infinitely often. On the contrary, the bound on lim sup means that, as $i \to \infty$, eventually the mean-square-error or the expected cost function

value at $\boldsymbol{w}_{k,i}$ would be uniformly smaller than the upper bound. For this reason, it is more critical to establish an upper bound on the limit superior rather than the limit inferior. It is for this purpose that Polyak-Ruppert averaging [106, 111] is applied in [97, 109, 132] to obtain a new time-averaged estimate at each node k as:

$$\boldsymbol{z}_{k,i} = \frac{1}{i}\sum_{t=1}^{i} \boldsymbol{w}_{k,t} \tag{4.18}$$

which is then shown to satisfy the lim sup bound:

$$\limsup_{i\to\infty} \mathbb{E}J^{\text{glob}}(\boldsymbol{z}_{k,i}) \leq J^{\text{glob}}(w^o) + \eta \tag{4.19}$$

This is a useful technique in enhancing the convergence behavior when the environment is stationary. However, over non-stationary environments, the technique is problematic since it reduces the adaptation and tracking ability of the algorithm because averaging of the estimates is performed over the entire history up to the current time i. Therefore, in terms of better adaptation ability, it is more favorable to seek estimates $\{\boldsymbol{w}_{k,i}\}$ that satisfy a "lim sup" bound directly on their mean-square-error. This objective can be achieved by adding a small regularization term. For example, we can convert a non-strongly convex function $J_l(w)$ to a strongly convex one by redefining $J_l(w)$ as $J_l(w) \leftarrow J_l(w) + \epsilon\|w\|^2$, where $\epsilon > 0$ is a small regularization factor. □

Remark 4.2. In Chapters 5–6, we relax Assumption 4.1 and only require the aggregate cost $J^{\text{glob}}(w)$ to be strongly convex. We show that the diffusion strategies (4.2)–(4.7) still achieve the convergence rate and steady-state performance of a centralized strategy. □

Remark 4.3. We further note that assumption (4.14) is a mix of the "relative

random noise" and "absolute random noise" model usually assumed in stochastic approximation [105]. Condition (4.14) implies that the gradient noise grows when the estimate is away from the optimum (large gradient). Condition (4.14) also states that even when the gradient vector is zero, there is still some residual noise variance σ_v^2. On the other hand, in [109, 125, 132], the variance of the gradient noise was instead assumed to be uniformly upper bounded, i.e.,

$$\mathbb{E}\{\|\boldsymbol{v}_{l,i}\|^2\} \leq \sigma_v^2 \qquad \text{or} \qquad \mathbb{E}\{\|\boldsymbol{v}_{l,i}\|^2|\mathcal{F}_{i-1}\} \leq \sigma_v^2 \tag{4.20}$$

with only an absolute noise term appearing in (4.20). Such an assumption is useful in constrained optimization over a compact set. However, for the unconstrained optimization problems that we consider here, we need a model that incorporates both "relative random noise" and "absolute random noise" [34, 105]. Without the relative noise term factor, the analyses and the gradient model would not be able to handle situations involving adaptation and learning (see the following Example 4.1). □

Example 4.1. Such a mix of "relative random noise" and "absolute random noise" is of practical importance. For instance, consider an example in which the loss function at node l is chosen to be of the following quadratic form:

$$Q_l(w, \{\boldsymbol{u}_{l,i}, \boldsymbol{d}_l(i)\}) = |\boldsymbol{d}_l(i) - \boldsymbol{u}_{l,i}w|^2$$

for some scalars $\{\boldsymbol{d}_l(i)\}$ and $1 \times M$ regression vectors $\{\boldsymbol{u}_{l,i}\}$. The corresponding cost function is then:

$$J_l(w) = \mathbb{E}|\boldsymbol{d}_l(i) - \boldsymbol{u}_{l,i}w|^2 \tag{4.21}$$

Assume further that the data $\{\boldsymbol{u}_{l,i}, \boldsymbol{d}_l(i)\}$ satisfy the linear regression model

$$\boldsymbol{d}_l(i) = \boldsymbol{u}_{l,i} w^o + \boldsymbol{z}_l(i) \tag{4.22}$$

where the regressors $\{\boldsymbol{u}_{l,i}\}$ are zero mean and independent over time with covariance matrix $R_{u,l} = \mathbb{E}\{\boldsymbol{u}_{l,i}^T \boldsymbol{u}_{l,i}\}$, and the noise sequence $\{\boldsymbol{z}_k(j)\}$ is also zero mean, white, with variance $\sigma_{z,k}^2$, and independent of the regressors $\{\boldsymbol{u}_{l,i}\}$ for all l, k, i, j. Then, as we pointed out in Example 2.1 in Chapter 2 that the gradient noise in this case can be expressed as:

$$\boldsymbol{v}_{l,i}(\boldsymbol{w}) = 2(R_{u,l} - \boldsymbol{u}_{l,i}^T \boldsymbol{u}_{l,i})(w^o - \boldsymbol{w}) - 2\boldsymbol{u}_{l,i}^T \boldsymbol{z}_l(i) \tag{4.23}$$

It can easily be verified that this noise satisfies both conditions stated in Assumption 4.2, namely, (4.13) and also:

$$\mathbb{E}\left\{\|\boldsymbol{v}_{l,i}(\boldsymbol{w})\|^2\right\} \leq 4\mathbb{E}\|R_{u,l} - \boldsymbol{u}_{l,i}^T \boldsymbol{u}_{l,i}\|^2 \cdot \mathbb{E}\|w^o - \boldsymbol{w}\|^2 + 4\sigma_{z,l}^2 \mathrm{Tr}(R_{u,l}) \tag{4.24}$$

for all $\boldsymbol{w} \in \mathcal{F}_{i-1}$. Note that both relative random noise and absolute random noise components appear in (4.24) and are necessary to model the statistical gradient perturbation even for quadratic costs. Such costs, and linear regression models of the form (4.22), arise frequently in the context of adaptive filters — see, e.g., [5, 23–26, 28, 64, 87–90, 116, 118, 124, 130]. □

4.3 Diffusion Adaptation Operators

To analyze the performance of the diffusion adaptation strategies, we first represent the mappings performed by (4.9)–(4.11) in terms of useful operators.

Definition 4.1 (Combination Operator)**.** *Suppose* $x = \mathrm{col}\{x_1, \ldots, x_N\}$ *is an*

arbitrary $N \times 1$ block column vector that is formed by stacking $M \times 1$ vectors $x_1, \ldots, x_N$ on top of each other. The combination operator $T_A : \mathbb{R}^{MN} \to \mathbb{R}^{MN}$ is defined as the linear mapping:

$$T_A(x) \triangleq (A^T \otimes I_M)\, x \tag{4.25}$$

where A is an $N \times N$ left-stochastic matrix, and $\otimes$ denotes the Kronecker product operation. □

Definition 4.2 (Gradient-Descent Operator)**.** *Consider the same $N \times 1$ block column vector x. Then, the gradient-descent operator $T_G : \mathbb{R}^{MN} \to \mathbb{R}^{MN}$ is the nonlinear mapping defined by:*

$$T_G(x) \triangleq \begin{bmatrix} x_1 - \mu_1 \sum_{l=1}^{N} c_{l1} \nabla_w J_l(x_1) \\ \vdots \\ x_N - \mu_N \sum_{l=1}^{N} c_{lN} \nabla_w J_l(x_N) \end{bmatrix} \tag{4.26}$$

□

Definition 4.3 (Power Operator)**.** *Consider the same $N \times 1$ block vector x. The power operator $P : \mathbb{R}^{MN} \to \mathbb{R}^N$ is defined as the mapping:*

$$P[x] \triangleq \mathrm{col}\{\|x_1\|^2, \ldots, \|x_N\|^2\} \tag{4.27}$$

where $\|\cdot\|$ denotes the Euclidean norm of a vector. □

We will use the power operator to study how error variances propagate after a specific operator $T_A(\cdot)$ or $T_G(\cdot)$ is applied to a random vector. We remark that we are using the notation "$P[\cdot]$" rather than "$P(\cdot)$" to highlight the fact that P is a mapping from $\mathbb{R}^{MN}$ to a lower dimensional space $\mathbb{R}^N$. In addition to the

above three operators, we define the following aggregate vector of gradient noise that depends on the state x:

$$\boldsymbol{v}(x) \triangleq -\mathrm{col}\Big\{\mu_1 \sum_{l=1}^{N} c_{l1}\boldsymbol{v}_l(x_1), \ldots, \mu_N \sum_{l=1}^{N} c_{lN}\boldsymbol{v}_l(x_N)\Big\} \tag{4.28}$$

where we are dropping the subscript i for simplicity. With these definitions, we can now represent the two combination steps (4.9) and (4.11) as two combination operators $T_{A_1}(\cdot)$ and $T_{A_2}(\cdot)$. We can also represent the adaptation step (4.10) by a gradient-descent operator perturbed by the noise operator (4.28):

$$\widehat{\boldsymbol{T}}_G(x) \triangleq T_G(x) + \boldsymbol{v}(x) \tag{4.29}$$

We can view $\widehat{\boldsymbol{T}}_G(x)$ as a random operator that maps each input $x \in \mathbb{R}^{MN}$ into an $\mathbb{R}^{MN}$ random vector, and we use boldface $\boldsymbol{T}$ to highlight this random nature. Let

$$\boldsymbol{w}_i \triangleq \mathrm{col}\{\boldsymbol{w}_{1,i}, \boldsymbol{w}_{2,i}, \ldots, \boldsymbol{w}_{N,i}\} \tag{4.30}$$

denote the vector that collects the estimators across all nodes. Then, the overall diffusion adaptation steps (4.9)–(4.11) that update $\boldsymbol{w}_{i-1}$ to $\boldsymbol{w}_i$ can be represented as a cascade composition of three operators:

$$\widehat{\boldsymbol{T}}_d(\cdot) \triangleq T_{A_2} \circ \widehat{\boldsymbol{T}}_G \circ T_{A_1}(\cdot) \tag{4.31}$$

where we use $\circ$ to denote the composition of any two operators, i.e., $T_1 \circ T_2(x) \triangleq T_1(T_2(x))$. If there is no gradient noise, then the diffusion adaptation operator

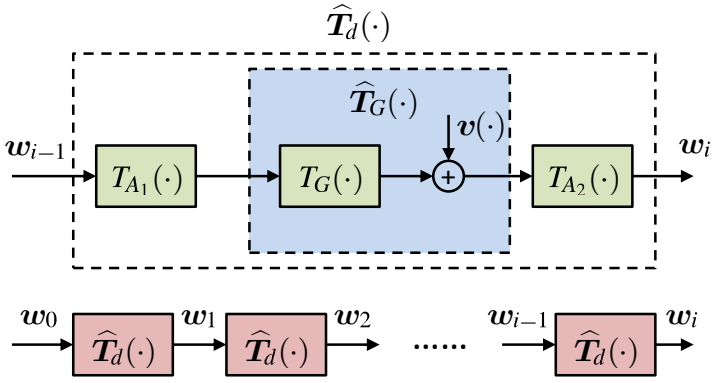


Figure 4.1: Representation of the diffusion adaptation strategy (4.9)–(4.11) in terms of operators. Each diffusion adaptation step can be viewed as a cascade composition of three operators: $T_{A_1}(\cdot)$, $T_G(\cdot)$, and $T_{A_2}(\cdot)$ with gradient perturbation $\boldsymbol{v}(\cdot)$. If $\boldsymbol{v}(\cdot) = 0$, then $\widehat{\boldsymbol{T}}_d(\cdot)$ becomes $T_d(\cdot)$.

(4.31) reduces to

$$T_d(\cdot) \triangleq T_{A_2} \circ T_G \circ T_{A_1}(\cdot) \tag{4.32}$$

In other words, the diffusion adaptation over the entire network with and without gradient noise can be described in the following compact forms:

$$\boldsymbol{w}_i = \widehat{\boldsymbol{T}}_d(\boldsymbol{w}_{i-1}) \tag{4.33}$$

$$w_i = T_d(w_{i-1}) \tag{4.34}$$

The combination operator $T_A(\cdot)$ aggregates the estimates from the neighborhood (social learning), while the gradient-descent operator $T_G(\cdot)$ incorporates information from the local gradient vector (self-learning). In Fig. 4.1, we show that each diffusion adaptation step can be represented as the cascade composition of three operators, with perturbation from the gradient noise operator. Next, in Lemma 4.1, we examine some of the properties of the operators $\{T_{A_1}, T_{A_2}, T_G\}$, which are

proved in Appendix 4.A.

Lemma 4.1 (Useful Properties)**.** *Consider $N\times 1$ block vectors $x = \mathrm{col}\{x_1,\ldots,x_N\}$ and $y = \mathrm{col}\{y_1,\ldots,y_N\}$ with $M\times 1$ entries $\{x_k, y_k\}$. Then, the operators $T_A(\cdot)$, $T_G(\cdot)$ and $P[\cdot]$ satisfy the following properties:*

1. *(Linearity): $T_A(\cdot)$ is a linear operator.*
2. *(Nonnegativity): $P[x] \succeq 0$.*
3. *(Scaling): For any scalar $a\in\mathbb{R}$, we have $P[ax] = a^2P[x]$.*
4. *(Convexity): suppose $x^{(1)},\ldots,x^{(K)}$ are $N\times 1$ block vectors formed in the same manner as x, and let $a_1,\ldots,a_K$ be non-negative real scalars that add up to one. Then,*

$$P[a_1x^{(1)}+\cdots+a_Kx^{(K)}] \preceq a_1P[x^{(1)}]+\cdots+a_KP[x^{(K)}] \tag{4.35}$$

5. *(Additivity): Suppose $\boldsymbol{x} = \mathrm{col}\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N\}$ and $\boldsymbol{y} = \mathrm{col}\{\boldsymbol{y}_1,\ldots,\boldsymbol{y}_N\}$ are $N\times 1$ block random vectors that satisfy $\mathbb{E}\boldsymbol{x}_k^T\boldsymbol{y}_k = 0$ for $k = 1,\ldots,N$. Then, $\mathbb{E}P[\boldsymbol{x}+\boldsymbol{y}] = \mathbb{E}P[\boldsymbol{x}] + \mathbb{E}P[\boldsymbol{y}]$.*
6. *(Variance relations):*

$$P[T_A(x)] \preceq A^TP[x] \tag{4.36}$$

$$P[T_G(x) - T_G(y)] \preceq \Gamma^2P[x-y] \tag{4.37}$$

where

$$\Gamma \triangleq \mathrm{diag}\{\gamma_1,\ldots,\gamma_N\} \tag{4.38}$$

$$\gamma_k \triangleq \max\{|1-\mu_k\sigma_{k,\max}|,\ |1-\mu_k\sigma_{k,\min}|\} \tag{4.39}$$

$$\sigma_{k,\min} \triangleq \sum_{l=1}^{N} c_{lk}\lambda_{l,\min}, \quad \sigma_{k,\max} \triangleq \sum_{l=1}^{N} c_{lk}\lambda_{l,\max} \tag{4.40}$$

7. *(Block Maximum Norm): The $\infty-$norm of $P[x]$ is the squared block maximum norm of x [115]:*

$$\|P[x]\|_{\infty} = \|x\|_{b,\infty}^2 \triangleq \big(\max_{1\le k\le N} \|x_k\|\big)^2 \tag{4.41}$$

8. *(Preservation of Inequality): Suppose vectors x, y and matrix F have non-negative entries, then $x \preceq y$ implies $Fx \preceq Fy$.* □

4.4 Transient Analysis

Using the operator representation developed above, we now analyze the transient behavior of the diffusion algorithm (4.9)–(4.11). From Fig. 4.1 and the previous discussion, we know that the stochastic recursion $\boldsymbol{w}_i = \widehat{\boldsymbol{T}}_d(\boldsymbol{w}_{i-1})$ is a perturbed version of the noise-free recursion $w_i = T_d(w_{i-1})$. Therefore, we first study the convergence of the noise free recursion, and then analyze the effect of gradient perturbation on the stochastic recursion.

The operator T_d is a continuous operator, which is guaranteed by the twice-differentiability of the cost functions. Therefore, if w_i converges to a vector w_∞, then this vector should be a fixed point of T_d [80, p.299]:

$$w_\infty = T_d(w_\infty) \tag{4.42}$$

We need to answer four questions pertaining to the fixed point. First, does the fixed point exist? Second, is it unique? Third, under which condition does the

recursion $w_i = T_d(w_{i-1})$ converge to the fixed point? Fourth, how far is the fixed point w_∞ away from the minimizer w^o of (2.1)? We answer the first two questions using the Banach Fixed Point Theorem (Contraction Theorem) [80, pp.2–9, pp.299–300]. Afterwards, we study convergence under gradient perturbation. The last question will be considered in the next subsection.

Definition 4.4 (Metric Space)**.** *A set X, whose elements we shall call points, is said to be a metric space if we can associate a real number $d(p,q)$ with any two points p and q of X, such that (i). $d(p,q) > 0$ if $p \neq q$, and $d(p,q) = 0$ if and only if $p = q$; (ii). $d(p,q) = d(q,p)$; (iii). $d(p,q) \leq d(p,r) + d(r,q)$, for any $r \in X$. Any function $d(p,q)$ with these three properties is called a distance function, or a metric, and we denote a metric space X with distance $d(\cdot,\cdot)$ as (X,d).* □

Definition 4.5 (Contraction)**.** *Let (X,d) be a metric space. A mapping $T : X \longrightarrow X$ is called a contraction on X if there is a positive real number $\delta < 1$ such that $d(T(x),T(y)) \leq \delta \cdot d(x,y)$ for all $x, y \in X$*

Lemma 4.2 (Banach Fixed Point Theorem [80])**.** *Consider a metric space (X,d), where $X \neq \emptyset$. Suppose that X is complete*[1] *and let $T : X \to X$ be a contraction. Then, T has precisely one fixed point.* □

As long as we can prove that the diffusion operator $T_d(\cdot)$ is a contraction, i.e., for any two points $x, y \in \mathbb{R}^{MN}$, after we apply the operator $T_d(\cdot)$, the distance between $T_d(x)$ and $T_d(y)$ scales down by a scalar that is uniformly bounded away from one, then the fixed point w_∞ defined in (4.42) exists and is unique. We now proceed to show that $T_d(\cdot)$ is a contraction operator in $X = \mathbb{R}^{MN}$ when the step-size parameters $\{\mu_k\}$ satisfy certain conditions.

[1] A metric space (X,d) is complete if any of its Cauchy sequences converges to a point in the space; a sequence $\{x_n\}$ is Cauchy in (X,d) if $\forall \epsilon > 0$, there exists N such that $d(x_n, x_m) < \epsilon$ for all $n,\ m > N$.

Theorem 4.1 (Fixed Point)**.** *Suppose the step-size parameters* $\{\mu_k\}$ *satisfy the following conditions*

$$0 < \mu_k < \frac{2}{\sigma_{k,\max}}, \qquad k = 1, 2, \ldots, N \tag{4.43}$$

Then, there exists a unique fixed point w_∞ *for the unperturbed diffusion operator* $T_d(\cdot)$ *in* (4.32)*.*

Proof. Let $x = \mathrm{col}\{x_1, \ldots, x_N\} \in \mathbb{R}^{MN\times 1}$ be formed by stacking $M \times 1$ vectors $x_1, \ldots, x_N$ on top of each other. Similarly, let $y = \mathrm{col}\{y_1, \ldots, y_N\}$. The distance function $d(x, y)$ that we will use is induced from the block maximum norm (5.110): $d(x, y) = \|x - y\|_{b,\infty} = \max_{1\le k\le N} \|x_k - y_k\|$. From the definition of the diffusion operator $T_d(\cdot)$ in (4.32), we have

$$\begin{aligned}
P[T_d(x) - T_d(y)] &\overset{(a)}{=} P\big[T_{A_2}\big(T_G \circ T_{A_1}(x) - T_G \circ T_{A_1}(y)\big)\big] \\
&\overset{(b)}{\preceq} A_2^T P\big[T_G \circ T_{A_1}(x) - T_G \circ T_{A_1}(y)\big] \\
&\overset{(c)}{\preceq} A_2^T \Gamma^2 P[T_{A_1}(x) - T_{A_1}(y)] \\
&\overset{(d)}{=} A_2^T \Gamma^2 P[T_{A_1}(x - y)] \\
&\overset{(e)}{\preceq} A_2^T \Gamma^2 A_1^T P[x - y]
\end{aligned} \tag{4.44}$$

where steps (a) and (d) are because of the linearity of $T_{A_1}(\cdot)$ and $T_{A_2}(\cdot)$, steps (b) and (e) are because of the variance relation property (4.36), and step (c) is due to the variance relation property (4.37). Taking the $\infty-$norm of both sides of (4.44), we have

$$\begin{aligned}
\|P[T_d(x) - T_d(y)]\|_\infty &\le \|A_2^T \Gamma^2 A_1^T\|_\infty \cdot \|P[x - y]\|_\infty \\
&\le \|\Gamma\|_\infty^2 \cdot \|P[x - y]\|_\infty
\end{aligned} \tag{4.45}$$

where, in the second inequality, we used the fact that $\|A_1^T\|_\infty = \|A_2^T\|_\infty = 1$ since A_1^T and A_2^T are right-stochastic matrices. Using property (5.110), we can conclude from (4.45) that: $\|T_d(x) - T_d(y)\|_{b,\infty} \leq \|\Gamma\|_\infty \cdot \|x - y\|_{b,\infty}$. Therefore, the operator $T_d(\cdot)$ is a contraction if $\|\Gamma\|_\infty < 1$, which, by substituting (4.38)–(4.39), becomes

$$|1 - \mu_k \sigma_{k,\max}| < 1, \ |1 - \mu_k \sigma_{k,\min}| < 1, \ k = 1, \ldots, N \tag{4.46}$$

and we arrive at the condition (4.43) on the step-sizes In other words, if condition (4.43) holds for each $k = 1, \ldots, N$, then $T_d(\cdot)$ is a contraction operator. By Lemma 4.2, the operator $T_d(\cdot)$ will have a unique fixed point w_∞ that satisfies equation (4.42). □

Given the existence and uniqueness of the fixed point, the third question to answer is if recursion $w_i = T_d(w_{i-1})$ converges to this fixed point. The answer is affimative under (4.43). However, we are not going to study this question separately. Instead, we will analyze the convergence of the more demanding stochastic recursion (4.33). Therefore, we now study how fast and how close the successive estimators $\{\boldsymbol{w}_i\}$ generated by recursion (4.33) approach w_∞. Once this issue is addressed, we will then examine how close w_∞ is to the desired w^o. Introduce the following mean-square-perturbation (MSP) vector at time i:

$$\text{MSP}_i \triangleq \mathbb{E} P[\boldsymbol{w}_i - w_\infty] \tag{4.47}$$

The k-th entry of MSP_i characterizes how far away the estimate $\boldsymbol{w}_{k,i}$ at node k and time i is from $w_{k,\infty}$ in the mean-square sense. To study the closeness of $\boldsymbol{w}_i$ to w_∞, we shall study how the quantity MSP_i evolves over time. By (4.33), (4.42)

and the definitions of $\widehat{\boldsymbol{T}}_d(\cdot)$ and $T_d(\cdot)$ in (4.31) and (4.32), we obtain

$$\begin{aligned}
\mathrm{MSP}_i &= \mathbb{E}P[\boldsymbol{w}_i - w_\infty] \\
&= \mathbb{E}P\big[T_{A_2}\circ\widehat{\boldsymbol{T}}_G\circ T_{A_1}(\boldsymbol{w}_{i-1}) - T_{A_2}\circ T_G\circ T_{A_1}(w_\infty)\big] \\
&\overset{(a)}{=} \mathbb{E}P\big[T_{A_2}\big(\widehat{\boldsymbol{T}}_G\circ T_{A_1}(\boldsymbol{w}_{i-1}) - T_G\circ T_{A_1}(w_\infty)\big)\big] \\
&\overset{(b)}{\preceq} A_2^T\mathbb{E}P\big[\widehat{\boldsymbol{T}}_G\circ T_{A_1}(\boldsymbol{w}_{i-1}) - T_G\circ T_{A_1}(w_\infty)\big] \\
&\overset{(c)}{=} A_2^T\mathbb{E}P\big[T_G\big(T_{A_1}(\boldsymbol{w}_{i-1})\big) - T_G\big(T_{A_1}(w_\infty)\big) + \boldsymbol{v}\big(T_{A_1}(\boldsymbol{w}_{i-1})\big)\big] \\
&\overset{(d)}{=} A_2^T\big\{\mathbb{E}P\big[T_G\big(T_{A_1}(\boldsymbol{w}_{i-1})\big) - T_G\big(T_{A_1}(w_\infty)\big)\big] + \mathbb{E}P\big[\boldsymbol{v}\big(T_{A_1}(\boldsymbol{w}_{i-1})\big)\big]\big\} \\
&\overset{(e)}{\preceq} A_2^T\Gamma^2\mathbb{E}P[T_{A_1}(\boldsymbol{w}_{i-1}) - T_{A_1}(w_\infty)] + A_2^T\mathbb{E}P[\boldsymbol{v}(T_{A_1}(\boldsymbol{w}_{i-1}))] \\
&\overset{(f)}{\preceq} A_2^T\Gamma^2A_1^T\cdot\mathbb{E}P[\boldsymbol{w}_{i-1} - w_\infty] + A_2^T\mathbb{E}P[\boldsymbol{v}(T_{A_1}(\boldsymbol{w}_{i-1}))] \\
&= A_2^T\Gamma^2A_1^T\cdot\mathrm{MSP}_{i-1} + A_2^T\mathbb{E}P[\boldsymbol{v}(T_{A_1}(\boldsymbol{w}_{i-1}))] \qquad (4.48)
\end{aligned}$$

where step (a) is by the linearity of $T_{A_1}(\cdot)$, steps (b) and (f) are by property (4.36), step (c) is by the substitution of (4.29), step (d) is by Property 5 in Lemma 4.1 and assumption (4.13), and step (e) is by (4.37). To proceed with the analysis, we establish the following lemma to bound the second term in (4.48).

Lemma 4.3 (Bound on Gradient Perturbation)**.** *It holds that*

$$\mathbb{E}P[\boldsymbol{v}(T_{A_1}(\boldsymbol{w}_{i-1}))] \preceq 4\alpha\lambda_{\max}^2\|C\|_1^2\cdot\Omega^2A_1^T\cdot\mathbb{E}P[\boldsymbol{w}_{i-1} - w_\infty] + \|C\|_1^2\Omega^2b_v \qquad (4.49)$$

where $\|\cdot\|_1$ *denotes the maximum absolute column sum and*

$$\begin{aligned}
\lambda_{\max} &\triangleq \max_{1\le k\le N}\lambda_{k,\max} \qquad (4.50) \\
b_v &\triangleq 4\alpha\lambda_{\max}^2A_1^TP[w_\infty - \mathbb{1}_N\otimes w^o] \\
&\quad + \max_{1\le k\le N}\{2\alpha\|\nabla_w J_k(w^o)\|^2 + \sigma_v^2\}\cdot\mathbb{1}_N \qquad (4.51)
\end{aligned}$$

$$\Omega \triangleq \operatorname{diag}\{\mu_1, \ldots, \mu_N\} \tag{4.52}$$

Proof. By the definition of $\boldsymbol{v}(\boldsymbol{x})$ in (4.28) with $\boldsymbol{x} = T_{A_1}(\boldsymbol{w}_{i-1})$ being a random vector, we get

$$\mathbb{E}P[\boldsymbol{v}(\boldsymbol{x})] = \begin{bmatrix} \mu_1^2 \mathbb{E}\big\| \sum_{l=1}^N c_{l1} \boldsymbol{v}_l(\boldsymbol{x}_1), \big\|^2 \\ \vdots \\ \mu_N^2 \mathbb{E}\big\| \sum_{l=1}^N c_{lN} \boldsymbol{v}_l(\boldsymbol{x}_N) \big\|^2 \end{bmatrix} \tag{4.53}$$

For each block in (4.53), using Jensen's inequality, we have

$$\begin{aligned} \mathbb{E}\big\| \sum_{l=1}^N c_{lk} \boldsymbol{v}_l(\boldsymbol{x}_k) \big\|^2 &= \big(\sum_{l=1}^N c_{lk} \big)^2 \cdot \mathbb{E}\big\| \sum_{l=1}^N \frac{c_{lk}}{\sum_{l=1}^N c_{lk}} \boldsymbol{v}_l(\boldsymbol{x}_k) \big\|^2 \\ &\leq \big(\sum_{l=1}^N c_{lk} \big)^2 \cdot \sum_{l=1}^N \frac{c_{lk}}{\sum_{l=1}^N c_{lk}} \mathbb{E}\|\boldsymbol{v}_l(\boldsymbol{x}_k)\|^2 \\ &\leq \|C\|_1 \sum_{l=1}^N c_{lk} \big[\alpha \mathbb{E}\|\nabla_w J_l(\boldsymbol{x}_k)\|^2 + \sigma_v^2 \big] \end{aligned} \tag{4.54}$$

where we used (4.14) in the last step. Using (4.120),

$$\nabla_w J_l(\boldsymbol{x}_k) = \nabla_w J_l(w^o) + \big[\int_0^1 \nabla_w^2 J_l\big(w^o + t(\boldsymbol{x}_k - w^o)\big) dt \big] (\boldsymbol{x}_k - w^o) \tag{4.55}$$

From (4.121) and the norm inequality $\|x+y\|^2 \leq 2\|x\|^2 + 2\|y\|^2$, we obtain

$$\begin{aligned} \|\nabla_w J_l(\boldsymbol{x}_k)\|^2 &\leq 2\|\nabla_w J_l(w^o)\|^2 + 2\lambda_{l,\max}^2 \cdot \|\boldsymbol{x}_k - w^o\|^2 \\ &\leq 2\|\nabla_w J_l(w^o)\|^2 + 2\lambda_{\max}^2 \cdot \|\boldsymbol{x}_k - w^o\|^2 \end{aligned} \tag{4.56}$$

Substituting (4.56) into (4.54), we obtain

$$\begin{aligned}
&\mathbb{E}\big\|\sum_{l=1}^{N} c_{lk}\boldsymbol{v}_l(\boldsymbol{x}_k)\big\|^2 \\
&\leq \|C\|_1\sum_{l=1}^{N} c_{lk}\big[2\alpha\lambda_{\max}^2\mathbb{E}\|\boldsymbol{x}_k-w^o\|^2+2\alpha\|\nabla_w J_l(w^o)\|^2+\sigma_v^2\big] \\
&\leq 2\alpha\lambda_{\max}^2\|C\|_1^2\cdot\mathbb{E}\|\boldsymbol{x}_k-w^o\|^2+\|C\|_1^2\cdot\overline{\sigma}_v^2 \qquad (4.57)
\end{aligned}$$

where $\overline{\sigma}_v^2 \triangleq \max\limits_{1\leq l\leq N}\{2\alpha\|\nabla_w J_l(w^o)\|^2+\sigma_v^2\}$. Substituting (4.57) and $\boldsymbol{x}=T_{A_1}(\boldsymbol{w}_{i-1})$ into (4.53) leads to

$$\begin{aligned}
&\mathbb{E}P[\boldsymbol{v}(T_{A_1}(\boldsymbol{w}_{i-1}))] \\
&\preceq \Omega^2\big\{2\alpha\|C\|_1^2\lambda_{\max}^2\cdot\mathbb{E}P[T_{A_1}(\boldsymbol{w}_{i-1})-\mathbb{1}_N\otimes w^o] \\
&\qquad +\|C\|_1^2\overline{\sigma}_v^2\mathbb{1}_N\big\} \\
&\overset{(a)}{=} \Omega^2\big\{2\alpha\|C\|_1^2\lambda_{\max}^2\cdot\mathbb{E}P\big[T_{A_1}(\boldsymbol{w}_{i-1})-T_{A_1}(\mathbb{1}_N\otimes w^o)\big] \\
&\qquad +\|C\|_1^2\overline{\sigma}_v^2\mathbb{1}_N\big\} \\
&\overset{(b)}{=} \Omega^2\big\{2\alpha\|C\|_1^2\lambda_{\max}^2\cdot\mathbb{E}P\big[T_{A_1}\big(\boldsymbol{w}_{i-1}-\mathbb{1}_N\otimes w^o\big)\big] \\
&\qquad +\|C\|_1^2\overline{\sigma}_v^2\mathbb{1}_N\big\} \\
&\overset{(c)}{\preceq} \Omega^2\big\{2\alpha\|C\|_1^2\lambda_{\max}^2A_1^T\cdot\mathbb{E}P[\boldsymbol{w}_{i-1}-\mathbb{1}_N\otimes w^o] \\
&\qquad +\|C\|_1^2\overline{\sigma}_v^2\mathbb{1}_N\big\} \\
&\overset{(d)}{=} \Omega^2\Big\{2\alpha\|C\|_1^2\lambda_{\max}^2A_1^T\cdot 4\mathbb{E}P\Big[\frac{\boldsymbol{w}_{i-1}-w_\infty}{2}+\frac{w_\infty-\mathbb{1}_N\otimes w^o}{2}\Big] \\
&\qquad +\|C\|_1^2\overline{\sigma}_v^2\mathbb{1}_N\Big\} \\
&\overset{(e)}{\preceq} \Omega^2\big\{2\alpha\|C\|_1^2\lambda_{\max}^2A_1^T\cdot\big(2\mathbb{E}P[\boldsymbol{w}_{i-1}-w_\infty] \\
&\qquad +2P[w_\infty-\mathbb{1}_N\otimes w^o]\big)+\|C\|_1^2\overline{\sigma}_v^2\mathbb{1}_N\big\} \\
&= 4\alpha\|C\|_1^2\lambda_{\max}^2\cdot\Omega^2A_1^T\cdot\mathbb{E}P[\boldsymbol{w}_{i-1}-w_\infty]+\|C\|_1^2\Omega^2\cdot b_v \qquad (4.58)
\end{aligned}$$

where step (a) is due to the fact that A_1^T is right-stochastic so that $T_{A_1}(\mathbb{1}_N \otimes w^o) = \mathbb{1}_N \otimes w^o$, step (b) is because of the linearity of $T_{A_1}(\cdot)$, step (c) is due to property (4.36), step (d) is a consequence of Property 3 of Lemma 4.1, and step (e) is due to the convexity property (5.102). $\square$

Substituting (4.49) into (4.48), we obtain

$$\boxed{\text{MSP}_i \preceq A_2^T \Gamma_d A_1^T \cdot \text{MSP}_{i-1} + \|C\|_1^2 \cdot A_2^T \Omega^2 b_v} \tag{4.59}$$

where

$$\Gamma_d \triangleq \Gamma^2 + 4\alpha\lambda_{\max}^2 \|C\|_1^2 \cdot \Omega^2 \tag{4.60}$$

The following theorem gives the stability conditions on the inequality recursion (4.59) and derives both asymptotic and non-asymptotic bounds for MSP.

Theorem 4.2 (Mean-Square Stability and Bounds)**.** *Suppose $A_2^T \Gamma_d A_1^T$ is a stable matrix, i.e., $\rho(A_2^T \Gamma_d A_1^T) < 1$, where $\rho(\cdot)$ denotes the spectral radius of its matrix argument. Then, the following non-asymptotic bound holds for all $i \geq 0$:*

$$\text{MSP}_i \preceq (A_2^T \Gamma_d A_1^T)^i [\text{MSP}_0 - \text{MSP}_\infty^{\text{ub}}] + \text{MSP}_\infty^{\text{ub}} \tag{4.61}$$

where $\text{MSP}_\infty^{\text{ub}}$ is the asymptotic upper bound on MSP defined as

$$\text{MSP}_\infty^{\text{ub}} \triangleq \|C\|_1^2 (I_N - A_2^T \Gamma_d A_1^T)^{-1} A_2^T \Omega^2 b_v \tag{4.62}$$

And, as $i \to \infty$, we have the following asymptotic bound

$$\limsup_{i \to \infty} \text{MSP}_i \preceq \text{MSP}_\infty^{\text{ub}} \tag{4.63}$$

Furthermore, a sufficient condition that guarantees the stability of the matrix $A_2^T \Gamma_d A_1^T$ *is that*

$$0 < \mu_k < \min\left\{ \frac{\sigma_{k,\max}}{\sigma_{k,\max}^2 + 4\alpha\lambda_{\max}^2 \|C\|_1^2},\ \frac{\sigma_{k,\min}}{\sigma_{k,\min}^2 + 4\alpha\lambda_{\max}^2 \|C\|_1^2} \right\} \tag{4.64}$$

for all $k = 1, \ldots, N$*, where* $\sigma_{k,\max}$ *and* $\sigma_{k,\min}$ *were defined earlier in* (4.40)*.*

Proof. Iterating inequality (4.59), we obtain

$$\mathrm{MSP}_i \preceq (A_2^T \Gamma_d A_1^T)^i \mathrm{MSP}_0 + \|C\|_1^2 \cdot \big[\sum_{j=0}^{i-1} (A_2^T \Gamma_d A_1^T)^j \big] A_2^T \Omega^2 b_v \tag{4.65}$$

For the second term in (4.65), we note that $(I + X + \cdots + X^{i-1})(I - X) = I - X^i$. If X is a stable matrix so that $(I - X)$ is invertible, then it leads to $\sum_{j=0}^{i-1} X^j = (I - X^i)(I - X)^{-1}$. Using this relation and given that the matrix $A_2^T \Gamma_d A_1^T$ is stable, we can express (4.65) as

$$\begin{aligned} \mathrm{MSP}_i &\preceq (A_2^T \Gamma_d A_1^T)^i \mathrm{MSP}_0 \\ &\quad + \|C\|_1^2 \cdot \big[I_N - (A_2^T \Gamma_d A_1^T)^i \big] (I_N - A_2^T \Gamma_d A_1^T)^{-1} A_2^T \Omega^2 b_v \\ &= (A_2^T \Gamma_d A_1^T)^i [\mathrm{MSP}_0 - \mathrm{MSP}_\infty^{\mathrm{ub}}] + \mathrm{MSP}_\infty^{\mathrm{ub}} \end{aligned} \tag{4.66}$$

Letting $i \to \infty$ on both sides of the above inequality, we get $\limsup_{i \to \infty} \mathrm{MSP}_i \preceq \mathrm{MSP}_\infty^{\mathrm{ub}}$. In the last step, we need to show that the conditions on the step-sizes $\{\mu_k\}$ guarantee stability of the matrix $A_2^T \Gamma_d A_1^T$. Note that the spectral radius of a matrix is upper bounded by its matrix norms. Therefore,

$$\begin{aligned} \rho(A_2^T \Gamma_d A_1^T) &\leq \|A_2^T \Gamma_d A_1^T\|_\infty \\ &\leq \|A_2^T\|_\infty \cdot \|\Gamma_d\|_\infty \cdot \|A_1^T\|_\infty \end{aligned}$$

$$= \|\Gamma_d\|_\infty$$
$$= \left\|\Gamma^2 + 4\alpha\lambda_{\max}^2\|C\|_1^2 \cdot \Omega^2\right\|_\infty$$

If the right-hand side of the above inequality is strictly less than one, then the matrix $A_2^T\Gamma_d A_2^T$ is stable. Using (4.38)–(4.39), this condition is satisfied by the following quadratic inequalities in μ_k :

$$(1-\mu_k\sigma_{k,\max})^2 + \mu_k^2 \cdot 4\alpha\lambda_{\max}^2\|C\|_1^2 < 1 \tag{4.67}$$
$$(1-\mu_k\sigma_{k,\min})^2 + \mu_k^2 \cdot 4\alpha\lambda_{\max}^2\|C\|_1^2 < 1 \tag{4.68}$$

for all $k = 1, \ldots, N$. Solving the above inequalities, we obtain condition (4.64).

□

The non-asymptotic bound (4.61) characterizes how the MSP at each node evolves over time. It shows that the MSP converges to steady state at a geometric rate determined by the spectral radius of the matrix $A_2^T\Gamma_d A_1^T$. The transient term is determined by the difference between the initial MSP and the steay-state MSP. At steady state, the MSP is upper bounded by $\text{MSP}_\infty^{\text{ub}}$. We now examine closely how small the steady-state MSP can be for small step-size parameters $\{\mu_k\}$. Taking the $\infty-$norm of both sides of (4.62) and using the Neuman series $(I_N - A_2^T\Gamma_d A_1^T)^{-1} = \sum_{j=0}^\infty (A_2^T\Gamma_d A_1^T)^j$, we obtain

$$\begin{aligned}
\|\text{MSP}_\infty^{\text{ub}}\|_\infty &= \left\|\|C\|_1^2 \cdot (I_N - A_2^T\Gamma_d A_1^T)^{-1} \cdot A_2^T\Omega^2 b_v\right\|_\infty \\
&\leq \|C\|_1^2 \cdot \Big(\sum_{j=0}^\infty \|A_2^T\|_\infty^j \cdot \|\Gamma_d\|_\infty^j \cdot \|A_1^T\|_\infty^j\Big) \cdot \|A_2^T\|_\infty \cdot \|\Omega\|_\infty^2 \cdot \|b_v\|_\infty \\
&\overset{(a)}{\leq} \|C\|_1^2 \cdot \Big(\sum_{j=0}^\infty \|\Gamma_d\|_\infty^j\Big) \cdot \big(\max_{1\leq k\leq N}\mu_k\big)^2 \cdot \|b_v\|_\infty
\end{aligned}$$

$$= \frac{\|C\|_1^2 \cdot \|b_v\|_\infty}{1 - \|\Gamma_d\|_\infty} \cdot \big(\max_{1 \le k \le N} \mu_k \big)^2 \tag{4.69}$$

where step (a) is because A_1^T and A_2^T are right-stochastic matrices so that their $\infty-$norms (maximum absolute row sum) are one. Let $\mu_{\max}$ and $\mu_{\min}$ denote the maximum and minimum values of $\{\mu_k\}$, respectively, and let $\beta \triangleq \mu_{\min}/\mu_{\max}$. For sufficiently small step-sizes, i.e.,

$$0 < \mu_k < \frac{2}{\sigma_{k,\max} + \sigma_{k,\min}} \tag{4.70}$$

by the definitions of Γ_d and Γ in (4.60) and (4.38), we have

$$\begin{aligned}
\|\Gamma_d\|_\infty &\le \|\Gamma\|_\infty^2 + 4\alpha\lambda_{\max}\|C\|_1^2 \cdot \|\Omega\|_\infty^2 \\
&\overset{(a)}{=} \max_{1 \le k \le N} \{|1 - \mu_k \sigma_{k,\min}|^2\} + 4\alpha\lambda_{\max}\mu_{\max}^2\|C\|_1^2 \\
&\le 1 - 2\mu_{\min}\sigma_{\min} + \mu_{\max}^2(\sigma_{\max}^2 + 4\alpha\lambda_{\max}\|C\|_1^2) \\
&= 1 - 2\beta\mu_{\max}\sigma_{\min} + \mu_{\max}^2(\sigma_{\max}^2 + 4\alpha\lambda_{\max}\|C\|_1^2)
\end{aligned} \tag{4.71}$$

where $\sigma_{\max}$ and $\sigma_{\min}$ are the maximum and minimum values of $\{\sigma_{k,\max}\}$ and $\{\sigma_{k,\min}\}$, respectively, and step (a) holds for sufficiently small step-sizes satisfying (4.70). Note that (4.69) is a monotonically increasing function of $\|\Gamma_d\|_\infty$. Substituting (4.71) into (4.69), we get

$$\boxed{\begin{aligned}
&\limsup_{i \to \infty} \|\mathrm{MSP}_i\|_\infty \le \|\mathrm{MSP}_\infty^{\mathrm{ub}}\|_\infty \\
&\le \frac{\|C\|_1^2 \cdot \|b_v\|_\infty \cdot \mu_{\max}}{2\beta\sigma_{\min} - \mu_{\max}(\sigma_{\max}^2 + 4\alpha\lambda_{\max}\|C\|_1^2)} \sim O(\mu_{\max})
\end{aligned}} \tag{4.72}$$

Note that, for sufficiently small step-sizes, the right-hand side of (4.72) is approximately $\frac{\|C\|_1^2 \cdot \|b_v\|_\infty}{2\beta\sigma_{\min}}\mu_{\max}$, which is on the order of $O(\mu_{\max})$. In other words, the

steady-state MSP can be made be arbitrarily small for small step-sizes, and the estimators $\boldsymbol{w}_i = \text{col}\{\boldsymbol{w}_{1,i}, \ldots, \boldsymbol{w}_{N,i}\}$ will be close to the fixed point w_∞ (in the mean-square sense) even under gradient perturbations. To understand how close the estimate $\boldsymbol{w}_{k,i}$ at each node k is to the optimal solution w^o, a natural question to consider is how close the fixed point w_∞ is to $\mathbb{1}_N \otimes w^o$, which we study next.

4.5 Bias Analysis

Our objective is to examine how large $\|\mathbb{1}_N \otimes w^o - w_\infty\|^2$ is when the step-sizes are small. We carry out the analysis in two steps: first, we derive an expression for $\tilde{w}_\infty \triangleq \mathbb{1}_N \otimes w^o - w_\infty$, and then we derive the conditions that guarantee small bias.

To begin with, recall that w_∞ is the fixed point of $T_d(\cdot)$, to which the recursion $w_i = T_d(w_{i-1})$ converges. In other words, both $w_{k,i}$ and $w_{k,i-1}$ converge to $w_{k,\infty}$, which is the kth block of the vector w_∞. By (4.2), this implies that $\phi_{k,i-1}$ also converges to its limit, denoted by $\phi_{k,\infty}$. And by (4.3), the convergence of $\phi_{k,i-1}$ further guarantees the convergence of $\psi_{k,i}$ to its limit $\psi_{k,\infty}$. Also note that $T_d(\cdot)$ is an operator representation of the recursions (4.2)–(4.4). We let $i \to \infty$ on both sides of (4.2)–(4.4) and obtain

$$\phi_{k,\infty} = \sum_{l=1}^{N} a_{1,lk}\, w_{l,\infty} \tag{4.73}$$

$$\psi_{k,\infty} = \phi_{k,\infty} - \mu_k \sum_{l=1}^{N} c_{lk} \nabla_w J_l(\phi_{k,\infty}) \tag{4.74}$$

$$w_{k,\infty} = \sum_{l=1}^{N} a_{2,lk}\, \psi_{l,\infty} \tag{4.75}$$

where $w_{k,\infty}$, $\phi_{k,\infty}$ and $\psi_{k,\infty}$ denote the limits of $w_{k,i}$, $\phi_{k,i}$ and $\psi_{k,i}$ as $i \to \infty$,

respectively. Introduce the following bias vectors at node k

$$\tilde{w}_{k,\infty} \triangleq w^o - w_{k,\infty},\ \tilde{\phi}_{k,\infty} \triangleq w^o - \phi_{k,\infty},\ \tilde{\psi}_{k,\infty} \triangleq w^o - \psi_{k,\infty} \tag{4.76}$$

Subtracting each equation of (4.73)–(4.75) from w^o and using relation $\nabla_w J_l(\phi_{k,\infty}) = \nabla_w J_l(w^o) - H_{lk,\infty}\tilde{\phi}_{k,\infty}$ that can be derived from Lemma 4.4 in Appendix 4.A, we obtain

$$\tilde{\phi}_{k,\infty} = \sum_{l=1}^{N} a_{1,lk}\ \tilde{w}_{l,\infty} \tag{4.77}$$

$$\tilde{\psi}_{k,\infty} = \Big[I_M - \mu_k \sum_{l=1}^{N} c_{lk} H_{lk,\infty}\Big]\tilde{\phi}_{k,\infty} + \mu_k \sum_{l=1}^{N} c_{lk}\nabla_w J_l(w^o) \tag{4.78}$$

$$\tilde{w}_{k,\infty} = \sum_{l=1}^{N} a_{2,lk}\ \tilde{\psi}_{l,\infty} \tag{4.79}$$

where $H_{lk,\infty}$ is a positive semi-definite symmetric matrix defined as

$$H_{lk,\infty}\ \triangleq\ \int_0^1 \nabla_w^2 J_l\big(w^o - t\sum_{l=1}^{N} a_{1,lk}\tilde{w}_{l,\infty}\big)\big)dt \tag{4.80}$$

Introduce the following global vectors and matrices

$$\tilde{w}_\infty \triangleq \mathbb{1}_N \otimes w^o - w_\infty = \mathrm{col}\{\tilde{w}_{1,\infty},\ldots,\tilde{w}_{N,\infty}\} \tag{4.81}$$

$$\mathcal{A}_1 \triangleq A_1 \otimes I_M,\quad \mathcal{A}_2 \triangleq A_2 \otimes I_M,\quad \mathcal{C} \triangleq C \otimes I_M, \tag{4.82}$$

$$\mathcal{M} \triangleq \mathrm{diag}\{\mu_1,\ldots,\mu_N\} \otimes I_M \tag{4.83}$$

$$\mathcal{R}_\infty \triangleq \sum_{l=1}^{N} \mathrm{diag}\big\{c_{l1}H_{l1,\infty},\cdots,c_{lN}H_{lN,\infty}\big\}, \tag{4.84}$$

$$g^o \triangleq \mathrm{col}\{\nabla_w J_1(w^o),\ldots,\nabla_w J_N(w^o)\} \tag{4.85}$$

Then, expressions (4.77), (4.79) and (4.78) lead to

$$\boxed{\tilde{w}_\infty = \left[I_{MN} - \mathcal{A}_2^T \left(I_{MN} - \mathcal{M}\mathcal{R}_\infty\right) \mathcal{A}_1^T\right]^{-1} \mathcal{A}_2^T \mathcal{M}\mathcal{C}^T g^o} \tag{4.86}$$

Theorem 4.3 (Bias at Small Step-sizes)**.** *Suppose that the $N \times N$ matrix product $\bar{A} \triangleq A_1 A_2$ is a primitive left-stochastic matrix, so that its eigenvalue of largest magnitude is one with multiplicity one, and all other eigenvalues are strictly smaller than one. Let θ denote the right eigenvector of $\bar{A}$ of eigenvalue one and whose entries are normalized to add up to one, i.e., $\mathbb{1}^T\theta = 1$. Furthermore, assume the following condition holds:*

$$\theta^T A_2^T \Omega C^T = c_0 \mathbb{1}^T \tag{4.87}$$

where $\Omega \triangleq \mathrm{diag}\{\mu_1, \ldots, \mu_N\}$ was defined earlier in Lemma 4.3, and c_0 is some constant. Then,

$$\|\tilde{w}_\infty\|^2 = \|\mathbb{1}_N \otimes w^o - w_\infty\|^2 \sim O(\mu_{\max}^2) \tag{4.88}$$

Proof. See Appendix 4.B. □

Let $\bar{A} = [\bar{a}_{lk}]$ denote the entries of $\bar{A}$. The matrix $\bar{A}$ is a primitive left-stochastic matrix if the network is connected (not necessarily fully connected) and there is at least one $\bar{a}_{kk} > 0$ for some node k, i.e., $\bar{A}^T\mathbb{1} = \mathbb{1}$ and there exists a finite integer j_o such that all entries of $\bar{A}^{j_o}$ are strictly positive. It then follows from the Perron-Frobenius Theorem [68] that the matrix $\bar{A} = A_1 A_2$ has an eigenvalue equal to one of multiplicity one while all other eigenvalues are strictly less than one. Obviously, $\mathbb{1}^T$ is a left eigenvector of $A_1 A_2$ corresponding to the eigenvalue at one. For the right eigenvector θ that corresponds to the eigenvalue at one,

the Perron-Frobenius Theorem further ensures that all entries of θ are positive. Furthermore, if condition (4.87) holds as well, then Theorem 4.3 implies that the bias would become arbitrarily small for small step-size. For condition (4.87) to hold, one choice is to require the matrices A_1^T and A_2^T to be doubly stochastic, and all nodes to use the same step-size μ, namely, $\Omega = \mu I_N$. In that case, the matrix $A_2^T A_1^T$ is doubly-stochastic so that the left eigenvector of eigenvalue one is $\theta^T = \mathbb{1}^T$ and (4.87) holds. The matrix A_1^T or A_2^T does not need to be doubly stochastic or the step-size parameters $\{\mu_k\}$ do not need to be the same across nodes. We can also have other possible choices that satisfy (4.87). Consider the ATC case where $A_2 = A$, $A_1 = I$ and $C = I$, where A is a left-stochastic matrix with nonnegative entries. Then, θ is the right eigenvector of $A_1 A_2 = A$ corresponding to the eigenvalue at one. In this case, condition (4.87) becomes $\theta^T \Omega = c_0 \mathbb{1}^T$, which is equivalent to $\theta_k \mu_k = c_0$, where θ_k is the kth entry of the vector θ. Therefore, if we choose to use different step-sizes μ_k at different nodes, then we need to choose the right eigenvector θ of the matrix A at eigenvalue one to be

$$\theta = c_0 \cdot \mathrm{col}\left\{\mu_1^{-1}, \mu_2^{-1}, \ldots, \mu_N^{-1}\right\} \tag{4.89}$$

where $c_0 = (\sum_{k=1}^N \mu_k^{-1})^{-1}$ so that $\theta^T \mathbb{1} = 1$. Such a left-stochastic matrix A can be implemented by using the Metropolis-Hasting rule [18, 62, 146].

Finally, we combine the results from Theorems 4.2 and 4.3 to bound the mean-square-error (MSE) of the estimators $\{\boldsymbol{w}_{k,i}\}$ from the desired solution w^o. Introduce the $N \times 1$ MSE vector

$$\mathrm{MSE}_i \triangleq \mathbb{E}P[\tilde{\boldsymbol{w}}_i] = \mathrm{col}\big\{\mathbb{E}\|\tilde{\boldsymbol{w}}_{1,i}\|^2, \ldots, \mathbb{E}\|\tilde{\boldsymbol{w}}_{N,i}\|^2\big\} \tag{4.90}$$

where $\tilde{\boldsymbol{w}}_{k,i} \triangleq w^o - \boldsymbol{w}_{k,i}$. Using Properties 3–4 in Lemma 4.1, we obtain

$$\begin{aligned}
\text{MSE}_i &= \mathbb{E}P\big[2\big(\frac{\mathbb{1}_N \otimes w^o - w_\infty}{2} + \frac{w_\infty - \boldsymbol{w}_i}{2}\big)\big] \\
&\preceq 2P[\tilde{w}_\infty] + 2\,\mathbb{E}P[w_\infty - \boldsymbol{w}_i] \\
&= 2P[\tilde{w}_\infty] + 2\,\text{MSP}_i
\end{aligned} \tag{4.91}$$

Taking the $\infty-$norm of both sides of above inequality and using property (5.110), we obtain

$$\begin{aligned}
\limsup_{i\to\infty} \|\text{MSE}_i\|_\infty &\leq 2\|P[\tilde{w}_\infty]\|_\infty + 2\limsup_{i\to\infty} \|\text{MSP}_i\|_\infty \\
&= 2\|\tilde{w}_\infty\|_{b,\infty}^2 + 2\limsup_{i\to\infty} \|\text{MSP}_i\|_\infty \\
&\sim O(\mu_{\max}^2) + O(\mu_{\max})
\end{aligned} \tag{4.92}$$

where in the last step, we used (4.72) and (4.88), and the fact that all vector norms are equivalent. Therefore, as the step-sizes become small, the MSEs become small and the estimates $\{\boldsymbol{w}_{k,i}\}$ get arbitrarily close to the optimal solution w^o. We also observe that, for small step-sizes, the dominating steady-state error is MSP, which is caused by the gradient noise and is on the order of $O(\mu_{\max})$. On the other hand, the bias term is a high order component, i.e., $O(\mu_{\max}^2)$, and can be ignored.

The fact that the bias term $\tilde{w}_\infty$ is small also gives us a useful approximation for $\mathcal{R}_\infty$ in (4.84). Since $\tilde{w}_\infty = \text{col}\{\tilde{w}_{1,\infty}, \ldots, \tilde{w}_{N,\infty}\}$ is small for small step-sizes, the matrix $H_{lk,\infty}$ defined in (4.80) can be approximated as $H_{lk,\infty} \approx \nabla_w^2 J_l(w^o)$. Then, by definition (4.84), we have

$$\boxed{\mathcal{R}_\infty \approx \sum_{l=1}^{N} \text{diag}\big\{c_{l1}\nabla_w^2 J_l(w^o), \ldots, c_{lN}\nabla_w^2 J_l(w^o)\big\}} \tag{4.93}$$

Expressing (4.93) is useful for evaluating closed-form expressions of the steady-state MSE in sequel.

4.6 Steady-State Performance

So far, we derived inequalities (4.92) to bound the steady-state performance, and showed that, for small step-sizes, the solution at each node k approaches the same optimal solution w^o. In this section, we derive closed-form expressions (rather than bounds) for the steady-state MSE at small step-sizes. Introduce the error vectors[2]

$$\tilde{\boldsymbol{\phi}}_{k,i} \triangleq w^o - \boldsymbol{\phi}_{k,i},\ \tilde{\boldsymbol{\psi}}_{k,i} \triangleq w^o - \boldsymbol{\psi}_{k,i},\ \tilde{\boldsymbol{w}}_{k,i} \triangleq w^o - \boldsymbol{w}_{k,i} \tag{4.94}$$

and the following global random quantities

$$\tilde{\boldsymbol{w}}_i \triangleq \mathrm{col}\{\tilde{\boldsymbol{w}}_{1,i}, \ldots, \tilde{\boldsymbol{w}}_{N,i}\} \tag{4.95}$$

$$\boldsymbol{\mathcal{R}}_{i-1} \triangleq \sum_{l=1}^{N} \mathrm{diag}\big\{c_{l1}\boldsymbol{H}_{l1,i-1}, \cdots, c_{lN}\boldsymbol{H}_{lN,i-1}\big\} \tag{4.96}$$

$$\boldsymbol{H}_{lk,i-1} \triangleq \int_0^1 \nabla_w^2 J_l\Big(w^o - t\sum_{l=1}^{N} a_{1,lk}\tilde{\boldsymbol{w}}_{l,i-1}\Big)dt \tag{4.97}$$

$$\boldsymbol{g}_i \triangleq \sum_{l=1}^{N} \mathrm{col}\big\{c_{l1}\boldsymbol{v}_{l,i}(\boldsymbol{\phi}_{1,i-1}), \cdots, c_{lN}\boldsymbol{v}_{l,i}(\boldsymbol{\phi}_{N,i-1})\big\} \tag{4.98}$$

Then, we can establish that

$$\tilde{\boldsymbol{w}}_i = \mathcal{A}_2^T[I_{MN} - \mathcal{M}\boldsymbol{\mathcal{R}}_{i-1}]\mathcal{A}_1^T\tilde{\boldsymbol{w}}_{i-1} + \mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o + \mathcal{A}_2^T\mathcal{M}\boldsymbol{g}_i \tag{4.99}$$

[2] We always use the notation $\tilde{w} = w^o - w$ to denote the error relative to w^o. For the error between w and the fixed point w_∞, we do not define a separate notation, but instead write $w_\infty - w$ explicitly to avoid confusion.

According to (4.92), the error $\tilde{\boldsymbol{w}}_{k,i}$ at each node k would be small for small step-sizes and after long enough time. In other words, $\boldsymbol{w}_{k,i}$ is close to w^o. And recalling from (4.9) that $\boldsymbol{\phi}_{k,i-1}$ is a convex combination of $\{\boldsymbol{w}_{l,i}\}$, we conclude that the quantities $\{\boldsymbol{\phi}_{l,i-1}\}$ are also close to w^o. Therefore, we can approximate $\boldsymbol{H}_{lk,i-1}$, $\boldsymbol{\mathcal{R}}_{i-1}$ and $\boldsymbol{g}_i$ in (4.96)–(4.98) by

$$\boldsymbol{H}_{lk,i-1} \approx \int_0^1 \nabla_w^2 J_l(w^o) dt = \nabla_w^2 J_l(w^o) \tag{4.100}$$

$$\boldsymbol{\mathcal{R}}_{i-1} \approx \sum_{l=1}^{N} \mathrm{diag}\big\{c_{l1}\nabla_w^2 J_l(w^o), \ldots, c_{lN}\nabla_w^2 J_l(w^o)\big\} \approx \mathcal{R}_\infty \tag{4.101}$$

Then, the error recursion (4.99) can be approximated by

$$\boxed{\tilde{\boldsymbol{w}}_i = \mathcal{A}_2^T[I_{MN} - \mathcal{M}\mathcal{R}_\infty]\mathcal{A}_1^T\tilde{\boldsymbol{w}}_{i-1} + \mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o + \mathcal{A}_2^T\mathcal{M}\boldsymbol{g}_i} \tag{4.102}$$

Note that our analysis proceeds with the above approximate relations (4.100)–(4.101). In Chapter 6, we will perform a rigorous analysis of the steady-state performance under more relaxed conditions. Furthermore, we will also quantify the error introduced by these approximations and show that it only adds a high order correct term of $o(\mu_{\max})$, where $\mu_{\max}$ denotes the largest step-size across nodes and $o(\mu_{\max})$ denotes a strictly higher order term of $\mu_{\max}$.

First, let us examine the behavior of $\mathbb{E}\tilde{\boldsymbol{w}}_i$. Taking expectation of both sides of recursion (4.102), we obtain

$$\mathbb{E}\tilde{\boldsymbol{w}}_i = \mathcal{A}_2^T[I_{MN} - \mathcal{M}\mathcal{R}_\infty]\mathcal{A}_1^T\mathbb{E}\tilde{\boldsymbol{w}}_{i-1} + \mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o \tag{4.103}$$

This recursion converges when the matrix $\mathcal{A}_2^T[I_{MN} - \mathcal{M}\mathcal{R}_\infty]\mathcal{A}_1^T$ is stable, which is guaranteed by (4.43) (see Appendix C of [34]). Let $i \to \infty$ on both sides of

(4.103) so that

$$\boxed{\begin{aligned}\mathbb{E}\tilde{\boldsymbol{w}}_\infty &\triangleq \lim_{i\to\infty}\mathbb{E}\tilde{\boldsymbol{w}}_i \\ &= \left[I_{MN}-\mathcal{A}_2^T\left(I_{MN}-\mathcal{M}\mathcal{R}_\infty\right)\mathcal{A}_1^T\right]^{-1}\mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o\end{aligned}} \tag{4.104}$$

Note that $\mathbb{E}\tilde{\boldsymbol{w}}_\infty$ coincides with (4.86). By Theorem 4.3, we know that the squared norm of this expression is on the order of $O(\mu_{\max}^2)$ at small step-sizes — see (4.88).

Next, we derive closed-form expressions for the MSEs, i.e., $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$. Let R_v denote the covariance matrix of $\boldsymbol{g}_i$ evaluated at w^o as $i\to\infty$:

$$\begin{aligned}R_v \;&= \mathbb{E}\Big\{\Big[\sum_{l=1}^{N}\mathrm{col}\big\{c_{l1}\boldsymbol{v}_{l,i}(w^o),\cdots,c_{lN}\boldsymbol{v}_{l,i}(w^o)\big\}\Big] \\ &\quad\times\Big[\sum_{l=1}^{N}\mathrm{col}\big\{c_{l1}\boldsymbol{v}_{l,i}(w^o),\cdots,c_{lN}\boldsymbol{v}_{l,i}(w^o)\big\}\Big]^T\Big\}\end{aligned} \tag{4.105}$$

In practice, we can evaluate R_v from the expressions of $\{\boldsymbol{v}_{l,i}(w^o)\}$. Equating the squared *weighted* Euclidean "norm" of both sides of (4.102), applying the expectation operator with assumption (4.13), we can establish the following approximate variance relation at small step-sizes:

$$\begin{aligned}\mathbb{E}\|\tilde{\boldsymbol{w}}_i\|_\Sigma^2 \approx \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|_{\Sigma'}^2 &+ \mathrm{Tr}(\Sigma\mathcal{A}_2^T\mathcal{M}R_v\mathcal{M}\mathcal{A}_2) + \mathrm{Tr}\{\Sigma\mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o(\mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o)^T\} \\ &+ 2(\mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o)^T\cdot\Sigma\mathcal{A}_2^T\left(I_{MN}-\mathcal{M}\mathcal{R}_\infty\right)\mathcal{A}_1^T\mathbb{E}\tilde{\boldsymbol{w}}_{i-1}\end{aligned} \tag{4.106}$$

$$\Sigma' \approx \mathcal{A}_1(I_{MN}-\mathcal{M}\mathcal{R}_\infty)\mathcal{A}_2\Sigma\mathcal{A}_2^T(I_{MN}-\mathcal{M}\mathcal{R}_\infty)\mathcal{A}_1^T \tag{4.107}$$

where Σ is a positive semi-definite weighting matrix that we are free to choose and the notation $\|x\|_\Sigma^2$ denotes $x^*\Sigma x$. Let $\sigma=\mathrm{vec}(\Sigma)$ denote the vectorization operation that stacks the columns of a matrix Σ on top of each other. We shall use the notation $\|x\|_\sigma^2$ and $\|x\|_\Sigma^2$ interchangeably. Following the argument from [34],

we can rewrite (4.106) as

$$\mathbb{E}\|\tilde{\boldsymbol{w}}_i\|_{\sigma}^2 \approx \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|_{F\sigma}^2 + r^T\sigma + \sigma^T Q\,\mathbb{E}\tilde{\boldsymbol{w}}_{i-1} \tag{4.108}$$

where

$$F \triangleq \mathcal{A}_1[I_{MN} - \mathcal{M}\mathcal{R}_{\infty}]\mathcal{A}_2 \otimes \mathcal{A}_1[I_{MN} - \mathcal{M}\mathcal{R}_{\infty}]\mathcal{A}_2 \tag{4.109}$$

$$\mathcal{B} = \mathcal{A}_2^T[I_{MN} - \mathcal{M}\mathcal{R}_{\infty}]\mathcal{A}_1^T \tag{4.110}$$

$$r \triangleq \mathrm{vec}\big(\mathcal{A}_2^T\mathcal{M}R_v\mathcal{M}\mathcal{A}_2\big) + \mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o \otimes \mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o \tag{4.111}$$

$$Q \triangleq 2\mathcal{A}_2^T(I_{MN} - \mathcal{M}\mathcal{R}_{\infty})\mathcal{A}_1^T \otimes \mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o \tag{4.112}$$

We already established that $\mathbb{E}\tilde{\boldsymbol{w}}_{i-1}$ on the right-hand side of (4.108) converges to its limit $\mathbb{E}\tilde{\boldsymbol{w}}_{\infty}$ under condition (4.43). And, it was shown in [116, pp.344-346] that such recursion converges to a steady-state value if the matrix F is stable, i.e., $\rho(F) < 1$. This condition is guaranteed when the step-sizes are sufficiently small (or chosen according to (4.43)) — see the proof in Appendix 4.D. Letting $i \to \infty$ on both sides of expression (4.108), we obtain:

$$\boxed{\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_i\|_{(I-F)\sigma}^2 \approx \left(r + Q\,\mathbb{E}\tilde{\boldsymbol{w}}_{\infty}\right)^T \sigma} \tag{4.113}$$

We can now resort to (4.113) and use it to evaluate various performance metrics by choosing proper weighting matrices Σ (or σ). For example, the MSE of any node k can be obtained by computing $\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_i\|_T^2$ with a block weighting matrix T that has an identity matrix at block (k,k) and zeros elsewhere: $\lim\limits_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \lim\limits_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_i\|_T^2$. Denote the vectorized version of this matrix by $t_k \triangleq \mathrm{vec}(\mathrm{diag}(e_k) \otimes I_M)$, where e_k is a vector whose kth entry is one and zeros elsewhere. Then, if we select σ in (4.113) as $\sigma = (I - F)^{-1}t_k$, the term on the left-hand side becomes

the desired $\lim_{i\to\infty}\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$ and the MSE for node k is therefore given by:

$$\boxed{\text{MSE}_k \triangleq \lim_{i\to\infty}\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 \approx (r+Q\,\mathbb{E}\tilde{\boldsymbol{w}}_\infty)^T(I-F)^{-1}t_k} \tag{4.114}$$

If we are interested in the average network MSE, then it is given by

$$\overline{\text{MSE}} \triangleq \frac{1}{N}\sum_{k=1}^{N}\text{MSE}_k \tag{4.115}$$

4.7 Conclusion

In this chapter, we analyzed the mean-square-error performance of the diffusion strategy, and showed that the solution at each node gets arbitrarily close to the minimizer of the aggregate cost for small step-sizes. The result holds even when the minimizers of the individual costs are not necessarily equal to each other.

4.A Properties of the Operators

Properties 1-3 are straightforward from the definitions of $T_A(\cdot)$ and $P[\cdot]$. We therefore omit the proof for brevity, and start with property 4.

(Property 4: Convexity)

We can express each $N\times 1$ block vector $x^{(k)}$ in the form $x^{(k)} = \text{col}\{x_1^{(k)},\ldots,x_N^{(k)}\}$ for $k=1,\ldots,N$. Then, the convex combination of $x^{(1)},\ldots,x^{(N)}$ can be expressed as

$$\sum_{k=1}^{K} a_k\, x^{(k)} = \text{col}\Big\{\sum_{k=1}^{K} a_k x_1^{(k)},\ldots,\sum_{k=1}^{K} a_k x_N^{(k)}\Big\} \tag{4.116}$$

According to the definition of the operator $P[\cdot]$, and in view of the convexity of $\|\cdot\|^2$, we have

$$P\Big[\sum_{k=1}^{K} a_l\ x^{(k)}\Big] = \mathrm{col}\Big\{\big\|\sum_{k=1}^{K} a_l x_1^{(k)}\big\|^2, \ldots, \big\|\sum_{k=1}^{K} a_l x_N^{(k)}\big\|^2\Big\}$$
$$\preceq \mathrm{col}\Big\{\sum_{k=1}^{K} a_l \|x_1^{(k)}\|^2, \ldots, \sum_{k=1}^{K} a_l \|x_N^{(k)}\|^2\Big\} = \sum_{k=1}^{K} a_l\ P[x^{(k)}] \tag{4.117}$$

(Property 5: Additivity)

By the definition of $P[\cdot]$ and the assumption that $\mathbb{E}\boldsymbol{x}_k^T\boldsymbol{y}_k = 0$ for each $k = 1, \ldots, N$, we obtain

$$\mathbb{E}P[\boldsymbol{x}+\boldsymbol{y}] = \mathrm{col}\{\mathbb{E}\|\boldsymbol{x}_1+\boldsymbol{y}_1\|^2,\ \ldots,\ \mathbb{E}\|\boldsymbol{x}_N+\boldsymbol{y}_N\|^2\}$$
$$= \mathrm{col}\{\mathbb{E}\|\boldsymbol{x}_1\|^2+\mathbb{E}\|\boldsymbol{y}_1\|^2, \ldots, \mathbb{E}\|\boldsymbol{x}_N\|^2+\mathbb{E}\|\boldsymbol{y}_N\|^2\} = \mathbb{E}P[\boldsymbol{x}]+\mathbb{E}P[\boldsymbol{y}]$$

(Property 6: Variance Relations)

We first prove (4.36). From the definition of $T_A(\cdot)$ in (4.25) and the definition of $P[\cdot]$ in (4.27), we express

$$P[T_A(x)] = \mathrm{col}\Big\{\big\|\sum_{l=1}^{N} a_{l1}x_l\big\|^2,\ \ldots,\ \big\|\sum_{l=1}^{N} a_{lN}x_l\big\|^2\Big\} \tag{4.118}$$

Since $\|\cdot\|^2$ is a convex function and each sum inside the squared norm operator is a convex combination of $x_1, \ldots, x_N$ (A^T is right stochastic), then by Jensen's inequality [20, p.77]:

$$P[T_A(x)] \preceq \mathrm{col}\Big\{\sum_{l=1}^{N} a_{l1}\|x_l\|^2,\ \ldots,\ \sum_{l=1}^{N} a_{lN}\|x_l\|^2\Big\}$$
$$= A^T\mathrm{col}\{\|x_1\|^2,\ \ldots,\ \|x_N\|^2\} = A^T P[x] \tag{4.119}$$

Next, we proceed to prove (4.37). We need to call upon the following useful lemmas from [105, p.24], and Lemmas 1–2 in [34], respectively.

Lemma 4.4 (Mean-Value Theorem)**.** *For any twice-differentiable function $f(\cdot)$, it holds that*

$$\nabla f(y) = \nabla f(x) + \left[\int_0^1 \nabla^2 f\big(x+t(y-x)\big)dt\right](y-x) \tag{4.120}$$

where $\nabla^2 f(\cdot)$ denotes the symmetric Hessian of $f(\cdot)$. □

Lemma 4.5 (Bounds on the Integral of Hessian)**.** *Under Assumption 4.1, it holds that for any vectors x and y:*

$$\lambda_{l,\min} I_M \leq \int_0^1 \nabla_w^2 J_l(x+ty)dt \leq \lambda_{l,\max} I_M \tag{4.121}$$

$$\left\| I - \mu_k \sum_{l=1}^N c_{lk} \left[\int_0^1 \nabla_w^2 J_l(x+ty)dt\right]\right\| \leq \gamma_k \tag{4.122}$$

where $\|\cdot\|$ denotes the $2-$induced norm, and γ_k, $\sigma_{k,\min}$ and $\sigma_{k,\max}$ were defined in (4.39)–(4.40). □

By the definition of the operator $T_G(\cdot)$ in (4.26) and the expression (4.120), we express $T_G(x) - T_G(y)$ as

$$T_G(x) - T_G(y) = \begin{bmatrix} \Big[I_M - \mu_1 \sum_{l=1}^N c_{l1} \int_0^1 \nabla_w^2 J_l(y_1 + t(x_1 - y_1))dt\Big](x_1 - y_1) \\ \vdots \\ \Big[I_M - \mu_N \sum_{l=1}^N c_{lN} \int_0^1 \nabla_w^2 J_l(y_N + t(x_N - y_N))dt\Big](x_N - y_N) \end{bmatrix}$$

Therefore, using (4.122) and the definition of $P[\cdot]$ in (4.27), we obtain

$$\begin{aligned} P[T_G(x)-T_G(y)] &\preceq \mathrm{col}\big\{\gamma_1^2\|x_1-y_1\|^2,\ldots,\gamma_N^2\|x_N-y_N\|^2\big\} \\ &= \Gamma^2 P[x-y] \end{aligned} \tag{4.123}$$

(Property 7: Block Maximum Norm)

According to the definition of $P[\cdot]$ and the definition of block maximum norm [34], we have

$$\begin{aligned} \|P[x]\|_\infty &= \big\|\mathrm{col}\{\|x_1\|^2,\ \ldots,\ \|x_N\|^2\}\big\|_\infty \\ &= \max_{1\le k\le N}\|x_k\|^2 = \big(\max_{1\le k\le N}\|x_k\|\big)^2 = \|x\|_{b,\infty}^2 \end{aligned} \tag{4.124}$$

(Property 8: Preservation of Inequality)

To prove $Fx \preceq Fy$, it suffices to prove $0 \preceq F(y-x)$. Since $x \preceq y$, we have $0 \preceq y-x$, i.e., all entries of the vector $y-x$ are nonnegative. Furthermore, since all entries of the matrix F are nonnegative, the entries of the vector $F(y-x)$ are all nonnegative, which means $0 \preceq F(y-x)$.

4.B Bias at Small Step-Sizes

It suffices to show that $\lim_{\mu_{\max}\to 0}\|\mathbb{1}\otimes w^o - w_\infty\|/\mu_{\max} = \xi$ where ξ is a constant independent of $\mu_{\max}$. It is known that any matrix is similar to a Jordan canonical form [82]. Hence, there exists an invertible matrix Y such that $A_2^T A_1^T = YJY^{-1}$, where J is the Jordan canonical form of the matrix $A_2^T A_1^T$, and the columns of the matrix Y are the corresponding *right principal vectors* of various degrees [82, pp.82–88]; the right principal vector of degree one is the right eigenvector. Obviously, the matrices J and Y are independent of $\mu_{\max}$. Using the Kronecker

product property [82, p.140]: $(A \otimes B)(C \otimes D) = AC \otimes BD$, we obtain

$$\mathcal{A}_2^T \mathcal{A}_1^T = A_2^T A_1^T \otimes I_M = (Y \otimes I_M)(J \otimes I_M)(Y^{-1} \otimes I_M) \tag{4.125}$$

Denote $\mu_k = \beta_k \mu_{\max}$, where β_k is some positive scalar such that $0 < \beta_k \leq 1$. Substituting (4.125) into (4.86), we obtain

$$\begin{aligned} \mathbb{1} \otimes w^o - w_\infty &= \left[I_{MN} - \mathcal{A}_2^T \mathcal{A}_1^T + \mathcal{A}_2^T \mathcal{M} \mathcal{R}_\infty \mathcal{A}_1^T \right]^{-1} \mathcal{A}_2^T \mathcal{M} \mathcal{C}^T g^o \\ &= (Y \otimes I_M) \left[I_{MN} - J \otimes I_M + \mu_{\max} E \right]^{-1} \\ &\quad \times (Y^{-1} \otimes I_M) \mathcal{A}_2^T \mathcal{M} \mathcal{C}^T g^o \end{aligned} \tag{4.126}$$

where

$$E = (Y^{-1} \otimes I_M) \mathcal{A}_2^T \mathcal{M}_0 \mathcal{R}_\infty \mathcal{A}_1^T (Y \otimes I_M) \tag{4.127}$$

$$\mathcal{M}_0 \triangleq \mathcal{M}/\mu_{\max} = \operatorname{diag}\{\beta_1, \ldots, \beta_N\} \otimes I_M \tag{4.128}$$

where we shall define $\Omega_0 \triangleq \operatorname{diag}\{\beta_1, \ldots, \beta_N\}$. By (4.5), the matrix $A_2^T A_1^T$ is right-stochastic, and since $A_2^T A_1^T$ is primitive, it will have an eigenvalue of one that has multiplicity one and is strictly greater than all other eigenvalues [68]. Furthermore, the corresponding left and right eigenvectors are θ^T and $\mathbb{1}$, with $\theta^T \succ 0$ (all entries of the row vector θ^T are real positive numbers). For this reason, we can partition J, Y^{-1} and Y in the following block forms:

$$J = \operatorname{diag}\{1,\ J_0\},\ Y^{-1} = \operatorname{col}\left\{ \frac{\theta^T}{\theta^T \mathbb{1}},\ Y_R \right\},\ Y = [\mathbb{1}\ Y_L] \tag{4.129}$$

where J_0 is an $(N-1) \times (N-1)$ matrix that contains the Jordan blocks of eigenvalues strictly within unit circle, i.e., $\rho(J_0) < 1$. The first row of the matrix

Y^{-1} in (4.129) is normalized by $\theta^T\mathbb{1}$ so that $Y^{-1}Y = I$. (Note that $Y^{-1}Y = I$ requires the product of the first row of Y^{-1} and the first column of Y to be one: $\frac{\theta^T}{\theta^T\mathbb{1}}\mathbb{1} = 1$.) Substituting these partitionings into (4.127):

$$E = \begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix} \tag{4.130}$$

$$E_{11} \triangleq \big(\frac{\theta^T}{\theta^T\mathbb{1}} \otimes I_M\big)\mathcal{A}_2^T\mathcal{M}_0\mathcal{R}_\infty\mathcal{A}_1^T(\mathbb{1} \otimes I_M) \tag{4.131}$$

$$E_{12} \triangleq \big(\frac{\theta^T}{\theta^T\mathbb{1}} \otimes I_M\big)\mathcal{A}_2^T\mathcal{M}_0\mathcal{R}_\infty\mathcal{A}_1^T(Y_L \otimes I_M) \tag{4.132}$$

$$E_{21} \triangleq (Y_R \otimes I_M)\mathcal{A}_2^T\mathcal{M}_0\mathcal{R}_\infty\mathcal{A}_1^T(\mathbb{1} \otimes I_M) \tag{4.133}$$

$$E_{22} \triangleq (Y_R \otimes I_M)\mathcal{A}_2^T\mathcal{M}_0\mathcal{R}_\infty\mathcal{A}_1^T(Y_L \otimes I_M) \tag{4.134}$$

Observe that the matrices E_{11}, E_{12}, E_{21} and E_{22} are independent of $\mu_{\max}$. Substituting (4.129)-(4.130) into (4.126), we obtain

$$\begin{aligned} \mathbb{1}\otimes w^o - w_\infty = (Y\otimes I_M)&\begin{bmatrix} \mu_{\max}E_{11} & \mu_{\max}E_{12} \\ \mu_{\max}E_{21} & I - J_0\otimes I_M + \mu_{\max}E_{22} \end{bmatrix}^{-1} \\ &\times \begin{bmatrix} \frac{1}{\theta^T\mathbb{1}}(\theta^T \otimes I_M)\mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o \\ (Y_R \otimes I_M)\mathcal{A}_2^T\mathcal{M}\mathcal{C}^T g^o \end{bmatrix} \end{aligned} \tag{4.135}$$

Let us denote

$$\begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \triangleq \begin{bmatrix} \mu_{\max}E_{11} & \mu_{\max}E_{12} \\ \mu_{\max}E_{21} & I - J_0\otimes I_M + \mu_{\max}E_{22} \end{bmatrix}^{-1} \tag{4.136}$$

Furthermore, recalling that w^o is the minimizer of the global cost function (2.1), we have

$$\sum_{l=1}^{N} \nabla_w J_l(w^o) = 0 \quad \Leftrightarrow \quad (\mathbb{1}^T \otimes I_M)\, g^o = 0 \tag{4.137}$$

which, together with condition (4.87), implies that

$$(\theta^T \otimes I_M)\mathcal{A}_2^T \mathcal{M}\mathcal{C}^T g^o = (\theta^T A_2^T \Omega C^T \otimes I_M) g^o = c_0 (\mathbb{1}^T \otimes I_M) g^o = 0$$

where we also used the facts that $\mathcal{A}_2^T = A_2^T \otimes I_M$, $\mathcal{C}^T = C^T \otimes I_M$, $\mathcal{M} = \Omega \otimes I_M$ and the Kronecker product property: $(A \otimes B)(C \otimes D)$. Substituting the above result and (4.136) into (4.135) and using (4.128) lead to

$$\mathbb{1} \otimes w^o - w_\infty = \mu_{\max} (Y \otimes I_M) \begin{bmatrix} G_{12} \\ G_{22} \end{bmatrix} (Y_R A_2^T \Omega_0 C^T \otimes I_M) g^o \tag{4.138}$$

To proceed with analysis, we need to evaluate G_{12} and G_{22}. We call upon the relation from [82, pp.48]:

$$\begin{bmatrix} P & Q \\ U & V \end{bmatrix}^{-1} = \begin{bmatrix} P^{-1} + P^{-1}QSUP^{-1} & -P^{-1}QS \\ -SUP^{-1} & S \end{bmatrix} \tag{4.139}$$

where $S = (V - UP^{-1}Q)^{-1}$. To apply the above relation to (4.136), we first need to verify that E_{11} is invertible. By (4.131),

$$\begin{aligned} E_{11} &= \Big(\frac{\theta^T}{\theta^T \mathbb{1}} A_2^T \Omega_0 \otimes I_M\Big) \mathcal{R}_\infty (A_1^T \mathbb{1} \otimes I_M) \\ &= (z^T \otimes I_M) \mathcal{R}_\infty (\mathbb{1} \otimes I_M) = \sum_{k=1}^{N} z_k \sum_{l=1}^{N} c_{lk} H_{lk,\infty} \end{aligned} \tag{4.140}$$

where z_k denotes the kth entry of the vector $z \triangleq \Omega_0 A_2 \theta / \theta^T \mathbb{1}$ (note that all entries of z are non-negative, i.e., $z_k \geq 0$). Recall from (4.80) that $H_{lk,\infty}$ is a symmetric positive semi-definite matrix. Moreover, since z_k and c_{lk} are nonnegative, we can conclude from (4.140) that E_{11} is a symmetric positive semi-definite matrix. Next, we show that E_{11} is actually strictly positive definite. Applying (4.121) to the expression of $H_{lk,\infty}$ in (4.80), we obtain $H_{lk,\infty} \geq \lambda_{l,\min} I_M$. Substituting into (4.140):

$$\begin{aligned} E_{11} &\geq \left[\sum_{k=1}^{N} z_k \sum_{l=1}^{N} c_{lk} \lambda_{l,\min} \right] I_M \geq \left(\sum_{k=1}^{N} z_k \right) \min_{1 \leq k \leq N} \left\{ \sum_{l=1}^{N} c_{lk} \lambda_{l,\min} \right\} I_M \\ &= \frac{\mathbb{1}^T \Omega_0 A_2 \theta}{\theta^T \mathbb{1}} \cdot \min_{1 \leq k \leq N} \left\{ \sum_{l=1}^{N} c_{lk} \lambda_{l,\min} \right\} \cdot I_M \end{aligned} \tag{4.141}$$

Noting that the matrices Ω_0 and A_0 have nonnegative entries with some entries being positive, and that all entries of θ are positive, we have $(\mathbb{1}^T \Omega_0 A_2 \theta)/(\theta^T \mathbb{1}) > 0$. Furthermore, by Assumption 4.1, we know $\sum_{l=1}^{N} c_{lk} \lambda_{l,\min} > 0$ for each $k = 1, \ldots, N$. Therefore, we conclude that $E_{11} > 0$ and is invertible. Applying (4.139) to (4.136), we get

$$G_{12} = -E_{11}^{-1} E_{12} G_{22} \tag{4.142}$$

$$G_{22} = \left[I - J_0 \otimes I_M + \mu_{\max} (E_{22} - E_{21} E_{11}^{-1} E_{12}^T) \right]^{-1} \tag{4.143}$$

Substituting (4.143) into (4.138) leads to

$$\mathbb{1} \otimes w^o - w_\infty = \mu_{\max} (Y \otimes I_M) \begin{bmatrix} -E_{11}^{-1} E_{12} \\ I \end{bmatrix} G_{22} (Y_R A_2^T \Omega_0 C^T \otimes I_M) g^o$$

Substituting into $\lim_{\mu_{\max}\to 0} \|\mathbb{1}\otimes w^o - w_\infty\|/\mu_{\max}$, we get

$$\begin{aligned}
&\lim_{\mu_{\max}\to 0} \frac{\|\mathbb{1}\otimes w^o - w_\infty\|}{\mu_{\max}} \\
&= \lim_{\mu_{\max}\to 0} \left\| (Y\otimes I_M) \begin{bmatrix} -E_{11}^{-1}E_{12} \\ I \end{bmatrix} G_{22}(Y_R A_2^T \Omega_0 C^T \otimes I_M) g^o \right\|
\end{aligned}$$

Observe that the only term on the right-hand side of the above expression that depends on $\mu_{\max}$ is G_{22}. From its expression (4.143), we observe that, as $\mu_{\max}\to 0$, the matrix G_{22} tends to $(I - J_0\otimes I_M)^{-1}$, which is independent of $\mu_{\max}$. Therefore, the limit on the right-hand side is independent of $\mu_{\max}$.

4.C Block Maximum Norm of a Matrix

Consider a block matrix X with blocks of size $M\times M$ each. Its block maximum norm is defined as [127]:

$$\|X\|_{b,\infty} \triangleq \max_{x\neq 0} \frac{\|Xx\|_{b,\infty}}{\|x\|_{b,\infty}} \tag{4.144}$$

where the block maximum norm of a vector $x \triangleq \mathrm{col}\{x_1,\ldots,x_N\}$, formed by stacking N vectors of size M each on top of each other, is defined as [127]:

$$\|x\|_{b,\infty} \triangleq \max_{1\leq k\leq N} \|x_k\| \tag{4.145}$$

where $\|\cdot\|$ denotes the Euclidean norm of its vector argument.

Lemma 4.6 (Block maximum norm)**.** *If a block diagonal matrix*

$$X \triangleq \mathrm{diag}\{X_1,\ldots,X_N\} \in \mathbb{R}^{NM\times NM} \tag{4.146}$$

consists of N blocks along the diagonal with dimension $M \times M$ each, then the block maximum norm of X is bounded as

$$\|X\|_{b,\infty} \leq \max_{1\leq k\leq N} \|X_k\| \tag{4.147}$$

in terms of the 2-induced norms of $\{X_k\}$ (largest singular values). Moreover, if X is symmetric, then equality holds in (4.147).

Proof. Note that $Xx = \text{col}\{X_1x_1,\ldots,X_Nx_N\}$. Evaluating the block maximum norm of vector Xx leads to

$$\begin{aligned}\|Xx\|_{b,\infty} &= \max_{1\leq k\leq N} \|X_kx_k\| \\ &\leq \max_{1\leq k\leq N} \|X_k\| \cdot \|x_k\| \\ &\leq \max_{1\leq k\leq N} \|X_k\| \cdot \max_{1\leq k\leq N} \|x_k\| \end{aligned} \tag{4.148}$$

Substituting (4.148) and (4.145) into (4.144), we establish (4.147) as

$$\begin{aligned}\|X\|_{b,\infty} &\triangleq \max_{x\neq 0} \frac{\|Xx\|_{b,\infty}}{\|x\|_{b,\infty}} \\ &\leq \max_{x\neq 0} \frac{\max_{1\leq k\leq N} \|X_k\| \cdot \max_{1\leq k\leq N} \|x_k\|}{\max_{1\leq k\leq N} \|x_k\|} \\ &= \max_{1\leq k\leq N} \|X_k\| \end{aligned} \tag{4.149}$$

Next, we prove that, if all the diagonal blocks of X are symmetric, then equality should hold in (4.149). To do this, we only need to show that there exists an $x_0 \neq 0$, such that

$$\frac{\|Xx_0\|_{b,\infty}}{\|x_0\|_{b,\infty}} = \max_{1\leq k\leq N} \|X_k\| \tag{4.150}$$

which would mean that

$$\begin{aligned}\|X\|_{b,\infty} &\triangleq \max_{x\neq 0} \frac{\|Xx\|_{b,\infty}}{\|x\|_{b,\infty}} \\ &\geq \frac{\|Xx_0\|_{b,\infty}}{\|x_0\|_{b,\infty}} \\ &= \max_{1\leq k\leq N} \|X_k\| \end{aligned} \tag{4.151}$$

Then, combining inequalities (4.149) and (4.151), we would obtain desired equality that

$$\|X\|_{b,\infty} = \max_{1\leq k\leq N} \|X_k\| \tag{4.152}$$

when X is block diagonal and symmetric. Thus, without loss of generality, assume the maximum in (4.150) is achieved by X_1, i.e.,

$$\max_{1\leq k\leq N} \|X_k\| = \|X_1\|$$

For a symmetric X_k, its 2-induced norm $\|X_k\|$ (defined as the largest singular value of X_k) coincides with the spectral radius of X_k. Let λ_0 denote the eigenvalue of X_1 of largest magnitude, with the corresponding right eigenvector given by z_0. Then,

$$\max_{1\leq k\leq N} \|X_k\| = |\lambda_0|, \qquad X_1 z_0 = \lambda_0 z_0$$

We select $x_0 = \mathrm{col}\{z_0, 0, \ldots, 0\}$. Then, we establish (4.150) by:

$$\begin{aligned}\frac{\|Xx_0\|_{b,\infty}}{\|x_0\|_{b,\infty}} &= \frac{\|\mathrm{col}\{X_1 z_0, 0, \ldots, 0\}\|_{b,\infty}}{\|\mathrm{col}\{z_0, 0, \ldots, 0\}\|_{b,\infty}} \\ &= \frac{\|X_1 z_0\|}{\|z_0\|} = \frac{\|\lambda_0 z_0\|}{\|z_0\|} = |\lambda_0| = \max_{1\leq k\leq N} \|X_k\| \end{aligned}$$

□

4.D Stability of $\mathcal{B}$ and $\mathcal{F}$

Recall the definitions of the matrices $\mathcal{B}$ and $\mathcal{F}$ from (4.110) and (4.109):

$$\mathcal{B} = \mathcal{A}_2^T[I_{MN} - \mathcal{M}\mathcal{R}_\infty]\mathcal{A}_1^T \tag{4.153}$$

$$\begin{aligned}\mathcal{F} &= \big(\mathcal{A}_1[I_{MN} - \mathcal{M}\mathcal{R}_\infty]\mathcal{A}_2\big) \otimes \big(\mathcal{A}_1[I_{MN} - \mathcal{M}\mathcal{R}_\infty]\mathcal{A}_2\big) \\ &= \mathcal{B}^T \otimes \mathcal{B}^T\end{aligned} \tag{4.154}$$

From (4.153)–(4.154), we obtain (see Theorem 13.12 from [82, p.141]):

$$\rho(\mathcal{F}) = \rho(\mathcal{B}^T \otimes \mathcal{B}^T) = [\rho(\mathcal{B}^T)]^2 = [\rho(\mathcal{B})]^2 \tag{4.155}$$

where $\rho(\cdot)$ denotes the spectral radius of its matrix argument. Therefore, the stability of the matrix $\mathcal{F}$ is equivalent to the stability of the matrix $\mathcal{B}$, and we only need to examine the stability of $\mathcal{B}$. Now note that the block maximum norm (see the definition in Appendix 4.C) of the matrix $\mathcal{B}$ satisfies

$$\|\mathcal{B}\|_{b,\infty} \leq \|I_{MN} - \mathcal{M}\mathcal{R}_\infty\|_{b,\infty} \tag{4.156}$$

since the block maximum norms of $\mathcal{A}_1$ and $\mathcal{A}_2$ are one (see [127, p.4801]):

$$\big\|\mathcal{A}_1^T\big\|_{b,\infty} = 1, \qquad \big\|\mathcal{A}_2^T\big\|_{b,\infty} = 1 \tag{4.157}$$

Moreover, by noting that the spectral radius of a matrix is upper bounded by any matrix norm (Theorem 5.6.9, [68, p.297]) and that $I_{MN} - \mathcal{M}\mathcal{R}_\infty$ is symmetric

and block diagonal, we have

$$\rho(\mathcal{B}) \leq \|I_{MN} - \mathcal{M}\mathcal{R}_{\infty}\|_{b,\infty} = \rho(I_{MN} - \mathcal{M}\mathcal{R}_{\infty}) \tag{4.158}$$

Therefore, the stability of $\mathcal{B}$ is guaranteed by the stability of $I_{MN} - \mathcal{M}\mathcal{R}_{\infty}$. Next, we call upon the following lemma to bound $\|I_{MN} - \mathcal{M}\mathcal{R}_{\infty}\|_{b,\infty}$.

Lemma 4.7 (Norm of $I_{MN} - \mathcal{M}\mathcal{D}_{\infty}$)**.** *It holds that the matrix $\mathcal{R}_{\infty}$ defined in* (4.101) *satisfies*

$$\|I_{MN} - \mathcal{M}\mathcal{R}_{\infty}\|_{b,\infty} \leq \max_{1 \leq k \leq N} \gamma_k \tag{4.159}$$

where γ_k is defined in (4.39)*.*

Proof. Since $\mathcal{R}_{\infty}$ is block diagonal and symmetric, $I_{MN} - \mathcal{M}\mathcal{R}_{\infty}$ is also block diagonal with blocks $\{I_M - \mu_k \mathcal{R}_{k,\infty}\}$, where $\mathcal{R}_{k,\infty}$ denotes the kth diagonal block of $\mathcal{R}_{\infty}$. Then, from (4.147) in Lemma 4.6 in Appendix 4.C, it holds that

$$\|I_{MN} - \mathcal{M}\mathcal{R}_{\infty}\|_{b,\infty} = \max_{1 \leq k \leq N} \|I_M - \mu_k \mathcal{R}_{k,\infty}\| \tag{4.160}$$

By the definition of $\mathcal{R}_{\infty}$ in (4.101), and using condition (4.12) from Assumption 4.1, we have

$$\Big(\sum_{l=1}^{N} c_{l,k}\lambda_{l,\min}\Big) \cdot I_M \leq \mathcal{R}_{k,\infty} \leq \Big(\sum_{l=1}^{N} c_{l,k}\lambda_{l,\max}\Big) \cdot I_M$$

which implies that

$$I_M - \mu_k \mathcal{R}_{k,\infty} \geq \Big(1 - \mu_k \sum_{l=1}^{N} c_{l,k}\lambda_{l,\max}\Big) \cdot I_M \tag{4.161}$$

$$I_M - \mu_k \mathcal{R}_{k,\infty} \leq \left(1 - \mu_k \sum_{l=1}^{N} c_{l,k} \lambda_{l,\min}\right) \cdot I_M \tag{4.162}$$

Thus, $\|I_M - \mu_k \mathcal{R}_{k,\infty}\| \leq \gamma_k$. Substituting into (4.160), we get (4.159). □

Substituting (4.159) into (4.158), we get:

$$\rho(\mathcal{B}) \leq \max_{1 \leq k \leq N} \gamma_k \tag{4.163}$$

As long as $\max_{1 \leq k \leq N} \gamma_k < 1$, then all the eigenvalues of $\mathcal{B}$ will lie within the unit circle. By the definition of γ_k in (4.39), this is equivalent to requiring

$$|1 - \mu_k \sigma_{k,\max}| < 1, \qquad |1 - \mu_k \sigma_{k,\min}| < 1$$

for $k = 1, \ldots, N$, where $\sigma_{k,\max}$ and $\sigma_{k,\min}$ are defined in (4.40). These conditions are satisfied if we choose μ_k such that

$$0 < \mu_k < 2/\sigma_{k,\max}, \qquad k = 1, \ldots, N \tag{4.164}$$

which is obviously guaranteed for sufficiently small step-sizes (and also by condition (4.64)).

CHAPTER 5

Transient Analysis

5.1 Introduction

In Chapter 4, we analyzed the stability and performance of the diffusion algorithm under the assumption that *each cost function $J_k(w)$ is strongly convex*. In this chapter, and in Chapter 6, we relax this assumption and consider a general class of distributed strategies, which includes diffusion strategies [26,34,36,42,58,89,91,115,146] and consensus strategies [48,75–77,97,98,137] as special cases. Both classes of algorithms involve self-learning and social-learning steps. During self-learning, each agent updates its state using its local data. During social learning, each agent aggregates information from its neighbors. A useful feature that results from these localized interactions is that the network ends up exhibiting global patterns of behavior. For example, in distributed estimation and learning, each agent is able to attain the performance of centralized solutions by relying solely on local cooperation [75, 146]. We now study the resulting global learning behavior by addressing four important questions: (i) where does the distributed algorithm converge to? (ii) when does it converge? (iii) how fast does it converge? and (iv) how close does it converge to the intended point? We answer questions (i)–(iii) in this chapter and question (iv) in Chapter 6. We study these four questions by dissecting and characterizing the learning dynamics of the network in some great detail. An interesting con-

clusion that follows from our analysis is that the learning curve of a multi-agent system will be shown to exhibit *three* different phases. In the first phase (Transient Phase I), the convergence rate of the network is determined by the second largest eigenvalue of the combination matrix in magnitude, which is related to the degree of network connectivity. In the second phase (Transient Phase II), the convergence rate is determined by the entries of the right-eigenvector of the combination matrix corresponding to the eigenvalue at one. And, in the third phase (the steady-state phase) the mean-square performance of the algorithm turns out to depend on this same right-eigenvector in a revealing way. Even more surprisingly, we shall discover that the agents have the same learning behavior starting at Transient Phase II, and are able to achieve a performance level that matches that of a fully connected network or a centralized stochastic-gradient strategy. Actually, we shall show that the consensus and diffusion strategies can be represented as perturbed versions of a centralized *reference* recursion in a certain transform domain. We quantify the effect of the perturbations and establish the aforementioned properties for the various phases of the learning behavior of the networks. The results will reveal the manner by which the network topology influences performance in some unexpected ways.

There have been of course many insightful works in the literature on distributed strategies and their convergence behavior. In Sections 5.2.2 and 5.4.1 further ahead, we explain in what ways the current chapter extends these earlier investigations and what novel contributions this work leads to. In particular, it will be seen that several new insights are discovered that clarify how distributed networks learn. For the time being, in these introductory remarks, we limit ourselves to mentioning one important aspect of our development. Most prior studies on distributed optimization and estimation tend to focus on the performance and convergence of the algorithms under *diminishing* step-size con-

ditions [13, 48, 75–77, 84, 97, 109, 125, 137], or on convergence under deterministic conditions on the data [97]. This is perfectly fine for applications involving *static* optimization problems where the objective is to locate the fixed optimizer of some aggregate cost function of interest. In this thesis, however, we examine the learning behavior of distributed strategies under *constant* step-size conditions. This is because constant step-sizes are necessary to enable continuous adaptation, learning, and tracking in the presence of streaming data and drifting conditions. These features would enable the algorithms to perform well even when the location of the optimizer drifts with time. Nevertheless, the use of constant step-sizes enriches the dynamics of (stochastic-gradient) distributed algorithms in that the gradient update term does not die out with time anymore, in clear contrast to the diminishing step-size case where the influence of the gradient term is annihilated over time due to the decaying value of the step-size parameter. For this reason, more care is needed to examine the learning behavior of distributed strategies in the constant step-size regime since their updates remain continually active and the effect of gradient noise is always present. This work also generalizes and extends in non-trivial ways the studies in the previous chapters. For example, while Chapter 2 assumed that the individual costs of all agents have the *same* minimizer, and Chapter 4 assumed that each of these individual costs is strongly convex, these requirements are not needed in the current chapter and the next chapter: individual costs can have distinct minimizers and they do not even need to be convex (see the discussion after expression (5.32)). This fact widens significantly the class of distributed learning problems that are covered by our framework. Moreover, the network behavior is studied under less restrictive assumptions and for broader scenarios, including a close study of the various phases of evolution during the transient phase of the learning process. We also study a larger class of distributed strategies that includes diffusion and consensus

strategies as special cases.

To examine the learning behavior of adaptive networks under broader and more relaxed conditions than usual, we pursue a new analysis route by introducing a *reference* centralized recursion and by studying the perturbation of the diffusion and consensus strategies relative to this centralized solution over time. Insightful new results are obtained through this perturbation analysis. For example, we are now able to examine closely *both* the transient phase behavior and the steady-state phase behavior of the learning process and to explain how behavior in these two stages relate to the behavior of the centralized solution (see Fig. 5.2 further ahead). Among several other results, the mean-square-error expression (5.45) derived later in Chapter 6 following some careful analysis, which builds on the results of this chapter, is one of the new (compact and powerful) insights; it reveals how the performance of each agent is closely related to that of the centralized stochastic approximation strategy — see the discussion right after (5.45). As the reader will ascertain from the derivations in the appendices, arriving at these conclusions for a broad class of distributed strategies and under weaker conditions than usual is demanding and necessitates a careful study of the evolution of the error dynamics over the network and its stability. When all is said and done, Chapters 5–6 lead to several novel insights into the learning behavior of adaptive networks. The following presentation in this chapter is based on [37].

5.2 Problem Formulation

5.2.1 Distributed Strategies: Consensus and Diffusion

We consider a connected network of N agents that are linked together through a topology — see Fig. 5.1. Each agent k implements a distributed algorithm of

the following form to update its state vector from $\boldsymbol{w}_{k,i-1}$ to $\boldsymbol{w}_{k,i}$:

$$\boldsymbol{\phi}_{k,i-1} = \sum_{l=1}^{N} a_{1,lk}\boldsymbol{w}_{l,i-1} \tag{5.1}$$

$$\boldsymbol{\psi}_{k,i} = \sum_{l=1}^{N} a_{0,lk}\boldsymbol{\phi}_{l,i-1} - \mu_k\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{\phi}_{k,i-1}) \tag{5.2}$$

$$\boldsymbol{w}_{k,i} = \sum_{l=1}^{N} a_{2,lk}\boldsymbol{\psi}_{l,i} \tag{5.3}$$

where $\boldsymbol{w}_{k,i} \in \mathbb{R}^M$ is the state of agent k at time i, usually an estimate for the solution of some optimization problem, $\boldsymbol{\phi}_{k,i-1} \in \mathbb{R}^M$ and $\boldsymbol{\psi}_{k,i} \in \mathbb{R}^M$ are intermediate variables generated at node k before updating to $\boldsymbol{w}_{k,i}$, μ_k is a non-negative constant step-size parameter used by node k, and $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ is an $M \times 1$ update vector function at node k. In deterministic optimization problems, the update vectors $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ can be the gradient or Newton steps associated with the cost functions [97]. On the other hand, in stocastic approximation problems, such as adaptation, learning and estimation problems [26, 34, 36, 42, 48, 49, 58, 75, 77, 89, 91, 109, 115, 125, 130, 137, 146], the update vectors are usually computed from realizations of data samples that arrive sequentially at the nodes. In the stochastic setting, the quantities appearing in (5.1)–(5.3) become random and we use boldface letters to highlight their stochastic nature. In Example 5.1 below, we illustrate choices for $\hat{\boldsymbol{s}}_{k,i}(w)$ in different contexts.

The combination coefficients $a_{1,lk}$, $a_{0,lk}$ and $a_{2,lk}$ in (5.1)–(5.3) are nonnegative weights that each node k assigns to the information arriving from node l; these coefficients are required to satisfy:

$$\sum_{l=1}^{N} a_{1,lk} = 1, \quad \sum_{l=1}^{N} a_{0,lk} = 1, \quad \sum_{l=1}^{N} a_{2,lk} = 1 \tag{5.4}$$

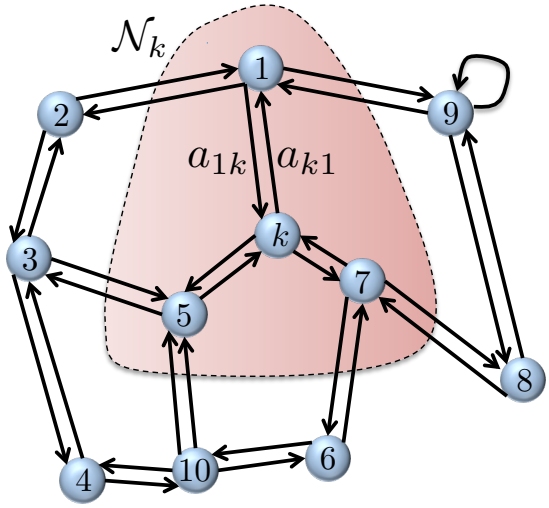


Figure 5.1: A network representing a multi-agent system. The set of all agents that can communicate with node k is denoted by $\mathcal{N}_k$.The edge linking any two agents is represented by two directed arrows to emphasize that information can flow in both directions.

$$a_{1,lk} \geq 0, \qquad a_{0,lk} \geq 0, \qquad a_{2,lk} \geq 0 \tag{5.5}$$

$$a_{1,lk} = a_{2,lk} = a_{0,lk} = 0, \qquad \text{if } l \notin \mathcal{N}_k \tag{5.6}$$

Observe from (5.6) that the combination coefficients are zero if $l \notin \mathcal{N}_k$, where $\mathcal{N}_k$ denotes the set of neighbors of node k. Therefore, each summation in (5.1)–(5.3) is actually confined to the neighborhood of node k. In algorithm (5.1)–(5.3), each node k first combines the states $\{\boldsymbol{w}_{l,i-1}\}$ from its neighbors and updates $\boldsymbol{w}_{k,i-1}$ to the intermediate variable $\boldsymbol{\phi}_{k,i-1}$. Then, the $\{\boldsymbol{\phi}_{l,i-1}\}$ from the neighbors are aggregated and updated to $\boldsymbol{\psi}_{k,i}$ along the opposite direction of $\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{\phi}_{k,i-1})$. Finally, the intermediate estimators $\{\boldsymbol{\psi}_{l,i}\}$ from the neighbors are combined to generate the new state $\boldsymbol{w}_{k,i}$ at node k.

Example 5.1. The distributed algorithm (5.1)–(5.3) can be applied to optimize

aggregate costs of the following form:

$$J^{\text{glob}}(w) = \sum_{k=1}^{N} J_k(w) \tag{5.7}$$

or to find Pareto-optimal solutions to multi-objective optimization problems, such as:

$$\min_{w} \{J_1(w), \ldots, J_N(w)\} \tag{5.8}$$

where $J_k(w)$ is an individual convex cost associated with each agent k. Optimization problems of the form (5.7)–(5.8) arise in various applications — see [26, 34, 36, 42, 48, 49, 51, 52, 58, 75–77, 84, 85, 89, 91, 96–98, 109, 115, 125, 130, 136, 137, 146]. Depending on the context, the update vector $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ may be chosen in different ways:

- In deterministic optimization problems, the expressions for $\{J_k(w)\}$ are known and the update vector $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ at node k is chosen as the deterministic gradient (column) vector $\nabla_w J_k(\cdot)$.

- In distributed estimation and learning, the individual cost function at each node k is usually selected as the expected value of some loss function $Q_k(\cdot,\cdot)$, i.e., $J_k(w) = \mathbb{E}\{Q_k(w, \boldsymbol{x}_{k,i})\}$ [34], where the expectation is with respect to the randomness in the data samples $\{\boldsymbol{x}_{k,i}\}$ collected at node k at time i. The exact expression for $\nabla_w J_k(w)$ is usually unknown since the probability distribution of the data is not known beforehand. In these situations, the update vector $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ is chosen as an instantaneous approximation for the true gradient vector, such as, $\hat{\boldsymbol{s}}_{k,i}(\cdot) = \widehat{\nabla_w J_k}(\cdot) = \nabla_w Q_k(\cdot, \boldsymbol{x}_{k,i})$. Note that the update vector $\hat{\boldsymbol{s}}_{k,i}(w)$ is now evaluated from the random data sample

$\boldsymbol{x}_{k,i}$. Therefore, it is also random and time dependent.

The update vectors $\{\hat{\boldsymbol{s}}_{k,i}(\cdot)\}$ may not necessarily be the gradients of cost functions or their stochastic approximations. They may take other forms for different reasons. For example, in [75], a certain gain matrix K is multiplied to the left of the stochastic gradient vector $\widehat{\nabla_w J_k}(\cdot)$ to make the estimator asymptotically efficient for a linear observation model. □

Returning to the general distributed strategy (5.1)–(5.3), we note that it can be specialized into various useful algorithms. We let A_1, A_0 and A_2 denote the $N \times N$ matrices that collect the coefficients $\{a_{1,lk}\}$, $\{a_{0,lk}\}$ and $\{a_{2,lk}\}$. Then, condition (5.4) is equivalent to

$$A_1^T \mathbb{1} = \mathbb{1}, \quad A_0^T \mathbb{1} = \mathbb{1}, \quad A_2^T \mathbb{1} = \mathbb{1} \tag{5.9}$$

where $\mathbb{1}$ is the $N \times 1$ vector with all its entries equal to one. Condition (5.9) means that the matrices $\{A_0, A_1, A_2\}$ are left-stochastic (i.e., the entries on each of their columns add up to one). Different choices for A_1, A_0 and A_2 correspond to different distributed strategies, as summarized in Table 5.1. Specifically, the traditional consensus [48, 75–77, 97, 98, 137] and diffusion (ATC and CTA) [26, 34, 36, 42, 89, 91, 115, 146] algorithms with *constant* step-sizes are given by the following iterations:

$$\text{Consensus}: \begin{cases} \boldsymbol{\phi}_{k,i-1} = \displaystyle\sum_{l \in \mathcal{N}_k} a_{0,lk} \boldsymbol{w}_{l,i-1} \\ \boldsymbol{w}_{k,i} = \boldsymbol{\phi}_{k,i-1} - \mu_k \hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}_{k,i-1}) \end{cases} \tag{5.10}$$

Table 5.1: Different choices for A_1, A_0 and A_2 correspond to different distributed strategies.

Distributed Strategeis	A_1	A_0	A_2	$A_1A_0A_2$
Consensus	I	A	I	A
ATC diffusion	I	I	A	A
CTA diffusion	A	I	I	A

$$\text{CTA diffusion}: \begin{cases} \boldsymbol{\phi}_{k,i-1} = \displaystyle\sum_{l\in\mathcal{N}_k} a_{1,lk}\boldsymbol{w}_{l,i-1} \\ \boldsymbol{w}_{k,i} = \boldsymbol{\phi}_{k,i-1} - \mu_k \hat{\boldsymbol{s}}_{k,i}(\boldsymbol{\phi}_{k,i-1}) \end{cases} \tag{5.11}$$

$$\text{ATC diffusion}: \begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu_k \hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{w}_{k,i} = \displaystyle\sum_{l\in\mathcal{N}_k} a_{2,lk}\boldsymbol{\psi}_{l,i} \end{cases} \tag{5.12}$$

Therefore, the convex combination steps appear in different locations in the consensus and diffusion implementations. For instance, observe that the consensus strategy (5.10) evaluates the update direction $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ at $\boldsymbol{w}_{k,i-1}$, which is the estimator *prior* to the aggregation, while the diffusion strategy (5.11) evaluates the update direction at $\boldsymbol{\phi}_{k,i-1}$, which is the estimator *after* the aggregation. In our analysis, we will proceed with the general form (5.1)–(5.3) to study all three schemes, and other possibilities, within a unifying framework.

We observe that there are two types of learning processes involved in the dynamics of each agent k: (i) self-learning in (5.2) from locally sensed data and (ii) social learning in (5.1) and (5.3) from neighbors. All nodes implement the same self- and social learning structure. As a result, the learning dynamics of all nodes in the network are coupled; knowledge exploited from local data at node k will be propagated to its neighbors and from there to their neighbors in

a diffusive learning process. It is expected that some global performance pattern will emerge from these localized interactions in the multi-agent system. In this chapter and the following Chapter 6, we address the following questions:

- Limit point: where does each state $\boldsymbol{w}_{k,i}$ converge to?
- Stability: under which conditions does convergence occur?
- Learning rate: how fast does convergence occur?
- Performance: how close does $\boldsymbol{w}_{k,i}$ get to the limit point?

We address the first three questions in this chapter, and examine the last question pertaining to performance in Chapter 6. We address the four questions by characterizing analytically the learning dynamics of the network to reveal the global behavior that emerges in the small step-size regime. The answers to these questions will provide useful and novel insights about how to tune the algorithm parameters in order to reach desired performance levels — see Sec. 6.6.

5.2.2 Relation to Prior Work

In comparison with the existing literature [13,21,48,71,75–77,84,97,109,125,126, 137], it is worth noting that most prior studies on distributed optimization algorithms focus on studying their performance and convergence under *diminishing* step-size conditions and for *doubly-stochastic* combination policies (i.e., matrices for which the entries on each of their columns *and* on each of their rows add up to one). These are of course useful conditions, especially when the emphasis is on solving *static* optimization problems. We focus instead on the case of *constant* step-sizes because, as explained earlier, they enable continuous adaptation and learning under drifting conditions; in contrast, diminishing step-sizes turn

off learning once they approach zero. By using constant step-sizes, the resulting algorithms are able to track *dynamic* solutions that may slowly drift as the underlying problem conditions change. Moreover, we do not limit the combination policies to be doubly-stochastic; we only require condition (5.9). It turns out that left-stochastic matrices lead to superior mean-square error performance (see, e.g., expression (6.66) and also [146]). Furthermore, constant step-sizes and left-stochastic combination policies enrich the learning dynamics of the network in interesting ways, as we are going to discover. In particular, under these conditions, we derive an interesting result that reveals how the topology of the network determines the limit point of the distributed algorithm. We will show that the combination weights steer the convergence point away from the expected solution and towards any of many possible Pareto optimal solutions. This is in contrast to commonly-used doubly-stochastic combination policies where the limit point of the network is fixed and cannot be changed regardless of the topology. We will show that the limit point is determined by the right eigenvector that is associated with the eigenvalue at one for the matrix product $A_1 A_0 A_2$. We will also be able to characterize in Chapter 6 how close each agent in the network gets to this limit point and to explain how the limit point plays the role of a Pareto optimal solution for a suitably defined aggregate cost function.

5.3 Modeling Assumptions

In this section, we collect the assumptions and definitions that are used in the analysis and explain why they are justified and how they relate to similar assumptions used in several prior studies in the literature. As the discussion will reveal, in most cases, the assumptions that we adopt here are relaxed (i.e., weaker) versions than conditions used before in the literature such as

in [11, 11, 34, 36, 52, 75, 77, 84, 97, 105, 109, 125, 137]. We do so in order to analyze the learning behavior of networks under conditions that are similar to what is normally assumed in the prior art, albeit ones that are generally less restrictive.

Assumption 5.1 (Strongly-connected network)**.** *The $N \times N$ matrix product $A \triangleq A_1 A_0 A_2$ is assumed to be a primitive left-stochastic matrix, i.e., $A^T \mathbb{1} = \mathbb{1}$ and there exists a finite integer j_o such that all entries of A^{j_o} are strictly positive.* □

This condition is satisfied for most networks and is not restrictive. Let $A = [a_{lk}]$ denote the entries of A. Assumption 5.1 is automatically satisfied if the product A corresponds to a connected network and there exists at least one $a_{kk} > 0$ for some node k (i.e., at least one node with a nontrivial self-loop) [115]. It then follows from the Perron-Frobenius Theorem [68] that the matrix $A_1 A_0 A_2$ has a single eigenvalue at one of multiplicity one and all other eigenvalues are strictly less than one in magnitude, i.e.,

$$1 = \lambda_1(A) > |\lambda_2(A)| \geq \cdots \geq |\lambda_N(A)| \tag{5.13}$$

Obviously, $\mathbb{1}^T$ is a left eigenvector for $A_1 A_0 A_2$ corresponding to the eigenvalue at one. Let θ denote the right eigenvector corresponding to the eigenvalue at one and whose entries are normalized to add up to one, i.e.,

$$A\theta = \theta, \qquad \mathbb{1}^T \theta = 1 \tag{5.14}$$

Then, the Perron-Frobenius Theorem further ensures that all entries of θ satisfy $0 < \theta_k < 1$. Note that, unlike [75,77,97,109,125,137], we do not require the matrix $A_1 A_0 A_2$ to be doubly-stochastic (in which case θ would be $\mathbb{1}/N$ and, therefore, all

its entries will be identical to each other). Instead, we will study the performance of the algorithms in the context of general left-stochastic matrices $\{A_1, A_0, A_2\}$ and we will examine the influence of (the generally non-equal entries of) θ on both the limit point and performance of the network.

Definition 5.1 (Step-sizes)**.** *Without loss of generality, we express the step-size at each node k as $\mu_k = \mu_{\max}\beta_k$, where $\mu_{\max} \triangleq \max\{\mu_k\}$ is the largest step-size, and $0 \leq \beta_k \leq 1$. We assume $\beta_k > 0$ for at least one k. Thus, observe that we are allowing the possibility of zero step-sizes by some of the agents.*

□

Definition 5.2 (Useful vectors)**.** *Let π and p be the following $N \times 1$ vectors:*

$$\pi \triangleq A_2\theta \tag{5.15}$$

$$p \triangleq \mathrm{col}\{\pi_1\beta_1, \ldots, \pi_N\beta_N\} \tag{5.16}$$

where π_k is the kth entry of the vector π. □

The vector p will play a critical role in the performance of the distributed strategy (5.1)–(5.3). Furthermore, we introduce the following assumptions on the update vectors $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ in (5.1)–(5.3).

Assumption 5.2 (Update vector: Randomness)**.** *There exists an $M \times 1$ deterministic vector function $s_k(w)$ such that, for all $M \times 1$ vectors $\boldsymbol{w}$ in the filtration $\mathcal{F}_{i-1}$ generated by the past history of iterates $\{\boldsymbol{w}_{k,j}\}$ for $j \leq i-1$ and all k, it holds that*

$$\mathbb{E}\,\{\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w})|\mathcal{F}_{i-1}\} = s_k(\boldsymbol{w}) \tag{5.17}$$

for all i, k. Furthermore, there exist $\alpha \geq 0$ and $\sigma_v^2 \geq 0$ such that for all i, k and $\boldsymbol{w} \in \mathcal{F}_{i-1}$:

$$\mathbb{E}\left\{\|\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}) - s_k(\boldsymbol{w})\|^2 \,\middle|\, \mathcal{F}_{i-1}\right\} \leq \alpha \cdot \|\boldsymbol{w}\|^2 + \sigma_v^2 \tag{5.18}$$

□

Condition (5.18) requires the conditional variance of the random update direction $\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w})$ to be bounded by the square-norm of $\boldsymbol{w}$. Condition (5.21) is a generalized version of Assumption 4.2 from Chapter 4; it is also a generalization of the assumptions from [11, 105, 109], where $\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w})$ was instead modeled as the following perturbed version of the true gradient vector:

$$\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}) = \widehat{\nabla_w J_k}(\boldsymbol{w}) = \nabla_w J_k(\boldsymbol{w}) + \boldsymbol{v}_{k,i}(\boldsymbol{w}) \tag{5.19}$$

with $s_k(\boldsymbol{w}) = \nabla_w J_k(\boldsymbol{w})$, in which case conditions (5.17)–(5.18) translate into the following requirements on the gradient noise $\boldsymbol{v}_{k,i}(\boldsymbol{w})$:

$$\mathbb{E}\left\{\boldsymbol{v}_{k,i}(\boldsymbol{w}) | \mathcal{F}_{i-1}\right\} = 0 \qquad \text{(zero mean)} \tag{5.20}$$

$$\mathbb{E}\left\{\|\boldsymbol{v}_{k,i}(\boldsymbol{w})\|^2 \,\middle|\, \mathcal{F}_{i-1}\right\} \leq \alpha \cdot \|\boldsymbol{w}\|^2 + \sigma_v^2 \tag{5.21}$$

In Example 4.1 of Chapter 4, we explained how these conditions are satisfied automatically in the context of mean-square-error adaptation over networks. Assumption 5.2 given by (5.17)–(5.18) is more general than (5.20)–(5.21) because we are allowing the update vector $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ to be constructed in forms other than (5.19). Furthermore, Assumption (5.21) is also more relaxed than the following

variant used in [11, 105]:

$$\mathbb{E}\left\{\|\boldsymbol{v}_{k,i}(\boldsymbol{w})\|^2 \,\middle|\, \mathcal{F}_{i-1}\right\} \leq \alpha\cdot\|\nabla_w J_k(\boldsymbol{w})\|^2+\sigma_v^2 \tag{5.22}$$

This is because (5.22) implies a condition of the form (5.21). Indeed, note that

$$\begin{aligned}
&\mathbb{E}\left\{\|\boldsymbol{v}_{k,i}(\boldsymbol{w})\|^2 \,\middle|\, \mathcal{F}_{i-1}\right\} \\
&\quad= \alpha\cdot\|\nabla_w J_k(\boldsymbol{w}) - \nabla_w J_k(0) + \nabla_w J_k(0)\|^2+\sigma_v^2 \\
&\quad\overset{(a)}{\leq} 2\alpha\cdot\|\nabla_w J_k(\boldsymbol{w}) - \nabla_w J_k(0)\|^2 + 2\alpha\|\nabla_w J_k(0)\|^2+\sigma_v^2 \\
&\quad\overset{(b)}{\leq} 2\alpha\lambda_U^2\cdot\|\boldsymbol{w}\|^2 + 2\alpha\|\nabla_w J_k(0)\|^2+\sigma_v^2 \\
&\quad\triangleq \alpha'\cdot\|\boldsymbol{w}\|^2 + \sigma_{v'}^2
\end{aligned} \tag{5.23}$$

where step (a) uses the relation $\|x+y\|^2 \leq 2\|x\|^2 + 2\|y\|^2$, and step (b) used (5.24) to be assumed next.

Assumption 5.3 (Update vector: Lipschitz)**.** *There exists a nonnegative λ_U such that for all $x, y \in \mathbb{R}^M$ and all k:*

$$\|s_k(x) - s_k(y)\| \leq \lambda_U\cdot\|x-y\| \tag{5.24}$$

where the subscript "U" in λ_U means "upper bound". □

A similar assumption to (5.24) was used before in the literature for the model (5.19) by requiring the gradient vector of the individual cost functions $J_k(w)$ to be Lipschitz [11, 52, 84, 105, 137]. Again, condition (5.24) is more general because we are not limiting the construction of the update direction to (5.19).

Assumption 5.4 (Update vector: Strong monotonicity)**.** *Let p_k denote the kth*

entry of the vector p defined in (5.16). *There exists $\lambda_L > 0$ such that for all $x, y \in \mathbb{R}^M$:*

$$(x-y)^T \cdot \sum_{k=1}^{N} p_k \Big[s_k(x) - s_k(y) \Big] \geq \lambda_L \cdot \|x-y\|^2 \tag{5.25}$$

where the subscript "L" in λ_L means "lower bound". □

Remark 5.1. Applying the Cauchy-Schwartz inequality [68, p.15] to the left-hand side of (5.25) and using (5.24), we deduce the following relation between λ_L and λ_U:

$$\lambda_U \cdot \|p\|_1 \geq \lambda_L \tag{5.26}$$

where $\|\cdot\|_1$ denotes the $1-$norm of the vector argument. □

The following lemma gives the equivalent forms of Assumptions 5.3–5.4 when the $\{s_k(w)\}$ happen to be differentiable.

Lemma 5.1 (Equivalent conditions on update vectors)**.** *Suppose $\{s_k(w)\}$ are differentiable in an open set $\mathcal{S} \subseteq \mathbb{R}^M$. Then, having conditions* (5.24) *and* (5.25) *hold on $\mathcal{S}$ is equivalent to the following conditions, respectively,*

$$\|\nabla_{w^T} s_k(w)\| \leq \lambda_U \tag{5.27}$$

$$\frac{1}{2}[H_c(w) + H_c^T(w)] \geq \lambda_L \cdot I_M \tag{5.28}$$

for any $w \in \mathcal{S}$, where $\|\cdot\|$ denotes the $2-$ induced norm (largest singular value) of its matrix argument and

$$H_c(w) \triangleq \sum_{k=1}^{n} p_k \nabla_{w^T} s_k(w) \tag{5.29}$$

Proof. See Appendix 5.A. □

Since in Assumptions 5.3–5.4 we require conditions (5.24) and (5.25) to be hold over the entire $\mathbb{R}^M$, then the equivalent conditions (5.27)–(5.28) will need to hold over the entire $\mathbb{R}^M$ when the $\{s_k(w)\}$ are differentiable. In the context of distributed optimization problems of the form (5.7)–(5.8) with twice-differentiable $J_k(w)$, where the stochastic gradient vectors are constructed as in (5.19), Lemma 5.1 implies that the above Assumptions 5.3–5.4 are equivalent to the following conditions on the Hessian matrix of each $J_k(w)$ [105, p.10]:

$$\left\|\nabla_w^2 J_k(w)\right\| \leq \lambda_U \tag{5.30}$$

$$\sum_{k=1}^{N} p_k \nabla_w^2 J_k(w) \geq \lambda_L I_M > 0 \tag{5.31}$$

Condition (5.31) is in turn equivalent to requiring the following weighted sum of the individual cost functions $\{J_k(w)\}$ to be strongly convex:

$$J^{\text{glob},\star}(w) \triangleq \sum_{k=1}^{N} p_k J_k(w) \tag{5.32}$$

We note that strong convexity conditions are prevalent in many studies on optimization techniques in the literature. For example, each of the individual costs $J_k(w)$ is assumed to be stronlgy convex in [125] in order to derive upper bounds on the limit superior ("lim sup") of the mean-square-error of the estimates $\boldsymbol{w}_{k,i}$ or the expected value of the cost function at $\boldsymbol{w}_{k,i}$. In comparison, the framework in this work does not require the individual costs to be strongly convex or even convex. Actually, some of the costs $\{J_k(w)\}$ can be non-convex as long as the aggregate cost (5.32) remains strongly convex. Such relaxed assumptions on the individual costs introduce challenges into the analysis, and we need to develop a

systematic approach to characterize the limiting behavior of adaptive networks under such less restrictive conditions.

Assumption 5.5 (Jacobian matrix: Lipschitz)**.** *Let w^o denote the limit point of the distributed strategy* (5.1)–(5.3)*, which is defined further ahead as the unique solution to* (5.35)*. Then, in a small neighborhood around w^o, we assume that $s_k(w)$ is differentiable with respect to w and satisfies*

$$\|\nabla_{w^T} s_k(w^o + \delta w) - \nabla_{w^T} s_k(w^o)\| \leq \lambda_H \cdot \|\delta w\| \tag{5.33}$$

for all $\|\delta w\| \leq r_H$ for some small r_H, and where λ_H is a nonnegative number independent of δw and w^o.

□

In the context of distributed optimization problems of the form (5.7)–(5.8) with twice-differentiable $J_k(w)$, where the stochastic gradient vectors are constructed as in (5.19), the above Assumption translates into the following Lipschitz Hessian condition:

$$\|\nabla_w^2 J_k(w^o + \delta w) - \nabla_w^2 J_k(w^o)\| \leq \lambda_H \cdot \|\delta w\| \tag{5.34}$$

Condition (5.33) is useful when we examine the convergence rate of the algorithm later in this article. It is also useful in deriving the steady-state mean-square-error expression (5.45) in Chapter 6.

5.4 Learning Behavior

5.4.1 Overview of Main Results

Before we proceed to the formal analysis, we first give a brief overview of the main results that we are going to establish in this chapter on the learning behavior of the distributed strategies (5.1)–(5.3) for sufficiently small step-sizes. The first major conclusion is that for general *left-stochastic* matrices $\{A_1, A_0, A_2\}$, the agents in the network will have their estimators $\boldsymbol{w}_{k,i}$ converge, in the mean-square-error sense, to the *same* vector w^o that corresponds to the unique solution of the following algebraic equation:

$$\sum_{k=1}^{N} p_k s_k(w) = 0 \tag{5.35}$$

For example, in the context of distributed optimization problems of the form (5.7), this result implies that for left-stochastic matrices $\{A_1, A_0, A_2\}$, the distributed strategies represented by (5.1)–(5.3) will *not* converge to the global minimizer of the original aggregate cost (5.7), which is the unique solution to the alternative algebraic equation

$$\sum_{k=1}^{N} \nabla_w J_k(w) = 0 \tag{5.36}$$

Instead, these distributed solutions will converge to the global minimizer of the *weighted* aggregate cost $J^{\mathrm{glob},\star}(w)$ defined by (5.32) in terms of the entries p_k, i.e., to the unique solution of

$$\sum_{k=1}^{N} p_k \nabla_w J_k(w) = 0 \tag{5.37}$$

Result (5.35) also means that the distributed strategies (5.1)–(5.3) converge to a Pareto optimal solution of the multi-objective problem (5.8); one Pareto solution for each selection of the topology parameters $\{p_k\}$. The distinction between the aggregate costs $J^{\rm glob}(w)$ and $J^{\rm glob,\star}(w)$ does not appear in earlier studies on distributed optimization [75, 77, 97, 109, 125, 137] mainly because these studies focus on *doubly-stochastic* combination matrices, for which the entries $\{p_k\}$ will all become equal to each other for uniform step-sizes $\mu_k \equiv \mu$ or $\mu_k(i) \equiv \mu(i)$. In that case, the minimizations of (5.7) and (5.32) become equivalent and the solution of (5.36) and (5.37) would then coincide. In other words, regardless of the choice of the doubly stochastic combination weights, when the $\{p_k\}$ are identical, the limit point will be unique and correspond to the solution of

$$\sum_{k=1}^{N} s_k(w) = 0 \tag{5.38}$$

In contrast, result (5.35) shows that left-stochastic combination policies add more flexibility into the behavior of the network. By selecting different combination weights, or even different topologies, the entries $\{p_k\}$ can be made to change and the limit point can be steered towards other desired Pareto optimal solutions.

The second major conclusion of the paper is that we will show in (5.145) further ahead that there always exist sufficiently small step-sizes such that the learning process over the network is mean-square stable. This means that the weight error vectors relative to w^o will satisfy

$$\limsup_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = O(\mu_{\max}) \tag{5.39}$$

so that the steady-state mean-square-error at each agent will be of the order of $O(\mu_{\max})$.

The third major conclusion of our analysis is that we will show that, during the convergence process towards the limit point w^o, the learning curve at each agent exhibits *three* distinct phases: Transient Phase I, Transient Phase II, and Steady-State Phase. These phases are illustrated in Fig. 5.2 and they are interpreted as follows. Let us first introduce a *reference* (centralized) procedure that is described by the following centralized-type recursion:

$$\bar{w}_{c,i} = \bar{w}_{c,i-1} - \mu_{\max} \sum_{k=1}^{N} p_k s_k(\bar{w}_{c,i-1}) \tag{5.40}$$

which is initialized at

$$\bar{w}_{c,0} = \sum_{k=1}^{N} \theta_k w_{k,0} \tag{5.41}$$

where θ_k is the kth entry of the eigenvector θ, $\mu_{\max}$, and $\{p_k\}$ are defined in Definitions 5.1–5.2, $w_{k,0}$ is the initial value of the distributed strategy at agent k, and $\bar{w}_{c,i}$ is an $M \times 1$ vector generated by the reference recursion (5.40). The three phases of the learning curve will be shown to have the following features:

- **Transient Phase I:**

  If agents are initialized at different values, then the estimates of the various agents will initially evolve in such a way to make each $\boldsymbol{w}_{k,i}$ get closer to the reference recursion $\bar{w}_{c,i}$. The rate at which the agents approach $\bar{w}_{c,i}$ will be determined by $|\lambda_2(A)|$, the second largest eigenvalue of A in magnitude. If the agents are initialized at the same value, say, e.g., $\boldsymbol{w}_{k,0} = 0$, then the learning curves start at Transient Phase II directly.

- **Transient Phase II:**

  In this phase, the trajectories of all agents are uniformly close to the tra-

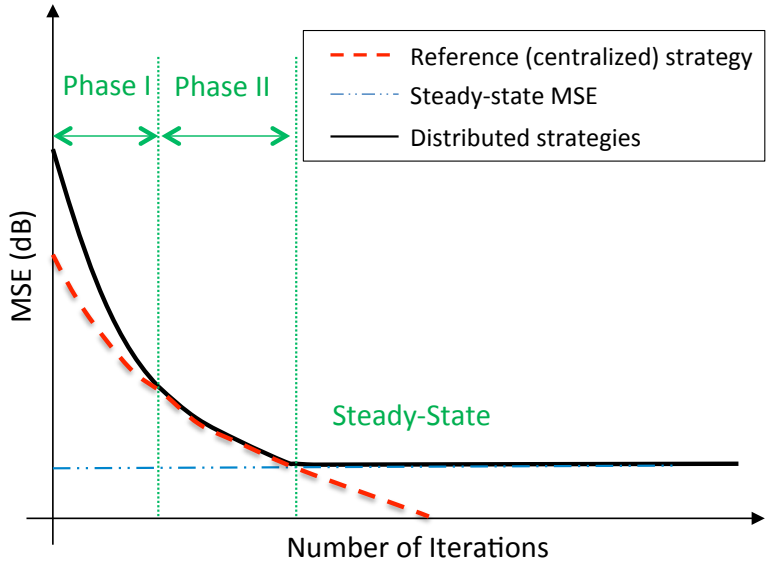


Figure 5.2: A typical mean-square-error (MSE) learning curve includes a transient stage that consists of two phases and a steady-state phase. The plot shows how the learning curve of a network of agents compares to the learning curve of a centralized reference solution. The analysis in this work, and in the following Chapter 6 characterizes in detail the parameters that determine the behavior of the network (rate, stability, and performance) during each phase of the learning process.

jectory of the reference recursion; they converge in a coordinated manner to steady-state. The learning curves at this phase are well modeled by the same reference recursion (5.40) since we will show in (5.150) that:

$$\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \|\tilde{w}_{c,i}\|^2 + O(\mu_{\max}^{1/2}) \cdot \gamma_c^i + O(\mu_{\max}) \tag{5.42}$$

Furthermore, for small step-sizes and during the later stages of this phase, $\bar{w}_{c,i}$ will be close enough to w^o and the convergence rate r will be shown to satisfy:

$$r = \big[\rho(I_M - \mu_{\max} H_c)\big]^2 + O\big((\mu_{\max}\epsilon)^{\frac{1}{2(M-1)}}\big) \tag{5.43}$$

where $\rho(\cdot)$ denotes the spectral radius of its matrix argument, ϵ is an arbitrarily small positive number, and H_c is the same matrix that results from evaluating (5.29) at $w = w^o$, i.e.,

$$H_c \triangleq \sum_{k=1}^{N} p_k H_k = H_c(w^o) \tag{5.44}$$

where $H_k \triangleq \nabla_{w^T} s_k(w^o)$.

- **Steady-State Phase:**

  The reference recursion (5.40) continues converging towards w^o so that $\|\tilde{w}_{c,i}\|^2 = \|w^o - \bar{w}_{c,i}\|^2$ will converge to zero ($-\infty$ dB in Fig. 5.2). However, for the distributed strategy (5.1)–(5.3), the mean-square-error $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \mathbb{E}\|w^o - \boldsymbol{w}_{k,i}\|^2$ at each agent k will converge to a *finite* steady-state value. We will be able to characterize this value in terms of the vector p in Chapter 6 as follows:

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \mu_{\max} \cdot \mathrm{Tr}\left\{X(p^T \otimes I_M)\mathcal{R}_v(p \otimes I_M)\right\} + o(\mu_{\max}) \tag{5.45}$$

  where X is the solution to the Lyapunov equation described later in (6.42) of Chapter 6 (when $\Sigma = I$). Expression (5.45) is a revealing result. It is a non-trivial extension of a classical result pertaining to the mean-square-error performance of stand-alone adaptive filters [54,57,72,141] to the more demanding context when a multitude of adaptive agents are coupled together in a cooperative manner through a network topology. This result has an important ramification, which we pursue in Chapter 6. We will show there that no matter how the agents are connected to each other, there is always a way to select the combination weights such that the performance of the network is invariant to the topology. This will also imply that, for any

connected topology, there is always a way to select the combination weights such that the performance of the network matches that of the centralized solution.

5.5 Study of Error Dynamics

5.5.1 Error Quantities

We shall examine the learning behavior of the distributed strategy (5.1)–(5.3) by examining how the perturbation between the distributed solution (5.1)–(5.3) and the reference solution (5.40) evolves over time — see Fig. 5.3. Specifically, let $\check{\boldsymbol{w}}_{k,i}$ denote the discrepancy between $\boldsymbol{w}_{k,i}$ and $\bar{w}_{c,i}$, i.e.,

$$\check{\boldsymbol{w}}_{k,i} \triangleq \boldsymbol{w}_{k,i} - \bar{w}_{c,i} \tag{5.46}$$

and let $\boldsymbol{w}_i$ and $\check{\boldsymbol{w}}_i$ denote the global vectors that collect the $\boldsymbol{w}_{k,i}$ and $\check{\boldsymbol{w}}_{k,i}$ from across the network, respectively:

$$\boldsymbol{w}_i \triangleq \mathrm{col}\{\boldsymbol{w}_{1,i}, \ldots, \boldsymbol{w}_{N,i}\} \tag{5.47}$$

$$\check{\boldsymbol{w}}_i \triangleq \mathrm{col}\{\check{\boldsymbol{w}}_{1,i}, \ldots, \check{\boldsymbol{w}}_{N,i}\} = \boldsymbol{w}_i - \mathbb{1} \otimes \bar{w}_{c,i} \tag{5.48}$$

It turns out that it is insightful to study the evolution of $\check{\boldsymbol{w}}_i$ in a *transformed* domain where it is possible to express the distributed recursion (5.1)–(5.3) as a perturbed version of the reference recursion (5.40).

Definition 5.3 (Network basis transformation)**.** *We define the transformation by introducing the Jordan canonical decomposition of the matrix* $A = A_1 A_0 A_2$.

Let

$$A^T = UDU^{-1} \tag{5.49}$$

where U is an invertible matrix whose columns correspond to the right-eigenvevtors of A^T, and D is a block Jordan matrix with a single eigenvalue at one with multiplicity one while all other eigenvalues are strictly less than one. The Kronecker form of A then admits the decomposition:

$$\mathcal{A}^T \triangleq A^T \otimes I_M = \mathcal{U}\mathcal{D}\mathcal{U}^{-1} \tag{5.50}$$

where

$$\mathcal{U} \triangleq U \otimes I_M, \qquad \mathcal{D} \triangleq D \otimes I_M \tag{5.51}$$

We use $\mathcal{U}$ to define the following basis transformation:

$$\boldsymbol{w}_i' \triangleq \mathcal{U}^{-1}\boldsymbol{w}_i = (U^{-1} \otimes I_M)\boldsymbol{w}_i \tag{5.52}$$

$$\check{\boldsymbol{w}}_i' \triangleq \mathcal{U}^{-1}\check{\boldsymbol{w}}_i = (U^{-1} \otimes I_M)\check{\boldsymbol{w}}_i \tag{5.53}$$

The relations between the quantities in transformations (5.52)–(5.53) *are illustrated in Fig. 5.3(a).* □

We can gain useful insight into the nature of this transformation by exploiting more directly the structure of the matrices $\mathcal{U}$, $\mathcal{D}$, and $\mathcal{U}^{-1}$. By Assumption 5.1, the matrix A^T has an eigenvalue one of multiplicity one, with the corresponding left- and right-eigenvectors being θ^T and $\mathbb{1}$, respectively. All other eigenvalues of D are strictly less than one in magnitude. Therefore, the matrices D, U, and

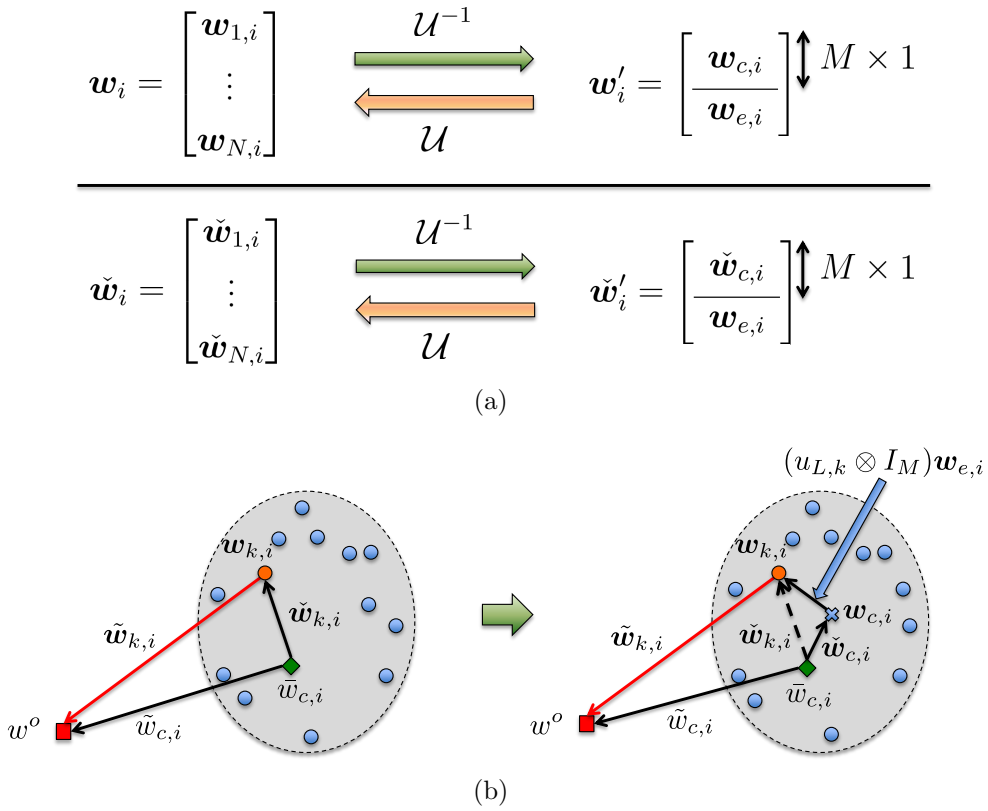


Figure 5.3: (a) Network basis transformation. (b) The diagrams show how the iterate $\boldsymbol{w}_{k,i}$ is decomposed relative to the reference $\bar{w}_{c,i}$ and relative to the centroid, $\boldsymbol{w}_{c,i}$, of the N iterates across the network.

U^{-1} can be partitioned as

$$D = \left[\begin{array}{c|c} 1 & \\ \hline & D_{N-1} \end{array}\right] \quad U = \left[\begin{array}{c|c} \mathbb{1} & U_L \end{array}\right] \quad U^{-1} = \left[\begin{array}{c} \theta^T \\ \hline U_R \end{array}\right] \tag{5.54}$$

where D_{N-1} is an $(N-1) \times (N-1)$ Jordan matrix with all diagonal entries strictly less than one in magnitude, U_L is an $N \times (N-1)$ matrix, and U_R is an $(N-1) \times N$ matrix. Then, the Kronecker forms $\mathcal{D}$, $\mathcal{U}$, and $\mathcal{U}^{-1}$ can be expressed

as

$$\mathcal{D} = \left[\begin{array}{c|c} I_M & \\ \hline & \mathcal{D}_{N-1} \end{array}\right], \; \mathcal{U} = \left[\mathbb{1}\otimes I_M \,\middle|\, \mathcal{U}_L\right], \; \mathcal{U}^{-1} = \left[\begin{array}{c} \theta^T \otimes I_M \\ \hline \mathcal{U}_R \end{array}\right] \tag{5.55}$$

where

$$\mathcal{U}_L \triangleq U_L \otimes I_M \tag{5.56}$$

$$\mathcal{U}_R \triangleq U_R \otimes I_M \tag{5.57}$$

$$\mathcal{D}_{N-1} \triangleq D_{N-1} \otimes I_M \tag{5.58}$$

It is important to note that $U^{-1}U = I_N$ and that

$$\theta^T \mathbb{1} = 1, \quad \theta^T U_L = 0, \quad U_R \mathbb{1} = 0, \quad U_R U_L = I_{N-1} \tag{5.59}$$

We first study the structure of $\boldsymbol{w}_i'$ defined in (5.52) using (5.54):

$$\boldsymbol{w}_i' = \mathrm{col}\{\underbrace{(\theta^T \otimes I_M)\boldsymbol{w}_i}_{\triangleq \boldsymbol{w}_{c,i}}, \; \underbrace{(U_R \otimes I_M)\boldsymbol{w}_i}_{\triangleq \boldsymbol{w}_{e,i}}\} \tag{5.60}$$

The two components $\boldsymbol{w}_{c,i}$ and $\boldsymbol{w}_{e,i}$ have useful interpretations. Recalling that θ_k denotes the kth entry of the vector θ, then $\boldsymbol{w}_{c,i}$ can be expressed as

$$\boldsymbol{w}_{c,i} = \sum_{k=1}^{N} \theta_k \boldsymbol{w}_{k,i} \tag{5.61}$$

As we indicated after Assumption 5.1, the entries $\{\theta_k\}$ are positive and add up to one. Therefore, $\boldsymbol{w}_{c,i}$ is a weighted average (i.e., the centroid) of the estimates $\{\boldsymbol{w}_{k,i}\}$ across all agents. To interpret $\boldsymbol{w}_{e,i}$, we examine the inverse mapping of

(5.52) from $\boldsymbol{w}_i'$ to $\boldsymbol{w}_i$ using the block structure of $\mathcal{U}$ in (5.54):

$$\begin{aligned}\boldsymbol{w}_i &= (U \otimes I_M)\boldsymbol{w}_i' \\ &= (\mathbb{1} \otimes I_M)\boldsymbol{w}_{c,i} + (U_L \otimes I_M)\boldsymbol{w}_{e,i} \\ &= \mathbb{1} \otimes \boldsymbol{w}_{c,i} + (U_L \otimes I_M)\boldsymbol{w}_{e,i} \end{aligned} \tag{5.62}$$

which implies that the individual estimates at the various agents satisfy:

$$\boldsymbol{w}_{k,i} = \boldsymbol{w}_{c,i} + (u_{L,k} \otimes I_M)\boldsymbol{w}_{e,i} \tag{5.63}$$

where $u_{L,k}$ denotes the kth row of the matrix U_L. The network basis transformation defined by (5.52) represents the cluster of iterates $\{\boldsymbol{w}_{k,i}\}$ by its centroid $\boldsymbol{w}_{c,i}$ and their positions $\{u_{L,k} \otimes I_M)\boldsymbol{w}_{e,i}\}$ relative to the centroid as shown in Fig. 5.3. The two parts, $\boldsymbol{w}_{c,i}$ and $\boldsymbol{w}_{e,i}$, of $\boldsymbol{w}_i'$ in (5.60) are the coordinates in this new transformed representation. Then, the actual error quantity $\tilde{\boldsymbol{w}}_{k,i}$ relative to w^o can be represented as

$$\begin{aligned}\tilde{\boldsymbol{w}}_{k,i} &= w^o - \bar{w}_{c,i} - (\boldsymbol{w}_{k,i} - \bar{w}_{c,i}) \\ &= w^o - \bar{w}_{c,i} - (\boldsymbol{w}_{c,i} + (u_{L,k} \otimes I_M)\boldsymbol{w}_{e,i} - \bar{w}_{c,i}) \end{aligned} \tag{5.64}$$

Introduce

$$\tilde{w}_{c,i} \triangleq w^o - \bar{w}_{c,i} \tag{5.65}$$

$$\check{\boldsymbol{w}}_{c,i} \triangleq \boldsymbol{w}_{c,i} - \bar{w}_{c,i} \tag{5.66}$$

Then, from (5.64) we arrive at the following critical relation for our analysis in

the sequel:

$$\tilde{\boldsymbol{w}}_{k,i} = \tilde{w}_{c,i} - \check{\boldsymbol{w}}_{c,i} - (u_{L,k} \otimes I_M)\boldsymbol{w}_{e,i} \tag{5.67}$$

This relation is also illustrated in Fig. 5.3. Then, the behavior of the error quantities $\{\tilde{\boldsymbol{w}}_{k,i}\}$ can be studied by examining $\tilde{w}_{c,i}$, $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$, respectively, which is pursued in Sec. 5.6 further ahead. The first term is the error between the reference recursion and w^o, which is studied in Theorems 5.1–5.3. The second quantity is the difference between the weighted centroid $\boldsymbol{w}_{c,i}$ of the cluster and the reference vector $\bar{w}_{c,i}$, and the third quantity characterizes the positions of the individual iterates $\{\boldsymbol{w}_{k,i}\}$ relative to the centroid $\boldsymbol{w}_{c,i}$. As long as the second and the third terms in (5.67), or equivalently, $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$, are small (which will be shown in Theorem 5.4), the behavior of each $\boldsymbol{w}_{k,i}$ can be well approximated by the behavior of the reference vector $\bar{w}_{c,i}$. Indeed, $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$ are the coordinates of the transformed vector $\check{\boldsymbol{w}}'_i$ defined by (5.53). To see this, we substitute (5.54) and (5.48) into (5.53) to get

$$\begin{aligned}\check{\boldsymbol{w}}'_i &= (U^{-1} \otimes I_M)(\boldsymbol{w}_i - \mathbb{1} \otimes \bar{w}_{c,i}) \\ &= \boldsymbol{w}'_i - (U^{-1}\mathbb{1}) \otimes \bar{w}_{c,i}\end{aligned} \tag{5.68}$$

Recalling (5.59) and the expression for U^{-1} in (5.54), we obtain

$$\begin{aligned}U^{-1}\mathbb{1} &= \mathrm{col}\{\theta^T\mathbb{1},\ U_R\mathbb{1}\} \\ &= \mathrm{col}\{1,\ 0_{N-1}\}\end{aligned} \tag{5.69}$$

where 0_{N-1} denotes an $(N-1) \times 1$ vector with all zero entries. Substituting

(5.69) and (5.60) into (5.68), we get

$$\check{\boldsymbol{w}}_i' = \mathrm{col}\{\boldsymbol{w}_{c,i} - \bar{w}_{c,i},\ \boldsymbol{w}_{e,i}\} = \mathrm{col}\{\check{\boldsymbol{w}}_{c,i},\ \boldsymbol{w}_{e,i}\} \tag{5.70}$$

Therefore, it suffices to study the dynamics of $\check{\boldsymbol{w}}_i'$ and its mean-square performance. We will establish joint recursions for $\boldsymbol{w}_{c,i}$ and $\boldsymbol{w}_{e,i}$ in Sec. 5.5.2, and joint recursions for $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$ in Sec. 5.5.3. Table 5.2 summarizes the definitions of the various quantities, the recursions that they follow, and their relations.

5.5.2 Signal Recursions

We now derive the joint recursion that describes the evolution of the quantities $\check{\boldsymbol{w}}_{c,i} = \boldsymbol{w}_{c,i} - \bar{w}_{c,i}$ and $\boldsymbol{w}_{e,i}$. Since $\bar{w}_{c,i}$ follows the reference recursion (5.40), it suffices to derive the joint recursion for $\boldsymbol{w}_{c,i}$ and $\boldsymbol{w}_{e,i}$. To begin with, we introduce the following global quantities:

$$\mathcal{A} = A \otimes I_M \tag{5.71}$$

$$\mathcal{A}_0 = A_0 \otimes I_M \tag{5.72}$$

$$\mathcal{A}_1 = A_1 \otimes I_M \tag{5.73}$$

$$\mathcal{A}_2 = A_2 \otimes I_M \tag{5.74}$$

$$\mathcal{M} = \Omega \otimes I_M \tag{5.75}$$

$$\Omega = \mathrm{diag}\{\mu_1, \ldots, \mu_N\} \tag{5.76}$$

We also let the notation $x = \mathrm{col}\{x_1, \ldots, x_N\}$ denote an arbitrary $N \times 1$ block column vector that is formed by stacking $M \times 1$ sub-vectors $x_1, \ldots, x_N$ on top of

Table 5.2: Summary of various iterates, error quantities, and their relations.

	Original system			**Transformed system**[a]			**Reference system**	
Quantity	$\boldsymbol{w}_{k,i}$	$\tilde{\boldsymbol{w}}_{k,i}$	$\check{\boldsymbol{w}}_{k,i}$	$\boldsymbol{w}_{c,i}$	$\check{\boldsymbol{w}}_{c,i}$	$\boldsymbol{w}_{e,i}$	$\bar{w}_{c,i}$	$\tilde{w}_{c,i}$
Definition	Iterate at agent k	$w^o - \boldsymbol{w}_{k,i}$	$\boldsymbol{w}_{k,i} - \bar{w}_{c,i}$	$\sum_{k=1}^{N} \theta_k \boldsymbol{w}_{k,i}$	$\boldsymbol{w}_{c,i} - \bar{w}_{c,i}$	$\mathcal{U}_R \boldsymbol{w}_i$	Ref. Iterate	$w^o - \bar{w}_{c,i}$
Recursion	Eqs. (5.1)–(5.3)			Eq. (5.89)	Eq. (5.97)	Eq. (5.98)	Eq. (5.40)	

[a] The transformation is defined by (5.52)–(5.53).

each other. We further define the following global update vectors:

$$\hat{\boldsymbol{s}}_i(x) \triangleq \text{col}\{\hat{\boldsymbol{s}}_{1,i}(x_1), \ldots, \hat{\boldsymbol{s}}_{N,i}(x_N)\} \tag{5.77}$$

$$s(x) \triangleq \text{col}\{s_1(x_1), \ldots, s_N(x_N)\} \tag{5.78}$$

Then, the general recursion for the distributed strategy (5.1)–(5.3) can be rewritten in terms of these extended quantities as follows:

$$\boldsymbol{w}_i = \mathcal{A}^T \boldsymbol{w}_{i-1} - \mathcal{A}_2^T \mathcal{M} \hat{\boldsymbol{s}}_i(\boldsymbol{\phi}_{i-1}) \tag{5.79}$$

where

$$\boldsymbol{\phi}_i \triangleq \text{col}\{\boldsymbol{\phi}_{1,i}, \ldots, \boldsymbol{\phi}_{N,i}\} \tag{5.80}$$

and is related to $\boldsymbol{w}_i$ and $\boldsymbol{w}'_i$ via the following relation

$$\boldsymbol{\phi}_i = \mathcal{A}_1^T \boldsymbol{w}_i = \mathcal{A}_1^T \mathcal{U} \boldsymbol{w}'_i \tag{5.81}$$

Applying the transformation (5.52) to both sides of (5.79), we obtain the transformed global recursion:

$$\boldsymbol{w}'_i = \mathcal{D} \boldsymbol{w}'_{i-1} - \mathcal{U}^{-1} \mathcal{A}_2^T \mathcal{M} \hat{\boldsymbol{s}}_i\,(\boldsymbol{\phi}_{i-1}) \tag{5.82}$$

We can now use the block structures in (5.54) and (5.60) to derive recursions for $\boldsymbol{w}_{c,i}$ and $\boldsymbol{w}_{e,i}$ from (5.82). Substituting (5.55) and (5.60) into (5.82), and using properties of Kronecker products [82, p.147], we obtain

$$\boldsymbol{w}_{c,i} = \boldsymbol{w}_{c,i-1} - (\theta^T \otimes I_M) \mathcal{A}_2^T \mathcal{M} \hat{\boldsymbol{s}}_i\,(\boldsymbol{\phi}_{i-1})$$

$$
\begin{aligned}
&= \boldsymbol{w}_{c,i-1} - (\theta^T A_2^T \Omega \otimes I_M)\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right) \\
&= \boldsymbol{w}_{c,i-1} - \mu_{\max} \cdot (p^T \otimes I_M)\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right)
\end{aligned} \tag{5.83}
$$

and

$$
\boldsymbol{w}_{e,i} = \mathcal{D}_{N-1}\boldsymbol{w}_{e,i-1} - \mathcal{U}_R \mathcal{A}_2^T \mathcal{M}\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right) \tag{5.84}
$$

where in the last step of (5.83) we used the relation

$$
\mu_{\max} \cdot p = \Omega A_2 \theta \tag{5.85}
$$

which follows from Definitions 5.1 and 5.2. Furthermore, by adding and subtracting identical factors, the term $\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right)$ that appears in (5.83) and (5.84) can be expressed as

$$
\begin{aligned}
\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right) = s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1}) &+ \underbrace{\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right) - s\left(\boldsymbol{\phi}_{i-1}\right)}_{\triangleq \boldsymbol{v}_i(\boldsymbol{\phi}_{i-1})} \\
&+ \underbrace{s\left(\boldsymbol{\phi}_{i-1}\right) - s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1})}_{\triangleq \boldsymbol{z}_{i-1}}
\end{aligned} \tag{5.86}
$$

where the first perturbation term $\boldsymbol{v}_i(\boldsymbol{\phi}_{i-1})$ consists of the difference between the true update vectors $\{s_k(\boldsymbol{\phi}_{k,i-1})\}$ and their stochastic approximations $\{\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{\phi}_{k,i-1})\}$, while the second perturbation term $\boldsymbol{z}_{i-1}$ represents the difference between the same $\{s_k(\boldsymbol{\phi}_{k,i-1})\}$ and $\{s_k(\boldsymbol{w}_{c,i-1})\}$. The subscript $i-1$ in $\boldsymbol{z}_{i-1}$ implies that this variable depends on data up to time $i-1$ and the subscript i in $\boldsymbol{v}_i(\boldsymbol{\phi}_{i-1})$ implies that its value depends on data up to time i (since, in general, $\hat{\boldsymbol{s}}_i(\cdot)$ can depend on data from time i — see Eq. (6.31) in Chapter 6 for an example). Then, $\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right)$

can be expressed as

$$\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right) = s(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1}) + \boldsymbol{v}_i + \boldsymbol{z}_{i-1} \tag{5.87}$$

Lemma 5.2 (Signal dynamics)**.** *In summary, the previous derivation shows that the weight iterates at each agent evolve according to the following dynamics:*

$$\boldsymbol{w}_{k,i} = \boldsymbol{w}_{c,i} + (u_{L,k}\otimes I_M)\boldsymbol{w}_{e,i} \tag{5.88}$$

$$\boldsymbol{w}_{c,i} = \boldsymbol{w}_{c,i-1} - \mu_{\max}\cdot (p^T\otimes I_M)\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right) \tag{5.89}$$

$$\boldsymbol{w}_{e,i} = \mathcal{D}_{N-1}\boldsymbol{w}_{e,i-1} - \mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right) \tag{5.90}$$

$$\hat{\boldsymbol{s}}_i\left(\boldsymbol{\phi}_{i-1}\right) = s(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1}) + \boldsymbol{v}_i + \boldsymbol{z}_{i-1} \tag{5.91}$$

□

5.5.3 Error Dynamics

To simplify the notation, we introduce the centralized operator $T_c : \mathbb{R}^M \to \mathbb{R}^M$ as the following mapping for any $x\in\mathbb{R}^M$:

$$\begin{aligned} T_c(x) &\triangleq x - \mu_{\max}\cdot (p^T\otimes I_M)\; s(\mathbb{1}\otimes x) \\ &= x - \mu_{\max}\sum_{k=1}^{N} p_k s_k(x) \end{aligned} \tag{5.92}$$

Substituting (5.87) into (5.89)–(5.90) and using (5.92), we find that we can rewrite (5.83) and (5.84) in the alternative form:

$$\boldsymbol{w}_{c,i} = T_c(\boldsymbol{w}_{c,i-1}) - \mu_{\max}\cdot (p^T\otimes I_M)\left[\boldsymbol{z}_{i-1} + \boldsymbol{v}_i\right] \tag{5.93}$$

$$\boldsymbol{w}_{e,i} = \mathcal{D}_{N-1}\boldsymbol{w}_{e,i-1} - \mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\left[s(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1} + \boldsymbol{v}_i\right] \tag{5.94}$$

Likewise, we can write the reference recursion (5.40) in the following compact form:

$$\bar{w}_{c,i} = T_c(\bar{w}_{c,i-1}) \tag{5.95}$$

Comparing (5.93) with (5.95), we notice that the recursion for the centroid vector, $\boldsymbol{w}_{c,i}$, follows the same update rule as the reference recursion except for the two driving perturbation terms $\boldsymbol{z}_{i-1}$ and $\boldsymbol{v}_i$. Therefore, we would expect the trajectory of $\boldsymbol{w}_{c,i}$ to be a perturbed version of that of $\bar{w}_{c,i}$. Let

$$\check{\boldsymbol{w}}_{c,i} \triangleq \boldsymbol{w}_{c,i} - \bar{w}_{c,i} \tag{5.96}$$

To obtain the dynamics of $\check{\boldsymbol{w}}_{c,i}$, we subtract (5.95) from (5.93).

Lemma 5.3 (Error dynamics)**.** *The error quantities that appear on the right-hand side of* (5.70) *evolve according to the following dynamics:*

$$\check{\boldsymbol{w}}_{c,i} = T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\,[\boldsymbol{z}_{i-1} + \boldsymbol{v}_i] \tag{5.97}$$

$$\boldsymbol{w}_{e,i} = \mathcal{D}_{N-1}\boldsymbol{w}_{e,i-1} - \mathcal{U}_R \mathcal{A}_2^T \mathcal{M}\,[s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1} + \boldsymbol{v}_i] \tag{5.98}$$

□

The analysis in sequel will study the dynamics of the variances of the error quantities $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$ based on (5.97)–(5.98). The main challenge is that these two recursions are coupled with each other through $\boldsymbol{z}_{i-1}$ and $\boldsymbol{v}_i$. To address the difficulty, we will extend the energy operator approach developed in [36] to the general scenario under consideration.

5.5.4 Energy Operator and Properties

To carry out the analysis, we need to introduce the following operators and their corresponding properties.

Definition 5.4 (Energy vector operator)**.** *Suppose* $x = \mathrm{col}\{x_1, \ldots, x_N\}$ *is an arbitrary* $N \times 1$ *block column vector that is formed by stacking* $M_0 \times 1$ *vectors* $x_1, \ldots, x_N$ *on top of each other. The energy vector operator* $P_{M_0} : \mathbb{C}^{M_0 N} \to \mathbb{R}^N$ *is defined as the mapping:*

$$P_{M_0}[x] \triangleq \mathrm{col}\{\|x_1\|^2, \ldots, \|x_N\|^2\} \tag{5.99}$$

where $\|\cdot\|$ *denotes the Euclidean norm of a vector.* □

Definition 5.5 (Norm matrix operator)**.** *Suppose* X *is an arbitrary* $K \times N$ *block matrix consisting of blocks* $\{X_{kn}\}$ *of size* $M_0 \times M_0$*:*

$$X = \begin{bmatrix} X_{11} & \cdots & X_{1N} \\ \vdots & & \vdots \\ X_{K1} & \cdots & X_{KN} \end{bmatrix} \tag{5.100}$$

The norm matrix operator $\bar{P}_{M_0} : \mathbb{C}^{M_0 K \times M_0 N} \to \mathbb{R}^{K \times N}$ *is defined as the mapping:*

$$\bar{P}_{M_0}[x] \triangleq \begin{bmatrix} \|X_{11}\| & \cdots & \|X_{1N}\| \\ \vdots & & \vdots \\ \|X_{K1}\| & \cdots & \|X_{KN}\| \end{bmatrix} \tag{5.101}$$

where $\|\cdot\|$ *denotes the* $2-$*induced norm of a matrix.* □

By default, we choose M_0 to be M, the size of the vector $\boldsymbol{w}_{k,i}$. In this case, we will drop the subscript in$P_{M_0}[\cdot]$ and use $P[\cdot]$ for convenience. However, in other

cases, we will keep the subscript to avoid confusion. Likewise, $\bar{P}_{M_0}[\cdot]$ characterizes the norms of different parts of a matrix it operates on. We will also drop the subscript if $M_0 = M$. Next, we state lemmas on properties of the operators $P_{M_0}[\cdot]$ and $\bar{P}_{M_0}[\cdot]$. We begin with some basic properties.

Lemma 5.4 (Basic properties)**.** *Consider $N\times 1$ block vectors $x = \mathrm{col}\{x_1, \ldots, x_N\}$ and $y = \mathrm{col}\{y_1, \ldots, y_N\}$ with $M \times 1$ entries $\{x_k, y_k\}$. Consider also the $K \times N$ block matrix X with blocks of size $M \times M$. Then, the operators $P[\cdot]$ and $\bar{P}[\cdot]$ satisfy the following properties:*

1. *(***Nonnegativity***):* $P[x] \succeq 0$, $\bar{P}[X] \succeq 0$.

2. *(***Scaling***): For any scalar $a \in \mathbb{C}$, we have $P[ax] = |a|^2 P[x]$ and $\bar{P}[aX] = |a| \cdot \bar{P}[X]$.*

3. *(***Convexity***): suppose $x^{(1)}, \ldots, x^{(K)}$ are $N \times 1$ block vectors formed in the same manner as x, $X^{(1)}, \ldots, X^{(K)}$ are $K \times N$ block matrices formed in the same manner as X, and let $a_1, \ldots, a_K$ be non-negative real scalars that add up to one. Then,*

$$P[a_1 x^{(1)} + \cdots + a_K x^{(K)}] \preceq a_1 P[x^{(1)}] + \cdots + a_K P[x^{(K)}] \tag{5.102}$$

$$\bar{P}[a_1 X^{(1)} + \cdots + a_K X^{(K)}] \preceq a_1 \bar{P}[X^{(1)}] + \cdots + a_K \bar{P}[X^{(K)}] \tag{5.103}$$

4. *(***Additivity***): Suppose $\boldsymbol{x} = \mathrm{col}\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ and $\boldsymbol{y} = \mathrm{col}\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$ are $N \times 1$ block random vectors that satisfy $\mathbb{E}\boldsymbol{x}_k^* \boldsymbol{y}_k = 0$ for $k = 1, \ldots, N$, where $*$ denotes complex conjugate transposition. Then,*

$$\mathbb{E}P[\boldsymbol{x} + \boldsymbol{y}] = \mathbb{E}P[\boldsymbol{x}] + \mathbb{E}P[\boldsymbol{y}] \tag{5.104}$$

5. *(***Triangular inequality***): Suppose X and Y are two $K\times N$ block matrices*

of same block size M. Then,

$$\bar{P}[X+Y] \preceq \bar{P}[X] + \bar{P}[Y] \tag{5.105}$$

6. ***(*Submultiplicity*):*** *Suppose X and Z are $K \times N$ and $N \times L$ block matrices of the same block size M, respectively. Then,*

$$\bar{P}[XZ] \preceq \bar{P}[X]\bar{P}[Z] \tag{5.106}$$

7. ***(*Kronecker structure*):*** *Suppose $X \in \mathbb{C}^{K\times N}$, $a \in \mathbb{C}^N$ and $b \in \mathbb{C}^M$. Then,*

$$\bar{P}[X \otimes I_M] = \bar{P}_1[X] \tag{5.107}$$

$$P[a \otimes b] = \|b\|^2 \cdot P_1[a] \tag{5.108}$$

where by definition, $\bar{P}_1[\cdot]$ and $P_1[\cdot]$ denote the operators that work on the scalar entries of their arguments. When X consists of nonnegative entries, relation (5.107) *becomes*

$$\bar{P}[X \otimes I_M] = X \tag{5.109}$$

8. ***(*Relation to norms*):*** *The $\infty-$norm of $P[x]$ is the squared block maximum norm of x:*

$$\|P[x]\|_\infty = \|x\|_{b,\infty}^2 \triangleq \big(\max_{1\le k\le N} \|x_k\|\big)^2 \tag{5.110}$$

Moreover, the sum of the entries in $P[x]$ is the squared Euclidean norm of

x:

$$\mathbb{1}^T P[x] = \|x\|^2 = \sum_{k=1}^{N} \|x_k\|^2 \tag{5.111}$$

9. *(***Inequality preservation***):* *Suppose vectors x, y and matrices F, G have nonnegative entries, then $x \preceq y$ implies $Fx \preceq Fy$, and $F \preceq G$ implies $Fx \preceq Gx$.*

10. *(***Upper bounds***):* *It holds that*

$$\bar{P}[X] \preceq \|\bar{P}[X]\|_1 \cdot \mathbb{1}\mathbb{1}^T \tag{5.112}$$

$$\bar{P}[X] \preceq \|\bar{P}[X]\|_\infty \cdot \mathbb{1}\mathbb{1}^T \tag{5.113}$$

where $\|\cdot\|_\infty$ denotes the $\infty-$induced norm of a matrix (maximum absolute row sum).

Proof. See Appendix 5.B. □

More importantly, the following variance relations hold for the energy and norm operators. These relations show how error variances propagate after a certain operator is applied to a random vector.

Lemma 5.5 (Variance relations)**.** *Consider $N \times 1$ block vectors $x = \mathrm{col}\{x_1, \ldots, x_N\}$ and $y = \mathrm{col}\{y_1, \ldots, y_N\}$ with $M \times 1$ entries $\{x_k, y_k\}$. The followig variance relations are satisfied by the energy vector operator $P[\cdot]$:*

1. *(***Linear transformation***):* *Given a $K \times N$ block matrix Q with the size of each block being $M \times M$, Qx defines a linear operator on x and its energy*

satisfies

$$P[Qx] \preceq \|\bar{P}[Q]\|_\infty \cdot \bar{P}[Q]\ P[x] \tag{5.114}$$

$$\preceq \|\bar{P}[Q]\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \cdot P[x] \tag{5.115}$$

As a special case, for a left-stochastic $N \times N$ matrix A, we have

$$P[(A^T \otimes I_M)x] \preceq A^T P[x] \tag{5.116}$$

2. ***(*Update operation*):*** *The global update vector defined by* (5.78) *satisfies the following variance relation:*

$$P[s(x) - s(y)] \preceq \lambda_U^2 P[x - y] \tag{5.117}$$

3. ***(*Centralized operation*):*** *The centralized operator $T_c(x)$ defined by* (5.92) *satisfies the following variance relations:*

$$P[T_c(x) - T_c(y)] \preceq \gamma_c^2 \cdot P[x - y] \tag{5.118}$$

$$P[T_c(x) - T_c(y)] \succeq (1 - 2\mu_{\max}\|p\|_1\lambda_U) \cdot P[x - y] \tag{5.119}$$

where

$$\gamma_c \triangleq 1 - \mu_{\max}\lambda_L + \frac{1}{2}\mu_{\max}^2\|p\|_1^2\lambda_U^2 \tag{5.120}$$

Moreover, it follows from (5.26) *that*

$$\begin{aligned}\gamma_c &\geq 1 - \mu_{\max}\lambda_L + \frac{1}{2}\mu_{\max}^2\lambda_L^2 \\ &= (1 - \frac{1}{2}\mu_{\max}\lambda_L)^2 + \frac{1}{4}\mu_{\max}^2\lambda_L^2 > 0\end{aligned} \tag{5.121}$$

4. ***(*Stable Jordan operation*):*** *Suppose D_L is an $L \times L$ Jordan matrix of the following block form:*

$$D_L \triangleq \mathrm{diag}\{D_{L,2}, \ldots, D_{L,n_0}\} \tag{5.122}$$

where the nth $L_n \times L_n$ Jordan block is defined as (note that $L = L_2 + \cdots + L_{n_0}$)

$$D_{L,n} \triangleq \begin{bmatrix} d_n & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & d_n \end{bmatrix} \tag{5.123}$$

We further assume D_L to be stable with $0 \leq |d_{n_0}| \leq \cdots \leq |d_2| < 1$. Then, for any $L \times 1$ vectors x' and y', we have

$$P_1[D_L x' + y'] \preceq \Gamma_e \cdot P_1[x'] + \frac{2}{1 - |d_2|} \cdot P_1[y'] \tag{5.124}$$

where Γ_e is the $L \times L$ matrix defined as

$$\Gamma_e \triangleq \begin{bmatrix} |d_2| & \frac{2}{1-|d_2|} & & \\ & \ddots & \ddots & \\ & & \ddots & \frac{2}{1-|d_2|} \\ & & & |d_2| \end{bmatrix} \tag{5.125}$$

5. ***(*Stable Kronecker Jordan operator*):*** *Suppose $\mathcal{D}_L = D_L \otimes I_M$, where D_L is the $L \times L$ Jordan matrix defined in* (5.122)–(5.123)*. Then, for any*

$LM \times 1$ vectors x_e and y_e, we have

$$P[\mathcal{D}_L x_e + y_e] \preceq \Gamma_e \cdot P[x_e] + \frac{2}{1-|d_2|} \cdot P[y_e] \tag{5.126}$$

Proof. See Appendix 5.C. □

5.6 Transient Analysis

Using the energy operators and the various properties, we can now examine the transient behavior of the learning curve more closely. Recall from (5.67) that $\tilde{\boldsymbol{w}}_{k,i}$ consists of three parts: the error of the reference recursion, $\tilde{w}_{c,i}$, the difference between the centroid and the reference, $\check{\boldsymbol{w}}_{c,i}$, and the position of individual iterates relative to the centroid, $(u_{L,k} \otimes I_M)\boldsymbol{w}_{e,i}$. The main objective in the sequel is to study the convergence of the reference error, $\tilde{w}_{c,i}$, and establish non-asymptotic bounds for the mean-square values of $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$, which will allow us to understand how fast and how close the iterates at the individual agents, $\{\boldsymbol{w}_{k,i}\}$, get to the reference recursion. Recalling from (5.70) that $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$ are the two blocks of the transformed vector $\check{\boldsymbol{w}}'_i$ defined by (5.53), we can examine instead the evolution of

$$\begin{aligned}\check{\mathcal{W}}'_i &\triangleq \mathbb{E}P[\check{\boldsymbol{w}}'_i] = \mathrm{col}\left\{\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}], \mathbb{E}P[\boldsymbol{w}_{e,i}]\right\} \\ &= \mathrm{col}\left\{\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2, \mathbb{E}P[\boldsymbol{w}_{e,i}]\right\}\end{aligned} \tag{5.127}$$

Specifically, we will study the convergence of $\tilde{w}_{c,i}$ in Sec. 5.6.1, the stability of $\check{\mathcal{W}}'_i$ in Sec. 5.6.2, and the two transient phases of $\tilde{\boldsymbol{w}}_{k,i}$ in Sec. 5.6.3.

5.6.1 Limit Point

Before we proceed to study $\check{\mathcal{W}}_i'$, we state the following theorems on the existence of a limit point and on the convergence of the reference recursion (5.95).

Theorem 5.1 (Limit point)**.** *Given Assumptions 5.3–5.4, there exists a unique $M \times 1$ vector w^o that solves*

$$\sum_{k=1}^{N} p_k s_k(w^o) = 0 \tag{5.128}$$

where p_k is the kth entry of the vector p defined in (5.16)*.*

Proof. See Appendix 5.D. □

Theorem 5.2 (Convergence of the reference recursion)**.** *Let $\tilde{w}_{c,i} \triangleq w^o - \bar{w}_{c,i}$ denote the error vector of the reference recursion* (5.95)*. Then, the following non-asymptotic bound on the squared error holds for all $i \geq 0$:*

$$(1-2\mu_{\max}\|p\|_1\lambda_U)^i \cdot \|\tilde{w}_{c,0}\|^2 \leq \|\tilde{w}_{c,i}\|^2 \leq \gamma_c^{2i} \cdot \|\tilde{w}_{c,0}\|^2 \tag{5.129}$$

Furthermore, if the following condition on the step-size holds

$$0 < \mu_{\max} < \frac{2\lambda_L}{\|p\|_1^2 \lambda_U^2} \tag{5.130}$$

then, the iterate $\tilde{w}_{c,i}$ converges to zero.

Proof. See Appendix 5.E. □

Note from (5.129) that, when the step-size is sufficiently small, the reference recursion (5.40) converges at a geometric rate between $1 - 2\mu_{\max}\|p\|_1\lambda_U$ and

$\gamma_c^2 = 1 - 2\mu_{\max}\lambda_L + o(\mu_{\max})$. We can get a more precise characterization of the convergence rate of the reference recursion.

Theorem 5.3 (Convergence rate of the reference recursion)**.** *Specifically, for any small $\epsilon > 0$, there exists a time instant i_0 such that, for $i \geq i_0$, the error vector $\tilde{w}_{c,i}$ converges to zero at the following rate:*

$$r = \big[\rho(I_M - \mu_{\max}H_c)\big]^2 + O\big((\mu_{\max}\epsilon)^{\frac{1}{2(M-1)}}\big) \tag{5.131}$$

Proof. See Appendix 5.F. □

Note that since (5.131) holds for arbitrary $\epsilon > 0$, we can choose ϵ to be an arbitrarily small positive number. Therefore, the convergence rate of the reference recursion is arbitrarily close to $[\rho(I_M - \mu_{\max}H_c)]^2$.

5.6.2 Mean-Square Stability

Now we apply the properties from Lemmas 5.4–5.5 to derive an inequality recursion for the transformed energy vector $\check{\mathcal{W}}_i' = \mathbb{E}P[\check{\boldsymbol{w}}_i']$. The results are summarized in the following lemma.

Lemma 5.6 (Inequality recursion for $\check{\mathcal{W}}_i'$)**.** *The $N \times 1$ vector $\check{\mathcal{W}}_i'$ defined by* (5.127) *satisfies the following relation for all time instants:*

$$\check{\mathcal{W}}_i' \preceq \Gamma\check{\mathcal{W}}_{i-1}' + \mu_{\max}^2 b_v \tag{5.132}$$

where

$$\Gamma \triangleq \Gamma_0 + \mu_{\max}^2\psi_0 \cdot \mathbb{1}\mathbb{1}^T \in \mathbb{R}^{N\times N} \tag{5.133}$$

$$\Gamma_0 \triangleq \begin{bmatrix} \gamma_c & \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T \\ 0 & \Gamma_e \end{bmatrix} \in \mathbb{R}^{N \times N} \tag{5.134}$$

$$b_v \triangleq \mathrm{col}\{b_{v,c},\ b_{v,e} \cdot \mathbb{1}\} \in \mathbb{R}^N \tag{5.135}$$

and Γ_e *is an* $(N-1) \times (N-1)$ *matrix of the same form as* (5.125) *(i.e., with the same structure and entries with* d_2 *replaced by* $|\lambda_2(A)|$*). The scalars* ψ_0*,* $h_c(\mu)$*,* $b_{v,c}$ *and* $b_{v,e}$ *are defined as*

$$\begin{aligned} \psi_0 \triangleq \max \Big\{ & 4\alpha \|p\|_1^2,\ \ 4\alpha \|p\|_1^2 \cdot \big\| \bar{P}[\mathcal{A}_1^T \mathcal{U}_L] \big\|_\infty^2, \\ & 4N \cdot \big\| \bar{P}[\mathcal{U}_R \mathcal{A}_2^T] \big\|_\infty^2 \lambda_U^2 \left(\frac{3}{1 - |\lambda_2(A)|} + \frac{\alpha}{\lambda_U^2} \right), \\ & 4N \cdot \big\| \bar{P}[\mathcal{U}_R \mathcal{A}_2^T] \big\|_\infty^2 \cdot \big\| \bar{P}[\mathcal{A}_1^T \mathcal{U}_L] \big\|_\infty^2 \lambda_U^2 \\ & \cdot \left(\frac{1}{1 - |\lambda_2(A)|} + \frac{\alpha}{\lambda_U^2} \right) \Big\} \end{aligned} \tag{5.136}$$

$$h_c(\mu_{\max}) \triangleq \|p\|_1^2 \cdot \big\| \bar{P}[\mathcal{A}_1^T \mathcal{U}_L] \big\|_\infty^2 \lambda_U^2 \cdot \Big[\frac{1}{\lambda_L - \frac{1}{2} \mu_{\max} \|p\|_1^2 \lambda_U^2} \Big] \tag{5.137}$$

$$b_{v,c} \triangleq \|p\|_1^2 \cdot \big[4\alpha(\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2 \big] \tag{5.138}$$

$$\begin{aligned} b_{v,e} \triangleq N \big\| \bar{P}[\mathcal{U}_R \mathcal{A}_2^T] \big\|_\infty^2 \Bigg(& 12 \frac{\lambda_U^2 \|\tilde{w}_{c,0}\|^2 + \|g^o\|_\infty}{1 - |\lambda_2(A)|} \\ & + 4\alpha(\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2 \Bigg) \end{aligned} \tag{5.139}$$

where $g^o \triangleq P[s(\mathbb{1} \otimes w^o)]$.

Proof. See Appendix 5.G. □

From (5.133)–(5.134), we see that as the step-size $\mu_{\max}$ becomes small, we have $\Gamma \approx \Gamma_0$, since the second term in the expression for Γ depends on the square of the step-size. Moreover, note that Γ_0 is an upper triangular matrix. Therefore, $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$ are weakly coupled for small step-sizes; $\mathbb{E}P[\boldsymbol{w}_{e,i}]$ evolves on its own,

but it will seep into the evolution of $\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}]$ via the off-diagonal term in Γ_0, which is $O(\mu_{\max})$. This insight is exploited to establish a non-asymptotic bound on $\check{\mathcal{W}}_i' = \mathrm{col}\{\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2, \mathbb{E}P[\boldsymbol{w}_{e,i}]\}$ in the following theorem.

Theorem 5.4 (Non-asymptotic bound for $\check{\mathcal{W}}_i'$)**.** *Suppose the matrix* Γ *defined in* (5.133) *is stable, i.e.,* $\rho(\Gamma) < 1$*. Then, the following non-asymptotic bound holds for all* $i \geq 0$*:*

$$\begin{aligned}\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}] \preceq\ & \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T(\gamma_c I - \Gamma_e)^{-1}\left(\gamma_c^i I - \Gamma_e^i\right)\mathcal{W}_{e,0} \\ & + \check{\mathcal{W}}_{c,\infty}^{\mathrm{ub}'} \qquad (5.140)\\ \mathbb{E}P[\boldsymbol{w}_{e,i}] \preceq\ & \Gamma_e^i \mathcal{W}_{e,0} + \check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'} \qquad (5.141)\end{aligned}$$

where $\mathcal{W}_{e,0} \triangleq \mathbb{E}P[\boldsymbol{w}_{e,0}]$*,* $\check{\mathcal{W}}_{c,\infty}^{\mathrm{ub}'}$ *and* $\check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'}$ *are the* $\limsup$ *bounds of* $\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}]$ *and* $\mathbb{E}P[\boldsymbol{w}_{e,i}]$*, respectively:*

$$\begin{aligned}\check{\mathcal{W}}_{c,\infty}^{\mathrm{ub}'} =\ & \mu_{\max} \cdot \frac{\psi_0\big(\lambda_L + h_c(0)\big)\mathbb{1}^T(I - \Gamma_e)^{-1}\mathcal{W}_{e,0} + b_{v,c}\lambda_L}{\lambda_L^2} \\ & + o(\mu_{\max}) \qquad (5.142)\\ \check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'} =\ & \mu_{\max}^2 \cdot \frac{\psi_0\big(\lambda_L + h_c(0)\big)\mathbb{1}^T(I - \Gamma_e)^{-1}\mathcal{W}_{e,0} + b_{v,e}\lambda_L}{\lambda_L} \\ & \times (I - \Gamma_e)^{-1}\mathbb{1} + o(\mu_{\max}^2) \qquad (5.143)\end{aligned}$$

where $o(\cdot)$ *denotes strictly higher order terms, and* $h_c(0)$ *is the value of* $h_c(\mu_{\max})$ *(see* (5.137)*) evaluated at* $\mu_{\max} = 0$*. An important implication of* (5.140) *and* (5.142) *is that*

$$\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}] \leq O(\mu_{\max}), \qquad \forall i \geq 0 \qquad (5.144)$$

Furthermore, a sufficient condition that guarantees the stability of the matrix Γ

is that

$$0 < \mu_{\max} < \min \left\{ \frac{\lambda_L}{\frac{1}{2}\|p\|_1^2 \lambda_U^2 + \frac{1}{3}\psi_0 \left(\frac{1-|\lambda_2(A)|}{2}\right)^{-2N}}, \right. \\ \sqrt{\frac{3(1-|\lambda_2(A)|)^{2N+1}}{2^{2N+2}\psi_0}}, \\ \left. \frac{\lambda_L}{\|p\|_1^2 \lambda_U^2 \left(\|\bar{P}_1[A^T U_L]\|_\infty^2 + \frac{1}{2}\right)} \right\} \tag{5.145}$$

Proof. See Appendix 5.I. □

Corollary 5.1 (Asymptotic bounds)**.** *It holds that*

$$\limsup_{i\to\infty} \mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2 \leq O(\mu_{\max}) \tag{5.146}$$

$$\limsup_{i\to\infty} \mathbb{E}\|\boldsymbol{w}_{e,i}\|^2 \leq O(\mu_{\max}^2) \tag{5.147}$$

Proof. The bound (5.146) holds since $\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2 = \mathbb{E}P[\check{\boldsymbol{w}}_{c,i}] \leq O(\mu_{\max})$ for all $i \geq 0$ according to (5.144). Furthermore, inequality (5.147) holds because

$$\begin{aligned} \limsup_{i\to\infty} \mathbb{E}\|\boldsymbol{w}_{e,i}\|^2 &\overset{(a)}{=} \limsup_{i\to\infty} \mathbb{1}^T \mathbb{E}P[\boldsymbol{w}_{e,i}] \\ &\overset{(b)}{\preceq} \mathbb{1}^T \check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'} \\ &\overset{(c)}{=} O(\mu_{\max}^2) \end{aligned} \tag{5.148}$$

where step (a) uses property (5.111), step (b) uses (5.141), and step (c) uses (5.143). □

Finally, we present following main theorem that characterizes the difference between the learning curve of $\tilde{\boldsymbol{w}}_{k,i}$ at each agent k and that of $\tilde{w}_{c,i}$ generated by the reference recursion (5.95).

Theorem 5.5 (Learning behavior of $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$)**.** *Suppose the stability condition* (5.145) *holds. Then, the difference between the learning curve of the mean-square-error* $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$ *at each agent* k *and the learning curve of* $\|\tilde{w}_{c,i}\|^2$ *is bounded as*

$$\begin{aligned}
&\big|\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 - \|\tilde{w}_{c,i}\|^2\big| \\
&\quad\leq 2\|u_{L,k}\otimes I_M\|^2\cdot \mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0} \\
&\qquad + 2\|\tilde{w}_{c,0}\|\cdot\|u_{L,k}\otimes I_M\|\cdot\sqrt{\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0}} \\
&\qquad + \gamma_c^i\cdot O(\mu_{\max}^{\frac{1}{2}}) + O(\mu_{\max}) \qquad\qquad \text{for all } i\geq 0
\end{aligned} \tag{5.149}$$

where γ_c^i *was defined earlier in* (5.120)*.*

Proof. See Appendix 5.K. □

5.6.3 Interpretation of Results

The result established in Theorem 5.5 is significant because it allows us to examine the learning behavior of $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$. Note that the third and fourth terms are small for small step-size parameter $\mu_{\max}$. Moreover, the first and second terms in (5.149) converge to zero at the rates of $\rho(\Gamma_e) = |\lambda_2(A)|$, the second largest magnitude eigenvalue of the combination matrix A, and $\sqrt{|\lambda_2(A)|}$, respectively. For sufficiently small step-sizes, these two rates will be faster than the convergence rate of $\|\tilde{w}_{c,i}\|^2$, which is between $1-2\mu_{\max}\|p\|_1\lambda_U$ and $r_c^2 = 1-2\mu_{\max}\lambda_L + o(\lambda_{\max})$ during the initial stages of adaptation and then $[\rho(I_M-\mu_{\max}H_c)]^2$ later on. Therefore, in Transient Phase I, the first and second terms in (5.149) converge to zero at a faster rate than $\|\tilde{w}_{c,i}\|^2$. Then, in Transient Phase II, we have

$$\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \|\tilde{w}_{c,i}\|^2 + O(\mu_{\max}^{1/2})\cdot\gamma_c^i + O(\mu_{\max}) \tag{5.150}$$

Table 5.3: Behavior of error quantities in different phases.

Error quantity	**Transient Phase I**		**Transient Phase II**		**Steady-State** [c]
	Convergence rate r [a]	**Value**	**Convergence rate** r [b]	**Value**	**Value**
$\|\tilde{w}_{c,i}\|^2$	$1-2\mu_{\max}\|p\|_1\lambda_U \le r \le \gamma_c^2$	$\gg O(\mu_{\max})$	$r=[\rho(I_M-\mu_{\max}H_c)]^2$	$\gg O(\mu_{\max})$	0
$\mathbb{E}\|\tilde{\boldsymbol{w}}_{c,i}\|^2$	converged	$O(\mu_{\max})$	converged	$O(\mu_{\max})$	$O(\mu_{\max})$
$\mathbb{E}P[\tilde{\boldsymbol{w}}_{e,i}]$	$r \le \lvert\lambda_2(A)\rvert$	$\gg O(\mu_{\max})$	converged	$O(\mu_{\max}^2)$	$O(\mu_{\max}^2)$
$\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$	Multiple modes	$\gg O(\mu_{\max})$	$r=[\rho(I_M-\mu_{\max}H_c)]^2$	$\gg O(\mu_{\max})$	$O(\mu_{\max})$

[a] γ_c is defined in (5.120), and $\gamma_c^2 = 1-2\mu_{\max}\lambda_L + o(\mu_{\max})$.
[b] We only show the leading term of the convergence rate for r. More precise expression can be found in (5.131).
[c] Closer studies of the steady-state performance can be found in Chapter 6 and [38].

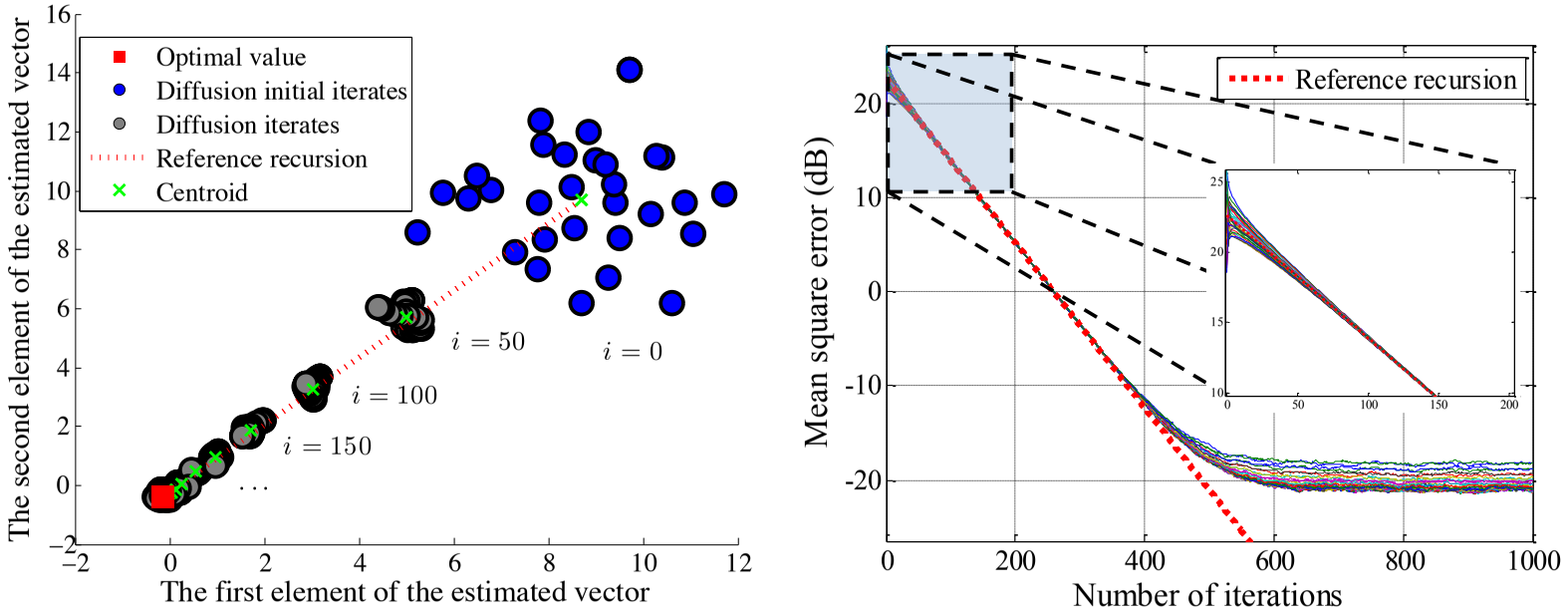


Figure 5.4: The evolution and learning curves of various quantities in a diffusion LMS adaptive network, where $M = 2$, and the regressors are spatially and temporally white, and isotropic across agents. (a) The evolution of the iterates $\{\boldsymbol{w}_{k,i}\}$ at all agents, the centroid $\boldsymbol{w}_{c,i}$, and the reference recursion $\bar{w}_{c,i}$ on the two-dimensional solution space; the horizontal axis and vertical axis are the first and second elements of $\boldsymbol{w}_{k,i}$, respectively. The clusters of $\{\boldsymbol{w}_{k,i}\}$ are plotted every 50 iterations. (b) The MSE learning curves, averaged over 1000 trials, for the iterates $\{\boldsymbol{w}_{k,i}\}$ at all agents, and the reference recursion $\bar{w}_{c,i}$. The zoom-in region shows the learning curves for different agents, which quick shrink together in Phase I.

so that the convergence rate of $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$ is the same as that of $\|\tilde{w}_{c,i}\|^2$ given by (5.131). Afterwards, as $i \to \infty$, we have $\|\tilde{w}_{c,i}\|^2 \to 0$ and taking the lim sup of both sides of (5.149) implies

$$\begin{aligned}\limsup_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 &= \limsup_{i\to\infty} \left|\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 - \|\tilde{w}_{c,i}\|^2\right| \\ &\leq O(\mu_{\max}) \end{aligned} \tag{5.151}$$

We will go a step further and evaluate this steady-state MSE for small step-sizes in Chapter 6. Therefore, $\boldsymbol{w}_{k,i}$ converges to w^o with a small steady-state MSE that is on the order of $O(\mu_{\max})$. And the steady-state MSE can be made arbitrarily small for small step-sizes.

Furthermore, the results established in Theorems 5.1–5.4 reveal the evolution of the three components, $\tilde{w}_{c,i}$, $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$ in (5.67) during the three distinct phases of the learning curve. From (5.140), the centroid $\boldsymbol{w}_{c,i}$ of the distributed algorithm (5.1)–(5.3) stays close to $\bar{w}_{c,i}$ over the entire time for sufficiently small step-sizes since the mean-square error $\mathbb{E}\|\boldsymbol{w}_{c,i} - \bar{w}_{c,i}\|^2 = \mathbb{E}P[\check{\boldsymbol{w}}_{c,i}]$ is always of the order of $O(\mu_{\max})$. However, $\mathcal{W}_{e,0} = \mathbb{E}P[\boldsymbol{w}_{e,i}]$ in (5.141) is not necessarily small at the beginning. This is because, as we pointed out in (5.63) and Fig. 5.3, $\boldsymbol{w}_{e,i}$ characterizes the deviation of the agents from their centroid. If the agents are initialized at different values, then $\mathbb{E}P[\boldsymbol{w}_{e,0}] \neq 0$, and it takes some time for $\mathbb{E}P[\boldsymbol{w}_{e,i}]$ to decay to a small value of $O(\mu_{\max}^2)$. By (5.141), the rate at which $\mathbb{E}P[\boldsymbol{w}_{e,i}]$ decays is $\rho(\Gamma_e) = |\lambda_2(A)|$. On the other hand, recall from Theorems 5.2–5.3 that the error of the reference recursion, $\tilde{w}_{c,i}$ converges at a rate between $1 - 2\mu_{\max}\|p\|_1\lambda_U$ and $r_c^2 = 1 - 2\mu_{\max}\lambda_L + o(\lambda_{\max})$ at beginning and then $[\rho(I_M - \mu_{\max}H_c)]^2$ later on, which is slower than the convergence rate of $\mathbb{E}P[\boldsymbol{w}_{e,i}]$ for small step-size $\mu_{\max}$. Now, returning to relation (5.67):

$$\tilde{\boldsymbol{w}}_{k,i} = \tilde{w}_{c,i} - \check{\boldsymbol{w}}_{c,i} - (u_{L,k} \otimes I_M)\boldsymbol{w}_{e,i} \tag{5.152}$$

this means that during the initial stage of adaptation, the third term in (5.152) decays to $O(\mu_{\max}^2)$ at a faster rate than the first term, although $\tilde{w}_{c,i}$ will eventually converge to zero. Recalling from (5.63) and Fig. 5.3 that $\boldsymbol{w}_{e,i}$ characterizes the deviation of the agents from their centroid, the decay of $\boldsymbol{w}_{e,i}$ implies that the agents are coordinating with each other so that their estimates $\boldsymbol{w}_{k,i}$ are close to the same $\boldsymbol{w}_{c,i}$ — we call this stage Transient Phase I. Moreover, as we just pointed out, the term $\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}]$ is $O(\mu_{\max})$ over the entire time domain so that the second term in (5.152) is always small. This also means that the centroid of the cluster in Fig. 5.3, i.e., $\boldsymbol{w}_{c,i}$, is always close to the reference recursion $\bar{w}_{c,i}$ since $\check{\boldsymbol{w}}_{c,i} = \boldsymbol{w}_{c,i} - \bar{w}_{c,i}$ is always small. Now that $\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2$ is $O(\mu_{\max})$ and $\mathbb{E}P[\boldsymbol{w}_{e,i}]$ is $O(\mu_{\max}^2)$, the error $\tilde{\boldsymbol{w}}_{k,i}$ at each agent k is mainly dominated by the first term, $\tilde{w}_{c,i}$, in (5.152), and the estimates $\{\boldsymbol{w}_{k,i}\}$ at different agents converge together at the same rate as the reference recursion, given by (5.131), to steady-state — we call this stage Transient Phase II. Furthermore, if $\mathcal{W}_{e,0} = 0$, i.e., the iterates $\boldsymbol{w}_{k,i}$ are initialized at the same value (e.g., zero vector), then (5.141) shows that $\mathbb{E}P[\boldsymbol{w}_{e,i}]$ is $O(\mu_{\max}^2)$ over the entire time domain so that the learning dynamics start at Transient Phase II directly. Finally, all agents reach the third phase, steady-state, where $\tilde{w}_{c,i} \to 0$ and $\tilde{\boldsymbol{w}}_{k,i}$ is dominated by the second and third terms in (5.152) so that $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$ becomes $O(\mu_{\max})$. We summarize the above results in Table 5.3 and illustrate the evolution of the quantities in the simulated example in Fig. 5.4. We observe from (5.4) that the radius of the cluster shrinks quickly at the early stage of the transient phase, and then converges towards the optimal solution.

5.6.4 Discussion on the Limit Point and the Fixed Point

We now discuss the relation between the iterate $\boldsymbol{w}_{k,i}$ at agent k, the limit point w^o, and the fixed point $w_{k,\infty}$ (defined in (5.167) and (5.170a)–(5.170c) further ahead). First, recall that w^o is the unique solution to the algebraic equation (5.128):

$$w^o: \quad \sum_{k=1}^{N} p_k s_k(w) = 0 \tag{5.153}$$

To define the fixed point $w_{k,\infty}$, we first introduce the following deterministic recursion, which uses the actual $s_k(w)$ instead of $\hat{\boldsymbol{s}}_{k,i}(w)$ in (5.1)–(5.3):

$$\phi_{k,i-1} = \sum_{l=1}^{N} a_{1,lk} w_{l,i-1} \tag{5.154a}$$

$$\psi_{k,i} = \sum_{l=1}^{N} a_{0,lk} \phi_{l,i-1} - \mu_k s_k(\phi_{k,i-1}) \tag{5.154b}$$

$$w_{k,i} = \sum_{l=1}^{N} a_{2,lk} \psi_{l,i} \tag{5.154c}$$

Introduce the following global vectors and matrices:

$$\phi_i \triangleq \mathrm{col}\{\phi_{1,i}, \ldots, \phi_{N,i}\} \tag{5.155}$$

$$\psi_i \triangleq \mathrm{col}\{\psi_{1,i}, \ldots, \psi_{N,i}\} \tag{5.156}$$

$$w_i \triangleq \mathrm{col}\{w_{1,i}, \ldots, w_{N,i}\} \tag{5.157}$$

$$s(\phi_{i-1}) \triangleq \mathrm{col}\{s_1(\phi_{1,i-1}), \ldots, s_N(\phi_{N,i-1})\} \tag{5.158}$$

$$\mathcal{A}_1 \triangleq A_1 \otimes I_M \tag{5.159}$$

$$\mathcal{A}_2 \triangleq A_2 \otimes I_M \tag{5.160}$$

$$\mathcal{A}_0 \triangleq A_0 \otimes I_M \tag{5.161}$$

$$\mathcal{M} \triangleq \text{diag}\{\mu_1 I_M, \ldots, \mu_N I_M\} \tag{5.162}$$

Then, recursions (5.154a)–(5.154c) can be written as

$$\phi_{i-1} = \mathcal{A}_1 w_{i-1} \tag{5.163}$$

$$\psi_i = \mathcal{A}_0 \phi_{i-1} - \mathcal{M} s(\phi_{i-1}) \tag{5.164}$$

$$w_i = \mathcal{A}_2 \psi_i \tag{5.165}$$

which leads to

$$w_i = \mathcal{A}_2 \mathcal{A}_0 \mathcal{A}_1 w_{i-1} - \mathcal{A}_2 \mathcal{M} s(\mathcal{A}_1 w_{i-1}) \tag{5.166}$$

The fixed point w_∞ of the deterministic recursion (5.166) is the solution to the following algebraic equation:

$$w_\infty = \mathcal{A}_2 \mathcal{A}_0 \mathcal{A}_1 w_\infty - \mathcal{A}_2 \mathcal{M} s(\mathcal{A}_1 w_\infty) \tag{5.167}$$

Note that, if the deterministic recursion (5.166) is stable, then it will converge to the fixed point w_∞. In Chapter 4, we proved that the recursion (5.167) is contractive so that there exists a unique fixed point for the deterministic diffusion recursion ($A_0 = I$ and $s_k(w) = \nabla_w J_k(w)$) under the assumption that each cost function $J_k(w)$ is strongly convex. Proving the result under the general conditions assumed in this chapter could be an interesting extension of the work. Introduce

$$\phi_\infty = \mathcal{A}_1 w_\infty \tag{5.168}$$

$$\psi_\infty = \mathcal{A}_0 \phi_\infty - \mathcal{M} s(\phi_\infty) \tag{5.169}$$

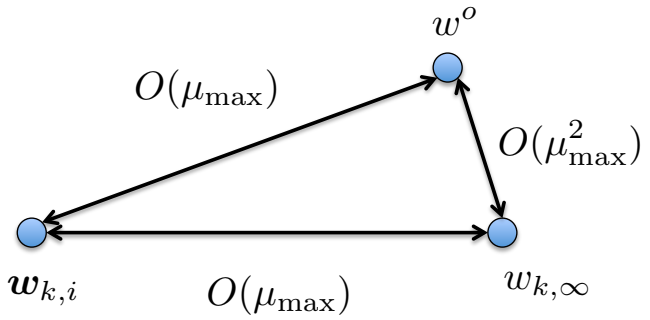


Figure 5.5: Relations between the fixed point $w_{k,\infty}$, the iterate $\boldsymbol{w}_{k,i}$, and the limit point w^o. In steady-state, the mean-square-error between $\boldsymbol{w}_{k,i}$ and w^o is $O(\mu_{\max})$, the mean-square-error between $\boldsymbol{w}_{k,i}$ and $w_{k,\infty}$ is $O(\mu_{\max})$, and the square-error (i.e., the bias) between $w_{k,\infty}$ and w^o is $O(\mu_{\max}^2)$.

and let $\phi_{k,\infty}$, $\psi_{k,\infty}$, and $w_{k,\infty}$ denote the kth sub-vector of ϕ_∞, ψ_∞, and w_∞. Then, the global fixed point equation (5.167) can also be written in the following form for each agent k:

$$\phi_{k,\infty} = \sum_{l=1}^{N} a_{1,lk} w_{l,\infty} \tag{5.170a}$$

$$\psi_{k,\infty} = \sum_{l=1}^{N} a_{0,lk} \phi_{l,\infty} - \mu_k s_k(\phi_{k,\infty}) \tag{5.170b}$$

$$w_{k,\infty} = \sum_{l=1}^{N} a_{2,lk} \psi_{l,\infty} \tag{5.170c}$$

We now proceed to discuss the relationship between $\boldsymbol{w}_{k,i}$, $w_{k,\infty}$, and w^o.

In this chapter, we showed in (5.151) that the iterate $\boldsymbol{w}_{k,i}$ at each agent k is close to w^o in steady-state with mean-square-error being $O(\mu_{\max})$:

$$\limsup_{i\to\infty} \mathbb{E}\|\boldsymbol{w}_{k,i} - w^o\|^2 = O(\mu_{\max}) \tag{5.171}$$

In Chapter 4, we performed the mean-square-error analysis using a different approach for diffusion strategies ($A_0 = I$) with $s_k(w) = \nabla_w J_k(w)$ and $\hat{\boldsymbol{s}}_{k,i}(w) =$

$\widehat{\nabla_w J_k}(w)$ under the assumption that each cost function $J_k(w)$ is strongly convex. Specifically, we analyzed the mean-square-error of $\boldsymbol{w}_{k,i}$ relative the fixed point $w_{k,\infty}$ and show it is $O(\mu_{\max})$ (see (4.72) and (4.47)). Then, by performing bias analysis, we showed that the square-error between the fixed point $w_{k,\infty}$ and the limit point w^o is $O(\mu_{\max}^2)$ (see (4.88)). Combining these two parts together, we arrive at the same result as (5.171) (see (4.92)):

$$\begin{aligned}\limsup_{i\to\infty}\mathbb{E}\|\boldsymbol{w}_{k,i}-w^o\|^2 &= \limsup_{i\to\infty}\mathbb{E}\|(\boldsymbol{w}_{k,i}-w_{k,\infty})+(w_{k,\infty}-w^o)\|^2\\ &\le \limsup_{i\to\infty}\left[2\cdot\mathbb{E}\|\boldsymbol{w}_{k,i}-w_{k,\infty}\|^2\right]+\limsup_{i\to\infty}\left[2\cdot\|w_{k,\infty}-w^o\|^2\right]\\ &= O(\mu_{\max})+O(\mu_{\max}^2)=O(\mu_{\max})\end{aligned}\tag{5.172}$$

The relations between $\boldsymbol{w}_{k,i}$, $w_{k,\infty}$, and w^o are illustrated in Fig. 5.5. In steady-state, the mean-square-error between $\boldsymbol{w}_{k,i}$ and w^o is $O(\mu_{\max})$, the mean-square-error between $\boldsymbol{w}_{k,i}$ and $w_{k,\infty}$ is $O(\mu_{\max})$, and the square-error between $w_{k,\infty}$ and w^o (i.e., the bias) is $O(\mu_{\max}^2)$. When the recursion is a deterministic recursion using the exact $s_k(w)$ instead of $\hat{\boldsymbol{s}}_{k,i}(w)$, then the iterate $w_{k,i}$ (we use regular font instead of $\boldsymbol{w}_{k,i}$ to highlight its deterministic nature in this case) converges to $w_{k,\infty}$ in steady-state. In this case, the error between $w_{k,i}$ and w^o in steady-state becomes the error between $w_{k,\infty}$ and w^o, i.e., the bias, which is $O(\mu_{\max}^2)$.

5.7 Conclusion

In this chapter, we studied the learning behavior of adaptive networks under fairly general conditions. We showed that, in the constant and small step-size regime, a typical learning curve of each agent exhibits three phases: Transient Phase I, Transient Phase II, and Steady-state Phase. A key observation is that,

the second and third phases approach the performance of a centralized strategy. Furthermore, we showed that the right eigenvector of the combination matrix corresponding to the eigenvalue at one influences the limit point, the convergence rate, and the steady-state mean-square-error (MSE) performance in a critical way. Analytical expressions that illustrate these effects were derived. Various implications were discussed and illustrative examples were also considered.

5.A Proof of Lemma 5.1

First, we establish that conditions (5.27) and (5.28) imply (5.24) and (5.25), respectively. Using the mean-value theorem [105, p.6], we have for any $x, y \in \mathcal{S}$:

$$\begin{aligned}\|s_k(x) - s_k(y)\| &= \left\| \int_0^1 \nabla_{w^T} s_k(y + t(x-y))dt \cdot (x-y) \right\| \\ &\leq \int_0^1 \|\nabla_{w^T} s_k(y + t(x-y))\| \, dt \cdot \|x-y\| \\ &\leq \lambda_U \cdot \|x-y\| \end{aligned} \tag{5.173}$$

where we used the fact that $y + t(x-y) = tx + (1-t)y \in \mathcal{S}$ given $x, y \in \mathcal{S}$ and $0 \leq t \leq 1$. Likewise, we have

$$\begin{aligned}&(x-y)^T \cdot \sum_{k=1}^N p_k [s_k(x) - s_k(y)] \\ &= (x-y)^T \cdot \sum_{k=1}^N p_k \int_0^1 \nabla_{w^T} s_k(y + t(x-y))dt \cdot (x-y) \\ &= (x-y)^T \cdot \int_0^1 \sum_{k=1}^N p_k \nabla_{w^T} s_k(y + t(x-y))dt \cdot (x-y) \\ &\overset{(6.22)}{=} (x-y)^T \cdot H_c(y + t(x-y)) \cdot (x-y) \\ &= (x-y)^T \cdot \frac{H_c(y + t(x-y)) + H_c^T(y + t(x-y))}{2} \cdot (x-y)\end{aligned}$$

$$\geq \lambda_L \cdot \|x - y\|^2 \tag{5.174}$$

Next, we establish the reverse direction that conditions (5.24) and (5.25) imply (5.27) and (5.28). Choosing $x = w + t \cdot \delta w$ and $y = w$ in (5.24) for any $\delta w \neq 0$ and any small positive t, we get

$$\begin{aligned}
&\|s_k(w + t \cdot \delta w) - s_k(w)\| \leq t \cdot \lambda_U \cdot \|\delta w\| \\
\Rightarrow \quad &\lim_{t \to 0^+} \left\| \frac{s_k(w + t \cdot \delta w) - s_k(w)}{t} \right\| \leq \lambda_U \cdot \|\delta w\| \\
\Rightarrow \quad &\left\| \lim_{t \to 0^+} \frac{s_k(w + t \cdot \delta w) - s_k(w)}{t} \right\| \leq \lambda_U \cdot \|\delta w\| \\
\Rightarrow \quad &\|\nabla_{w^T} s_k(w) \delta w\| \leq \lambda_U \cdot \|\delta w\| \\
\Rightarrow \quad &\|\nabla_{w^T} s_k(w)\| \triangleq \sup_{\delta w \neq 0} \frac{\|\nabla_{w^T} s_k(w) \delta w\|}{\|\delta w\|} \leq \lambda_U
\end{aligned} \tag{5.175}$$

Likewise, choosing $x = w + t \cdot \delta w$ and $y = w$ in (5.25) for any $\delta w \neq 0$ and any small positive t, we obtain

$$\begin{aligned}
&t \cdot \delta w^T \cdot \sum_{k=1}^{N} p_k [s_k(w + t \cdot \delta w) - s_k(w)] \geq t^2 \cdot \lambda_L \cdot \|\delta w\|^2 \\
&\quad \Rightarrow \quad \delta w^T \cdot \sum_{k=1}^{N} p_k \left(\lim_{t \to 0^+} \frac{s_k(w + t \cdot \delta w) - s_k(w)}{t} \right) \\
&\qquad\qquad \geq \lambda_L \cdot \|\delta w\|^2 \\
&\quad \Rightarrow \quad \delta w^T \cdot \sum_{k=1}^{N} p_k \nabla_{w^T} s_k(w) \cdot \delta w \geq \lambda_L \cdot \|\delta w\|^2 \\
&\quad \Rightarrow \quad \delta w^T H_c(w) \delta w \geq \lambda_L \cdot \|\delta w\|^2 \\
&\quad \Rightarrow \quad \delta w^T \frac{H_c(w) + H_c^T(w)}{2} \delta w \geq \lambda_L \cdot \|\delta w\|^2 \\
&\quad \Rightarrow \quad \frac{H_c(w) + H_c^T(w)}{2} \geq \lambda_L \cdot I_M
\end{aligned} \tag{5.176}$$

5.B Proof of Lemma 5.4

Properties 1-2 are straightforward from the definitions of $P[\cdot]$ and $\bar{P}[\cdot]$. Property 4 was proved in Chapter 4. We establish the remaining properties.

(***Property 3: Convexity***) The convexity of $P[\cdot]$ has already been proven in Chapter 4. We now establish the convexity of the operator $\bar{P}[\cdot]$. Let $X_{qn}^{(k)}$ denote the $(q,n)-$th $M \times M$ block of the matrix $X^{(k)}$, where $q = 1, \ldots, K$ and $n = 1, \ldots, N$. Then,

$$\begin{aligned}
&\bar{P}\left[\sum_{k=1}^{K} a_k X^{(k)}\right] \\
&\quad \preceq \begin{bmatrix} \sum_{k=1}^{K} a_k \left\| X_{11}^{(k)} \right\| & \cdots & \sum_{k=1}^{K} a_k \left\| X_{1N}^{(k)} \right\| \\ \vdots & & \vdots \\ \sum_{k=1}^{K} a_k \left\| X_{K1}^{(k)} \right\| & \cdots & \sum_{k=1}^{K} a_k \left\| X_{KN}^{(k)} \right\| \end{bmatrix} \\
&\quad = \sum_{k=1}^{K} a_k \bar{P}[X^{(k)}]
\end{aligned} \tag{5.177}$$

(***Property 5: Triangular inequality***) Let X_{qn} and Y_{qn} denote the $(q,n)-$th $M \times M$ blocks of the matrices X and Y, respectively, where $q = 1, \ldots, K$ and $n = 1, \ldots, N$. Then, by the triangular inequality of the matrix norm $\|\cdot\|$, we have

$$\begin{aligned}
\bar{P}\left[X+Y\right] &\preceq \begin{bmatrix} \|X_{11}\| + \|Y_{11}\| & \cdots & \|X_{1N}\| + \|Y_{1N}\| \\ \vdots & & \vdots \\ \|X_{K1}\| + \|Y_{K1}\| & \cdots & \|X_{KN}\| + \|Y_{KN}\| \end{bmatrix} \\
&= \bar{P}[X] + \bar{P}[Y]
\end{aligned} \tag{5.178}$$

(*Property 6: Submultiplicity*) Let X_{kn} and Z_{nl} be the $(k,n)-$th and $(n,l)-$th $M \times M$ blocks of X and Z, respectively. Then, the $(k,l)-$th $M \times M$ block of the matrix product XZ, denoted by $[XZ]_{k,l}$, is

$$[XZ]_{k,l} = \sum_{n=1}^{N} X_{kn} Z_{nl} \tag{5.179}$$

Therefore, the $(k,l)-$th entry of the matrix $\bar{P}[XZ]$ can be bounded as

$$\left[\bar{P}[XZ]\right]_{k,l} = \left\| \sum_{n=1}^{N} X_{kn} Z_{nl} \right\| \leq \sum_{n=1}^{N} \|X_{kn}\| \cdot \|Z_{nl}\| \tag{5.180}$$

Note that $\|X_{kn}\|$ and $\|Z_{nl}\|$ are the $(k,n)-$th and $(n,l)-$th entries of the matrices $\bar{P}[X]$ and $\bar{P}[Z]$, respectively. The right-hand side of the above inequality is therefore the $(k,l)-$th entry of the matrix product $\bar{P}[X]\bar{P}[Z]$. Therefore, we obtain

$$\bar{P}[XZ] \preceq \bar{P}[X]\bar{P}[Z] \tag{5.181}$$

(*Property 7: Kronecker structure*) For (5.107), we note that the $(k,n)-$th $M \times M$ block of $X \otimes I_M$ is $x_{kn} I_M$. Therefore, by the definition of $\bar{P}[\cdot]$, we have

$$\bar{P}[X \otimes I_M] = \begin{bmatrix} |x_{11}| & \cdots & |x_{1N}| \\ \vdots & & \\ |x_{K1}| & \cdots & |x_{KN}| \end{bmatrix} = \bar{P}_1[X] \tag{5.182}$$

In the special case when X consists of nonnegative entries, $\bar{P}_1[X] = X$, and we recover (5.109). To prove (5.108), we let $a = \text{col}\{a_1, \ldots, a_N\}$ and $b =$

$\text{col}\{b_1, \ldots, b_M\}$. Then, by the definitin of $P[\cdot]$, we have

$$P[a \otimes b] = \text{col}\{|a_1|^2 \cdot \|b\|^2, \ldots, |a_N|^2 \cdot \|b\|^2\} = \|b\|^2 \cdot P_1[a] \tag{5.183}$$

(***Property 8: Relation to norms***) Relations (5.110) and (5.111) are straightforward and follow from the definition.

(***Property 9: Inequality preservation***) The proof that $x \preceq y$ implies $Fx \preceq Fy$ can be found in Chapter 4. We now prove that $F \preceq G$ implies $Fx \preceq Gx$. This can be proved by showig that $(G - F)x \succeq 0$, which is true because all entries of $G - F$ and x are nonnegative due to $F \preceq G$ and $x \succeq 0$.

(***Property 10: Upper bounds***) By the definition of $\bar{P}[X]$ in (5.101), we get

$$\begin{aligned}
\bar{P}[X] &\preceq \left(\max_{l,k} \|X_{lk}\| \right) \cdot \mathbb{1}\mathbb{1}^T \\
&\preceq \max_{l} \left(\sum_{k=1}^{N} \|X_{lk}\| \right) \cdot \mathbb{1}\mathbb{1}^T \\
&= \|\bar{P}[X]\|_\infty \cdot \mathbb{1}\mathbb{1}^T
\end{aligned} \tag{5.184}$$

Likewise, we can establish that $\bar{P}[X] \preceq \|\bar{P}[X]\|_1 \cdot \mathbb{1}\mathbb{1}^T$.

5.C Proof of Lemma 5.5

(***Property 1: Linear transformation***) Let Q_{kn} be the (k, n)-th $M \times M$ block of Q. Then

$$P[Qx] = \text{col}\left\{ \left\| \sum_{n=1}^{N} Q_{1n} x_n \right\|^2, \ldots, \left\| \sum_{n=1}^{N} Q_{Kn} x_n \right\|^2 \right\} \tag{5.185}$$

Using the convexity of $\|\cdot\|^2$, we have the following bound on each n-th entry:

$$\begin{aligned}
\left\|\sum_{n=1}^{N} Q_{kn}x_n\right\|^2 &\overset{(a)}{=} \left[\sum_{n=1}^{N}\|Q_{kn}\|\right]^2 \cdot \left\|\sum_{n=1}^{N}\frac{\|Q_{kn}\|}{\sum_{l=1}^{N}\|Q_{kl}\|}\cdot\frac{Q_{kn}}{\|Q_{kn}\|}x_n\right\|^2 \\
&\overset{(b)}{\leq} \left[\sum_{n=1}^{N}\|Q_{kn}\|\right]^2 \cdot \sum_{n=1}^{N}\frac{\|Q_{kn}\|}{\sum_{l=1}^{N}\|Q_{kl}\|}\cdot\frac{\|Q_{kn}\|^2}{\|Q_{kn}\|^2}\|x_n\|^2 \\
&= \left[\sum_{n=1}^{N}\|Q_{kn}\|\right]\cdot\sum_{n=1}^{N}\|Q_{kn}\|\cdot\|x_n\|^2 \\
&\leq \max_k\left[\sum_{n=1}^{N}\|Q_{kn}\|\right]\cdot\sum_{n=1}^{N}\|Q_{kn}\|\cdot\|x_n\|^2 \\
&= \|\bar{P}[Q]\|_\infty\cdot\sum_{n=1}^{N}\|Q_{kn}\|\cdot\|x_n\|^2 \qquad (5.186)
\end{aligned}$$

where in step (b) we applied Jensen's inequlity to $\|\cdot\|^2$. Note that if some $\|Q_{kn}\|$ in step (a) is zero, we eliminate the corresponding term from the sum and it can be verified that the final result still holds. Substituting into (5.185), we establish (5.114). The special case (5.116) can be obtained by using $\bar{P}[A^T\otimes I_M] = A^T$ and that $\|A^T\|_\infty = 1$ (left-stochastic) in (5.114). Finally, the upper bound (5.115) can be proved by applying (5.113) to $\bar{P}[Q]$.

(*Property 2: Update operation*) By the definition of $P[\cdot]$ and the Lipschitz Assumption 5.3, we have

$$\begin{aligned}
P[s(x)-s(y)] &= \mathrm{col}\{\|s_1(x_1)-s_1(y_1)\|^2,\ldots\|s_N(x_N)-s_N(y_N)\|^2\} \\
&\preceq \mathrm{col}\{\lambda_U^2\cdot\|x_1-y_1\|^2,\ldots\lambda_U^2\cdot\|x_N-y_N\|^2\} \\
&= \lambda_U^2\cdot P[x-y] \qquad (5.187)
\end{aligned}$$

(***Property 3: Centralized opertion***) Since $T_c : \mathbb{R}^M \to \mathbb{R}^M$, the output of $P[T_c(x) - T_c(y)]$ becomes a scalar. From the definition, we get

$$\begin{aligned}
P\left[T_c(x) - T_c(y)\right] &= \left\| x - y - \mu_{\max} \cdot \left(p^T \otimes I_M\right) \left[s(\mathbb{1} \otimes x) - s(\mathbb{1} \otimes y)\right] \right\|^2 \\
&= \left\| x - y - \mu_{\max} \cdot \sum_{k=1}^{N} p_k \left[s_k(x) - s_k(y)\right] \right\|^2 \\
&= \|x - y\|^2 - 2\mu_{\max} \cdot (x - y)^T \sum_{k=1}^{N} p_k \left[s_k(x) - s_k(y)\right] \\
&\quad + \mu_{\max}^2 \left\| \left(p^T \otimes I_M\right) \left[s(\mathbb{1} \otimes x) - s(\mathbb{1} \otimes y)\right] \right\|^2 \\
&= \|x - y\|^2 - 2\mu_{\max} \cdot (x - y)^T \sum_{k=1}^{N} p_k \left[s_k(x) - s_k(y)\right] \\
&\quad + \mu_{\max}^2 P\left[\left(p^T \otimes I_M\right) \left[s(\mathbb{1} \otimes x) - s(\mathbb{1} \otimes y)\right]\right] \qquad (5.188)
\end{aligned}$$

We first prove the upper bound (5.118) as follows:

$$\begin{aligned}
P\left[T_c(x) - T_c(y)\right] &= \left\| x - y - \mu_{\max} \cdot \sum_{k=1}^{N} p_k [s_k(x) - s_k(y)] \right\|^2 \\
&= \|x - y\|^2 - 2\mu_{\max} \cdot (x - y)^T \sum_{k=1}^{N} p_k [s_k(x) - s_k(y)] \\
&\quad + \mu_{\max}^2 \cdot \Big\| \sum_{k=1}^{N} p_k [s_k(x) - s_k(y)] \Big\|^2 \\
&\overset{(6.18)}{\leq} \|x - y\|^2 - 2\mu_{\max} \cdot \lambda_L \cdot \|x - y\|^2 \\
&\quad + \mu_{\max}^2 \cdot \Big\| \sum_{k=1}^{N} p_k [s_k(x) - s_k(y)] \Big\|^2 \\
&\leq \|x - y\|^2 - 2\mu_{\max} \cdot \lambda_L \cdot \|x - y\|^2 \\
&\quad + \mu_{\max}^2 \cdot \Big[\sum_{k=1}^{N} p_k \|s_k(x) - s_k(y)\| \Big]^2 \\
&\overset{(5.24)}{\leq} \|x - y\|^2 - 2\mu_{\max} \cdot \lambda_L \cdot \|x - y\|^2
\end{aligned}$$

$$
\begin{aligned}
&+ \mu_{\max}^2 \cdot \left[\sum_{k=1}^{N} p_k \cdot \lambda_U \cdot \|x-y\|\right]^2 \\
&= \|x-y\|^2 - 2\mu_{\max} \cdot \lambda_L \cdot \|x-y\|^2 \\
&\quad + \mu_{\max}^2 \cdot \|p\|_1^2 \lambda_U^2 \cdot \|x-y\|^2 \\
&= (1 - 2\mu_{\max}\lambda_L + \mu_{\max}^2 \lambda_U^2 \|p\|_1^2) \cdot \|x-y\|^2 \\
&\leq \left(1 - \mu_{\max}\lambda_L + \frac{1}{2}\mu_{\max}^2 \lambda_U^2 \|p\|_1^2\right)^2 \cdot \|x-y\|^2 \qquad (5.189)
\end{aligned}
$$

where in the last step we used the relation $(1-x) \leq (1-\frac{1}{2}x)^2$.

Next, we prove the lower bound (5.119). From (5.188), we notice that the last term in (5.188) is always nonnegative so that

$$
\begin{aligned}
P\left[T_c(x) - T_c(y)\right] &\succeq \|x-y\|^2 - 2\mu_{\max} \cdot (x-y)^T \sum_{k=1}^{N} p_k \left[s_k(x) - s_k(y)\right] \\
&\overset{(a)}{\succeq} \|x-y\|^2 - 2\mu_{\max} \cdot \|x-y\| \cdot \Big\| \sum_{k=1}^{N} p_k \left[s_k(x) - s_k(y)\right] \Big\| \\
&\succeq \|x-y\|^2 - 2\mu_{\max} \cdot \|x-y\| \cdot \sum_{k=1}^{N} p_k \|s_k(x) - s_k(y)\| \\
&\overset{(b)}{\succeq} \|x-y\|^2 - 2\mu_{\max} \cdot \|x-y\| \cdot \sum_{k=1}^{N} p_k \lambda_U \|x-y\| \\
&= (1 - 2\mu_{\max}\lambda_U \|p\|_1) \cdot \|x-y\|^2 \qquad (5.190)
\end{aligned}
$$

where in step (a), we used the Cauchy-Schwartz inequality $x^T y \leq |x^T y| \leq \|x\| \cdot \|y\|$, and in step (b) we used (5.24).

(***Property 4: Stable Jordan operator***) First, we notice that matrix $D_{L,n}$ can be written as

$$
D_{L,n} = d_n \cdot I_{L_n} + \Theta_{L_n} \qquad (5.191)
$$

where Θ_{L_n} is an $L_n \times L_n$ strictly upper triangular matrix of the following form:

$$\Theta_{L_n} \triangleq \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{bmatrix} \tag{5.192}$$

Define the following matrices:

$$\Lambda_L \triangleq \text{diag}\{d_2 I_{L_2}, \ldots, d_{n_0} I_{L_{n_0}}\} \tag{5.193}$$

$$\Theta'_L \triangleq \text{diag}\{\Theta_{L_2}, \ldots, \Theta_{L_{n_0}}\} \tag{5.194}$$

Then, the original Jordan matrix D_L can be expressed as

$$D_L = \Lambda_L + \Theta'_L \tag{5.195}$$

so that

$$\begin{aligned} P_1[D_L x' + y'] &= P_1\left[\Lambda_L x' + \Theta'_L x' + y'\right] \\ &= P_1\Bigg[|d_2| \cdot \frac{1}{|d_2|}\Lambda_L x' + \frac{1-|d_2|}{2} \cdot \frac{2}{1-|d_2|}\Theta'_L x' \\ &\qquad + \frac{1-|d_2|}{2} \cdot \frac{2}{1-|d_2|} y'\Bigg] \\ &\overset{(a)}{\preceq} |d_2| \cdot P_1\left[\frac{1}{|d_2|}\Lambda_L x'\right] + \frac{1-|d_2|}{2} \cdot P_1\left[\frac{2}{1-|d_2|}\Theta'_L x'\right] \\ &\quad + \frac{1-|d_2|}{2} \cdot P_1\left[\frac{2}{1-|d_2|} y'\right] \\ &\overset{(b)}{=} \frac{1}{|d_2|} \cdot P_1\left[\Lambda_L x'\right] + \frac{2}{1-|d_2|} \cdot P_1\left[\Theta'_L x'\right] + \frac{2}{1-|d_2|} \cdot P_1\left[y'\right] \\ &\overset{(c)}{\preceq} \frac{\|\bar{P}_1[\Lambda_L]\|_\infty}{|d_2|} \cdot \bar{P}_1[\Lambda_L] \cdot P_1\left[x'\right] \end{aligned}$$

$$
\begin{aligned}
&+\frac{2\|\bar{P}_1[\Theta'_L]\|_\infty}{1-|d_2|}\cdot\bar{P}_1[\Theta'_L]\cdot P_1\left[x'\right]+\frac{2}{1-|d_2|}\cdot P_1\left[y'\right]\\
&\overset{(d)}{\preceq}\bar{P}_1[\Lambda_L]\cdot P_1\left[x'\right]+\frac{2}{1-|d_2|}\cdot\Theta'_L\cdot P_1\left[x'\right]+\frac{2}{1-|d_2|}\cdot P_1\left[y'\right]\\
&\overset{(e)}{\preceq}|d_2|\cdot I_L\cdot P_1\left[x'\right]+\frac{2}{1-|d_2|}\cdot\Theta_L\cdot P_1\left[x'\right]+\frac{2}{1-|d_2|}\cdot P_1\left[y'\right]\\
&\overset{(f)}{=}\Gamma_e\cdot P_1\left[x'\right]+\frac{2}{1-|d_2|}\cdot P_1\left[y'\right] \qquad (5.196)
\end{aligned}
$$

where step (a) uses the convexity property (5.102), step (b) uses the scaling property, step (c) uses variance relation (5.114), step (d) uses $\|\bar{P}_1[\Lambda_L]\|_\infty = |d_2|$, $\bar{P}_1[\Theta'_L] = \Theta'_L$ and $\|\bar{P}_1[\Theta'_L]\|_\infty = \|\Theta'_L\|_\infty = 1$, step (e) uses $\bar{P}_1[\Lambda_L] \preceq |d_2|\cdot I_L$ and $\Theta'_L \preceq \Theta_L$, where Θ_L denotes a matrix of the same form as (5.192) but of size $L\times L$, step (f) uses the definition of the matrix Γ_e in (5.125). The above derivation assumes $|d_2|\neq 0$. When $|d_2|=0$, we can verify that the above inequality still holds. To see this, we first notice that when $|d_2|=0$, the relation $0\le|d_{n_0}|\le\cdots\le|d_2|$ implies that $d_{n_0}=\cdots=d_2=0$ so that $\Lambda_L=0$ and $D_L=\Theta'_L$ — see (5.193) and (5.195). Therefore, similar to the steps (a)–(f) in (5.196), we get

$$
\begin{aligned}
P_1[D_Lx'+y'] &= P_1[\Theta'_Lx'+y']\\
&= P_1\big[\frac{1}{2}\cdot 2\Theta'_Lx'+\frac{1}{2}\cdot 2y'\big]\\
&\preceq \frac{1}{2}\cdot P_1[2\Theta'_Lx']+\frac{1}{2}\cdot P_1[2y']\\
&= \frac{1}{2}\cdot 2^2\cdot P_1[\Theta'_Lx']+\frac{1}{2}\cdot 2^2\cdot P_1[y']\\
&= 2P_1[\Theta'_Lx']+2P_1[y']\\
&\preceq 2\|\bar{P}_1[\Theta'_L]\|_\infty\cdot\bar{P}_1[\Theta'_L]P_1[x']+2P_1[y']\\
&= 2\Theta'_LP_1[x']+2P_1[y']\\
&\preceq 2\Theta_LP_1[x']+2P_1[y'] \qquad (5.197)
\end{aligned}
$$

By (5.125), we have $\Gamma_e = 2\Theta_L$ when $|d_2| = 0$. Therefore, the above expression is the same as the one on the right-hand side of (5.196).

(***Property 5: Stable Kronecker Jordan operator***) Using (5.195) we have

$$
\begin{aligned}
P\left[\mathcal{D}_L x_e + y_e\right] &= P\left[(\Lambda_L \otimes I_M)x_e + (\Theta'_L \otimes I_M)x_e + y_e\right] \\
&= P\Big[|d_2| \cdot \frac{1}{|d_2|} \cdot (\Lambda_L \otimes I_M)x_e + \frac{1-|d_2|}{2} \cdot \frac{2}{1-|d_2|} \cdot (\Theta'_L \otimes I_M)x_e \\
&\qquad + \frac{1-|d_2|}{2} \cdot \frac{2}{1-|d_2|} \cdot y_e\Big] \\
&\overset{(a)}{\preceq} |d_2| \cdot P\Big[\frac{1}{|d_2|} \cdot (\Lambda_L \otimes I_M)x_e\Big] \\
&\quad + \frac{1-|d_2|}{2} \cdot P\Big[\frac{2}{1-|d_2|} \cdot (\Theta'_L \otimes I_M)x_e\Big] \\
&\quad + \frac{1-|d_2|}{2} \cdot P\Big[\frac{2}{1-|d_2|} \cdot y_e\Big] \\
&\overset{(b)}{=} \frac{1}{|d_2|} \cdot P[(\Lambda_L \otimes I_M)x_e] + \frac{2}{1-|d_2|} \cdot P[(\Theta'_L \otimes I_M)x_e] \\
&\quad + \frac{2}{1-|d_2|} \cdot P[y_e] \\
&\overset{(c)}{\preceq} \frac{\|\bar{P}[(\Lambda_L \otimes I_M)]\|_\infty}{|d_2|} \cdot \bar{P}[(\Lambda_L \otimes I_M)] \cdot P[x_e] \\
&\quad + \frac{2\|\bar{P}[\Theta'_L \otimes I_M]\|_\infty}{1-|d_2|} \cdot \bar{P}[\Theta'_L \otimes I_M] \cdot P[x_e] + \frac{2}{1-|d_2|} \cdot P[y_e] \\
&\overset{(d)}{\preceq} \bar{P}[(\Lambda_L \otimes I_M)] \cdot P[x_e] + \frac{2}{1-|d_2|} \cdot \Theta'_L \cdot P[x_e] \\
&\quad + \frac{2}{1-|d_2|} \cdot P[y_e] \\
&\overset{(e)}{\preceq} |d_2| \cdot I_L \cdot P[x_e] + \frac{2}{1-|d_2|} \cdot \Theta_L \cdot P[x_e] + \frac{2}{1-|d_2|} \cdot P[y_e] \\
&\overset{(f)}{=} \Gamma_e \cdot P[x_e] + \frac{2}{1-|d_2|} \cdot P[y_e]
\end{aligned}
\tag{5.198}
$$

where step (a) uses the convexity property (5.102), step (b) uses the scaling

property, step (c) uses variance relation (5.114), step (d) uses $\|\bar{P}[\Lambda_L \otimes I_M]\|_\infty = |d_2|$ and $\bar{P}[\Theta'_L \otimes I_M] = \Theta'_L$, step (e) uses $\bar{P}[\Lambda_L \otimes I_M] \preceq |d_2| \cdot I_L$ and $\Theta'_L \preceq \Theta_L$, and step (f) uses the definition of the matrix Γ_e in (5.125). Likewise, we can also verify that the above inequalty holds for the case $|d_2| = 0$.

5.D Proof of Theorem 5.1

Consider the following operator:

$$T_0(w) \triangleq w - \frac{\lambda_L}{\|p\|_1^2 \lambda_U^2} \sum_{k=1}^{N} p_k s_k(w) \tag{5.199}$$

As long as we are able to show that $T_0(w)$ is a strict contraction mapping, i.e., $\forall\, x, y,\ \|T_0(x) - T_0(y)\| \leq \gamma_0 \|x - y\|$ with $\gamma_o < 1$, then we can invoke the Banach fixed point theorem [80, pp.299-300] to conclude that there exists a unique w^o such that $w^o = T_0(w^o)$, i.e.,

$$w^o = w^o - \frac{\lambda_L}{\|p\|_1^2 \lambda_U^2} \sum_{k=1}^{N} p_k s_k(w^o) \ \Leftrightarrow\ \sum_{k=1}^{N} p_k s_k(w^o) = 0 \tag{5.200}$$

as desired. Now, to show that $T_0(\cdot)$ defined in (5.199) is indeed a contraction, we compare $T_0(\cdot)$ with $T_c(\cdot)$ in (5.92) and observe that $T_0(w)$ has the same form as $T_c(\cdot)$ if we set $\mu_{\max} = \frac{\lambda_L}{\|p\|_1^2 \lambda_U^2}$ in (5.92). Therefore, calling upon property (5.118) and using $\mu_{\max} = \frac{\lambda_L}{\|p\|_1^2 \lambda_U^2}$ in the expression for γ_c in (5.120), we obtain

$$\begin{aligned} P[T_0(x) - T_0(y)] &\preceq \left(1 - \frac{\lambda_L}{\|p\|_1^2 \lambda_U^2} \lambda_L + \frac{1}{2} \left(\frac{\lambda_L}{\|p\|_1^2 \lambda_U^2}\right)^2 \|p\|_1^2 \lambda_U^2\right)^2 \cdot P[x - y] \\ &= \left(1 - \frac{1}{2} \frac{\lambda_L^2}{\|p\|_1^2 \lambda_U^2}\right)^2 \cdot P[x - y] \end{aligned} \tag{5.201}$$

By the definition of $P[\cdot]$ in (5.99), the above inequality is equivalent to

$$\|T_0(x) - T_0(y)\|^2 \leq \left(1 - \frac{1}{2}\frac{\lambda_L^2}{\|p\|_1^2 \lambda_U^2}\right)^2 \cdot \|x - y\|^2 \tag{5.202}$$

It remains to show that $|1 - \lambda_L^2/(2\|p\|_1^2\lambda_U^2)| < 1$. By (5.26) and the fact that λ_L, $\|p\|_1^2$ and λ_U^2 are positive, we have

$$\frac{1}{2} < 1 - \frac{1}{2}\frac{\lambda_L^2}{\|p\|_1^2 \lambda_U^2} < 1 \tag{5.203}$$

Therefore, $T_0(w)$ is a strict contraction mapping.

5.E Proof of Theorem 5.2

By Theorem 5.1, w^o is the unique solution to equation (5.128). Subtracting both sides of (5.128) from w^o, we recognize that w^o is also the unique solution to the following equation:

$$w^o = w^o - \mu_{\max} \sum_{k=1}^{N} p_k s_k(w^o) \tag{5.204}$$

so that $w^o = T_c(w^o)$. Applying property (5.118), we obtain

$$\begin{aligned}
\|\tilde{w}_{c,i}\|^2 &= P[w^o - \bar{w}_{c,i}] \\
&= P[T_c(w^o) - T_c(\bar{w}_{c,i-1})] \\
&\preceq \gamma_c^2 \cdot P[w^o - \bar{w}_{c,i-1}] \\
&\preceq \gamma_c^{2i} \cdot P[w^o - \bar{w}_{c,0}] \\
&= \gamma_c^{2i} \cdot \|\tilde{w}_{c,0}\|^2
\end{aligned} \tag{5.205}$$

Since $\gamma_c > 0$, the upper bound on the right-hand side will converge to zero if $\gamma_c < 1$. From its definition (5.120), this condition is equivalent to requiring

$$1 - \mu_{\max}\lambda_L + \frac{1}{2}\mu_{\max}^2 \|p\|_1^2 \lambda_U^2 < 1 \tag{5.206}$$

Solving the above quadratic inequality in $\mu_{\max}$, we obtain (5.130). On the other hand, to prove the lower bound in (5.130), we apply (5.119) and obtain

$$\begin{aligned}
\|\tilde{w}_{c,i}\|^2 &= P[w^o - \bar{w}_{c,i}] \\
&= P[T_c(w^o) - T_c(\bar{w}_{c,i-1})] \\
&\succeq (1 - 2\mu_{\max}\|p\|_1\lambda_U) \cdot P[w^o - \bar{w}_{c,i-1}] \\
&\succeq (1 - 2\mu_{\max}\|p\|_1\lambda_U)^i \cdot P[w^o - \bar{w}_{c,0}] \\
&= (1 - 2\mu_{\max}\|p\|_1\lambda_U)^i \cdot \|\tilde{w}_{c,0}\|^2
\end{aligned} \tag{5.207}$$

5.F Proof of Theorem 5.3

Since (5.129) already establishes that $\bar{w}_{c,i}$ approaches w^o asymptotically (so that $\tilde{w}_{c,i} \to 0$), and since from Assumption 5.5 we know that $s_k(w)$ is differentiable when $\|\tilde{w}_{c,i}\| \leq r_H$ for large enough i, we are justified to use the mean-value theorem [105, p.24] to obtain the following useful relation:

$$\begin{aligned}
&s_k(\bar{w}_{c,i-1}) - s_k(w^o) \\
&\quad = -\left[\int_0^1 \nabla_{w^T} s_k(w^o - t\tilde{w}_{c,i-1})dt\right] \tilde{w}_{c,i-1} \\
&\quad = -\nabla_{w^T} s_k(w^o) \cdot \tilde{w}_{c,i-1} \\
&\qquad - \int_0^1 \left[\nabla_{w^T} s_k(w^o - t\tilde{w}_{c,i-1}) - \nabla_{w^T} s_k(w^o)\right] dt \cdot \tilde{w}_{c,i-1}
\end{aligned} \tag{5.208}$$

Therefore, subtracting w^o from both sides of (5.40) and using (5.128) we get,

$$\begin{aligned}\tilde{w}_{c,i} &= \tilde{w}_{c,i-1} + \mu_{\max}\sum_{k=1}^{N} p_k(s_k(\bar{w}_{c,i-1}) - s_k(w^o)) \\ &= [I - \mu_{\max}H_c]\tilde{w}_{c,i-1} - \mu_{\max}\cdot e_{i-1} \end{aligned} \tag{5.209}$$

where

$$H_c \triangleq \sum_{k=1}^{N} p_k \nabla_{w^T} s_k(w^o) \tag{5.210}$$

$$e_{i-1} \triangleq \sum_{k=1}^{N} p_k \int_0^1 \big[\nabla_{w^T} s_k(w^o - t\tilde{w}_{c,i-1}) - \nabla_{w^T} s_k(w^o)\big]dt \cdot \tilde{w}_{c,i-1} \tag{5.211}$$

Furthermore, the perturbation term e_{i-1} satisfies the following bound:

$$\begin{aligned}\|e_{i-1}\| &\leq \sum_{k=1}^{N} p_k \int_0^1 \big\|\nabla_{w^T} s_k(w^o - t\tilde{w}_{c,i-1}) - \nabla_{w^T} s_k(w^o)\big\|dt \cdot \|\tilde{w}_{c,i-1}\| \\ &\leq \sum_{k=1}^{N} p_k \int_0^1 \lambda_H \cdot t \cdot \|\tilde{w}_{c,i-1}\|dt \cdot \|\tilde{w}_{c,i-1}\| \\ &= \frac{1}{2}\|p\|_1 \lambda_H \cdot \|\tilde{w}_{c,i-1}\|^2 \end{aligned} \tag{5.212}$$

Evaluating the weighted Euclidean norm of both sides of (5.209), we get

$$\|\tilde{w}_{c,i}\|_{\Sigma}^2 = \|\tilde{w}_{c,i-1}\|_{B_c^T\Sigma B_c}^2 - 2\mu_{\max}\cdot\tilde{w}_{c,i-1}^T B_c^T \Sigma e_{i-1} + \mu_{\max}^2 \cdot \|e_{i-1}\|_{\Sigma}^2 \tag{5.213}$$

where

$$B_c = I - \mu_{\max}H_c \tag{5.214}$$

Moreover, $\|x\|_{\Sigma}^2 = x^T\Sigma x$, and Σ is an arbitrary positive semi-definite weighting matrix. The second and third terms on the right-hand side of (5.213) satisfy the following bounds:

$$\begin{aligned}\left|\tilde{w}_{c,i-1}^T B_c^T \Sigma e_{i-1}\right| &\overset{(a)}{\leq} \|\tilde{w}_{c,i-1}\| \cdot \|B_c^T\| \cdot \|\Sigma\| \cdot \|e_{i-1}\| \\ &\leq \|\tilde{w}_{c,i-1}\| \cdot \|B_c^T\| \cdot \mathrm{Tr}(\Sigma) \cdot \|e_{i-1}\| \\ &\leq \|\tilde{w}_{c,i-1}\| \cdot \|B_c^T\| \cdot \mathrm{Tr}(\Sigma) \cdot \frac{\lambda_H \|p\|_1}{2} \cdot \|\tilde{w}_{c,i-1}\|^2 \end{aligned} \tag{5.215}$$

and

$$\begin{aligned}\|e_{i-1}\|_{\Sigma}^2 &\overset{(b)}{\leq} \|\Sigma\| \cdot \|e_{i-1}\|^2 \\ &\leq \mathrm{Tr}(\Sigma) \cdot \|e_{i-1}\|^2 \\ &\leq \mathrm{Tr}(\Sigma) \cdot \frac{\lambda_H^2 \|p\|_1^2}{4} \cdot \|\tilde{w}_{c,i-1}\|^4 \end{aligned} \tag{5.216}$$

where for steps (a) and (b) in the above inequalities we used the property $\|\Sigma\| \leq \rho(\Sigma) \leq \mathrm{Tr}(\Sigma)$ for real symmetric (or Hermitian) nonnegative-definite Σ. Now, for any given small $\epsilon > 0$, there exists i_0 such that, for $i \geq i_0$, we have $\|\tilde{w}_{c,i-1}\| \leq \epsilon$ so that

$$\left|\tilde{w}_{c,i-1}^T B_c^T \Sigma e_{i-1}\right| \leq \epsilon \cdot \|B_c^T\| \cdot \mathrm{Tr}(\Sigma) \cdot \frac{\lambda_H \|p\|_1}{2} \cdot \|\tilde{w}_{c,i-1}\|^2 \tag{5.217}$$

$$\|e_{i-1}\|_{\Sigma}^2 \leq \epsilon^2 \cdot \mathrm{Tr}(\Sigma) \cdot \frac{\lambda_H^2 \|p\|_1^2}{4} \cdot \|\tilde{w}_{c,i-1}\|^2 \tag{5.218}$$

Substituting (5.217)–(5.218) into (5.213), we obtain

$$\|\tilde{w}_{c,i-1}\|_{B_c^T \Sigma B_c - \Delta}^2 \leq \|\tilde{w}_{c,i}\|_{\Sigma}^2 \leq \|\tilde{w}_{c,i-1}\|_{B_c^T \Sigma B_c + \Delta}^2 \tag{5.219}$$

where

$$\begin{aligned}\Delta &\triangleq \mu_{\max}\epsilon\cdot\lambda_H\|p\|_1\cdot\big[\|B_c^T\|+\mu_{\max}\epsilon\frac{\lambda_H\|p\|_1}{4}\big]\cdot\mathrm{Tr}(\Sigma)\cdot I_M\\ &= O(\mu_{\max}\epsilon)\cdot\mathrm{Tr}(\Sigma)\cdot I_M\end{aligned}\tag{5.220}$$

Let $\sigma=\mathrm{vec}(\Sigma)$ denote the vectorization operation that stacks the columns of a matrix Σ on top of each other. We shall use the notation $\|x\|_\sigma^2$ and $\|x\|_\Sigma^2$ interchangeably to denote the weighted squared Euclidean norm of a vector. Using the Kronecker product property [82, p.147]: $\mathrm{vec}(U\Sigma V)=(V^T\otimes U)\mathrm{vec}(\Sigma)$, we can vectorize the matrices $B_c^T\Sigma B_c+\Delta$ and $B_c^T\Sigma B_c-\Delta$ in (5.219) as $\mathcal{F}_+\sigma$ and $\mathcal{F}_-\sigma$, respectively, where

$$\begin{aligned}\mathcal{F}_+ &\triangleq B_c^T\otimes B_c^T+\mu_{\max}\epsilon\cdot\lambda_H\|p\|_1\cdot\big[\|B_c^T\|+\mu_{\max}\epsilon\frac{\lambda_H\|p\|_1}{4}\big]qq^T\\ &= B_c^T\otimes B_c^T+O(\mu_{\max}\epsilon)\end{aligned}\tag{5.221}$$

$$\begin{aligned}\mathcal{F}_- &\triangleq B_c^T\otimes B_c^T-\mu_{\max}\epsilon\cdot\lambda_H\|p\|_1\cdot\big[\|B_c^T\|+\mu_{\max}\epsilon\frac{\lambda_H\|p\|_1}{4}\big]qq^T\\ &= B_c^T\otimes B_c^T-O(\mu_{\max}\epsilon)\end{aligned}\tag{5.222}$$

where $q\triangleq\mathrm{vec}(I_M)$, and we have used the fact that $\mathrm{Tr}(\Sigma)=\mathrm{Tr}(\Sigma I_M)=\mathrm{vec}(I_M)^T\mathrm{vec}(\Sigma)=q^T\sigma$. In this way, we can write relation (5.219) as

$$\|\tilde{w}_{c,i-1}\|_{\mathcal{F}_-\sigma}^2\le\|\tilde{w}_{c,i}\|_\sigma^2\le\|\tilde{w}_{c,i-1}\|_{\mathcal{F}_+\sigma}^2\tag{5.223}$$

Using a state-space technique from [116, pp.344-346], we conclude that $\|\tilde{w}_{c,i}\|_\Sigma^2$ converges at a rate that is between $\rho(\mathcal{F}_-)$ and $\rho(\mathcal{F}_+)$. Recalling from (5.221)–(5.222) that $\mathcal{F}_+$ and $\mathcal{F}_-$ are perturbed matrices of $B_c^T\otimes B_c^T$, and since the perturbation term is $O(\epsilon\mu_{\max})$ which is small for small ϵ, we would expect the spectral radii of $\mathcal{F}_+$ and $\mathcal{F}_-$ to be small perturbations of $\rho(B_c^T\otimes B_c^T)$. This claim is

justified below.

Lemma 5.7 (Perturbation of spectral radius)**.** *Let $\epsilon \ll 1$ be a sufficiently small positive number. For any $M \times M$ matrix X, the spectral radius of the perturbed matrix $X + E$ for $E = O(\epsilon)$ is*

$$\rho(X + E) = \rho(X) + O\big(\epsilon^{\frac{1}{2(M-1)}}\big) \tag{5.224}$$

Proof. Let $X = TJT^{-1}$ be the Jordan canonical form of the matrix X. Without loss of generality, we consider the case where there are two Jordan blocks:

$$J = \text{diag}\{J_1, J_2\} \tag{5.225}$$

where $J_1 \in \mathbb{R}^{L\times L}$ and $J_2 \in \mathbb{R}^{(M-L)\times(M-L)}$ are Jordan blocks of the form

$$J_k = \begin{bmatrix} \lambda_k & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_k \end{bmatrix} \tag{5.226}$$

with $|\lambda_1| > |\lambda_2|$. Since $X + E$ is similar to $T^{-1}(X + E)T$, the matrix $X + E$ has the same set of eigenvalues as $J + E_0$ where

$$E_0 \triangleq T^{-1}ET = O(\epsilon) \tag{5.227}$$

Let

$$\epsilon_0 \triangleq \epsilon^{\frac{1}{2(M-1)}} \tag{5.228}$$

$$D_{\epsilon_0} \triangleq \text{diag}\left\{1, \epsilon_0, \ldots, \epsilon_0^{M-1}\right\} \tag{5.229}$$

Then, by similarity again, the matrix $J + E_0$ has the same set of eigenvalues as

$$D_{\epsilon_0}^{-1}(J + E_0)D_{\epsilon_0} = D_{\epsilon_0}^{-1}JD_{\epsilon_0} + E_1 \tag{5.230}$$

where $E_1 \triangleq D_{\epsilon_0}^{-1}E_0D_{\epsilon_0}$. Note that the ∞-induced norm (the maximum absolute row sum) of E_1 is bounded by

$$\begin{aligned}\|E_1\|_\infty &\leq \|D_{\epsilon_0}^{-1}\|_\infty \cdot \|E_0\|_\infty \cdot \|D_{\epsilon_0}\|_\infty \\ &= \frac{1}{\epsilon_0^{M-1}} \cdot O(\epsilon) \cdot 1 = \frac{1}{\epsilon^{\frac{1}{2}}} \cdot O(\epsilon) = O(\epsilon^{\frac{1}{2}})\end{aligned} \tag{5.231}$$

and that

$$D_{\epsilon_0}^{-1}JD_{\epsilon_0} = \text{diag}\{J_1', J_2'\} \tag{5.232}$$

where

$$J_k' = \begin{bmatrix} \lambda_k & \epsilon_0 & & \\ & \ddots & \ddots & \\ & & \ddots & \epsilon_0 \\ & & & \lambda_k \end{bmatrix} \tag{5.233}$$

Then, by appealing to Geršgorin Theorem [68, p.344], we conclude that the eigenvalues of the matrix $D_{\epsilon_0}^{-1}JD_{\epsilon_0} + E_1$, which are also the eigenvalues of the matrices $J + E_0$ and $X + E$, lie inside the union of the Geršgorin discs, namely,

$$\bigcup_{m=1}^{M} \mathcal{G}_m \tag{5.234}$$

where $\mathcal{G}_m$ is the mth Geršgorin disc defined as

$$\begin{aligned}\mathcal{G}_m &\triangleq \begin{cases} \left\{\lambda:\ |\lambda-\lambda_1| \le \epsilon_0 + \sum_{\ell=1}^{M} |E_{1,m\ell}|\right\}, & 1 \le m \le L \\ \left\{\lambda:\ |\lambda-\lambda_2| \le \epsilon_0 + \sum_{\ell=1}^{M} |E_{1,m\ell}|\right\}, & L < m \le M \end{cases} \\ &= \begin{cases} \left\{\lambda:\ |\lambda-\lambda_1| \le O\big(\epsilon^{\frac{1}{2(M-1)}}\big)\right\}, & 1 \le m \le L \\ \left\{\lambda:\ |\lambda-\lambda_2| \le O\big(\epsilon^{\frac{1}{2(M-1)}}\big)\right\}, & L < m \le M \end{cases} \end{aligned} \tag{5.235}$$

and where $E_{1,m\ell}$ denotes the (m,ℓ)-th entry of the matrix E_1. In the last step we used (5.228) and (5.231). Observe from (5.235) that there are two clusters of Geršgorin discs that are centered around λ_1 and λ_2, respectively, and have radii on the order of $O(\epsilon^{\frac{1}{2(M-1)}})$. A further statement from Geršgorin theorem shows that if the these two clusters of discs happen to be disjoint, which is true in our case since $|\lambda_1| > |\lambda_2|$ and we can select ϵ to be sufficiently small to ensure this property. Then there are exactly L eigenvalues of $X+E$ in $\cup_{m=1}^{L}\mathcal{G}_m$ while the remaining $M-L$ eigenvalues are in $\cup_{m=M-L}^{M}\mathcal{G}_m$. From $|\lambda_1| > |\lambda_2|$, we conclude that the largest eigenvalue of $D_{\epsilon_0}^{-1} J D_{\epsilon_0} + E_1$ is $\lambda_1 + O(\epsilon^{\frac{1}{2(M-1)}})$, which establishes (5.224).

□

Using (5.224) for $\mathcal{F}_+$ and $\mathcal{F}_-$ in (5.221)–(5.222), we conclude that

$$\rho(\mathcal{F}_+) = \big[\rho(I_M - \mu_{\max} H_c)\big]^2 + O\big((\mu_{\max}\epsilon)^{\frac{1}{2(M-1)}}\big) \tag{5.236}$$

$$\rho(\mathcal{F}_-) = \big[\rho(I_M - \mu_{\max} H_c)\big]^2 + O\big((\mu_{\max}\epsilon)^{\frac{1}{2(M-1)}}\big) \tag{5.237}$$

which holds for arbitrarily small ϵ. Since the convergence rate of $\|\tilde{w}_{c,i}\|^2$ is between $\rho(\mathcal{F}_+)$ and $\rho(\mathcal{F}_-)$, we arrive at (5.131).

5.G Proof of Lemma 5.6

From the definition in (5.127), it suffices to establish a joint inequality recursion for both $\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}]$ and $\mathbb{E}P[\boldsymbol{w}_{e,i}]$. To begin with, we state the following bounds on the perturbation terms in (5.86).

Lemma 5.8 (Bounds on the perturbation terms)**.** *The following bounds hold for any $i \geq 0$.*

$$P[\boldsymbol{z}_{i-1}] \preceq \lambda_U^2 \cdot \left\|\bar{P}_1[A_1^T U_L]\right\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \cdot P[\boldsymbol{w}_{e,i-1}] \tag{5.238}$$

$$P[s(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1})] \preceq 3\lambda_U^2 \cdot P[\check{\boldsymbol{w}}_{c,i-1}] \cdot \mathbb{1} + 3\lambda_U^2 \|\tilde{w}_{c,0}\|^2 \cdot \mathbb{1} + 3g^o \tag{5.239}$$

$$\begin{aligned}\mathbb{E}\{P[\boldsymbol{v}_i]|\mathcal{F}_{i-1}\} \preceq\ & 4\alpha \cdot \mathbb{1} \cdot P[\check{\boldsymbol{w}}_{c,i-1}] \\ & + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \\ & + \left[4\alpha \cdot (\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2\right] \cdot \mathbb{1}\end{aligned} \tag{5.240}$$

$$\begin{aligned}\mathbb{E}P[\boldsymbol{v}_i] \preceq\ & 4\alpha \cdot \mathbb{1} \cdot \mathbb{E}P[\check{\boldsymbol{w}}_{c,i-1}] \\ & + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\ & + \left[4\alpha \cdot (\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2\right] \cdot \mathbb{1}\end{aligned} \tag{5.241}$$

where $P[\check{\boldsymbol{w}}_{c,i-1}] = \|\check{\boldsymbol{w}}_{c,i-1}\|^2$, and $g^o \triangleq P[s(\mathbb{1} \otimes w^o)]$.

Proof. See Appendix 5.H. □

Now, we derive an inequality recursion for $\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}]$ from (5.97). Note that

$$\begin{aligned}\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}] &= \mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2 \\ &= \mathbb{E}P\big[T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1} \\ &\qquad - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{v}_i)\big] \\ &\overset{(a)}{=} \mathbb{E}P\left[T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\right]\end{aligned}$$

$$
\begin{aligned}
&+ \mu_{\max}^2 \cdot \mathbb{E}P\left[(p^T \otimes I_M)\boldsymbol{v}_i)\right] \\
&= \mathbb{E}P\Bigg[\gamma_c \cdot \frac{1}{\gamma_c}\left(T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1})\right) \\
&\quad + (1-\gamma_c) \cdot \frac{-\mu_{\max}}{1-\gamma_c}(p^T \otimes I_M)\boldsymbol{z}_{i-1}\Bigg] \\
&+ \mu_{\max}^2 \cdot \mathbb{E}P\left[(p^T \otimes I_M)\boldsymbol{v}_i)\right] \\
&\overset{(b)}{\preceq} \gamma_c \cdot \frac{1}{\gamma_c^2}\mathbb{E}P\left[T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1})\right] \\
&+ (1-\gamma_c) \cdot \frac{\mu_{\max}^2}{(1-\gamma_c)^2}\mathbb{E}P\left[(p^T \otimes I_M)\boldsymbol{z}_{i-1}\right] \\
&+ \mu_{\max}^2\mathbb{E}P\left[(p^T \otimes I_M)\boldsymbol{v}_i\right] \\
&\overset{(c)}{\preceq} \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] + \frac{\mu_{\max}^2}{1-\gamma_c}\mathbb{E}P\left[(p^T \otimes I_M)\boldsymbol{z}_{i-1}\right] \\
&+ \mu_{\max}^2\mathbb{E}P\left[(p^T \otimes I_M)\boldsymbol{v}_i\right] \\
&= \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] + \frac{\mu_{\max}^2}{1-\gamma_c}\mathbb{E}\left\|(p^T \otimes I_M)\boldsymbol{z}_{i-1}\right\|^2 \\
&+ \mu_{\max}^2\mathbb{E}\left\|(p^T \otimes I_M)\boldsymbol{v}_i\right\|^2 \\
&\overset{(d)}{=} \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] + \frac{\mu_{\max}^2}{1-\gamma_c}\mathbb{E}\left\|\sum_{k=1}^N p_k\boldsymbol{z}_{k,i-1}\right\|^2 \\
&+ \mu_{\max}^2\mathbb{E}\left\|\sum_{k=1}^N p_k\boldsymbol{v}_{k,i}\right\|^2 \\
&= \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] \\
&+ \frac{\mu_{\max}^2}{1-\gamma_c}\cdot\left(\sum_{l=1}^N p_l\right)^2\cdot\mathbb{E}\left\|\sum_{k=1}^N \frac{p_k}{\sum_{l=1}^N p_l}\boldsymbol{z}_{k,i-1}\right\|^2 \\
&+ \mu_{\max}^2\cdot\left(\sum_{l=1}^N p_l\right)^2\cdot\mathbb{E}\left\|\sum_{k=1}^N \frac{p_k}{\sum_{l=1}^N p_l}\boldsymbol{v}_{k,i}\right\|^2 \\
&\overset{(e)}{\leq} \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] \\
&+ \frac{\mu_{\max}^2}{1-\gamma_c}\cdot\left(\sum_{l=1}^N p_l\right)^2\cdot\sum_{k=1}^N \frac{p_k}{\sum_{l=1}^N p_l}\mathbb{E}\left\|\boldsymbol{z}_{k,i-1}\right\|^2
\end{aligned}
$$

$$
\begin{aligned}
&+\mu_{\max}^2 \cdot \left(\sum_{l=1}^{N} p_l\right)^2 \cdot \sum_{k=1}^{N} \frac{p_k}{\sum_{l=1}^{N} p_l} \mathbb{E}\left\|\boldsymbol{v}_{k,i}\right\|^2 \\
&= \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] + \frac{\mu_{\max}^2}{1-\gamma_c} \cdot \left(\sum_{l=1}^{N} p_l\right) \cdot \sum_{k=1}^{N} p_k \mathbb{E}\left\|\boldsymbol{z}_{k,i-1}\right\|^2 \\
&\quad + \mu_{\max}^2 \cdot \left(\sum_{l=1}^{N} p_l\right) \cdot \sum_{k=1}^{N} p_k \mathbb{E}\left\|\boldsymbol{v}_{k,i}\right\|^2 \\
&= \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] + \frac{\mu_{\max}^2}{1-\gamma_c} \cdot \|p\|_1 \cdot p^T \mathbb{E}P[\boldsymbol{z}_{i-1}] \\
&\quad + \mu_{\max}^2 \cdot \|p\|_1 \cdot p^T \mathbb{E}P[\boldsymbol{v}_i] \\
&\overset{(f)}{=} \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] + \frac{\mu_{\max} \cdot \|p\|_1}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2 \lambda_U^2} \cdot p^T \mathbb{E}P[\boldsymbol{z}_{i-1}] \\
&\quad + \mu_{\max}^2 \cdot \|p\|_1 \cdot p^T \mathbb{E}P[\boldsymbol{v}_i] \\
&\overset{(g)}{\preceq} \gamma_c \cdot \mathbb{E}P\left[\check{\boldsymbol{w}}_{c,i-1}\right] \\
&\quad + \frac{\mu_{\max}\|p\|_1}{\lambda_L - \mu_{\max}\frac{1}{2}\|p\|_1^2 \lambda_U^2} \\
&\quad \cdot p^T \left\{\lambda_U^2 \cdot \left\|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\right\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \cdot \mathbb{E}P[\boldsymbol{w}_{e,i-1}]\right\} \\
&\quad + \mu_{\max}^2 \cdot \|p\|_1 \cdot p^T \Big\{4\alpha \cdot \mathbb{1} \cdot \mathbb{E}P[\check{\boldsymbol{w}}_{c,i-1}] \\
&\qquad + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\
&\qquad + \left[4\alpha \cdot (\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2\right] \cdot \mathbb{1}\Big\} \\
&\overset{(h)}{=} \left[\gamma_c + \mu_{\max}^2 \cdot 4\alpha \|p\|_1^2\right] \cdot \mathbb{E}P[\check{\boldsymbol{w}}_{c,i-1}] \\
&\quad + \|p\|_1^2 \cdot \left\|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\right\|_\infty^2 \cdot \lambda_U^2 \\
&\qquad \cdot \left[\frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2 \lambda_U^2} + 4\mu_{\max}^2 \frac{\alpha}{\lambda_U^2}\right] \cdot \mathbb{1}^T \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\
&\quad + \mu_{\max}^2 \cdot \|p\|_1^2 \cdot \left[4\alpha\left(\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2\right) + \sigma_v^2\right]
\end{aligned} \tag{5.242}
$$

where step (a) uses the additivity property in Lemma 5.4 since the definition of $\boldsymbol{z}_{i-1}$ and $\boldsymbol{v}_i$ in (5.86) and the definition of $\boldsymbol{w}_{c,i-1}$ in (5.60) imply that $\boldsymbol{z}_{i-1}$ and

$\boldsymbol{w}_{c,i-1}$ depend on all $\{\boldsymbol{w}_j\}$ for $j \leq i-1$, meaning that the cross terms are zero:

$$\begin{aligned}
&\mathbb{E}[\boldsymbol{v}_i \boldsymbol{z}_{i-1}^T] = \mathbb{E}\left\{\mathbb{E}\left[\boldsymbol{v}_i|\mathcal{F}_{i-1}\right]\boldsymbol{z}_{i-1}^T\right\} = 0 \\
&\mathbb{E}\big\{\boldsymbol{v}_i[T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1})]^T\big\} \\
&\qquad = \mathbb{E}\left\{\mathbb{E}\left[\boldsymbol{v}_i|\mathcal{F}_{i-1}\right][T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1})]^T\right\} = 0
\end{aligned}$$

Step (b) uses the convexity property in Lemma 5.4, step (c) uses the variance property (5.118), step (d) uses the notation $\boldsymbol{z}_{k,i-1}$ and $\boldsymbol{v}_{k,i}$ to denote the kth $M \times 1$ block of the $NM \times 1$ vector $\boldsymbol{z}_{i-1}$ and $\boldsymbol{v}_i$, respectively, step (e) applies Jensen's inequality to the convex function $\|\cdot\|^2$, step (f) substitutes expression (5.120) for γ_c, step (g) substitutes the bounds for the perturbation terms from (5.238), (5.239), and (5.241), step (h) uses the fact that $p^T\mathbb{1} = \|p\|_1$.

Next, we derive the bound for $\mathbb{E}P[\boldsymbol{w}_{e,i}]$ from the recursion for $\boldsymbol{w}_{e,i}$ in (5.94):

$$\begin{aligned}
&\mathbb{E}P[\boldsymbol{w}_{e,i}] \\
&\quad = \mathbb{E}P\big[\mathcal{D}_{N-1}\boldsymbol{w}_{e,i-1} - \mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\, s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}) \\
&\qquad\quad - \mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\boldsymbol{z}_{i-1} - \mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\boldsymbol{v}_i\big] \\
&\quad \overset{(a)}{=} \mathbb{E}P\left[\mathcal{D}_{N-1}\boldsymbol{w}_{e,i-1} - \mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\ \left(s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1}\right)\right] \\
&\qquad + \mathbb{E}P\left[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\boldsymbol{v}_i\right] \\
&\quad \overset{(b)}{\preceq} \Gamma_e \cdot \mathbb{E}P\left[\boldsymbol{w}_{e,i-1}\right] \\
&\qquad + \frac{2}{1-|\lambda_2(A)|}\cdot\mathbb{E}P\left[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\big(s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1}\big)\right] \\
&\qquad + \mathbb{E}P\left[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\boldsymbol{v}_i\right] \\
&\quad \overset{(c)}{\preceq} \Gamma_e \cdot \mathbb{E}P\left[\boldsymbol{w}_{e,i-1}\right] \\
&\qquad + \frac{2}{1-|\lambda_2(A)|}\cdot\left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}]\right\|_\infty^2
\end{aligned}$$

$$
\begin{aligned}
&\quad \cdot \mathbb{1}\mathbb{1}^T \cdot \mathbb{E}P\left[s(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1}\right] \\
&\quad + \left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}]\right\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \cdot \mathbb{E}P\left[\boldsymbol{v}_i\right] \\
&\overset{(d)}{\preceq} \Gamma_e \cdot \mathbb{E}P\left[\boldsymbol{w}_{e,i-1}\right] \\
&\quad + \mu_{\max}^2 \cdot \frac{4\left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^2}{1-|\lambda_2(A)|}\cdot \mathbb{1}\mathbb{1}^T \\
&\quad \cdot\left\{\mathbb{E}P\left[s(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1})\right] + \mathbb{E}P\left[\boldsymbol{z}_{i-1}\right]\right\} \\
&\quad + \mu_{\max}^2 \cdot \left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \cdot \mathbb{E}P\left[\boldsymbol{v}_i\right] \\
&\overset{(e)}{\preceq} \left[\Gamma_e + 4\mu_{\max}^2 \cdot \left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^2 \cdot \left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2 \lambda_U^2 N\right. \\
&\qquad \left.\times \left(\frac{1}{1-|\lambda_2(A)|} + \frac{\alpha}{\lambda_U^2}\right)\mathbb{1}\mathbb{1}^T\right]\cdot \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\
&\quad + 4\mu_{\max}^2 \cdot \left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^2 \lambda_U^2 N\left(\frac{3}{1-|\lambda_2(A)|} + \frac{\alpha}{\lambda_U^2}\right) \\
&\qquad \cdot \mathbb{1}\cdot \mathbb{E}\|\check{\boldsymbol{w}}_{c,i-1}\|^2 \\
&\quad + \mu_{\max}^2 \cdot \left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^2 \cdot \left[12\frac{\lambda_U^2\|\tilde{w}_{c,0}\|^2 N + \mathbb{1}^T g^o}{1-|\lambda_2(A)|}\right. \\
&\qquad \left. + N[4\alpha(\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2]\right]\cdot \mathbb{1} \\
&\overset{(f)}{\preceq} \left[\Gamma_e + 4\mu_{\max}^2 \cdot \left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^2 \cdot \left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2 \lambda_U^2 N\right. \\
&\qquad \left.\times \left(\frac{1}{1-|\lambda_2(A)|} + \frac{\alpha}{\lambda_U^2}\right)\mathbb{1}\mathbb{1}^T\right]\cdot \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\
&\quad + 4\mu_{\max}^2 \cdot \left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^2 \lambda_U^2 N\left(\frac{3}{1-|\lambda_2(A)|} + \frac{\alpha}{\lambda_U^2}\right) \\
&\qquad \cdot \mathbb{1}\cdot \mathbb{E}\|\check{\boldsymbol{w}}_{c,i-1}\|^2 \\
&\quad + \mu_{\max}^2 \cdot N\left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^2 \cdot \left[12\frac{\lambda_U^2\|\tilde{w}_{c,0}\|^2 + \|g^o\|_\infty}{1-|\lambda_2(A)|}\right. \\
&\qquad \left. + 4\alpha(\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2\right]\cdot \mathbb{1}
\end{aligned}
\tag{5.243}
$$

where step (a) uses the additivity property in Lemma 5.4 since the definition of

$\boldsymbol{z}_{i-1}$ and $\boldsymbol{v}_i$ in (5.86) and the definitions of $\boldsymbol{w}_{c,i-1}$ and $\boldsymbol{w}_{e,i-1}$ in (5.60) imply that $\boldsymbol{z}_{i-1}$, $\boldsymbol{w}_{c,i-1}$ and $\boldsymbol{w}_{e,i-1}$ depend on all $\{\boldsymbol{w}_j\}$ for $j \leq i-1$, meanng that the cross terms between $\boldsymbol{v}_i$ and all other terms are zero, just as in step (a) of (5.242), step (b) uses the variance relation of stable Kronecker Jordan operators from (5.126) with $d_2 = \lambda_2(A)$, step (c) uses the variance relation of linear operator (5.115), step (d) uses the submultiplictive property (5.106) and $P[x+y] \preceq 2P[x] + 2P[y]$ derived from the convexity property (5.102) and the scaling property in (5.238), (5.239), and (5.241), step (e) substitutes the bounds on the perturbation terms from (5.238)–(5.241), and step (f) uses the inequality $|\mathbb{1}^T g^o| \leq N\|g^o\|_\infty$.

Finally, using the quantities defined in (5.136)–(5.139), we can rewrite recursions (5.242) and (5.243) as

$$\begin{aligned}
\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}] \preceq\ & (\gamma_c + \mu_{\max}^2\psi_0)\cdot \mathbb{E}P[\check{\boldsymbol{w}}_{c,i-1}] \\
& + (\mu_{\max} h_c(\mu_{\max}) + \mu_{\max}^2\psi_0)\cdot \mathbb{1}^T \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\
& + \mu_{\max}^2 b_{v,c} \qquad (5.244) \\
\mathbb{E}P[\boldsymbol{w}_{e,i}] \preceq\ & \mu_{\max}^2\psi_0 \mathbb{1}\cdot \mathbb{E}P[\check{\boldsymbol{w}}_{c,i-1}] \\
& + (\Gamma_e + \mu_{\max}^2\psi_0 \mathbb{1}\mathbb{1}^T)\cdot \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\
& + \mu_{\max}^2 b_{v,e}\cdot \mathbb{1} \qquad (5.245)
\end{aligned}$$

where $\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}] = \mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2$. Using the matrices and vectors defined in (5.133)–(5.135), we can write the above two recursions in a joint form as in (5.132).

5.H Proof of Lemma 5.8

First, we establish the bound for $P[\boldsymbol{z}_{i-1}]$ in (5.238). Substituting (5.62) and (5.81) into the definition of $\boldsymbol{z}_{i-1}$ in (5.86) we get:

$$\begin{aligned}
P[\boldsymbol{z}_{i-1}] &\preceq P\big[s\left(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}+(A_1^TU_L\otimes I_M)\boldsymbol{w}_{e,i-1}\right)-s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})\big] \\
&\overset{(a)}{\preceq} \lambda_U^2\cdot P\left[(A_1^TU_L\otimes I_M)\boldsymbol{w}_{e,i-1}\right] \\
&\overset{(b)}{\preceq} \lambda_U^2\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2\cdot\mathbb{1}\mathbb{1}^T\cdot P[\boldsymbol{w}_{e,i-1}] \qquad (5.246)
\end{aligned}$$

where step (a) uses the variance relation (5.117), and step (b) uses property (5.115).

Next, we prove the bound on $P[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})]$. It holds that

$$\begin{aligned}
P[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})] &= P\Big[\frac{1}{3}\cdot 3\big(s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})-s(\mathbb{1}\otimes\bar{w}_{c,i-1})\big) \\
&\quad+\frac{1}{3}\cdot 3\big(s(\mathbb{1}\otimes\bar{w}_{c,i-1})-s(\mathbb{1}\otimes w^o)\big) \\
&\quad+\frac{1}{3}\cdot 3\cdot s\big(\mathbb{1}\otimes w^o\big)\Big] \\
&\overset{(a)}{\preceq} \frac{1}{3}\cdot P\big[3\big(s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})-s(\mathbb{1}\otimes\bar{w}_{c,i-1})\big)\big] \\
&\quad+\frac{1}{3}\cdot P\big[3\big(s(\mathbb{1}\otimes\bar{w}_{c,i-1})-s(\mathbb{1}\otimes w^o)\big)\big] \\
&\quad+\frac{1}{3}\cdot P\big[3\cdot s(\mathbb{1}\otimes w^o)\big] \\
&\overset{(b)}{=} 3P\big[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})-s(\mathbb{1}\otimes\bar{w}_{c,i-1})\big] \\
&\quad+3P\big[s(\mathbb{1}\otimes\bar{w}_{c,i-1})-s(\mathbb{1}\otimes w^o)\big]+3P\big[s(\mathbb{1}\otimes w^o)\big] \\
&\overset{(c)}{\preceq} 3\lambda_U^2\cdot P\big[\mathbb{1}\otimes(\boldsymbol{w}_{c,i-1}-\bar{w}_{c,i-1})\big] \\
&\quad+3\lambda_U^2\cdot P[\mathbb{1}\otimes(\bar{w}_{c,i-1}-w^o)]+3P[s(\mathbb{1}\otimes w^o)] \\
&\overset{(d)}{=} 3\lambda_U^2\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^2\cdot\mathbb{1}+3\lambda_U^2\cdot\|\bar{w}_{c,i-1}-w^o\|^2\cdot\mathbb{1}
\end{aligned}$$

$$
\begin{aligned}
&+ 3P[s(\mathbb{1}\otimes w^o)] \\
&\overset{(e)}{\preceq} 3\lambda_U^2\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^2\cdot\mathbb{1}+3\lambda_U^2\|\tilde{w}_{c,0}\|^2\cdot\mathbb{1}+3P[s(\mathbb{1}\otimes w^o)] \qquad (5.247)
\end{aligned}
$$

where step (a) uses the convexity property (5.102), step (b) uses the scaling property in Lemma 5.4, step (c) uses the variance relation (5.117), step (d) uses property (5.108), and step (e) uses the bound (5.129) and the fact that $\gamma_c < 1$.

Finally, we establish the bounds on $P[\boldsymbol{v}_i]$ in (5.240)–(5.241). Introduce the $MN\times 1$ vector $\boldsymbol{x}$:

$$
\boldsymbol{x} \triangleq \mathbb{1}\otimes\boldsymbol{w}_{c,i-1} + \mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1} \equiv \boldsymbol{\phi}_{i-1} \qquad (5.248)
$$

We partitin $\boldsymbol{x}$ in block form as $\boldsymbol{x} = \mathrm{col}\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N\}$, where each $\boldsymbol{x}_k$ is $M\times 1$. Then, by the definition of $\boldsymbol{v}_i$ from (5.86), we have

$$
\begin{aligned}
\mathbb{E}\{P[\boldsymbol{v}_i]|\mathcal{F}_{i-1}\} &= \mathbb{E}\{P[\hat{\boldsymbol{s}}_i(\boldsymbol{x}) - s(\boldsymbol{x})]|\mathcal{F}_{i-1}\} \\
&= \mathrm{col}\big\{\mathbb{E}\left[\|\hat{\boldsymbol{s}}_{1,i}(\boldsymbol{x}_1) - s_1(\boldsymbol{x}_1)\|^2\big|\mathcal{F}_{i-1}\right], \\
&\qquad \ldots, \mathbb{E}\left[\|\hat{\boldsymbol{s}}_{N,i}(\boldsymbol{x}_N) - s_N(\boldsymbol{x}_N)\|^2\big|\mathcal{F}_{i-1}\right]\big\} \\
&\overset{(a)}{\preceq} \mathrm{col}\big\{\alpha\cdot\|\boldsymbol{x}_1\|^2+\sigma_v^2,\ \ldots,\ \alpha\cdot\|\boldsymbol{x}_N\|^2+\sigma_v^2\big\} \\
&= \alpha\cdot P[\boldsymbol{x}] + \sigma_v^2\mathbb{1} \qquad (5.249)
\end{aligned}
$$

where step (a) uses Assumption (5.18). Now we bound $P[\boldsymbol{x}]$:

$$
\begin{aligned}
P[\boldsymbol{x}] &= P\left[\mathbb{1}\otimes\boldsymbol{w}_{c,i-1} + \mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1}\right] \\
&= P\Big[\frac{1}{4}\cdot 4\cdot\mathbb{1}\otimes(\boldsymbol{w}_{c,i-1}-\bar{w}_{c,i-1})+\frac{1}{4}\cdot 4\cdot\mathbb{1}\otimes(\bar{w}_{c,i-1}-w^o) \\
&\qquad +\frac{1}{4}\cdot 4\cdot\mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1}+\frac{1}{4}\cdot 4\cdot\mathbb{1}\otimes w^o\Big]
\end{aligned}
$$

$$
\begin{aligned}
&= P\Big[\frac{1}{4}\cdot 4\cdot \mathbb{1}\otimes \check{\boldsymbol{w}}_{c,i-1} + \frac{1}{4}\cdot 4\cdot \mathbb{1}\otimes \tilde{w}_{c,i-1} \\
&\qquad + \frac{1}{4}\cdot 4\cdot \mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1} + \frac{1}{4}\cdot 4\cdot \mathbb{1}\otimes w^o\Big] \\
&\overset{(a)}{\preceq} \frac{1}{4}\cdot 4^2\cdot P[\mathbb{1}\otimes \check{\boldsymbol{w}}_{c,i-1}] + \frac{1}{4}\cdot 4^2\cdot P[\mathbb{1}\otimes \tilde{w}_{c,i-1}] \\
&\quad + \frac{1}{4}\cdot 4^2\cdot P[\mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1}] + \frac{1}{4}\cdot 4^2\cdot P[\mathbb{1}\otimes w^o] \\
&\overset{(b)}{=} 4\cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2\cdot \mathbb{1} + 4\cdot \|\tilde{w}_{c,i-1}\|^2\cdot \mathbb{1} \\
&\quad + 4\cdot P[\mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1}] + 4\cdot \|w^o\|^2\cdot \mathbb{1} \\
&\overset{(c)}{\preceq} 4\cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2\cdot \mathbb{1} + 4\cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\cdot \mathbb{1}\mathbb{1}^T\cdot P[\boldsymbol{w}_{e,i-1}] \\
&\quad + 4\cdot \|\tilde{w}_{c,0}\|^2\cdot \mathbb{1} + 4\cdot \|w^o\|^2\cdot \mathbb{1}
\end{aligned}
\tag{5.250}
$$

where step (a) uses the convexity property (5.102) and the scaling property in Lemma 5.4, step (b) uses the Kronecker property (5.108), step (c) uses the variance relation (5.114) and the bound (5.129). Substituting (5.250) into (5.249), we obtain (5.240), and taking expectation of (5.240) with respect to $\mathcal{F}_{i-1}$ leads to (5.241).

5.I Proof of Theorem 5.4

Assume initially that the matrix Γ is stable (we show further ahead how the step-size parameter $\mu_{\max}$ can be selected to ensure this property). Then, we can iterate the inequality recursion (5.132) and obtain

$$
\begin{aligned}
\check{\mathcal{W}}_i' &\preceq \Gamma^i\check{\mathcal{W}}_0' + \mu_{\max}^2\sum_{j=0}^{i-1}\Gamma^j b_v \\
&\preceq \sum_{j=0}^{\infty}\Gamma^j\check{\mathcal{W}}_0' + \mu_{\max}^2\sum_{j=0}^{\infty}\Gamma^j b_v
\end{aligned}
$$

$$\preceq (I-\Gamma)^{-1}(\check{\mathcal{W}}_0' + \mu_{\max}^2 b_v) \tag{5.251}$$

where the first two inequalities use the fact that all entries of Γ are nonnegative. Moreover, substituting (5.133) into (5.132), we get

$$\check{\mathcal{W}}_i' \preceq \Gamma_0 \check{\mathcal{W}}_{i-1}' + \mu_{\max}^2 \psi_0 \mathbb{1}\mathbb{1}^T \check{\mathcal{W}}_{i-1}' + \mu_{\max}^2 b_v \tag{5.252}$$

Substituting (5.251) into the second term on the right-hand side of (5.252) leads to

$$\check{\mathcal{W}}_i' \preceq \Gamma_0 \check{\mathcal{W}}_{i-1}' + \mu_{\max}^2 \cdot c_v(\mu_{\max}) \tag{5.253}$$

where

$$c_v(\mu_{\max}) \triangleq \psi_0 \cdot \mathbb{1}^T (I-\Gamma)^{-1}(\check{\mathcal{W}}_0' + \mu_{\max}^2 b_v) \cdot \mathbb{1} + b_v \tag{5.254}$$

Now iterating (5.253) leads to the following non-asymptotic bound:

$$\check{\mathcal{W}}_i{}' \preceq \Gamma_0^i \check{\mathcal{W}}_0' + \sum_{j=0}^{i-1} \mu_{\max}^2 \Gamma_0^j \cdot c_v(\mu_{\max}) \preceq \Gamma_0^i \check{\mathcal{W}}_0' + \check{\mathcal{W}}_\infty^{\mathrm{ub}'} \tag{5.255}$$

where

$$\check{\mathcal{W}}_\infty^{\mathrm{ub}'} \triangleq \mu_{\max}^2 (I-\Gamma_0)^{-1} \cdot c_v(\mu_{\max}) \tag{5.256}$$

We now derive the non-asymptotic bounds (5.140)–(5.141) from (5.255). To this end, we need to study the structure of the term $\Gamma_0^i \check{\mathcal{W}}_0'$. Our approach relies on applying the unilateral $z-$transform to the causal matrix sequence $\{\Gamma_0^i, i \geq 0\}$ to

get

$$\Gamma_0(z) \triangleq \mathcal{Z}\left\{\Gamma_0^i\right\} = z(zI - \Gamma_0)^{-1} \tag{5.257}$$

since Γ_0 is a stable matrix. Note from (5.134) that Γ_0 is a 2×2 block upper triangular matrix. Substituting (5.134) into the above expression and using the formula for inverting 2×2 block upper triangular matrices (see formula (4) in [82, p.48]), we obtain

$$\Gamma_0(z) = \begin{bmatrix} \frac{z}{z-\gamma_c} & \mu_{\max} h_c(\mu_{\max}) \cdot \frac{z}{z-\gamma_c} \cdot \mathbb{1}^T (zI - \Gamma_e)^{-1} \\ 0 & z(zI - \Gamma_e)^{-1} \end{bmatrix} \tag{5.258}$$

Next we compute the inverse $z-$transform to obtain Γ_0^i. Thus, observe that the inverse $z-$transform of the $(1,1)$ entry, the $(2,1)$ block, and the $(2,2)$ block are the causal sequences γ_c^i, 0, and Γ_e^i, respectively. For the $(1,2)$ block, it can be expressed in partial fractions as

$$\begin{aligned} &\mu_{\max} h_c(\mu_{\max}) \cdot \frac{z}{z-\gamma_c} \cdot \mathbb{1}^T (zI - \Gamma_e)^{-1} \\ &= \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T (\gamma_c I - \Gamma_e)^{-1} \left(\frac{z}{z-\gamma_c} I - z(zI - \Gamma_e)^{-1} \right) \end{aligned}$$

from which we conclude that the inverse $z-$transform of the $(1,2)$ block is

$$\mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T (\gamma_c I - \Gamma_e)^{-1} \left(\gamma_c^i I - \Gamma_e^i\right), \quad i \geq 0 \tag{5.259}$$

It follows that

$$\Gamma_0^i = \begin{bmatrix} \gamma_c^i & \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T (\gamma_c I - \Gamma_e)^{-1} \left(\gamma_c^i I - \Gamma_e^i\right) \\ 0 & \Gamma_e^i \end{bmatrix} \tag{5.260}$$

Furthermore, as indicated by (5.41) in Sec. 5.4.1, the reference recursion (5.40) is initialized at the centroid of the network, i.e., $\bar{w}_{c,0} = \sum_{k=1}^{N} \theta_k w_{k,0}$. This fact, together with (5.61) leads to $\bar{w}_{c,0} = w_{c,0}$, which means that $\check{\boldsymbol{w}}_{c,0} = 0$. As a result, we get the following form for $\mathcal{W}_0'$:

$$\check{\mathcal{W}}_0' = \operatorname{col}\{0,\ \mathbb{E}P[\boldsymbol{w}_{e,0}]\} \tag{5.261}$$

Multiplying (5.260) to the left of (5.261) gives

$$\Gamma_0^i \check{\mathcal{W}}_0' = \begin{bmatrix} \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T (\gamma_c I - \Gamma_e)^{-1} (\gamma_c^i I - \Gamma_e^i) \mathcal{W}_{e,0} \\ \Gamma_e^i \mathcal{W}_{e,0} \end{bmatrix} \tag{5.262}$$

where $\mathcal{W}_{e,0} = \mathbb{E}P[\boldsymbol{w}_{e,0}]$. Substituting (5.262) into (5.255), we obtain

$$\begin{aligned} \check{\mathcal{W}}_i' \preceq & \begin{bmatrix} \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T (\gamma_c I - \Gamma_e)^{-1} (\gamma_c^i I - \Gamma_e^i)\, \mathbb{E}P[\boldsymbol{w}_{e,0}] \\ \Gamma_e^i\ \mathbb{E}P[\boldsymbol{w}_{e,0}] \end{bmatrix} \\ & + \check{\mathcal{W}}_\infty^{\mathrm{ub}'} \end{aligned} \tag{5.263}$$

Finally, we study the behavior of the asymptotic bound $\check{\mathcal{W}}_\infty^{\mathrm{ub}'}$ by calling upon the following lemma.

Lemma 5.9 (Useful matrix expressions)**.** *It holds that*

$$\begin{aligned} \mathbb{1}^T (I - \Gamma)^{-1} &= \zeta(\mu_{\max}) \\ &\quad \cdot \begin{bmatrix} \frac{\mu_{\max}^{-1}}{\lambda_L - \frac{\mu_{\max}}{2} \|p\|_1^2 \lambda_U^2} & \left(1 + \frac{h_c(\mu_{\max})}{\lambda_L - \frac{\mu_{\max}}{2} \|p\|_1^2 \lambda_U^2}\right) \mathbb{1}^T (I - \Gamma_e)^{-1} \end{bmatrix} \end{aligned} \tag{5.264}$$

$$(I - \Gamma_0)^{-1} = \begin{bmatrix} \frac{\mu_{\max}^{-1}}{\lambda_L - \frac{\mu_{\max}}{2} \|p\|_1^2 \lambda_U^2} & \frac{h_c(\mu_{\max})}{\lambda_L - \frac{\mu_{\max}}{2} \|p\|_1^2 \lambda_U^2} \mathbb{1}^T (I - \Gamma_e)^{-1} \\ 0 & (I - \Gamma_e)^{-1} \end{bmatrix} \tag{5.265}$$

where

$$\zeta(\mu_{\max}) = \Bigg\{1 - \psi_0 \cdot \Big[\frac{\mu_{\max}}{\lambda_L - \frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2} + \mu_{\max}^2\left(1 + \frac{h_c(\mu_{\max})}{\lambda_L - \frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2}\right)\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{1}\Big]\Bigg\}^{-1} \tag{5.266}$$

Proof. See Appendix 5.J. □

Substituting (5.254), (5.261), (5.264) and (5.265) into (5.256) and after some algebra, we obtain

$$\begin{aligned}\check{\mathcal{W}}_{\infty}^{\mathrm{ub}'} = \psi_0 \cdot \zeta(\mu_{\max}) f(\mu_{\max}) & \cdot \begin{bmatrix} \mu_{\max}\frac{1+\mu_{\max}h_c(\mu_{\max})\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{1}}{\lambda_L - \frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2} \\ \mu_{\max}^2 \cdot (I-\Gamma_e)^{-1}\mathbb{1} \end{bmatrix} \\ & + \begin{bmatrix} \mu_{\max}\frac{b_{v,c}+\mu_{\max}h_c(\mu_{\max})\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{1}b_{v,e}}{\lambda_L - \frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2} \\ \mu_{\max}^2 b_{v,e} \cdot (I-\Gamma_e)^{-1}\mathbb{1} \end{bmatrix}\end{aligned} \tag{5.267}$$

where

$$\begin{aligned} f(\mu_{\max}) \triangleq & \frac{\mu_{\max} b_{v,c}}{\lambda_L - \frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2} \\ & + \left(1 + \frac{h_c(\mu_{\max})}{\lambda_L - \frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2}\right) \cdot \mathbb{1}^T(I-\Gamma_e)^{-1} \\ & \times \left(\mathbb{E}P[\boldsymbol{w}_{e,0}] + \mu_{\max}^2 \mathbb{1} b_{v,e}\right) \end{aligned} \tag{5.268}$$

Introduce

$$\begin{aligned}\check{\mathcal{W}}_{c,\infty}^{\mathrm{ub}'} \triangleq & \psi_0 \cdot \zeta(\mu_{\max}) f(\mu_{\max}) \\ & \cdot \mu_{\max}\frac{1 + \mu_{\max}h_c(\mu_{\max})\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{1}}{\lambda_L - \frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2}\end{aligned}$$

$$+\mu_{\max}\frac{b_{v,c}+\mu_{\max}h_c(\mu_{\max})\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{1}b_{v,e}}{\lambda_L-\frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2} \tag{5.269}$$

$$\check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'}\triangleq\psi_0\cdot\zeta(\mu_{\max})f(\mu_{\max})\cdot\mu_{\max}^2\cdot(I-\Gamma_e)^{-1}\mathbb{1}$$
$$+\mu_{\max}^2 b_{v,e}\cdot(I-\Gamma_e)^{-1}\mathbb{1} \tag{5.270}$$

Then, we have

$$\check{\mathcal{W}}_{\infty}^{\mathrm{ub}'}=\mathrm{col}\{\check{\mathcal{W}}_{c,\infty}^{\mathrm{ub}'},\ \check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'}\} \tag{5.271}$$

Substituting (5.271) into (5.263), we conclude (5.140)–(5.141). Now, to prove (5.142)–(5.143), it suffices to prove

$$\lim_{\mu_{\max}\to 0}\frac{\check{\mathcal{W}}_{c,\infty}^{\mathrm{ub}'}}{\mu_{\max}}=\frac{\psi_0\big(\lambda_L+h_c(0)\big)\mathbb{1}^T(I-\Gamma_e)^{-1}\mathcal{W}_{e,0}+b_{v,c}\lambda_L}{\lambda_L^2} \tag{5.272}$$

$$\lim_{\mu_{\max}\to 0}\frac{\check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'}}{\mu_{\max}^2}=\frac{\psi_0\big(\lambda_L+h_c(0)\big)\mathbb{1}^T(I-\Gamma_e)^{-1}\mathcal{W}_{e,0}+b_{v,e}\lambda_L}{\lambda_L}\cdot(I-\Gamma_e)^{-1}\mathbb{1} \tag{5.273}$$

Substituting (5.269) and (5.270) into the left-hand side of (5.272) and (5.273), respectively, we get

$$\begin{aligned}\lim_{\mu_{\max}\to 0}\frac{\check{\mathcal{W}}_{c,\infty}^{\mathrm{ub}'}}{\mu_{\max}}&=\lim_{\mu_{\max}\to 0}\left\{\psi_0\cdot\zeta(\mu_{\max})f(\mu_{\max})\cdot\frac{1+\mu_{\max}h_c(\mu_{\max})\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{1}}{\lambda_L-\frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2}\right.\\&\qquad\left.+\frac{b_{v,c}+\mu_{\max}h_c(\mu_{\max})\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{1}b_{v,e}}{\lambda_L-\frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2}\right\}\\&=\psi_0\cdot\zeta(0)f(0)\cdot\frac{1}{\lambda_L}+\frac{b_{v,c}}{\lambda_L}\\&\overset{(a)}{=}\psi_0\cdot 1\cdot\left[\left(1+\frac{h_c(0)}{\lambda_L}\right)\cdot\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{E}P[\boldsymbol{w}_{e,0}]\right]\cdot\frac{1}{\lambda_L}+\frac{b_{v,c}}{\lambda_L}\\&=\frac{\psi_0\big(\lambda_L+h_c(0)\big)\mathbb{1}^T(I-\Gamma_e)^{-1}\mathcal{W}_{e,0}+b_{v,c}\lambda_L}{\lambda_L^2}\end{aligned} \tag{5.274}$$

$$\lim_{\mu_{\max}\to 0}\frac{\check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'}}{\mu_{\max}^2}=\lim_{\mu_{\max}\to 0}\big\{\psi_0\cdot\zeta(\mu_{\max})f(\mu_{\max})\cdot(I-\Gamma_e)^{-1}\mathbb{1}+b_{v,e}\cdot(I-\Gamma_e)^{-1}\mathbb{1}\big\}$$

$$
\begin{aligned}
&= \psi_0 \cdot \zeta(0) f(0) \cdot (I - \Gamma_e)^{-1} \mathbb{1} + b_{v,e} \cdot (I - \Gamma_e)^{-1} \mathbb{1} \\
&\overset{(b)}{=} \psi_0 \cdot \left[\left(1 + \frac{h_c(0)}{\lambda_L} \right) \cdot \mathbb{1}^T (I - \Gamma_e)^{-1} \mathbb{E} P[\boldsymbol{w}_{e,0}] \right] \cdot (I - \Gamma_e)^{-1} \mathbb{1} \\
&\quad + b_{v,e} \cdot (I - \Gamma_e)^{-1} \mathbb{1} \\
&= \frac{\psi_0 \big(\lambda_L + h_c(0) \big) \mathbb{1}^T (I - \Gamma_e)^{-1} \mathcal{W}_{e,0} + b_{v,e} \lambda_L}{\lambda_L} \cdot (I - \Gamma_e)^{-1} \mathbb{1} \qquad (5.275)
\end{aligned}
$$

where steps (a) and (b) use the expressions for $\zeta(\mu_{\max})$ and $f(\mu_{\max})$ from (5.266) and (5.268).

Now we proceed to prove (5.144). We already know that the second term on the right-hand side of (5.140), $\check{\mathcal{W}}_{c,\infty}^{\mathrm{ub}'}$, is $O(\mu_{\max})$ because of (5.142). Therefore, we only need to show that the first term on the right-hand side of (5.140) is $O(\mu_{\max})$ for all $i \geq 0$. To this end, it suffices to prove that

$$
\lim_{\mu_{\max} \to 0} \frac{\left\| \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T (\gamma_c I - \Gamma_e)^{-1} \left(\gamma_c^i I - \Gamma_e^i \right) \mathcal{W}_{e,0} \right\|}{\mu_{\max}} \leq \text{constant} \qquad (5.276)
$$

where the constant on the right-hand side should be independent of i. This can be proved as below:

$$
\begin{aligned}
&\lim_{\mu_{\max} \to 0} \frac{\left\| \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T (\gamma_c I - \Gamma_e)^{-1} \left(\gamma_c^i I - \Gamma_e^i \right) \mathcal{W}_{e,0} \right\|}{\mu_{\max}} \\
&= \lim_{\mu_{\max} \to 0} \left\| h_c(\mu_{\max}) \cdot \mathbb{1}^T (\gamma_c I - \Gamma_e)^{-1} \left(\gamma_c^i I - \Gamma_e^i \right) \mathcal{W}_{e,0} \right\| \\
&\leq \lim_{\mu_{\max} \to 0} |h_c(\mu_{\max})| \cdot \|\mathbb{1}\| \cdot \|(\gamma_c I - \Gamma_e)^{-1}\| \\
&\qquad \cdot \left(\left\| \gamma_c^i I \right\| + \left\| \Gamma_e^i \right\| \right) \cdot \|\mathcal{W}_{e,0}\| \\
&\overset{(a)}{\leq} \lim_{\mu_{\max} \to 0} |h_c(\mu_{\max})| \cdot N \cdot \|(\gamma_c I - \Gamma_e)^{-1}\| \\
&\qquad \cdot \Big(1 + C_e \cdot \big(\rho(\Gamma_e) + \epsilon \big)^i \Big) \cdot \|\mathcal{W}_{e,0}\| \\
&\overset{(b)}{\leq} \lim_{\mu_{\max} \to 0} |h_c(\mu_{\max})| \cdot N \cdot \|(\gamma_c I - \Gamma_e)^{-1}\| \cdot (1 + C_e) \cdot \|\mathcal{W}_{e,0}\|
\end{aligned}
$$

$$\begin{aligned}
&\overset{(c)}{=} |h_c(0)| \cdot N \cdot \|(I-\Gamma_e)^{-1}\| \cdot [1+C_e] \cdot \|\mathcal{W}_{e,0}\| \\
&= \text{constant}
\end{aligned} \tag{5.277}$$

where step (a) uses $\gamma_c = 1-\mu_{\max}\lambda_L+\frac{1}{2}\mu_{\max}^2\lambda_L^2 < 1$ for sufficiently small step-sizes, and uses the property that for any small $\epsilon > 0$ there exists a constant C such that $\|X^i\| \leq C \cdot [\rho(X)+\epsilon]^i$ for all $i \geq 0$ [105, p.38], step (b) uses the fact that $\rho(\Gamma_e) = |\lambda_2(A)| < 1$ so that $\rho(\Gamma_e)+\epsilon < 1$ for small ϵ (e.g., $\epsilon = (1-\rho(\Gamma_e))/2$), and step (c) uses $\gamma_c = 1-\mu_{\max}\lambda_L+\frac{1}{2}\mu_{\max}^2\lambda_L^2 \to 1$ when $\mu_{\max} \to 0$.

It remains to prove that condition (5.145) guaranees the stability of the matrix Γ, i.e., $\rho(\Gamma) < 1$. First, we introduce the diagonal matrices $D_{\epsilon,0} \triangleq \text{diag}\{\epsilon, \cdots, \epsilon^{N-1}\}$ and $D_\epsilon = \text{diag}\{1, D_{\epsilon,0}\}$, where ϵ is chosen to be

$$\epsilon \triangleq \frac{1}{4}(1-|\lambda_2(A)|)^2 \leq \frac{1}{4} \tag{5.278}$$

It holds that $\rho(\Gamma) = \rho(D_\epsilon^{-1}\Gamma D_\epsilon)$ since similarity transformations do not alter eigenvalues. By the definition of Γ in (5.133), we have

$$D_\epsilon^{-1}\Gamma D_\epsilon = D_\epsilon^{-1}\Gamma_0 D_\epsilon + \mu_{\max}^2\psi_0 \cdot D_\epsilon^{-1}\mathbb{1}\mathbb{1}^T D_\epsilon \tag{5.279}$$

We now recall that the spectral radius of a matrix is upper bounded by any of its matrix norms. Thus, taking the $1-$norm (the maximum absolute column sum of the matrix) of both sides of the above expression and using the triangle inequality and the fact that $0 < \epsilon \leq 1/4$, we get

$$\begin{aligned}
\rho(\Gamma) &= \rho(D_\epsilon^{-1}\Gamma D_\epsilon) \\
&\leq \|D_\epsilon^{-1}\Gamma_0 D_\epsilon\|_1 + \|\mu_{\max}^2\psi_0 \cdot D_\epsilon^{-1}\mathbb{1}\mathbb{1}^T D_\epsilon\|_1 \\
&\leq \|D_\epsilon^{-1}\Gamma_0 D_\epsilon\|_1 + \mu_{\max}^2\psi_0 \cdot \|D_\epsilon^{-1}\mathbb{1}\mathbb{1}^T D_\epsilon\|_1
\end{aligned}$$

$$
\begin{aligned}
&= \|D_\epsilon^{-1}\Gamma_0 D_\epsilon\|_1 + \mu_{\max}^2\psi_0 \cdot \left(1 + \epsilon^{-1} + \cdots + \epsilon^{-(N-1)}\right) \\
&= \|D_\epsilon^{-1}\Gamma_0 D_\epsilon\|_1 + \mu_{\max}^2\psi_0 \cdot \frac{1-\epsilon^{-N}}{1-\epsilon^{-1}} \\
&= \|D_\epsilon^{-1}\Gamma_0 D_\epsilon\|_1 + \mu_{\max}^2\psi_0 \cdot \frac{\epsilon(\epsilon^{-N}-1)}{1-\epsilon} \\
&\leq \|D_\epsilon^{-1}\Gamma_0 D_\epsilon\|_1 + \mu_{\max}^2\psi_0 \cdot \frac{\frac{1}{4}(\epsilon^{-N}-1)}{1-\frac{1}{4}} \\
&\leq \|D_\epsilon^{-1}\Gamma_0 D_\epsilon\|_1 + \frac{1}{3}\mu_{\max}^2\psi_0\epsilon^{-N}
\end{aligned} \tag{5.280}
$$

Moreover, we can use (5.134) to write:

$$
D_\epsilon^{-1}\Gamma_0 D_\epsilon = \begin{bmatrix} \gamma_c & \mu_{\max}h_c(\mu_{\max})\mathbb{1}^T D_{\epsilon,0} \\ 0 & D_{\epsilon,0}^{-1}\Gamma_e D_{\epsilon,0} \end{bmatrix} \tag{5.281}
$$

where (recall the expression for Γ_e from (5.125) where we replace d_2 by $\lambda_2(A)$):

$$
D_{\epsilon,0}^{-1}\Gamma_e D_{\epsilon,0} = \begin{bmatrix} |\lambda_2(A)| & \frac{1-|\lambda_2(A)|}{2} & & \\ & \ddots & \ddots & \\ & & \ddots & \frac{1-|\lambda_2(A)|}{2} \\ & & & |\lambda_2(A)| \end{bmatrix} \tag{5.282}
$$

$$
\mu_{\max}h_c(\mu_{\max})\mathbb{1}^T D_{\epsilon,0} = \mu_{\max}h_c(\mu_{\max})\begin{bmatrix} \epsilon & \cdots & \epsilon^{N-1} \end{bmatrix} \tag{5.283}
$$

Therefore, the 1-norm of $D_\epsilon^{-1}\Gamma_0 D_\epsilon$ can be evaluatd as

$$
\begin{aligned}
\|D_{\epsilon,0}^{-1}\Gamma_0 D_{\epsilon,0}\|_1 = \max\Big\{&\gamma_c,\ |\lambda_2(A)| + \mu_{\max}h_c(\mu_{\max})\epsilon, \\
&\frac{1+|\lambda_2(A)|}{2} + \mu_{\max}h_c(\mu_{\max})\epsilon^2, \cdots, \\
&\frac{1+|\lambda_2(A)|}{2} + \mu_{\max}h_c(\mu_{\max})\epsilon^{N-1}\Big\}
\end{aligned} \tag{5.284}
$$

Since $0 < \epsilon \leq 1/4$, we have $\epsilon > \epsilon^2 > \cdots > \epsilon^{N-1} > 0$. Therefore,

$$\|D_{\epsilon,0}^{-1}\Gamma_0 D_{\epsilon,0}\|_1 = \max\Big\{\gamma_c,\ |\lambda_2(A)| + \mu_{\max}h_c(\mu)\epsilon, \\ \frac{1+|\lambda_2(A)|}{2} + \mu_{\max}h_c(\mu_{\max})\epsilon^2\Big\} \tag{5.285}$$

Substituting the above expression for $\|D_{\epsilon,0}^{-1}\Gamma_0 D_{\epsilon,0}\|_1$ into (5.280) leads to

$$\rho(\Gamma) \leq \max\Big\{\gamma_c,\ |\lambda_2(A)| + \mu_{\max}h_c(\mu_{\max})\epsilon, \\ \frac{1+|\lambda_2(A)|}{2} + \mu_{\max}h_c(\mu_{\max})\epsilon^2\Big\} + \frac{1}{3}\mu_{\max}^2\psi_0\epsilon^{-N} \tag{5.286}$$

We recall from (5.121) that $\gamma_c > 0$. To ensure $\rho(\Gamma) < 1$, it suffices to require that $\mu_{\max}$ is such that the following conditions are satisfied:

$$\gamma_c + \frac{1}{3}\mu_{\max}^2\psi_0\epsilon^{-N} < 1 \tag{5.287}$$

$$|\lambda_2(A)| + \mu_{\max}h_c(\mu_{\max})\epsilon + \frac{1}{3}\mu_{\max}^2\psi_0\epsilon^{-N} < 1 \tag{5.288}$$

$$\frac{1+|\lambda_2(A)|}{2} + \mu_{\max}h_c(\mu_{\max})\epsilon^2 + \frac{1}{3}\mu_{\max}^2\psi_0\epsilon^{-N} < 1 \tag{5.289}$$

We now solve these three inequalities to get a condition on $\mu_{\max}$. Substituting the expression for γ_c from (5.120) into (5.287), we get

$$1 - \mu_{\max}\lambda_L + \mu_{\max}^2\Big(\frac{1}{3}\psi_0\epsilon^{-N} + \frac{1}{2}\|p\|_1^2\lambda_U^2\Big) < 1 \tag{5.290}$$

the solution of which is given by

$$0 < \mu_{\max} < \frac{\lambda_L}{\frac{1}{3}\psi_0\epsilon^{-N} + \frac{1}{2}\|p\|_1^2\lambda_U^2} \tag{5.291}$$

For (5.288)–(5.289), if we substitute the expression for $h_c(\mu_{\max})$ from (5.137) into

(5.288)–(5.289), we get a third-order inequality in $\mu_{\max}$, which is difficult to solve in closed-form. However, inequalities (5.288)–(5.289) can be guaranteed by the following conditions:

$$\mu_{\max} h_c(\mu_{\max})\epsilon < \frac{(1-|\lambda_2(A)|)^2}{4}, \ \frac{\mu_{\max}^2 \psi_0 \epsilon^{-N}}{3} < \frac{1-|\lambda_2(A)|}{4} \tag{5.292}$$

This is because we would then have:

$$\begin{aligned} &|\lambda_2(A)| + \mu_{\max} h_c(\mu_{\max})\epsilon + \frac{1}{3}\mu_{\max}^2 \psi_0 \epsilon^{-N} \\ &\quad < |\lambda_2(A)| + \frac{(1-|\lambda_2(A)|)^2}{4} + \frac{1-|\lambda_2(A)|}{4} \\ &\quad \leq |\lambda_2(A)| + \frac{1-|\lambda_2(A)|}{4} + \frac{1-|\lambda_2(A)|}{4} \\ &\quad = \frac{1+|\lambda_2(A)|}{2} < 1 \end{aligned} \tag{5.293}$$

Likewise, by the fact that $0 < \epsilon \leq 1/4 < 1$,

$$\begin{aligned} &\frac{1+|\lambda_2(A)|}{2} + \mu_{\max} h_c(\mu_{\max})\epsilon^2 + \frac{1}{3}\mu_{\max}^2 \psi_0 \epsilon^{-N} \\ &\quad < \frac{1+|\lambda_2(A)|}{2} + \mu_{\max} h_c(\mu_{\max})\epsilon + \frac{1}{3}\mu_{\max}^2 \psi_0 \epsilon^{-N} \\ &\quad < \frac{1+|\lambda_2(A)|}{2} + \frac{(1-|\lambda_2(A)|)^2}{4} + \frac{1-|\lambda_2(A)|}{4} \\ &\quad \leq \frac{1+|\lambda_2(A)|}{2} + \frac{1-|\lambda_2(A)|}{4} + \frac{1-|\lambda_2(A)|}{4} \\ &\quad = 1 \end{aligned} \tag{5.294}$$

Substituting (5.137) and (5.278) into (5.292), we find that the latter conditions

are satisfied for

$$\begin{cases} 0 < \mu_{\max} < \dfrac{\lambda_L}{\|p\|_1^2 \lambda_U^2 \left(\|\bar{P}[\mathcal{A}^T \mathcal{U}_2]\|_\infty^2 + \frac{1}{2}\right)} \\ \\ 0 < \mu_{\max} < \sqrt{\dfrac{3(1-|\lambda_2(A)|)^{2N+1}}{2^{2N+2}\psi_0}} \end{cases} \tag{5.295}$$

Combining (5.287), (5.289) and (5.295) , we arrive at condition (5.145).

5.J Proof of Lemma 5.9

Applying the matrix inversion lemma [68] to (5.133), we get

$$\begin{aligned} (I-\Gamma)^{-1} &= (I - \Gamma_0 - \mu_{\max}^2 \psi_0 \cdot \mathbb{1}\mathbb{1}^T)^{-1} \\ &= (I-\Gamma_0)^{-1} + \frac{\mu_{\max}^2 \psi_0 \cdot (I-\Gamma_0)^{-1}\mathbb{1}\mathbb{1}^T (I-\Gamma_0)^{-1}}{1-\mu_{\max}^2 \psi_0 \cdot \mathbb{1}^T (I-\Gamma_0)^{-1}\mathbb{1}} \end{aligned} \tag{5.296}$$

so that

$$\mathbb{1}^T (I-\Gamma)^{-1} = \frac{1}{1-\mu_{\max}^2 \psi_0 \cdot \mathbb{1}^T (I-\Gamma_0)^{-1}\mathbb{1}} \cdot \mathbb{1}^T (I-\Gamma_0)^{-1} \tag{5.297}$$

By (5.134), the matrix Γ_0 is a 2×2 block upper triangular matrix whose inverse is given by

$$(I-\Gamma_0)^{-1} = \begin{bmatrix} (1-\gamma_c)^{-1} & \frac{\mu_{\max} h_c(\mu_{\max})}{1-\gamma_c} \mathbb{1}^T (I-\Gamma_e)^{-1}) \\ 0 & (I-\Gamma_e)^{-1} \end{bmatrix} \tag{5.298}$$

Substituting (5.120) into the above expression leads to (5.265). Furthermore, from (5.265), we have

$$\mathbb{1}^T(I-\Gamma_0)^{-1}\mathbb{1} = \frac{\mu_{\max}^{-1}}{\lambda_L - \frac{\mu_{\max}}{2}\|p\|_1^2\lambda_U^2} + \left(1+\frac{h_c(\mu_{\max})}{\lambda_L-\frac{\mu}{2}\|p\|_1^2\lambda_U^2}\right)\mathbb{1}^T(I-\Gamma_e)^{-1}\mathbb{1} \tag{5.299}$$

Substituting (5.299) into (5.297), we obtain (5.264).

5.K Proof of Theorem 5.5

Taking the squared Euclidean norm of both sides of (5.67) and applying the expectation operator, we obtain

$$\begin{aligned}\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \|\tilde{w}_{c,i}\|^2 &+ \mathbb{E}\,\|\check{\boldsymbol{w}}_{c,i} + (u_{L,k}\otimes I_M)\boldsymbol{w}_{e,i}\|^2 \\ &- 2\tilde{w}_{c,i}^T\left[\mathbb{E}\check{\boldsymbol{w}}_{c,i} + (u_{L,k}\otimes I_M)\boldsymbol{w}_{e,i}\right]\end{aligned} \tag{5.300}$$

which means that, for all $i \geq 0$,

$$\begin{aligned}&\left|\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 - \|\tilde{w}_{c,i}\|^2\right| \\ &= \Big|\mathbb{E}\,\|\check{\boldsymbol{w}}_{c,i} + (u_{L,k}\otimes I_M)\boldsymbol{w}_{e,i}\|^2 \\ &\quad - 2\tilde{w}_{c,i}^T\big[\mathbb{E}\check{\boldsymbol{w}}_{c,i} + (u_{L,k}\otimes I_M)\mathbb{E}\boldsymbol{w}_{e,i}\big]\Big| \\ &\overset{(a)}{\leq} \mathbb{E}\,\|\check{\boldsymbol{w}}_{c,i} + (u_{L,k}\otimes I_M)\boldsymbol{w}_{e,i}\|^2 \\ &\quad + 2\|\tilde{w}_{c,i}\|\cdot\big[\,\|\mathbb{E}\check{\boldsymbol{w}}_{c,i}\| + \|u_{L,k}\otimes I_M\|\cdot\|\mathbb{E}\boldsymbol{w}_{e,i}\|\big] \\ &\overset{(b)}{\leq} 2\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2 + 2\|u_{L,k}\otimes I_M\|^2\cdot\mathbb{E}\|\boldsymbol{w}_{e,i}\|^2 \\ &\quad + 2\|\tilde{w}_{c,i}\|\cdot\big[\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\| + \|u_{L,k}\otimes I_M\|\cdot\mathbb{E}\|\boldsymbol{w}_{e,i}\|\big]\end{aligned}$$

$$
\begin{aligned}
&\overset{(c)}{\leq} 2\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2 + 2\|u_{L,k}\otimes I_M\|^2\cdot\mathbb{E}\|\boldsymbol{w}_{e,i}\|^2 \\
&\quad + 2\|\tilde{w}_{c,i}\|\cdot\big[\sqrt{\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2} + \|u_{L,k}\otimes I_M\|\cdot\sqrt{\mathbb{E}\|\boldsymbol{w}_{e,i}\|^2}\big] \\
&\overset{(d)}{=} 2\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}] + 2\|u_{L,k}\otimes I_M\|^2\cdot\mathbb{1}^T\mathbb{E}P[\boldsymbol{w}_{e,i}] \\
&\quad + 2\|\tilde{w}_{c,i}\|\cdot\Big[\sqrt{\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}]} + \|u_{L,k}\otimes I_M\|\cdot\sqrt{\mathbb{1}^T\mathbb{E}P[\boldsymbol{w}_{e,i}]}\Big] \\
&\overset{(e)}{\leq} O(\mu_{\max}) + 2\|u_{L,k}\otimes I_M\|^2\cdot\big(\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0} + \mathbb{1}^T\check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'}\big) \\
&\quad + 2\gamma_c^i\|\tilde{w}_{c,0}\|\cdot\Big[O(\mu_{\max}^{\frac{1}{2}}) \\
&\qquad\quad + \|u_{L,k}\otimes I_M\|\cdot\sqrt{\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0} + \mathbb{1}^T\check{\mathcal{W}}_{e,\infty}^{\mathrm{ub}'}}\Big] \\
&\overset{(f)}{\leq} O(\mu_{\max}) + 2\|u_{L,k}\otimes I_M\|^2\cdot\big(\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0} + O(\mu_{\max}^2)\big) \\
&\quad + 2\gamma_c^i\|\tilde{w}_{c,0}\|\cdot\big[O(\mu_{\max}^{\frac{1}{2}}) \\
&\qquad\quad + \|u_{L,k}\otimes I_M\|\cdot\left(\sqrt{\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0}} + \sqrt{O(\mu_{\max}^2)}\right)\big] \\
&= 2\|u_{L,k}\otimes I_M\|^2\cdot\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0} \\
&\qquad\quad + 2\gamma_c^i\cdot\|\tilde{w}_{c,0}\|\cdot\|u_{L,k}\otimes I_M\|\cdot\sqrt{\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0}} \\
&\quad + 2\gamma_c^i\|\tilde{w}_{c,0}\|\cdot\Big[O(\mu_{\max}^{\frac{1}{2}}) + \|u_{L,k}\otimes I_M\|\cdot O(\mu_{\max})\Big] \\
&\quad + O(\mu_{\max}) + O(\mu_{\max}^2) \\
&\overset{(g)}{\leq} 2\|u_{L,k}\otimes I_M\|^2\cdot\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0} \\
&\quad + 2\|\tilde{w}_{c,0}\|\cdot\|u_{L,k}\otimes I_M\|\cdot\sqrt{\mathbb{1}^T\Gamma_e^i\mathcal{W}_{e,0}} \\
&\quad + \gamma_c^i\cdot O(\mu_{\max}^{\frac{1}{2}}) + O(\mu_{\max})
\end{aligned}
\tag{5.301}
$$

where step (a) used Cauchy-Schwartz inequality, step (b) used $\|x+y\|^2 \leq 2\|x\|^2 + 2\|y\|^2$, step (c) applied Jensen's inequality to the concave function $\sqrt{\cdot}$, step (d) used property (5.111), step (e) substituted the non-asymptotic bounds (5.140) and (5.141) and the fact that $\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}] \leq O(\mu_{\max})$ for *all* $i \geq 0$ from (5.144), step (f) used (5.143) and the fact that $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$ for $x, y \geq 0$, and step (g)

used $\gamma_c < 1$ for sufficiently small step-sizes (guaranteed by (5.130)).

CHAPTER 6

Performance Analysis

6.1 Introduction

In Chapter 5, we carried out a detailed transient analysis of the learning behavior of multi-agent networks, which have applications in different contexts [7, 19, 26, 29, 34, 36, 42, 46, 48, 49, 51, 52, 58, 67, 70, 75–77, 84, 85, 89, 91, 96–98, 100, 102, 107, 109, 110, 113–115, 125, 130, 136, 137, 140, 146]. The analysis revealed interesting results about the learning abilities of distributed strategies when *constant* step-sizes are used to ensure continuous tracking of drifts in the data. It was noted that when constant step-sizes are employed to drive the learning process, the dynamics of the distributed strategies is modified in a critical manner. Specifically, components that relate to gradient noise are not annihilated any longer, as happens when diminishing step-sizes are used. These noise components remain persistently active throughout the adaptation process and it becomes necessary to examine their impact on network performance, such as examining questions of the following nature: (a) can these persistent noise components drive the network unstable? (b) can the degradation in performance be controlled and minimized? (c) what is the size of the degradation? Motivated by these questions, we provided in Chapter 5 detailed answers to the following three inquiries: (i) where does the distributed strategy converge to? (ii) under what conditions on the data and network topology does it converge? (iii) and what are the rates of convergence

of the learning process? In particular, we showed in Chapter 5 that there always exist sufficiently small constant step-sizes that ensure the mean-square convergence of the learning process to a well-defined limit point even in the presence of persistent gradient noise. We characterized this limit point as the *unique* fixed point solution of a nonlinear algebraic equation consisting of the weighted sum of individual update functions — see Eq. (6.5) further ahead. The scaling weights $\{p_k\}$ that appear in this equation were shown to be determined by the entries of the right-eigenvector θ of the network combination policy corresponding to the eigenvalue at one (also called the Perron eigenvector; its entries are normalized to add up to one and are all strictly positive for strongly-connected networks) — see Eq. (6.6). The analysis from Chapter 5 further revealed that the learning curve of the multi-agent system exhibits *three* distinct phases. In the first phase (Transient Phase I), the convergence rate of the network is determined by the second largest eigenvalue of the combination policy in magnitude, which is related to the degree of network connectivity. In the second phase (Transient Phase II), the convergence rate is determined by the Perron eigenvector. And, in the third phase (the steady-state phase) the mean-square error (MSE) performance attains a bound on the order of $O(\mu_{\max})$, where $\mu_{\max}$ is the largest step-size among all agents.

In this chapter, we address in some detail two additional questions related to network performance, namely, iv) how close do the individual agents get to the limit point of the network? and v) can the system of networked agents be made to match the learning performance of a centralized solution where all information is collected and processed centrally by a fusion center? In the process of answering these questions, we shall derive a closed-form expression for the steady-state MSE of each agent — see (6.14) further ahead. Expression (6.14) turns out to be a revealing result; it amounts to a non-trivial extension of a classical result for

stand-alone adaptive agents [54, 57, 72, 141] to the more demanding context of networked agents and for cost functions that are not necessarily quadratic or of the mean-square-error type. As we are going to explain in the sequel, relation (6.14) captures the effect of the network topology (through the $\{p_k\}$), gradient noise (through the covariance matrix $\mathcal{R}_v$) , and data characteristics (through the Lyapunov solution X) in an integrated manner and shows how these various factors influence performance. Result (6.14) applies to connected networks under fairly general conditions and for fairly general aggregate cost functions.

We shall also explain later in Sections 6.5 and 6.6 of this chapter that, as long as the network is strongly connected, a left-stochastic combination matrix can always be constructed to have any desired Perron-eigenvector — see expressions (6.66) and (6.73). This observation has an important ramification for the following reason. Starting from any collection of N agents, there exists a finite number of topologies that can link these agents together. And for each possible topology, there are infinitely many combination policies that can be used to train the network. Since the performance of the network is dependent on the Perron-eigenvector of its combination policy, one of the important conclusions that will follow is that regardless of the network topology, there will always exist choices for the respective combination policies such that the steady-state performance of all topologies can be made identical to each other to first-order in $\mu_{\max}$. In other words, no matter how the agents are connected to each other, there is always a way to select the combination weights such that the performance of the network is invariant to the topology. This will also mean that, for any connected topology, there is always a way to select the combination weights such that the performance of the network matches that of the centralized stochastic-approximation (since a centralized solution can be viewed as corresponding to a fully-connected network). The following presentation in this chapter is based on [38].

6.2 Family of Distributed Strategies

6.2.1 Distributed Strategies: Consensus and Diffusion

We consider a connected network of N agents that are linked together through a topology — see Fig. 5.1 in Chapter 5. Each agent k implements a distributed algorithm of the following form to update its state vector from $\boldsymbol{w}_{k,i-1}$ to $\boldsymbol{w}_{k,i}$:

$$\boldsymbol{\phi}_{k,i-1} = \sum_{l=1}^{N} a_{1,lk}\boldsymbol{w}_{l,i-1} \tag{6.1}$$

$$\boldsymbol{\psi}_{k,i} = \sum_{l=1}^{N} a_{0,lk}\boldsymbol{\phi}_{l,i-1} - \mu_k\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{\phi}_{k,i-1}) \tag{6.2}$$

$$\boldsymbol{w}_{k,i} = \sum_{l=1}^{N} a_{2,lk}\boldsymbol{\psi}_{l,i} \tag{6.3}$$

where $\boldsymbol{w}_{k,i} \in \mathbb{R}^M$ is the state of agent k at time i, usually an estimate for the solution of some optimization problem, $\boldsymbol{\phi}_{k,i-1} \in \mathbb{R}^M$ and $\boldsymbol{\psi}_{k,i} \in \mathbb{R}^M$ are intermediate variables generated at node k before updating to $\boldsymbol{w}_{k,i}$, μ_k is a non-negative constant step-size parameter used by node k, and $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ is an $M \times 1$ update vector function at node k. We explained in Chapter 5 that in deterministic optimization problems, the update vectors $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ can be selected as the gradient or Newton steps associated with the individual utility functions at the agents [97]. On the other hand, in stocastic approximation problems, such as adaptation, learning and estimation problems [26,34,36,42,48,49,58,75,77,89,91,109,115,125, 130, 137, 146], the update vectors $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ are usually computed from realizations of data samples that arrive sequentially at the nodes. In the stochastic setting, the quantities appearing in (6.1)–(6.3) become random variables and we shall use boldface letters to highlight their stochastic nature. In Example 5.1 of Chapter 5, we illustrated various choices for $\hat{\boldsymbol{s}}_{k,i}(w)$ in different contexts.

The combination coefficients $a_{1,lk}$, $a_{0,lk}$ and $a_{2,lk}$ in (6.1)–(6.3) are nonnegative convex-combination weights that each node k assigns to the information arriving from node l and will be zero if agent l is not in the neighborhood of agent k. Therefore, each summation in (6.1)–(6.3) is actually confined to the neighborhood of node k. We let A_1, A_0 and A_2 denote the $N \times N$ matrices that collect the coefficients $\{a_{1,lk}\}$, $\{a_{0,lk}\}$ and $\{a_{2,lk}\}$. Then, the matrices A_1, A_0 and A_2 satisfy

$$A_1^T \mathbb{1} = \mathbb{1}, \quad A_0^T \mathbb{1} = \mathbb{1}, \quad A_2^T \mathbb{1} = \mathbb{1} \tag{6.4}$$

where $\mathbb{1}$ is the $N \times 1$ vector with all its entries equal to one. Condition (6.4) means that the matrices $\{A_0, A_1, A_2\}$ are left-stochastic (i.e., the entries on each of their columns add up to one). We also explained in Chapter 5 that different choices for A_1, A_0 and A_2 correspond to different distributed strategies, such as the traditional consensus [48, 75–77, 97, 98, 137] and diffusion (ATC and CTA) [26, 34, 36, 42, 89, 91, 115, 146] algorithms. In our analysis, we will proceed with the general form (6.1)–(6.3) to study all three schemes, and other possibilities, within a unifying framework.

6.2.2 Review of the Main Results from Chapter 5

Due the coupled nature of the social and self-learning steps in (6.1)–(6.3), information derived from local data at agent k will be propagated to its neighbors and from there to their neighbors in a diffusive learning process. It is expected that some global performance pattern will emerge from these localized interactions in the multi-agent system. As mentioned in the introductory remarks, in Chapter 5 and in this chapter, we examine the following five questions:

- Limit point: where does each state $\boldsymbol{w}_{k,i}$ converge to?

- Stability: under which condition does convergence occur?
- Learning rate: how fast does convergence occur?
- Performance: how close does $\boldsymbol{w}_{k,i}$ get to the limit point?
- Generalization: can $\boldsymbol{w}_{k,i}$ match the performance of a centralized solution?

In Chapter 5, we addressed the first three questions in detail and derived expressions that fully characterize the answer in each case. One of the major and interesting conclusions established in Chapter 5 is that for general *left-stochastic* matrices $\{A_1, A_0, A_2\}$, the agents in the network will have their iterates $\boldsymbol{w}_{k,i}$ converge, in the mean-square-error sense, to the *same* limit vector w^o that corresponds to the unique solution of the following algebraic equation:

$$\sum_{k=1}^{N} p_k s_k(w) = 0 \tag{6.5}$$

where the update functions $s_k(\cdot)$ are defined further ahead in (6.15) as the conditional means of the update directions $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ used in (6.1)–(6.3), and each positive coefficient p_k is the kth entry of the following vector:

$$p = \text{col}\left\{\frac{\mu_1}{\mu_{\max}}\pi_1, \ldots, \frac{\mu_N}{\mu_{\max}}\pi_N\right\} \tag{6.6}$$

Here, $\mu_{\max}$ is the largest step-size among all agents, π_k is the kth entry of the vector $\pi \triangleq A_2\theta$, and θ is the right eigenvector of $A \triangleq A_1A_0A_2$ corresponding to the eigenvalue at one with its entries normalized to add up to one, i.e.,

$$A\theta = \theta, \qquad \mathbb{1}^T\theta = 1 \tag{6.7}$$

We refer to θ as the Perron eigenvector of A. The unique solution w^o of (6.5) has the interpretation of a Pareto optimal solution corresponding to the weights $\{p_k\}$ [20, 36, 37]. By selecting different combination policies A, or even different topologies, the entries $\{p_k\}$ can be made to change (since θ will change) and the limit point w^o resulting from (6.5) can be steered towards different Pareto optimal solutions.

The second major conclusion from Chapter 5 is that, during the convergence process towards the limit point w^o, the learning curve at each agent exhibits *three* distinct phases (see Fig. 5.2 in Chapter 5): Transient Phase I, Transient Phase II, and Steady-State Phase. These phases were shown in Chapter 5 to have the following features:

- **Transient Phase I:**

  If the agents are initialized at different values, then the iterates at the various agents will initially evolve in such a way to make each $\boldsymbol{w}_{k,i}$ get closer to the following *reference* (centralized) recursion $\bar{w}_{c,i}$:

$$\bar{w}_{c,i} = \bar{w}_{c,i-1} - \mu_{\max} \sum_{k=1}^{N} p_k s_k(\bar{w}_{c,i-1}) \tag{6.8}$$

  which is initialized at

$$\bar{w}_{c,0} = \sum_{k=1}^{N} \theta_k w_{k,0} \tag{6.9}$$

  where $w_{k,0}$ is the initial value of the distributed strategy at agent k. The rate at which the agents approach $\bar{w}_{c,i}$ is determined by $|\lambda_2(A)|$, the second largest eigenvalue of A in magnitude. If the agents are initialized at the same value, say, e.g., $\boldsymbol{w}_{k,0} = 0$, then the learning curves start at Transient

Phase II directly.

- **Transient Phase II:**

  In this phase, the trajectories of all agents are uniformly close to the trajectory of the reference recursion; they converge in a coordinated manner to steady-state. The learning curves at this phase are well modeled by the same reference recursion (6.8) since we showed in (6.10) from Chapter 5 that:

$$\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \|\tilde{w}_{c,i}\|^2 + O(\mu_{\max}^{1/2}) \cdot \gamma_c^i + O(\mu_{\max}) \tag{6.10}$$

  where the error vectors are defined by $\tilde{\boldsymbol{w}}_{k,i} = w^o - \boldsymbol{w}_{k,i}$ and $\tilde{w}_{c,i} = w^o - \bar{w}_{c,i}$. Furthermore, for small step-sizes and during the later stages of this phase, $\bar{w}_{c,i}$ will be close enough to w^o and the convergence rate r was shown in expression (5.131) from Chapter 5 to be given by

$$r = \big[\rho(I_M - \mu_{\max} H_c)\big]^2 + O\big((\mu_{\max}\epsilon)^{\frac{1}{2(M-1)}}\big) \tag{6.11}$$

  where $\rho(\cdot)$ denotes the spectral radius of its matrix argument, ϵ is an arbitrarily small positive number, and H_c is defined as the aggregate (Hessian-type) sum:

$$H_c \triangleq \sum_{k=1}^{N} p_k \nabla_{w^T} s_k(w^o) \tag{6.12}$$

- **Steady-State Phase:**

  The reference recursion (6.8) continues converging towards w^o so that $\|\tilde{w}_{c,i}\|^2$ will converge to zero ($-\infty$ dB in Fig. 5.2 of Chapter 5). However, for the distributed strategy (6.1)–(6.3), the mean-square-error $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2$ at each

agent k will converge to a *finite* steady-state value that is on the order of $O(\mu_{\max})$:

$$\limsup_{i\to} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 \leq O(\mu_{\max}) \tag{6.13}$$

Note that the bound (6.13) provides a partial answer to the fourth question we are interested in, namely, how close the $\boldsymbol{w}_{k,i}$ get to the network limit point w^o. Expression (6.13) indicates that the mean-square error is on the order of $\mu_{\max}$. However, in this chapter, we will examine this mean-square error more closely and provide a more accurate characterization of the steady-state MSE value by deriving a closed-form expression for it. In particular, we will be able to characterize this MSE value in terms of the vector p as follows:

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \mu_{\max} \cdot \mathrm{Tr}\left\{X(p^T \otimes I_M) \cdot \mathcal{R}_v \cdot (p \otimes I_M)\right\} + o(\mu_{\max}) \tag{6.14}$$

where X is the solution to a certain Lyapunov equation described later in (6.42) (when $\Sigma = I$), $\mathcal{R}_v$ is a gradient noise covariance matrix defined below in (6.25), and $o(\mu_{\max})$ denotes a strictly higher order term of $\mu_{\max}$. Expression (6.14) is a most revealing result; it captures the effect of the network topology through the eigenvector p, and it captures the effects of gradient noise and data characteristics through the matrices $\mathcal{R}_v$ and X, respectively. Expression (6.14) is a non-trivial extension of a classical and famous result pertaining to the mean-square-error performance of stand-alone adaptive agents [54,57,72,141] to the more demanding context of networked agents. In particular, it can be easily verified that (6.14) reduces to the well-known $\mu M \sigma_v^2/2$ expression for the mean-square deviation of single LMS learners when the network size is set to $N = 1$ and the topology is removed [54,57,72,141]. However, expression (6.14) is not limited to single agents

or to mean-square-error costs. It applies to rather general connected networks and to fairly general cost functions.

6.2.3 Relation to Prior Work

As pointed out in Chapter 5 (see Sec. 5.2.2), most prior works in the literature [13,21,48,71,75–77,84,97,109,125,126,137] focus on studying the performance and convergence of their respective distributed strategies under *diminishing* step-size conditions and for *doubly-stochastic* combination policies. In contrast, we focus on *constant* step-sizes in order to enable continuous adaptation and learning under drifting conditions. We also focus on *left-stochastic combination matrices* in order to induce flexibility about the network limit point; this is because doubly-stochastic policies force the network to converge to the *same* limit point, while left-stochastic policies enable the networks to converge to any of infinitely many Pareto optimal solutions. Moreover, the value of the limit point can be controlled through the selection of the Perron eigenvector.

Furthermore, the performance of distributed strategies has usually been characterized in terms of bounds on their steady-state mean-square-error performance — see, e.g., [71,84,97,109,125,126,137]. In Chapter 5 of the work, as a byproduct of our study of the three stages of the learning process, we were able to derive performance bounds for the steady-state MSE of a fairly general class of distributed strategies under broader (weaker) conditions than normally considered in the literature. In this chapter, we push the analysis noticeably further and derive a closed-form expression for the steady-state MSE in the slow adaptation regime, such as expression (6.14), which captures in an integrated manner how various network parameters (topology, combination policy, utilities) influence performance.

Other useful and related works in the literature appear in [13, 75–77]. These works, however, study the distribution of the error vector in steady-state under *diminishing* step-size conditions and using central limit theorem (CLT) arguments. They showed a Gaussian distribution for the error quantities in steady-state and derived an expression for the error variance but their expression naturally tends to zero as $i \to \infty$ since, under the conditions assumed in these works, the error vector $\tilde{\boldsymbol{w}}_{k,i}$ approaches zero almost surely. Such results are possible because, in the diminishing step-size case, the influence of gradient noise is annihilated by the decaying step-size. However, in the *constant* step-size regime, the influence of gradient noise is always present and seeps into the operation of the algorithm. In this case, the error vector does *not* approach zero any longer and its variance approaches instead a steady-state *positive-definite* value. Our objective is to characterize this steady-state value and to examine how it is influenced by the network topology, by the persistent gradient noise conditions, and by the data characteristics and utility functions. In the constant step-size regime, CLT arguments cannot be employed anymore because the Gaussianity result does not hold any longer. Indeed, reference [145] illustrates this situation clearly; it derived an expression for the characteristic function of the limiting error distribution in the case of mean-square-error estimation and it was shown that the distribution is not Gaussian. For these reasons, the analysis in this work is based on alternative techniques that do not pursue any specific form for the steady-state distribution and that rely instead on the use of energy conservation arguments [34, 115, 116]. As the analysis and detailed derivations in the appendices show, this is a formidable task to pursue due to the coupling among the agents and the persistent noise conditions. Nevertheless, under certain conditions that are generally weaker than similar conditions used in related contexts in the literature, we will be able to derive accurate expressions for the network MSE performance and its convergence

rate in small constant step-size regime.

6.3 Modeling Assumptions

In this section, we first recall the assumptions used in Chapter 5 and then introduce two conditions that are required to carry out the MSE analysis in this chapter. We already explained in Sec. 5.3 of Chapter 5 how the assumptions listed below relate to, and extend, similar conditions used in the literature.

Assumption 6.1 (Strongly-connected network)**.** *The $N \times N$ matrix product $A \triangleq A_1 A_0 A_2$ is assumed to be a primitive left-stochastic matrix, i.e., $A^T \mathbb{1} = \mathbb{1}$ and there exists a finite integer j_o such that all entries of A^{j_o} are strictly positive.*

□

Assumption 6.2 (Update vector: Randomness)**.** *There exists an $M \times 1$ deterministic vector function $s_k(w)$ such that, for all $M \times 1$ vectors $\boldsymbol{w}$ in the filtration $\mathcal{F}_{i-1}$ generated by the past history of iterates $\{\boldsymbol{w}_{k,j}\}$ for $j \leq i-1$ and all k, it holds that*

$$\mathbb{E}\left\{\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w})|\mathcal{F}_{i-1}\right\} = s_k(\boldsymbol{w}) \tag{6.15}$$

for all i, k. Furthermore, there exist $\alpha \geq 0$ and $\sigma_v^2 \geq 0$ such that for all i, k and $\boldsymbol{w} \in \mathcal{F}_{i-1}$:

$$\mathbb{E}\left\{\|\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}) - s_k(\boldsymbol{w})\|^2 \,\middle|\, \mathcal{F}_{i-1}\right\} \leq \alpha \cdot \|\boldsymbol{w}\|^2 + \sigma_v^2 \tag{6.16}$$

□

Assumption 6.3 (Update vector: Lipschitz)**.** *There exists a nonnegative* λ_U *such that for all* $x, y \in \mathbb{R}^M$ *and all* k*:*

$$\|s_k(x) - s_k(y)\| \leq \lambda_U \cdot \|x - y\| \tag{6.17}$$

where the subscript "U*" in* λ_U *means "upper bound".* □

Assumption 6.4 (Update vector: Strong monotonicity)**.** *Let* p_k *denote the* k*th entry of the vector* p *defined in* (6.6)*. There exists* $\lambda_L > 0$ *such that for all* $x, y \in \mathbb{R}^M$*:*

$$(x - y)^T \cdot \sum_{k=1}^{N} p_k \Big[s_k(x) - s_k(y) \Big] \geq \lambda_L \cdot \|x - y\|^2 \tag{6.18}$$

where the subscript "L*" in* λ_L *means "lower bound".* □

Assumption 6.5 (Jacobian matrix: Lipschitz)**.** *Let* w^o *denote the limit point of the distributed strategy* (6.1)–(6.3)*, which was defined earlier as the unique solution to* (6.5) *and was characterized in Theorem 5.1. Then, in a small neighborhood around* w^o*, we assume that* $s_k(w)$ *is differentiable with respect to* w *and satisfies*

$$\|\nabla_{w^T} s_k(w^o + \delta w) - \nabla_{w^T} s_k(w^o)\| \leq \lambda_H \cdot \|\delta w\| \tag{6.19}$$

for all $\|\delta w\| \leq r_H$ *for some small* r_H*, and where* λ_H *is a nonnegative number independent of* δw *and* w^o*.*

□

The following lemma gives the equivalent forms of Assumptions 6.3–6.4 when

the $\{s_k(w)\}$ happen to be differentiable.

Lemma 6.1 (Equivalent conditions on update vectors)**.** *Suppose $\{s_k(w)\}$ are differentiable in an open set $\mathcal{S} \subseteq \mathbb{R}^M$. Then, having conditions* (6.17) *and* (6.18) *hold on $\mathcal{S}$ is equivalent to the following conditions, respectively,*

$$\|\nabla_{w^T} s_k(w)\| \leq \lambda_U \tag{6.20}$$

$$\frac{1}{2}[H_c(w) + H_c^T(w)] \geq \lambda_L \cdot I_M \tag{6.21}$$

for any $w \in \mathcal{S}$, where $\|\cdot\|$ denotes the $2-$ induced norm (largest singular value) of its matrix argument and

$$H_c(w) \triangleq \sum_{k=1}^{n} p_k \nabla_{w^T} s_k(w) \tag{6.22}$$

Proof. See Appendix 5.A in Chapter 5. □

Next, we introduce two new assumptions on $\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w})$, which are needed for the MSE analysis of this chapter. Assumption 6.6 below has been used before in the stochastic approximation literature — see, for example, [112] and Eq. (6.2) in Theorem 6.1 of [99, p.147].

Assumption 6.6 (Second-order moment of gradient noise)**.** *Let $\boldsymbol{v}_i(x)$ denote the $MN \times 1$ global vector that collects the statistical fluctuations in the stochastic update vectors across all agents:*

$$\boldsymbol{v}_i(x) \triangleq \mathrm{col}\{\hat{\boldsymbol{s}}_{1,i}(x_1) - s_1(x_1),\ \ldots,\ \hat{\boldsymbol{s}}_{N,i}(x_N) - s_N(x_N)\} \tag{6.23}$$

where we are using the vector x to denote a block vector consisting of entries x_k of size $M \times 1$ each, i.e., $x \triangleq \mathrm{col}\{x_1, \ldots, x_N\}$. For any $\boldsymbol{x}_k \in \mathcal{F}_{i-1}$, $1 \leq k \leq N$,

we introduce the covariance matrix:

$$\mathcal{R}_{v,i}(\boldsymbol{x}) \triangleq \mathbb{E}\left\{\boldsymbol{v}_i(\boldsymbol{x})\boldsymbol{v}_i^T(\boldsymbol{x})\big|\mathcal{F}_{i-1}\right\} \tag{6.24}$$

where, again, we are using the notation $\boldsymbol{x}$ *to refer to the block vector* $\boldsymbol{x} = col\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ *with stochastic entries of size* $M \times 1$ *each. Note that* $\mathcal{R}_{v,i}(\boldsymbol{x})$ *generally depends on time* i*. This is because the distribution of* $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ *given* $\mathcal{F}_{i-1}$ *usually varies with time. We assume that, in the limit, this second-order moment of the distribution becomes invariant and tends to a constant value when evaluated at* $\boldsymbol{x} = \mathbb{1} \otimes w^o$*:*

$$\lim_{i\to\infty} \mathcal{R}_{v,i}(\mathbb{1} \otimes w^o) \triangleq \mathcal{R}_v \tag{6.25}$$

Furthermore, in a small neighborhood around $\mathbb{1} \otimes w^o$*, we assume that there exists a* $\lambda_v \geq 0$ *and a* $0 < \kappa \leq 4$ *such that for all* $i \geq 0$*:*

$$\|\mathcal{R}_{v,i}(\mathbb{1} \otimes w^o + \delta x) - \mathcal{R}_{v,i}(\mathbb{1} \otimes w^o)\| \leq \lambda_v \cdot \|\delta x\|^\kappa \tag{6.26}$$

for all $\|\delta x\| \leq r_V$ *for some* r_V*.* □

Example 6.1. We illustrate how Assumption 6.6 holds automatically in the context of distributed least-mean-squares estimation. Suppose each agent k receives a stream of data samples $\{\boldsymbol{u}_{k,i}, \boldsymbol{d}_k(i)\}$ that are generated by the following linear model:

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^o + \boldsymbol{n}_k(i) \tag{6.27}$$

where the $1 \times M$ regressors $\{\boldsymbol{u}_{k,i}\}$ are zero mean and independent over time and space with covariance matrix $R_{u,k} = \mathbb{E}\{\boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i}\} \geq 0$ and the noise sequence

$\{\boldsymbol{n}_l(j)\}$ is also zero mean, white, with variance $\sigma_{n,l}^2$, and independent of the regressors $\{\boldsymbol{u}_{k,i}\}$ for all l, k, i, j. The objective is to estimate the $M \times 1$ parameter vector w^o by minimizing the following global cost function

$$J^{\text{glob}}(w) = \sum_{k=1}^{N} J_k(w) \tag{6.28}$$

where

$$J_k(w) = \mathbb{E}|\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}w|^2 \tag{6.29}$$

In this case, the actual gradient vector when evaluated at an $M \times 1$ vector x_k is given by

$$s_k(x_k) = \nabla_w \mathbb{E}|\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}x_k|^2 \tag{6.30}$$

and it can be replaced by the instantaneous approximation

$$\hat{\boldsymbol{s}}_{k,i}(x_k) = -2\boldsymbol{u}_{l,i}^T[\boldsymbol{d}_l(i) - \boldsymbol{u}_{l,i}x_k] \tag{6.31}$$

(Recall from (6.2) that the stochastic gradient at each agent k is evaluated at $\phi_{k,i-1}$ and in this case $x_k = \phi_{k,i-1}$.) It follows that the gradient noise vector $\boldsymbol{v}_{k,i}(x_k)$ evaluated at x_k, at each agent k is given by

$$\boldsymbol{v}_{k,i}(x_k) = 2(R_{u,k} - \boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i})(w^o - x_k) - 2\boldsymbol{u}_{k,i}^T\boldsymbol{n}_k(i) \tag{6.32}$$

and it is straightforward to verify that

$$\mathcal{R}_{v,i}(\mathbb{1} \otimes w^o) = \text{diag}\{4\sigma_{n,1}^2 R_{u,1}, \cdots, 4\sigma_{n,N}^2 R_{u,N}\} \tag{6.33}$$

which is independent of i and, therefore, condition (6.25) holds with $\mathcal{R}_v$ given by (6.33). Furthermore, condition (6.26) is also satisfied. Indeed, let $x = \mathrm{col}\{x_1, \ldots, x_N\} \in \mathbb{R}^{MN}$, and from (6.32) we find that

$$\mathcal{R}_{v,i}(x) = \mathrm{diag}\{G_1, \ldots, G_N\} + \mathcal{R}_{v,i}(\mathbb{1} \otimes w^o) \tag{6.34}$$

where each G_k is a function of $w^o - x_k$ and is given by

$$\begin{aligned} G_k \triangleq 4 \cdot \mathbb{E}\big\{ & (R_{u,k} - \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i})(w^o - x_k) \\ & (w^o - x_k)^T (R_{u,k} - \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i})^T \big\} \end{aligned} \tag{6.35}$$

Note that

$$\|G_k\| \leq 4 \cdot \mathbb{E}\left\| R_{u,k} - \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i} \right\|^2 \cdot \|w^o - x_k\|^2 \tag{6.36}$$

so that

$$\begin{aligned} & \big\| \mathcal{R}_{v,i}(x) - \mathcal{R}_{v,i}(\mathbb{1} \otimes w^o) \big\| \\ & = \max_{1 \leq k \leq N} \|G_k\| \\ & \leq \max_{1 \leq k \leq N} \big\{ 4 \cdot \mathbb{E} \|R_{u,k} - \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i}\|^2 \cdot \|w^o - x_k\|^2 \big\} \\ & \leq \max_{1 \leq k \leq N} \big\{ 4 \cdot \mathbb{E} \|R_{u,k} - \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i}\|^2 \big\} \cdot \max_{1 \leq k \leq N} \|w^o - x_k\|^2 \\ & \leq \max_{1 \leq k \leq N} \big\{ 4 \cdot \mathbb{E} \|R_{u,k} - \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i}\|^2 \big\} \cdot \sum_{k=1}^{N} \|w^o - x_k\|^2 \\ & = \max_{1 \leq k \leq N} \big\{ 4 \cdot \mathbb{E} \|R_{u,k} - \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i}\|^2 \big\} \cdot \|\mathbb{1} \otimes w^o - x\|^2 \end{aligned} \tag{6.37}$$

In other words, condition (6.26) holds for the least-mean-squares estimation case.

□

Assumption 6.7 (Fourth-order moment of gradient noise)**.** *There exist nonnegative numbers α_4 and σ_{v4}^2 such that for any $M \times 1$ random vector $\boldsymbol{w} \in \mathcal{F}_{i-1}$,*

$$\mathbb{E}\left\{\|\boldsymbol{v}_{k,i}(\boldsymbol{w})\|^4 \big| \mathcal{F}_{i-1}\right\} \leq \alpha_4 \cdot \|\boldsymbol{w}\|^4 + \sigma_{v4}^4 \tag{6.38}$$

□

This assumption will be used in the analysis for constant step-size adaptation to arrive at accurate expressions for the steady-state MSE of the agents. By assuming that the fourth-order moment of the gradient noise is bounded as in (6.38), it becomes possible to derive MSE expressions that can be shown to be at most $O(\mu_{\max}^{\min(3/2,1+\kappa/2)})$ away from the actual MSE performance. When the step-sizes are sufficiently small, the size of the term $\mu_{\max}^{\min(3/2,1+\kappa/2)}$ is even smaller and, for all practical purposes, this term is negligible — see expression (6.1) in Theorem 6.1.

Example 6.2. It turns out that condition (6.38) is automatically satisfied in the context of distributed least-mean-squares estimation. We continue with the setting of Example 6.1. From expression (6.32), we have that for any $M \times 1$ random vector $\boldsymbol{w} \in \mathcal{F}_{i-1}$,

$$\begin{aligned}
\|\boldsymbol{v}_{k,i}(\boldsymbol{w})\|^4 &= 16\left\|(R_{u,k} - \boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i})(w^o - \boldsymbol{w}) - \boldsymbol{u}_{k,i}^T\boldsymbol{n}_k(i)\right\|^4 \\
&\overset{(a)}{\leq} 16 \times 8\Big(\left\|R_{u,k} - \boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|^4 \cdot \|w^o - \boldsymbol{w}\|^4 + \|\boldsymbol{u}_{k,i}\|^4 \cdot \|\boldsymbol{n}_k(i)\|^4\Big) \\
&\overset{(b)}{\leq} 128\Big(8\left\|R_{u,k} - \boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|^4 \cdot \|\boldsymbol{w}\|^4 + 8\left\|R_{u,k} - \boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|^4 \cdot \|w^o\|^4 \\
&\qquad + \|\boldsymbol{u}_{k,i}\|^4 \cdot \|\boldsymbol{n}_k(i)\|^4\Big)
\end{aligned} \tag{6.39}$$

where steps (a) and (b) use the inequality $\|x+y\|^4 \leq 8\|x\|^4 + 8\|y\|^4$, which can be obtained by applying Jensen's inequality to the convex function $\|\cdot\|^4$. Applying

the expectation operator conditioned on $\mathcal{F}_{i-1}$, we obtain

$$\begin{aligned}
&\mathbb{E}\left\{\|\boldsymbol{v}_{k,i}(\boldsymbol{w})\|^4|\mathcal{F}_{i-1}\right\} \\
&\quad\overset{(a)}{\leq} 1024\cdot\mathbb{E}\left\{\left\|R_{u,k}-\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|^4|\mathcal{F}_{i-1}\right\}\cdot\|\boldsymbol{w}\|^4+ \\
&\qquad 1024\cdot\mathbb{E}\left\{\left\|R_{u,k}-\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|^4|\mathcal{F}_{i-1}\right\}\cdot\|w^o\|^4+ \\
&\qquad 128\cdot\mathbb{E}\left\{\|\boldsymbol{u}_{k,i}\|^4|\mathcal{F}_{i-1}\right\}\cdot\mathbb{E}\left\{\|\boldsymbol{n}_k(i)\|^4|\mathcal{F}_{i-1}\right\} \\
&\quad\overset{(b)}{=} 1024\cdot\mathbb{E}\left\{\left\|R_{u,k}-\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|^4\right\}\cdot\|\boldsymbol{w}\|^4+ \\
&\qquad 1024\cdot\mathbb{E}\left\{\left\|R_{u,k}-\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|^4\right\}\cdot\|w^o\|^4+ \\
&\qquad 128\cdot\mathbb{E}\,\|\boldsymbol{u}_{k,i}\|^4\cdot\mathbb{E}\,\|\boldsymbol{n}_k(i)\|^4 \\
&\quad\triangleq \alpha_4\cdot\|\boldsymbol{w}\|^4+\sigma_{v4}^4
\end{aligned}\tag{6.40}$$

where step (a) uses the fact that $\boldsymbol{w}\in\mathcal{F}_{i-1}$ and is thus determined given $\mathcal{F}_{i-1}$, and step (b) uses the fact that $\boldsymbol{u}_{k,i}$ and $\boldsymbol{v}_{k,i}(i)$ are independent of $\mathcal{F}_{i-1}$. □

6.4 Performance of Multi-Agent Learning Strategy

In this section, we are interested in evaluating $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|_\Sigma^2$ as $i\to\infty$ for arbitrary positive semi-definite weighting matrices Σ. The main result is summarized in the following theorem.

Theorem 6.1 (Steady-state performance)**.** *For small step-sizes, the weighted mean-square-error of the distributed strategy* (6.1)–(6.3) *(which includes diffusion and consensus algorithms as special cases) is given by*

$$\lim_{i\to\infty}\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|_\Sigma^2=\mu_{\max}\cdot\mathrm{Tr}\left\{X(p^T\otimes I_M)\cdot\mathcal{R}_v\cdot(p\otimes I_M)\right\}+O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big)\tag{6.41}$$

where Σ is any positive semi-definite weighting matrix, and X is the unique positive semi-definite solution to the following Lyapunov equation:

$$H_c^T X + X H_c = \Sigma \tag{6.42}$$

where H_c was defined earlier in (6.12)*. The unique solution of* (6.42) *can be represented by the integral expression [73, p.769]:*

$$X = \int_0^\infty e^{-H_c^T t} \cdot \Sigma \cdot e^{-H_c t} dt \tag{6.43}$$

Moreover, if Σ is strictly positive-definite, then X is also strictly positive-definite.

Proof. The argument is nontrivial and involves several steps. The details are provided in Appendix 6.A. □

Example 6.3. (Distributed stochastic gradient-descent: General case) When stochastic gradients are used to define the update directions $\hat{\boldsymbol{s}}_{k,i}(\cdot)$ in (6.1)–(6.3), then we can simplify the mean-square-error expression (6.41) as follows. We first substitute $s_k(w) = \nabla_w J_k(w)$ into (6.12) to obtain

$$H_c = \sum_{k=1}^{N} p_k \nabla_w^2 J_k(w^o) \tag{6.44}$$

Now the matrix H_c is the weighted sum of the Hessian matrices of the individual costs $\{J_k(w)\}$ and is therefore symmetric. Then, the Lyapunov equation (6.42) becomes

$$H_c X + X H_c = \Sigma \tag{6.45}$$

We have simple solutions to (6.45) for the following two choices of Σ:

1. When $\Sigma = I_M$, we have $X = \frac{1}{2}H_c^{-1}$ and

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \frac{\mu_{\max}}{2} \cdot \mathrm{Tr}\left\{H_c^{-1}(p^T \otimes I_M)\cdot\mathcal{R}_v\cdot(p\otimes I_M)\right\} + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \tag{6.46}$$

2. When $\Sigma = \frac{1}{2}H_c$, we have $X = \frac{1}{4}I_M$ and

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2_{H_c} = \frac{\mu_{\max}}{4} \cdot \mathrm{Tr}\left\{(p^T \otimes I_M)\cdot\mathcal{R}_v\cdot(p\otimes I_M)\right\} + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \tag{6.47}$$

□

Example 6.4. (Distributed stochastic gradient descent: Uncorrelated noise) *In the special case that the gradient noises at the different agents are uncorrelated with each other, then $\mathcal{R}_v$ is block diagonal and we write it as*

$$\mathcal{R}_v = \mathrm{diag}\{R_{v,1},\ldots,R_{v,N}\} \tag{6.48}$$

where $R_{v,k}$ is the $M\times M$ covariance matrix of the gradient noise at agent k. Then, the MSE expression (6.46) *at each agent k can be written as*

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \frac{\mu_{\max}}{2} \cdot \mathrm{Tr}\left\{\left(\sum_{k=1}^N p_k \nabla_w^2 J_k(w^o)\right)^{-1} \cdot \left(\sum_{k=1}^N p_k^2 R_{v,k}\right)\right\} + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \tag{6.49}$$

and expression (6.47) *for the weighted MSE becomes*

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2_{H_c} = \frac{\mu_{\max}}{4} \cdot \mathrm{Tr}\left\{\sum_{k=1}^N p_k^2 R_{v,k}\right\} + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \tag{6.50}$$

□

6.5 Performance of Centralized Solution

We conclude from (6.41) that the weighted mean-square-error at each node k will be the same across all agents in the network for small step-sizes. This is an important "equalization" effect. Moreover, as we now verify, the performance level given by (6.41) is close to the performance of a centralized strategy that collects all the data from the agents and processes them using the following recursion:

$$\boldsymbol{w}_{\text{cent},i} = \boldsymbol{w}_{\text{cent},i-1} - \mu_{\max} \sum_{k=1}^{N} p_k \hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w}_{\text{cent},i-1}) \tag{6.51}$$

To establish this fact, we first note that the performance of the above centralized strategy can be analyzed in the same manner as the distributed strategy. Indeed, let $\check{\boldsymbol{w}}_{\text{cent},i} \triangleq \boldsymbol{w}_{\text{cent},i} - \bar{w}_{c,i}$ denote the discrepancy between the above centralized recursion and reference recursion (6.8). Then, we obtain from (6.8) and (6.51) that

$$\check{\boldsymbol{w}}_{\text{cent},i} = T_c(\boldsymbol{w}_{\text{cent},i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{v}_i(\boldsymbol{w}_{\text{cent},i-1}) \tag{6.52}$$

where the operator $T_c(w)$ is defined as the following mapping from $\mathbb{R}^M$ to $\mathbb{R}^M$:

$$T_c(w) \triangleq w - \mu_{\max} \sum_{k=1}^{N} p_k s_k(w) \tag{6.53}$$

Comparing (6.52) with expression (6.80) from Chapter 5, we note that these two recursions take similar forms except for an additionl perturbation term $\boldsymbol{z}_{i-1}$ in

(6.80) of Chapter 5. Therefore, following the same line of transient analysis as in Chapter 5 and steady-state analysis as in the proof of Theorem 6.1 stated earlier, we can conclude that, in the small step-size regime, the transient behavior of the centralized strategy (6.51) is close to the reference recursion (6.8), and the steady-state performance is again given by (6.41).

Theorem 6.2 (Centralized performance)**.** *Suppose the step-size parameter* $\mu_{\max}$ *in the centralized recursion* (6.51) *satisfies the following condition*

$$0 < \mu_{\max} < \frac{\lambda_L}{\|p\|^2 \cdot \left(\frac{\lambda_U^2}{2} + 2\alpha\right)} \tag{6.54}$$

Then, the MSE term $\mathbb{E}\|\tilde{\boldsymbol{w}}_{\text{cent},i}\|^2$ *converges at the rate of*

$$r = \left[\rho(I_M - \mu_{\max} H_c)\right]^2 + O\big((\mu_{\max}\epsilon)^{\frac{1}{2(M-1)}}\big) \tag{6.55}$$

where ϵ *is an arbitrarily small positive number. Furthermore, in the small step-size regime, the steady-state MSE of* (6.51) *is also given by* (6.41)

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{\text{cent},i}\|_{\Sigma}^2 = \mu_{\max} \cdot \text{Tr}\left\{X(p^T \otimes I_M) \cdot \mathcal{R}_v \cdot (p \otimes I_M)\right\} + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \tag{6.56}$$

□

6.6 Benefits of Coopeartion

In this section, we illustrate the implications of the main results of this work in the context of distributed learning and distributed optimization. Consider a network of N connected agents, where each agent k receives a stream of data

$\{\boldsymbol{x}_{k,i}\}$ arising from some underlying distribution. The networked multi-agent system would like to extract from the distributed data some useful information about the underlying process. To measure the quality of the inference task, an individual cost function $J_k(w)$ is associated with each agent k, where w denotes an $M \times 1$ parameter vector. The agents are generally interested in minimizing some aggregate cost function of the form (6.28):

$$J^{\text{glob}}(w) = \sum_{k=1}^{N} J_k(w) \tag{6.57}$$

Based on whether the individual costs $\{J_k(w)\}$ share a common minimizer or not, we can classify problems of the form (6.57) into two broad categories.

6.6.1 Category I: Distributed Learning

In this case, the data streams $\{\boldsymbol{x}_{k,i}\}$ are assumed to be generated by (possibly different) distributions that nevertheless depend on the same parameter vector $w^o \in \mathbb{R}^M$. The objective is then to estimate this common parameter w^o in a distributed manner. To do so, we first need to associate with each agent k a cost function $J_k(w)$ that measures how well some arbitrary parameter w approximates w^o. The cost $J_k(w)$ should be such that w^o is one of its minimizers. More formally, let $\mathcal{W}_k^o$ denote the set of vectors that minimize the selected $J_k(w)$, then it is expected that

$$w^o \in \mathcal{W}_k^o \triangleq \Big\{ w : \arg\min_{w} J_k(w) \Big\} \tag{6.58}$$

for $k = 1, \ldots, N$. Since $J^{\text{glob}}(w)$ is assumed to be strongly convex, then the intersection of the sets $\mathcal{W}_k^o$ should contain the single element w^o:

$$w^o \in \mathcal{W}^o = \bigcap_{k=1}^{N} \mathcal{W}_k^o \tag{6.59}$$

The main motivation for cooperation in this case is that the data collected at each agent k may not be sufficient to uniquely identify w^o since w^o is not necessarily the unique element in $\mathcal{W}_k^o$; this happens, for example, when the individual costs $J_k(w)$ are not *strictly* convex. However, once the individual costs are aggregated into (6.57) and the aggregate function is strongly convex, then w^o is the unique element in $\mathcal{W}^o$. In this way, the cooperative minimization of $J^{\text{glob}}(w)$ allows the agents to estimate w^o.

6.6.1.1 Working under Partial Observation

Under the scenario described by (6.59), the solution of (6.5) agrees with the unique minimizer w^o for $J^{\text{glob}}(w)$ given by (6.57) regardless of the $\{p_k\}$ and, therefore, regardless of the combination policy A. Therefore, the results from Sec. 5.4 of Chapter 5 show that the iterate $\boldsymbol{w}_{k,i}$ at each agent k converges to this unique w^o at a centralized rate and the results from Sec. 6.4 of this chapter show that this iterate achieves the centralized steady-state MSE performance. Note that Assumption 6.4 can be satisfied without requiring each $J_k(w)$ to be strongly convex. Instead, we only require $J^{\text{glob}}(w)$ to be strongly convex. In other words, we do not need each agent to have complete information about w^o; we only need the network to have enough information to determine w^o uniquely. Although the individual agents in this case have partial information about w^o, the distributed strategies (6.1)–(6.3) enable them to attain the same performance

level as a centralized solution. The following example illustrates the idea in the context of distributed least-mean-squares estimation over networks.

Example 6.5. Consider Example 6.1 again. When the covariance matrix $R_{u,k} \triangleq \mathbb{E}[\boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i}]$ is rank deficient, then $J_k(w)$ in (6.29) would not be strongly convex and there would be infinitely many minimizers to $J_k(w)$. In this case, the information provided to agent k via (6.27) is not sufficient to determine w^o uniquely. However, if the global cost function is strongly convex, which can be verified to be equivalent to requiring:

$$\sum_{k=1}^{N} p_k R_{u,k} > \lambda_L I_M > 0 \tag{6.60}$$

then the information collected over the entire network is rich enough to learn the unique w^o. As long as (6.60) holds for one set of positive $\{p_k\}$, it will hold for all other $\{p_k\}$. A "network observability" condition similar to (6.60) was used in [75] to characterize the sufficiency of information over the network in the context of distributed estimation over linear models albeit with diminishing step-sizes. □

6.6.1.2 Optimizing the MSE Performance

Since the distributed strategies (6.1)–(6.3) converge to the minimizer w^o of (6.57) for any set of $\{p_k\}$, we can then consider selecting the $\{p_k\}$ to optimize the MSE performance. Consider the case where $H_k \equiv H$ and $\mu_k \equiv \mu$ and assume the gradient noises $\boldsymbol{v}_{k,i}(w)$ are asymptotically uncorrelated across the agents so that $\mathcal{R}_v$ from (6.25) is block diagonal with entries denoted by:

$$\mathcal{R}_v = \text{diag}\{R_{v,1}, \ldots, R_{v,N}\} \tag{6.61}$$

Then, we have $\beta_k = 1$, $p_k = \theta_k$ and

$$H_c = H = \nabla_w^2 J_1(w^o) = \cdots = \nabla_w^2 J_N(w^o) \tag{6.62}$$

in which case expression (6.46) becomes

$$\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 = \frac{\mu_{\max}}{2} \cdot \sum_{k=1}^{N} \theta_k^2 \mathrm{Tr}\left(H^{-1} R_{v,k}\right) + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \tag{6.63}$$

The optimal positive coefficients $\{\theta_k\}$ that minimize (6.63) subject to $\sum_{k=1}^{N} \theta_k = 1$ are given by

$$\theta_k^o = \frac{[\mathrm{Tr}(H^{-1}R_{v,k})]^{-1}}{\sum_{\ell=1}^{N}[\mathrm{Tr}(H^{-1}R_{v,\ell})]^{-1}}, \quad k = 1, \ldots, N \tag{6.64}$$

and, substituting into (6.63), the optimal MSE is then given by

$$\mathrm{MSE}^{\mathrm{opt}} = \frac{\mu_{\max}}{2} \cdot \left[\sum_{\ell=1}^{N} \frac{1}{\mathrm{Tr}(H^{-1}R_{v,\ell})}\right]^{-1} + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \tag{6.65}$$

The optimal Perron-eigenvector $\theta^o = \mathrm{col}\{\theta_1^o, \ldots, \theta_N^o\}$ can be implemented by selecting the combination policy A as the following Hasting's rule [18, 62, 146]:

$$a_{lk}^o = \begin{cases} \dfrac{(\theta_k^o)^{-1}}{\max\left\{|\mathcal{N}_k| \cdot (\theta_k^o)^{-1}, |\mathcal{N}_\ell| \cdot (\theta_\ell^o)^{-1}\right\}}, & \ell \in \mathcal{N}_k \backslash \{k\} \\ \\ 1 - \displaystyle\sum_{m \in \mathcal{N}_k \backslash \{k\}} a_{mk}^o, & \ell = k \end{cases} \tag{6.66}$$

where $|\mathcal{N}_k|$ denotes the cardinality of the set $\mathcal{N}_k$. The above combination matrix can be constructed in a decentralized manner, where each node only requires

information from its own neighbors. In practice, the noise covariance matrices $\{R_{v,\ell}\}$ need to be estimated from the local data and an adaptive estimation scheme is proposed in [146] to address this issue.

6.6.1.3 Matching Performance across Topologies

Note that the steady-state mean-square error depends on the vector p, which is determined by the Perron eigenvector θ of the matrix A. The above result implies that, as long as the network is strongly connected, i.e., Assumption 6.1 holds, a left-stochastic matrix A can always be constructed to have any desired Perron eigenvector θ with positive entries according to (6.66). Now, starting from any collection of N agents, there exists a finite number of topologies that can link these agents together. For each possible topology, there are infinitely many combination policies that can be used to train the network. One important conclusion that follows from the above results is that regardless of the topology, there always exists a choice for A such that the performance of all topologies are identical to each other to first-order in $\mu_{\max}$. In other words, no matter how the agents are connected to each other, there is always a way to select the combination weights such that the performance of the network is invariant to the topology. This also means that, for any connected topology, there is always a way to select the combination weights such that the performance of the network matches that of the centralized solution.

Example 6.6. We illustrate the result using the diffusion least-mean-square estimation context discussed earlier in Examples 6.1–6.2. Consider a network of 30 agents ($N = 30$), where each agent has access to a stream of data samples $\{\boldsymbol{u}_{k,i}, \boldsymbol{d}_k(i)\}$ that are generated by the linear model (6.27). As assumed in Example 6.1, the $1 \times M$ regressors $\{\boldsymbol{u}_{k,i}\}$ are zero mean and independent over time and

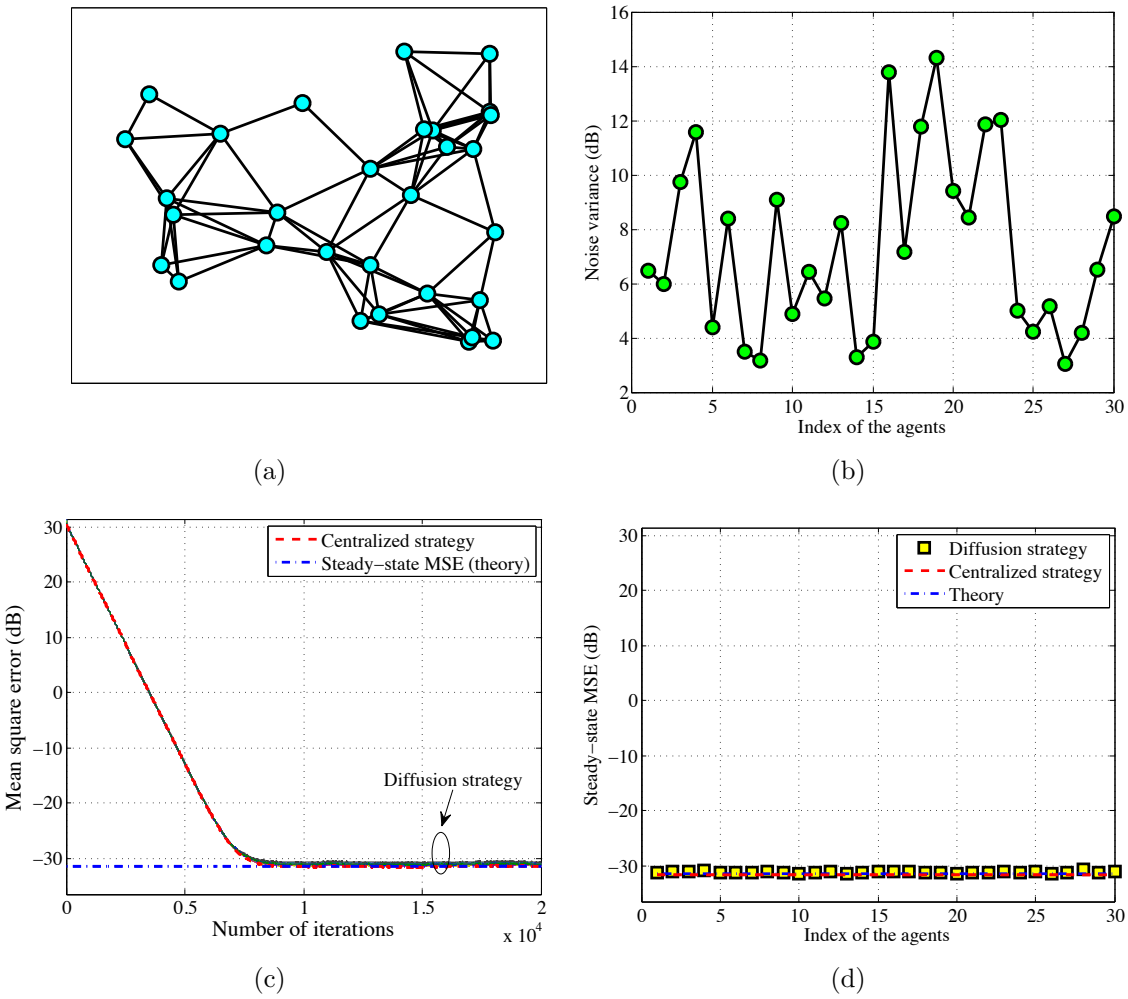


Figure 6.1: Comparing the performance of a 30-node diffusion LMS network with that of the centralized strategy (6.51), where $M = 10$, $\mu_k = 0.0005$ for all agents, and Hasting's rule (6.66) is used as the combination policy. The result is obtained by averaging over 1000 Monte Carlo experiments. (a) A randomly generated topology. (b) The noise profile across the network. (c) The learning curves for different agents in the diffusion LMS network, the centralized strategy, and the theoretical steady-state MSE. (d) The steady-state MSE of diffusion LMS, centralized strategy, and the theoretical value.

space with covariance matrix $R_{u,k}$, and the noise sequence $\{\boldsymbol{n}_l(j)\}$ is also zero mean, white, with variance $\sigma_{n,l}^2$, and independent of the regressors $\{\boldsymbol{u}_{k,i}\}$ for all l, k, i, j. In the simulation here, we consider the case where $M = 2$, $R_{u,k} = I_M$. In diffusion LMS estimation, each agent k uses (6.29) as its cost function $J_k(w)$ and (6.31) as the stochastic gradient vector $\hat{\boldsymbol{s}}_{k,i}(\cdot)$. Therefore, each agent k adopts the following recursion to adaptively estimate the model parameter w^o, which is the minimizer of the global cost function (6.28):

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i} + 2\mu_k \boldsymbol{u}_{k,i}^T [\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}] \tag{6.67}$$

$$\boldsymbol{w}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{lk} \boldsymbol{\psi}_{l,i} \tag{6.68}$$

We randomly generate a topology as shown in Fig. 6.1 (a) and noise variance profile across agents as shown in Fig. 6.1 (b). We choose $\mu_k \equiv \mu = 0.0005$ to be the step-size for all agents and Hasting's rule (6.66) as the combination policy. In the simulation, we assume the noise variances are known to the agents. Alternatively, they can also be estimated in an adaptive manner using approaches proposed in [146]. In Figs. 6.1 (c)–(d), we illustrate the learning curves and steady-state MSE of all agents, respectively, and compare them to the theoretical value and to the following centralized LMS strategy:

$$\boldsymbol{w}_{\text{cent},i} = \boldsymbol{w}_{\text{cent},i-1} + 2\mu \sum_{k=1}^{N} p_k \cdot \boldsymbol{u}_{k,i}^T [\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{\text{cent},i-1}] \tag{6.69}$$

where $p_k = \theta_k^o$ is given by (6.64). The results are obtained by averaging over 1000 Monte Carlo experiments. We observe from Fig. 6.1 (c) that the learning curves of all agents are close to each other and to the centralized strategy. Furthermore, Fig. 6.1 illustrates the equalization effect over the network; each agent in the network achieves almost the same steady-state MSE that is close to the

centralized strategy although the noise variances in the data are different across the agents. □

6.6.2 Category II: Distributed Optimization

In this case, we include situations where the individual costs $J_k(w)$ do not have a common minimizer, i.e., $\mathcal{W}^o = \emptyset$. The optimization problem should then be viewed as one of solving a multi-objective minimization problem

$$\min_{w} \{J_1(w), \ldots, J_N(w)\} \tag{6.70}$$

where $J_k(w)$ is an individual convex cost associated with each agent k. A vector w^o is said to be a Pareto optimal solution to (6.70) if there does not exist another vector w that is able to improve (i.e., reduce) any individual cost without degrading (increasing) some of the other costs. Pareto optimal solutions are not unique. The question we would like to address now is the following. Given individual costs $\{J_k(w)\}$ and a combination policy A, what is the limit point of the distributed strategies (6.1)–(6.3)? From Theorem 5.4 in Chapter 5, the distributed strategy (6.1)–(6.3) converges to the limit point w^o defined as the unique solution to (6.5). Substituting $s_k(w) = \nabla_w J_k(w)$ into (6.5), we obtain

$$\sum_{k=1}^{N} p_k \nabla_w J_k(w^o) = 0 \tag{6.71}$$

In other words, w^o is the minimizer of the following global cost function:

$$J^{\text{glob}}(w) = \sum_{k=1}^{N} p_k J_k(w) \tag{6.72}$$

It is shown in [20, pp.178–180] that the minimizer of (6.72) is a Pareto-optimal solution for the multi-objective optimization problem (6.70). And different choices for the vector p lead to different Pareto-optimal points on the tradeoff curve. Therefore, in order to converge to a certain Pareto-optimal point corresponding to a given set of positive coefficients $\{p_k\}$, we need to design a left-stochastic matrix A so that its Perron eigenvector leads to the $\{p_k\}$. This can be achieved by constructing A according to the following Hasting's rule:

$$a_{lk} = \begin{cases} \dfrac{p_k^{-1}}{\max\left\{|\mathcal{N}_k| \cdot p_k^{-1}, |\mathcal{N}_l| \cdot p_l^{-1}\right\}}, & l \in \mathcal{N}_k \backslash \{k\} \\ 1 - \displaystyle\sum_{m \in \mathcal{N}_k \backslash \{k\}} a_{mk}, & l = k \end{cases} \tag{6.73}$$

6.7 Conclusion

Along with Chapter 5, this chapter examined in some detail the mean-square performance, convergence, and stability of distributed strategies for adaptation and learning over graphs under *constant* step-size update rules. Keeping the step-size fixed allows the network to track drifts in the underlying data models, their statistical distributions, and even drifts in the utility functions. Earlier work [147] has shown that constant adaptation regimes endow networks with tracking abilities and derived results that quantify how the performance of adaptive networks is affected by the level of non-stationarity in the data. Similar conclusions extend to the general scenario studied in Chapter 5 and this chapter, which is the reason why step-sizes have been set to a constant value throughout our treatment. When this is done, the dynamics of the learning process is enriched in a nontrivial manner. This is because the effect of gradient noise does not die out anymore with time (in comparison, when diminishing step-sizes are used, gradi-

ent noise is annihilated by the decaying step-sizes). And since agents are coupled through their interactions over the network, it follows that their gradient noises will continually influence the performance of their neighbors. As a result, the network mean-square performance does not tend to zero anymore. Instead, it approaches a steady-state level. One of the main objectives of this chapter has been to quantify this level and to show explicitly how its value is affected by three parameters: the network topology, the gradient noise, and the data characteristics. As the analysis and the detailed derivations in the appendices of the current manuscript show, this is a formidable task to pursue due to the coupling among the agents. Nevertheless, under certain conditions that are generally weaker than similar conditions used in related contexts in the literature, we were able to derive accurate expressions for the network MSE performance and its convergence rate. For example, the MSE expression we derived is accurate in the first order term of $\mu_{\max}$. Once an MSE expression has been derived, we were then able to optimize it over the network topology (for the important case of uniform Hessian matrices across the network, as is common for example in machine learning [129] and mean-square-error estimation problems [116]). We were able to show that arbitrary connected topologies for the same set of agents can always be made to perform similarly. We were also able to show that arbitrary connected topologies for the same set of agents can be made to match the performance of a fully connected network. These are useful insights and they follow from the analytical results derived in this work.

6.A Proof of Theorem 6.1

The argument involves several steps, labeled steps 6.A.1 through 6.A.5, and relies also on intermediate results that are proven in this appendix. We start with step

6.A.1.

6.A.1 Relating the weighted MSE to the steady-state error covariance matrix Π_∞

Let $\Pi_i \triangleq \mathbb{E}\left\{\tilde{\boldsymbol{w}}_i\tilde{\boldsymbol{w}}_i^T\right\}$ denote the error covariance matrix of the global error vector

$$\tilde{\boldsymbol{w}}_i \triangleq \mathrm{col}\{\tilde{\boldsymbol{w}}_{1,i},\ldots,\tilde{\boldsymbol{w}}_{N,i}\} \tag{6.74}$$

where $\tilde{\boldsymbol{w}}_{k,i} \triangleq w^o - \tilde{\boldsymbol{w}}_{k,i}$. Note that if we are able to evaluate Π_i as $i \to \infty$, i.e., Π_∞, then we can obtain the steaty-state weighted mean-square-error for any individual agent via the following relation:

$$\begin{aligned}
\lim_{i\to\infty}\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|_\Sigma^2 &= \lim_{i\to\infty}\mathbb{E}\Big\{\mathrm{col}\{\tilde{\boldsymbol{w}}_{1,i},\ldots,\tilde{\boldsymbol{w}}_{N,i}\}^T \\
&\qquad \cdot \mathrm{diag}\{0,\ldots,\Sigma,\ldots,0\} \\
&\qquad \cdot \mathrm{col}\{\tilde{\boldsymbol{w}}_{1,i},\ldots,\tilde{\boldsymbol{w}}_{N,i}\}\Big\} \\
&= \lim_{i\to\infty}\mathbb{E}\left\{\tilde{\boldsymbol{w}}_i^T(E_{kk}\otimes\Sigma)\tilde{\boldsymbol{w}}_i\right\} \\
&= \lim_{i\to\infty}\mathbb{E}\left(\mathrm{Tr}\left\{\tilde{\boldsymbol{w}}_i\tilde{\boldsymbol{w}}_i^T(E_{kk}\otimes\Sigma)\right\}\right) \\
&= \lim_{i\to\infty}\mathrm{Tr}\left\{\mathbb{E}\left[\tilde{\boldsymbol{w}}_i\tilde{\boldsymbol{w}}_i^T\right](E_{kk}\otimes\Sigma)\right\} \\
&= \mathrm{Tr}\{\Pi_\infty(E_{kk}\otimes\Sigma)\}
\end{aligned} \tag{6.75}$$

where E_{kk} is an $M\times M$ matrix with (k,k)-entry equal to one and all other entries equal to zero. We could proceed with the analysis by deriving a recursion of $\tilde{\boldsymbol{w}}_i$ from (6.1)–(6.3) and examining the corresponding error covariance matrix, Π_i. However, we will take an alternative approach here by calling upon the following

decomposition of the error quantity $\tilde{\boldsymbol{w}}_{k,i}$ from Chapter 5 (see Eq. (6.76) therein):

$$\tilde{\boldsymbol{w}}_{k,i} = \tilde{w}_{c,i} - \check{\boldsymbol{w}}_{c,i} - (u_{L,k} \otimes I_M)\boldsymbol{w}_{e,i} \tag{6.76}$$

where $\tilde{w}_{c,i} \triangleq w^o - w_{c,i}$ denotes the error of the reference recursion (6.8) relative to w^o, the vectors $\check{\boldsymbol{w}}_{c,i}$ and $\boldsymbol{w}_{e,i}$ are the two transformed quantities introduced in Eqs. (5.70) and (5.60) in Chapter 5, and $u_{L,k}$ is the kth row of the matrix U_L which is a sub-matrix of the transform matrix introduced in Eq. (5.54) in Chapter 5. In particular, $\check{\boldsymbol{w}}_{c,i}$ represents the error of the centroid of the iterates $\{\boldsymbol{w}_{k,i}\}$ relative to the reference recursion:

$$\check{\boldsymbol{w}}_{c,i} \triangleq \boldsymbol{w}_{c,i} - \bar{w}_{c,i} \tag{6.77}$$

where the centroid $\boldsymbol{w}_{c,i}$ is defined as

$$\boldsymbol{w}_{c,i} \triangleq \sum_{k=1}^{N} \theta_k \boldsymbol{w}_{k,i} \tag{6.78}$$

and $(u_{L,k} \otimes I_M)\boldsymbol{w}_{e,i}$ represents the error of the iterate $\boldsymbol{w}_{k,i}$ at agent k relative to the centroid $\boldsymbol{w}_{c,i}$. The details and derivation of the decomposition (6.76) appear in Sec. 5.5.1 of Chapter 5. Relation (6.76) can also be written in the following equivalent global form:

$$\tilde{\boldsymbol{w}}_i = \mathbb{1} \otimes \tilde{w}_{c,i} - \mathbb{1} \otimes \check{\boldsymbol{w}}_{c,i} - (U_L \otimes I_M)\boldsymbol{w}_{e,i} \tag{6.79}$$

The major motivation to use (6.79) in our steady-state analysis is that the convergence results and non-asymptotic MSE bounds for each term in (6.79) will reveal that some quantities will either disappear or become higher order terms in steady-state for small step-sizes. In particular, we are going to show that the

mean-square-error of $\tilde{\boldsymbol{w}}_i$ is dominated by the mean-square-error of $\check{\boldsymbol{w}}_{c,i}$. Therefore, it will suffice to examine the mean-square-error of $\check{\boldsymbol{w}}_{c,i}$. We derived in expression (5.97) from Chapter 5 the following relation for $\check{\boldsymbol{w}}_{c,i}$:

$$\check{\boldsymbol{w}}_{c,i} = T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\,[\boldsymbol{z}_{i-1} + \boldsymbol{v}_i] \tag{6.80}$$

where

$$T_c(x) \triangleq x - \mu_{\max} \sum_{k=1}^{N} p_k s_k(x) \tag{6.81}$$

$$\boldsymbol{v}_i \triangleq \hat{\boldsymbol{s}}_i\,(\boldsymbol{\phi}_{i-1}) - s\,(\boldsymbol{\phi}_{i-1}) \tag{6.82}$$

$$\boldsymbol{z}_{i-1} \triangleq s\,(\boldsymbol{\phi}_{i-1}) - s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1}) \tag{6.83}$$

The two perturbation terms $\boldsymbol{v}_i(\boldsymbol{\phi}_{i-1})$ and $\boldsymbol{z}_{i-1}$ were further shown to satisfy the following bounds in Appendix 5.H in Chapter 5.

$$P[\boldsymbol{z}_{i-1}] \preceq \lambda_U^2 \cdot \left\|\bar{P}_1[A_1^T U_L]\right\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \cdot P[\boldsymbol{w}_{e,i-1}] \tag{6.84}$$

$$P[s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1})] \preceq 3\lambda_U^2 \cdot P[\check{\boldsymbol{w}}_{c,i-1}] \cdot \mathbb{1} + 3\lambda_U^2 \|\tilde{w}_{c,0}\|^2 \cdot \mathbb{1} + 3g^o \tag{6.85}$$

$$\begin{aligned} \mathbb{E}\{P[\boldsymbol{v}_i]|\mathcal{F}_{i-1}\} \preceq\ & 4\alpha \cdot \mathbb{1} \cdot P[\check{\boldsymbol{w}}_{c,i-1}] \\ & + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \\ & + \left[4\alpha \cdot (\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2\right] \cdot \mathbb{1} \end{aligned} \tag{6.86}$$

$$\begin{aligned} \mathbb{E}P[\boldsymbol{v}_i] \preceq\ & 4\alpha \cdot \mathbb{1} \cdot \mathbb{E}P[\check{\boldsymbol{w}}_{c,i-1}] \\ & + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\ & + \left[4\alpha \cdot (\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2\right] \cdot \mathbb{1} \end{aligned} \tag{6.87}$$

where $P[\check{\boldsymbol{w}}_{c,i-1}] = \|\check{\boldsymbol{w}}_{c,i-1}\|^2$, and $g^o \triangleq P[s(\mathbb{1} \otimes w^o)]$. We further showed in Eqs. (5.146) and (5.147) from Chapter 5 that

$$\limsup_{i\to\infty} \mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2 \leq O(\mu_{\max}) \tag{6.88}$$

$$\limsup_{i\to\infty} \mathbb{E}\|\boldsymbol{w}_{e,i}\|^2 \leq O(\mu_{\max}^2) \tag{6.89}$$

6.A.2 Approximation of Π_∞ by $\mathbb{1}\mathbb{1}^T \otimes \check{\Pi}_{c,\infty}$

In order to evaluate Π_∞, which is needed for (6.75), we first establish the following observation using (6.79): in steady-state, the error covariance matrix of $\tilde{\boldsymbol{w}}_i$ (i.e., Π_∞) is equal to the error covaraince matrix of the component $\mathbb{1} \otimes \check{\boldsymbol{w}}_{c,i}$ to the first order of $\mu_{\max}$. Indeed, let $\check{\Pi}_{c,i}$ denote the covariance matrix of $\check{\boldsymbol{w}}_{c,i}$, i.e., $\check{\Pi}_{c,i} \triangleq \mathbb{E}\{\check{\boldsymbol{w}}_{c,i}\check{\boldsymbol{w}}_{c,i}^T\}$. By (6.79), we have

$$\begin{aligned}
\Pi_i &= \mathbb{E}\left\{\tilde{\boldsymbol{w}}_i\tilde{\boldsymbol{w}}_i^T\right\} \\
&= \mathbb{1}\mathbb{1}^T \otimes [\tilde{w}_{c,i}\tilde{w}_{c,i}^T] + \mathbb{1}\mathbb{1}^T \otimes \check{\Pi}_{c,i} \\
&\quad + \mathbb{E}\left\{[(U_L \otimes I_M)\boldsymbol{w}_{e,i}][(U_L \otimes I_M)\boldsymbol{w}_{e,i}]^T\right\} \\
&\quad - (\mathbb{1} \otimes \tilde{w}_{c,i})\big(\mathbb{1} \otimes \mathbb{E}\check{\boldsymbol{w}}_{c,i} + (U_L \otimes I_M)\mathbb{E}\boldsymbol{w}_{e,i}\big)^T \\
&\quad - \big(\mathbb{1} \otimes \mathbb{E}\check{\boldsymbol{w}}_{c,i} + (U_L \otimes I_M)\mathbb{E}\boldsymbol{w}_{e,i}\big)(\mathbb{1} \otimes \tilde{w}_{c,i})^T \\
&\quad + \mathbb{E}\left\{(\mathbb{1} \otimes \check{\boldsymbol{w}}_{c,i})[(U_L \otimes I_M)\boldsymbol{w}_{e,i}]^T\right\} \\
&\quad + \mathbb{E}\left\{[(U_L \otimes I_M)\boldsymbol{w}_{e,i}](\mathbb{1} \otimes \check{\boldsymbol{w}}_{c,i})^T\right\}
\end{aligned} \tag{6.90}$$

so that

$$\begin{aligned}
\left\|\Pi_i - \mathbb{1}\mathbb{1}^T \otimes \check{\Pi}_{c,i}\right\| &\overset{(a)}{\leq} \left\|\mathbb{1}\mathbb{1}^T \otimes [\tilde{w}_{c,i}\tilde{w}_{c,i}^T]\right\| \\
&\quad + \left\|\mathbb{E}\left\{[(U_L \otimes I_M)\boldsymbol{w}_{e,i}][(U_L \otimes I_M)\boldsymbol{w}_{e,i}]^T\right\}\right\|
\end{aligned}$$

$$
\begin{aligned}
&\quad + 2\left\|\mathbb{1}\otimes\tilde{w}_{c,i}\right\|\cdot\left\|\mathbb{1}\otimes\mathbb{E}\check{\boldsymbol{w}}_{c,i}+(U_L\otimes I_M)\mathbb{E}\boldsymbol{w}_{e,i}\right\| \\
&\quad + 2\left\|\mathbb{E}\left\{(\mathbb{1}\otimes\check{\boldsymbol{w}}_{c,i})[(U_L\otimes I_M)\boldsymbol{w}_{e,i}]^T\right\}\right\| \\
&\overset{(b)}{\leq} \left\|\mathbb{1}\mathbb{1}^T\otimes[\tilde{w}_{c,i}\tilde{w}_{c,i}^T]\right\| + \mathbb{E}\left\|(U_L\otimes I_M)\boldsymbol{w}_{e,i}]\right\|^2 \\
&\quad + 2\left\|\mathbb{1}\otimes\tilde{w}_{c,i}\right\|\cdot\left\|\mathbb{1}\otimes\mathbb{E}\check{\boldsymbol{w}}_{c,i}+(U_L\otimes I_M)\mathbb{E}\boldsymbol{w}_{e,i}\right\| \\
&\quad + 2\left\|\mathbb{E}\left\{(\mathbb{1}\otimes\check{\boldsymbol{w}}_{c,i})[(U_L\otimes I_M)\boldsymbol{w}_{e,i}]^T\right\}\right\| \qquad (6.91)
\end{aligned}
$$

where step (a) uses triangular inequality, and step (b) applies Jensen's inequality $\|\mathbb{E}[\cdot]\| \leq \mathbb{E}\|\cdot\|$ to the convex function $\|\cdot\|$ and the inequality $\|xy^T\| \leq \|x\|\cdot\|y\|$. Taking lim sup of both sides as $i\to\infty$, we obtain

$$
\begin{aligned}
&\limsup_{i\to\infty}\left\|\Pi_i - \mathbb{1}\mathbb{1}^T\otimes\check{\Pi}_{c,i}\right\| \\
&\leq \limsup_{i\to\infty}\mathbb{E}\left\|(U_L\otimes I_M)\boldsymbol{w}_{e,i}]\right\|^2 \\
&\quad + \limsup_{i\to\infty}\left\{2\left\|\mathbb{E}\left\{(\mathbb{1}\otimes\check{\boldsymbol{w}}_{c,i})[(U_L\otimes I_M)\boldsymbol{w}_{e,i}]^T\right\}\right\|\right\} \qquad (6.92)
\end{aligned}
$$

since $\tilde{w}_{c,i}\to 0$ as $i\to\infty$ according to Theorem 5.2 in Chapter 5. We now bound the two terms on the right-hand side of (6.92) using (6.88)–(6.89) and show that they are higher order terms of $\mu_{\max}$. By (6.89), the first term on the right-hand side of (6.92) is $O(\mu_{\max}^2)$ because

$$
\begin{aligned}
&\limsup_{i\to\infty}\mathbb{E}\left\|(U_L\otimes I_M)\boldsymbol{w}_{e,i}]\right\|^2 \\
&\quad \leq \limsup_{i\to\infty}\left\|U_L\otimes I_M\right\|^2\cdot\mathbb{E}\|\boldsymbol{w}_{e,i}\|^2 \leq O(\mu_{\max}^2) \qquad (6.93)
\end{aligned}
$$

Moreover, for any random variables $\boldsymbol{x}$ and $\boldsymbol{y}$, we have $|\mathbb{E}\{\boldsymbol{x}\boldsymbol{y}\}|^2 \leq \mathbb{E}\{\boldsymbol{x}^2\}\cdot\mathbb{E}\{\boldsymbol{y}^2\}$. Applying this result to the last term in (6.92) we have

$$
\left\|\mathbb{E}\left[(\mathbb{1}\otimes\check{\boldsymbol{w}}_{c,i})[(U_L\otimes I_M)\boldsymbol{w}_{e,i}]^T\right]\right\|
$$

$$\leq \sqrt{\mathbb{E}\|\mathbb{1}\otimes\check{\boldsymbol{w}}_{c,i}\|^2\cdot\mathbb{E}\|(U_L\otimes I_M)\boldsymbol{w}_{e,i}\|^2} \tag{6.94}$$

Using (6.88) and (6.89), we conclude that

$$\limsup_{i\to\infty}\left\|\mathbb{E}\left[(\mathbb{1}\otimes\check{\boldsymbol{w}}_{c,i})[(U_L\otimes I_M)\boldsymbol{w}_{e,i}]^T\right]\right\| \leq O(\mu_{\max}^{3/2}) \tag{6.95}$$

Therefore, substituting (6.93) and (6.95) into (6.92), we conclude that

$$\limsup_{i\to\infty}\left\|\Pi_i-\mathbb{1}\mathbb{1}^T\otimes\check{\Pi}_{c,i}\right\| \leq O(\mu_{\max}^{3/2}) \tag{6.96}$$

so that, as claimed earlier, in steady-state ($i\to\infty$):

$$\Pi_\infty = \mathbb{1}\mathbb{1}^T\otimes\check{\Pi}_{c,\infty}+O(\mu_{\max}^{3/2}) \tag{6.97}$$

6.A.3 Approximation of $\check{\Pi}_{c,\infty}$ by $\check{\Pi}_{a,\infty}$

Now we evaluate the expression for $\check{\Pi}_{c,\infty}$. To do this, we rewrite expressions (6.80)–(6.83) for $\check{\boldsymbol{w}}_{c,i}$ as

$$\begin{aligned}\check{\boldsymbol{w}}_{c,i} &= \check{\boldsymbol{w}}_{c,i-1}-\mu_{\max}\sum_{k=1}^{N}p_k\left[s_k(\boldsymbol{w}_{c,i-1})-s_k(\bar{w}_{c,i-1})\right]-\mu_{\max}\cdot(p^T\otimes I_M)\left[\boldsymbol{z}_{i-1}+\boldsymbol{v}_i\right]\\ &= \left[I_M-\mu_{\max}H_c\right]\check{\boldsymbol{w}}_{c,i-1}-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{v}_i-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\\ &\quad-\mu_{\max}\big(s_c(\boldsymbol{w}_{c,i-1})-s_c(\bar{w}_{c,i-1})-H_c\check{\boldsymbol{w}}_{c,i-1}\big)\end{aligned} \tag{6.98}$$

where

$$H_c \triangleq \sum_{k=1}^{N}p_k\nabla_{w^T}s_k(w^o) \tag{6.99}$$

$$s_c(w) \triangleq \sum_{k=1}^{N} p_k s_k(w) \tag{6.100}$$

Next, we show that the mean-square-error between $\check{\boldsymbol{w}}_{c,i}$ generated by (6.98) and the $\check{\boldsymbol{w}}_{a,i}$ generated by the following auxiliary recursion is small for small step-sizes:

$$\check{\boldsymbol{w}}_{a,i} = [I_M - \mu_{\max} H_c]\, \check{\boldsymbol{w}}_{a,i-1} - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{v}_i \tag{6.101}$$

Indeed, subtracting (6.101) from (6.98) leads to

$$\begin{aligned}
\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i} = {} & [I_M - \mu_{\max} H_c]\,(\check{\boldsymbol{w}}_{c,i-1} - \check{\boldsymbol{w}}_{a,i-1}) \\
& - \mu_{\max}\big(s_c(\boldsymbol{w}_{c,i-1}) - s_c(\bar{w}_{c,i-1}) - H_c \check{\boldsymbol{w}}_{c,i-1}\big) \\
& - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}
\end{aligned} \tag{6.102}$$

We recall the definition of the scalar factor γ_c from Eq. (5.120) in Chapter 5:

$$\gamma_c \triangleq 1 - \mu_{\max}\lambda_L + \frac{1}{2}\mu_{\max}^2 \|p\|_1^2 \lambda_U^2 \tag{6.103}$$

Now evaluating the squared Euclidean norm of both sides of (6.102), we get

$$\begin{aligned}
& \|\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i}\|^2 \\
& = \Big\|\gamma_c \cdot \frac{1}{\gamma_c}\,[I_M - \mu_{\max} H_c]\,(\check{\boldsymbol{w}}_{c,i-1} - \check{\boldsymbol{w}}_{a,i-1}) \\
& \quad + \frac{1-\gamma_c}{2} \cdot \frac{-2\mu_{\max}}{1-\gamma_c} \cdot \big(s_c(\boldsymbol{w}_{c,i-1}) - s_c(\bar{w}_{c,i-1}) - H_c \check{\boldsymbol{w}}_{c,i-1}\big) \\
& \quad + \frac{1-\gamma_c}{2} \cdot \frac{-2\mu_{\max}}{1-\gamma_c} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\Big\|^2 \\
& \overset{(a)}{\leq} \gamma_c \cdot \Big\|\frac{1}{\gamma_c}\,[I_M - \mu_{\max} H_c]\,(\check{\boldsymbol{w}}_{c,i-1} - \check{\boldsymbol{w}}_{a,i-1})\Big\|^2 \\
& \quad + \frac{1-\gamma_c}{2} \cdot \Big\|\frac{-2\mu_{\max}}{1-\gamma_c} \cdot \big(s_c(\boldsymbol{w}_{c,i-1}) - s_c(\bar{w}_{c,i-1}) - H_c \check{\boldsymbol{w}}_{c,i-1}\big)\Big\|^2
\end{aligned}$$

$$
\begin{aligned}
&+\frac{1-\gamma_c}{2}\cdot\Big\|\frac{-2\mu_{\max}}{1-\gamma_c}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\Big\|^2\\
&\le\frac{1}{\gamma_c}\cdot\|I_M-\mu_{\max}H_c\|^2\cdot\|\check{\boldsymbol{w}}_{c,i-1}-\check{\boldsymbol{w}}_{a,i-1}\|^2\\
&\quad+\frac{2\mu_{\max}^2}{1-\gamma_c}\cdot\Big\|s_c(\boldsymbol{w}_{c,i-1})-s_c(\bar{w}_{c,i-1})-H_c\check{\boldsymbol{w}}_{c,i-1}\Big\|^2\\
&\quad+\frac{2\mu_{\max}^2}{1-\gamma_c}\cdot\|(p^T\otimes I_M)\|^2\cdot\|\boldsymbol{z}_{i-1}\|^2\\
&\overset{(b)}{=}\frac{1}{\gamma_c}\cdot\|B_c\|^2\cdot\|\check{\boldsymbol{w}}_{c,i-1}-\check{\boldsymbol{w}}_{a,i-1}\|^2\\
&\quad+\frac{2\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\Big\|s_c(\boldsymbol{w}_{c,i-1})-s_c(\bar{w}_{c,i-1})-H_c\check{\boldsymbol{w}}_{c,i-1}\Big\|^2\\
&\quad+\frac{2\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|(p^T\otimes I_M)\|^2\cdot\mathbb{1}^TP[\boldsymbol{z}_{i-1}]
\end{aligned}
\tag{6.104}
$$

where in step (a) we used the convexity of the squared norm $\|\cdot\|^2$, and in step (b) we introduced $B_c\triangleq I_M-\mu_{\max}H_c$. We now proceed to bound the three terms on the right-hand side of the above inequality. First note that

$$
\begin{aligned}
B_c^TB_c&=(I-\mu_{\max}H_c)^T(I-\mu_{\max}H_c)\\
&=I-\mu_{\max}(H_c+H_c^T)+\mu_{\max}^2H_c^TH_c
\end{aligned}
\tag{6.105}
$$

Under Assumption 6.5, conditions (6.20) and (6.21) hold in the ball $||\delta w||\le r_H$ around w^o. Recall from (6.99) that H_c is evaluated at w^o. Therefore, from (6.21) we have

$$
H_c+H_c^T\ge 2\lambda_L\cdot I_M \tag{6.106}
$$

and by (6.20), we have

$$
\|H_c\|=\left\|\sum_{k=1}^N p_k\nabla_{w^T}s_k(w^o)\right\|
$$

$$\leq \sum_{k=1}^{N} p_k \|\nabla_{w^T} s_k(w^o)\|$$
$$\leq \sum_{k=1}^{N} p_k \cdot \lambda_U = \|p\|_1 \cdot \lambda_U \tag{6.107}$$

Note further that $\|H_c\|^2 \equiv \lambda_{\max}(H_c^T H_c)$, where $\lambda_{\max}(\cdot)$ denotes the largest eigenvalue of the matrix argument. This implies that

$$0 \leq H_c^T H_c \leq \|p\|_1^2 \lambda_U^2 \cdot I_M \tag{6.108}$$

Substituting (6.106) and (6.108) into (6.105), we obtain

$$B_c^T B_c \leq \left(1 - 2\mu_{\max}\lambda_L + \mu_{\max}^2 \|p\|_1^2 \lambda_U^2\right) \cdot I \tag{6.109}$$

so that

$$\|B_c\|^2 \leq 1 - 2\mu_{\max}\lambda_L + \mu_{\max}^2 \|p\|_1^2 \lambda_U^2$$
$$\leq \left(1 - \mu_{\max}\lambda_L + \frac{1}{2}\mu_{\max}^2 \|p\|_1^2 \lambda_U^2\right)^2 = \gamma_c^2 \tag{6.110}$$

where in the last inequality we used $(1 - x) \leq (1 - \frac{1}{2}x)^2$. Next, we bound the second term on the right-hand side of (6.104). To do this, we need to bound it in two separate cases:

1. **Case 1:** $\|\tilde{w}_{c,i-1}\| + \|\check{\boldsymbol{w}}_{c,i-1}\| \leq r_H$

   This condition implies that, for any $0 \leq t \leq 1$, the vector $\bar{w}_{c,i-1} + t\check{\boldsymbol{w}}_{c,i-1}$ is inside a ball that is centered at w^o with radius r_H since:

$$\|(\bar{w}_{c,i-1} + t\check{\boldsymbol{w}}_{c,i-1}) - w^o\| = \|\tilde{w}_{c,i-1} + t\check{\boldsymbol{w}}_{c,i-1}\|$$
$$\leq \|\tilde{w}_{c,i-1}\| + t\|\check{\boldsymbol{w}}_{c,i-1}\|$$

$$\leq \|\tilde{w}_{c,i-1}\| + \|\check{\boldsymbol{w}}_{c,i-1}\|$$
$$\leq r_H \tag{6.111}$$

By Assumption 6.5, the function $s_k(w)$ is differentiable at $\bar{w}_{c,i-1} + t\check{\boldsymbol{w}}_{c,i-1}$ so that using the following mean-value theorem [105, p.6]:

$$s_k(\boldsymbol{w}_{c,i-1}) = s_k(\bar{w}_{c,i-1}) + \left(\int_0^1 \nabla_{w^T} s_k(\bar{w}_{c,i-1} + t\check{\boldsymbol{w}}_{c,i-1})dt\right) \cdot \check{\boldsymbol{w}}_{c,i-1} \tag{6.112}$$

Then, we have

$$\begin{aligned}
&\left\|s_c(\boldsymbol{w}_{c,i-1}) - s_c(\bar{w}_{c,i-1}) - H_c\check{\boldsymbol{w}}_{c,i-1}\right\|^2 \\
&= \Big\| \sum_{k=1}^N p_k[s_k(\boldsymbol{w}_{c,i-1}) - s_k(\bar{w}_{c,i-1})] - H_c\check{\boldsymbol{w}}_{c,i-1}\Big\|^2 \\
&= \Big\| \sum_{k=1}^N p_k \int_0^1 \nabla_{w^T} s_k(\bar{w}_{c,i-1}+t\check{\boldsymbol{w}}_{c,i-1})dt\cdot\check{\boldsymbol{w}}_{c,i-1} - \sum_{k=1}^N p_k \nabla_{w^T} s_k(w^o)\cdot\check{\boldsymbol{w}}_{c,i-1}\Big\|^2 \\
&= \Big\| \sum_{k=1}^N p_k \cdot \int_0^1 \big[\nabla_{w^T} s_k(\bar{w}_{c,i-1} + t\check{\boldsymbol{w}}_{c,i-1}) - \nabla_{w^T} s_k(w^o)\big]dt \cdot \check{\boldsymbol{w}}_{c,i-1}\Big\|^2 \\
&\leq \Big\{ \sum_{k=1}^N p_k \cdot \int_0^1 \big\|\nabla_{w^T} s_k(\bar{w}_{c,i-1} + t\check{\boldsymbol{w}}_{c,i-1}) - \nabla_{w^T} s_k(w^o)\big\|dt \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|\Big\}^2 \\
&\overset{(a)}{\leq} \Big\{ \sum_{k=1}^N p_k \cdot \int_0^1 \lambda_H \cdot \|(\bar{w}_{c,i-1} + t\check{\boldsymbol{w}}_{c,i-1}) - w^o\|dt \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|\Big\}^2 \\
&\leq \Big\{ \sum_{k=1}^N p_k \cdot \int_0^1 \lambda_H \cdot (\|\bar{w}_{c,i-1} - w^o\| + t\|\check{\boldsymbol{w}}_{c,i-1}\|)dt \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|\Big\}^2 \\
&\leq \Big\{ \sum_{k=1}^N p_k \cdot \int_0^1 \lambda_H \cdot (\|\bar{w}_{c,i-1} - w^o\| + \|\check{\boldsymbol{w}}_{c,i-1}\|)dt \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|\Big\}^2 \\
&= \Big\{ \sum_{k=1}^N p_k \cdot \lambda_H \cdot (\|\tilde{w}_{c,i-1}\| + \|\check{\boldsymbol{w}}_{c,i-1}\|) \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|\Big\}^2
\end{aligned}$$

$$
\begin{aligned}
&= \Big\{ \|p\|_1 \cdot \lambda_H \cdot (\|\tilde{w}_{c,i-1}\| + \|\check{\boldsymbol{w}}_{c,i-1}\|) \cdot \|\check{\boldsymbol{w}}_{c,i-1}\| \Big\}^2 \\
&= \|p\|_1^2 \cdot \lambda_H^2 \cdot (\|\tilde{w}_{c,i-1}\| + \|\check{\boldsymbol{w}}_{c,i-1}\|)^2 \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 \\
&\leq 2\|p\|_1^2 \cdot \lambda_H^2 \cdot (\|\tilde{w}_{c,i-1}\|^2 + \|\check{\boldsymbol{w}}_{c,i-1}\|^2) \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 \qquad (6.113)
\end{aligned}
$$

where step (a) uses Assumption 6.5 and the last inequality uses $(x+y)^2 \leq 2x^2 + 2y^2$.

2. **Case 2:** $\|\tilde{w}_{c,i-1}\| + \|\check{\boldsymbol{w}}_{c,i-1}\| > r_H$

   It holds that

$$
\begin{aligned}
&\big\| s_c(\boldsymbol{w}_{c,i-1}) - s_c(\bar{w}_{c,i-1}) - H_c \check{\boldsymbol{w}}_{c,i-1} \big\|^2 \\
&= \big\| \sum_{k=1}^{N} p_k \big[s_k(\boldsymbol{w}_{c,i-1}) - s_k(\bar{w}_{c,i-1}) - \nabla_{w^T} s_k(w^o) \cdot \check{\boldsymbol{w}}_{c,i-1} \big] \big\|^2 \\
&\leq \Big\{ \sum_{k=1}^{N} p_k \big\| s_k(\boldsymbol{w}_{c,i-1}) - s_k(\bar{w}_{c,i-1}) - \nabla_{w^T} s_k(w^o) \cdot \check{\boldsymbol{w}}_{c,i-1} \big\| \Big\}^2 \\
&\leq \Big\{ \sum_{k=1}^{N} p_k \big(\|s_k(\boldsymbol{w}_{c,i-1}) - s_k(\bar{w}_{c,i-1})\| + \|\nabla_{w^T} s_k(w^o)\| \cdot \|\check{\boldsymbol{w}}_{c,i-1}\| \big) \Big\}^2 \\
&\overset{(a)}{\leq} \Big\{ \sum_{k=1}^{N} p_k \cdot (\lambda_U \cdot \|\check{\boldsymbol{w}}_{c,i-1}\| + \lambda_U \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|) \Big\}^2 \\
&= \Big\{ 2\|p\|_1 \cdot \lambda_U \cdot \|\check{\boldsymbol{w}}_{c,i-1}\| \Big\}^2 \\
&\leq 4\|p\|_1^2 \cdot \lambda_U^2 \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 \\
&\overset{(b)}{\leq} 4\|p\|_1^2 \cdot \lambda_U^2 \cdot \left(\frac{\|\tilde{w}_{c,i-1}\| + \|\check{\boldsymbol{w}}_{c,i-1}\|}{r_H} \right)^2 \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 \\
&\overset{(c)}{\leq} 2\|p\|_1^2 \cdot \frac{4\lambda_U^2}{r_H^2} \cdot (\|\tilde{w}_{c,i-1}\|^2 + \|\check{\boldsymbol{w}}_{c,i-1}\|^2) \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 \qquad (6.114)
\end{aligned}
$$

where in step (a) we used (6.17) and (6.20), in step (b) we used the fact that $\|\tilde{w}_{c,i-1}\| + \|\check{\boldsymbol{w}}_{c,i-1}\| > r_H$ in the current case, and in step (c) we used the relation $\|x+y\|^2 \leq 2\|x\|^2 + 2\|y\|^2$.

Based on (6.113) and (6.114) from both cases, we have

$$\begin{aligned}&\left\|s_c(\boldsymbol{w}_{c,i-1}) - s_c(\bar{w}_{c,i-1}) - H_c\check{\boldsymbol{w}}_{c,i-1}\right\|^2\\&\quad\le 2\|p\|_1^2\cdot\lambda_{HU}^2\cdot(\|\tilde{w}_{c,i-1}\|^2+\|\check{\boldsymbol{w}}_{c,i-1}\|^2)\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^2\end{aligned}\tag{6.115}$$

where

$$\lambda_{HU}^2 \triangleq \max\left\{\lambda_H^2, \frac{4\lambda_U^2}{r_H^2}\right\}\tag{6.116}$$

The third term on the right-hand side of (6.104) can be bounded by (6.84). Therefore, substituting (6.110), (6.115) and (6.84) into (6.104) and applying the expectation operator, we get

$$\begin{aligned}\mathbb{E}\|\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i}\|^2 \le{}& \gamma_c\cdot\mathbb{E}\|\check{\boldsymbol{w}}_{c,i-1} - \check{\boldsymbol{w}}_{a,i-1}\|^2\\&+\frac{4\mu_{\max}\|p\|_1^2\lambda_{HU}^2}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\left(\mathbb{E}\|\check{\boldsymbol{w}}_{c,i-1}\|^2\cdot\|\tilde{w}_{c,i-1}\|^2 + \mathbb{E}\|\check{\boldsymbol{w}}_{c,i-1}\|^4\right)\\&+\frac{2N\mu_{\max}\|p^T\otimes I_M\|^2}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\lambda_U^2\cdot\|\bar{P}_1[A_1^TU_1]\|_\infty^2\cdot\mathbb{E}\|\boldsymbol{w}_{e,i-1}\|^2\end{aligned}\tag{6.117}$$

where in the last term on the right-hand side of (6.117) we used $\mathbb{1}^TP[x] = \|x\|^2$ from property (5.111) in Chapter 5. Recall from Theorem 5.2 in Chapter 5 that $\tilde{w}_{c,i-1}\to 0$, and from (6.88)–(6.89) that $\mathbb{E}\|\check{\boldsymbol{w}}_{c,i-1}\|^2 \le O(\mu_{\max})$ and $\mathbb{E}\|\boldsymbol{w}_{e,i-1}\|^2 \le O(\mu_{\max}^2)$ in steady-state. Moreover, we also have the following result regarding $\mathbb{E}\|\check{\boldsymbol{w}}_{c,i-1}\|^4$ in steady-state.

Lemma 6.2 (Asymptotic bound on the 4th order moment)**.** *Using Assumptions 6.1–6.7, it holds that*

$$\limsup_{i\to\infty}\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^4 \le O(\mu_{\max}^2)\tag{6.118}$$

$$\limsup_{i\to\infty} \mathbb{E}\|\boldsymbol{w}_{e,i}\|^4 \leq O(\mu_{\max}^4) \tag{6.119}$$

Proof. See Appendix 6.B. □

Therefore, inequality recursion (6.117) becomes

$$\mathbb{E}\|\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i}\|^2 \leq \gamma_c \cdot \mathbb{E}\|\check{\boldsymbol{w}}_{c,i-1} - \check{\boldsymbol{w}}_{a,i-1}\|^2 + O(\mu_{\max}^3) \tag{6.120}$$

As long as $\gamma_c < 1$, which is guaranteed by the stability condition (5.145) from Chapter 5, the above recursion (6.120) leads to

$$\begin{aligned}
\limsup_{i\to\infty} \mathbb{E}\|\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i}\|^2 &\leq \frac{1}{1-\gamma_c} \cdot O(\mu_{\max}^3) \\
&= \frac{1}{\mu_{\max} - \frac{1}{2}\mu_{\max}^2 \|p\|_1^2 \lambda_U^2} \cdot O(\mu_{\max}^3) \\
&= O(\mu_{\max}^2)
\end{aligned} \tag{6.121}$$

Based on (6.121), we can now show that the steady-state covariance matrix of $\check{\boldsymbol{w}}_{c,i}$ is equal to the covariance matrix of $\check{\boldsymbol{w}}_{a,i}$ plus a high order perturbation term. First, we have

$$\begin{aligned}
\check{\Pi}_{c,i} &= \mathbb{E}[\check{\boldsymbol{w}}_{c,i}\check{\boldsymbol{w}}_{c,i}^T] \\
&= \mathbb{E}[(\check{\boldsymbol{w}}_{a,i} + \check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})(\check{\boldsymbol{w}}_{a,i} + \check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T] \\
&= \mathbb{E}[\check{\boldsymbol{w}}_{a,i}\check{\boldsymbol{w}}_{a,i}^T] + \mathbb{E}[\check{\boldsymbol{w}}_{a,i}(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T] \\
&\quad + \mathbb{E}[(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})\check{\boldsymbol{w}}_{a,i}^T] + \mathbb{E}[(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T] \\
&= \check{\Pi}_{a,i} + \mathbb{E}[\check{\boldsymbol{w}}_{c,i}(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T] + \mathbb{E}[(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})\check{\boldsymbol{w}}_{c,i}^T] \\
&\quad - \mathbb{E}[(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T]
\end{aligned} \tag{6.122}$$

The second to the fourth terms in (6.122) are asymptotically high order terms of $\mu_{\max}$. Indeed, for the second term, we have as $i \to \infty$:

$$\begin{aligned}
&\limsup_{i\to\infty} \left\|\mathbb{E}[\check{\boldsymbol{w}}_{c,i}(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T]\right\| \\
&\qquad \le \limsup_{i\to\infty} \mathbb{E}\left\|\check{\boldsymbol{w}}_{c,i}(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T\right\| \\
&\qquad \le \limsup_{i\to\infty} \mathbb{E}[\|\check{\boldsymbol{w}}_{c,i}\| \cdot \|\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i}\|] \\
&\qquad \le \limsup_{i\to\infty} \sqrt{\mathbb{E}\|\check{\boldsymbol{w}}_{c,i}\|^2 \cdot \mathbb{E}\|\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i}\|^2} \\
&\qquad \le O(\mu_{\max}^{3/2})
\end{aligned} \tag{6.123}$$

Likewise, the third term in (6.122) is asymptotically $O(\mu_{\max}^{3/2})$. For the fourth term in (6.122), we have as $i \to \infty$:

$$\begin{aligned}
&\limsup_{i\to\infty} \left\|\mathbb{E}[(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T]\right\| \\
&\qquad \overset{(a)}{\le} \limsup_{i\to\infty} \mathbb{E}\left\|(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})(\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i})^T\right\| \\
&\qquad \overset{(b)}{\le} \limsup_{i\to\infty} \mathbb{E}\left\|\check{\boldsymbol{w}}_{c,i} - \check{\boldsymbol{w}}_{a,i}\right\|^2 \\
&\qquad \overset{(c)}{\le} O(\mu_{\max}^2)
\end{aligned} \tag{6.124}$$

where step (a) applies Jensen's inequality to the convex function $\|\cdot\|$, step (b) uses the relation $\|xy^T\| \le \|x\| \cdot \|y\|$, and step (c) uses (6.121). Substituting (6.123)–(6.124) into (6.122), we get,

$$\limsup_{i\to\infty} \left\|\check{\Pi}_{c,i} - \check{\Pi}_{a,i}\right\| \le O(\mu_{\max}^{3/2}) \tag{6.125}$$

or equivalently, in steady-state,

$$\check{\Pi}_{c,i} = \check{\Pi}_{a,i} + O(\mu_{\max}^{3/2}) \tag{6.126}$$

Combining with (6.97) we therefore find that

$$\Pi_\infty = \mathbb{1}\mathbb{1}^T \otimes \check{\Pi}_{a,i} \;+\; O(\mu_{\max}^{3/2}) \tag{6.127}$$

6.A.4 Evaluation of $\check{\Pi}_{a,\infty}$

We now proceed to evaluate $\check{\Pi}_{a,i}$ from recursion (6.101):

$$\begin{aligned}
\check{\Pi}_{a,i} &= B_c\check{\Pi}_{a,i-1}B_c^T + \mu_{\max}^2(p^T\otimes I_M)\mathbb{E}\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1})(p\otimes I_M) \\
&= B_c\check{\Pi}_{a,i-1}B_c^T + \mu_{\max}^2(p^T\otimes I_M)\mathbb{E}\mathcal{R}_{v,i}(\mathbb{1}\otimes w^o)(p\otimes I_M) \\
&\quad + \mu_{\max}^2(p^T\otimes I_M)\mathbb{E}\big[\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1}) - \mathcal{R}_{v,i}(\mathbb{1}\otimes w^o)\big](p\otimes I_M)
\end{aligned} \tag{6.128}$$

We will verify that the last perturbation term in (6.128) is also a high-order term in $\mu_{\max}$. First note that

$$\begin{aligned}
&\big\|\mu_{\max}^2(p^T\otimes I_M)\mathbb{E}\big[\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1}) - \mathcal{R}_{v,i}(\mathbb{1}\otimes w^o)\big](p\otimes I_M)\big\| \\
&\quad \le \mu_{\max}^2\cdot\|p\otimes I_M\|^2\cdot\mathbb{E}\big\|\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1}) - \mathcal{R}_{v,i}(\mathbb{1}\otimes w^o)\big\|
\end{aligned} \tag{6.129}$$

Next, we bound the rightmost term inside the expectation of (6.129). We also need to bound it in two separate cases before arriving at a universal bound:

1. **Case 1:** $\|\tilde{\boldsymbol{\phi}}_{i-1}\| \le r_V$

   By (6.26) in Assumption 6.6, we have

$$\big\|\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1}) - \mathcal{R}_{v,i}(\mathbb{1}\otimes w^o)\big\|$$

$$\leq \lambda_v \cdot \|\boldsymbol{\phi}_{i-1} - \mathbb{1} \otimes w^o\|^\kappa = \lambda_v \cdot \|\tilde{\boldsymbol{\phi}}_{i-1}\|^\kappa \tag{6.130}$$

2. **Case 2:** $\|\tilde{\boldsymbol{\phi}}_{i-1}\| > r_V$

   In this case, we have

$$\begin{aligned} &\big\|\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1}) - \mathcal{R}_{v,i}(\mathbb{1} \otimes w^o)\big\| \\ &\quad \leq \|\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1})\| + \|\mathcal{R}_{v,i}(\mathbb{1} \otimes w^o)\| \end{aligned} \tag{6.131}$$

   To proceed, we first bound $\|\mathcal{R}_{v,i}(\boldsymbol{w})\|$ as follows, where $\boldsymbol{w} \triangleq \text{col}\{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_N\}$. From the definition of $\mathcal{R}_{v,i}(\boldsymbol{w})$ in (6.24), we have

$$\begin{aligned} \|\mathcal{R}_{v,i}(\boldsymbol{w})\| &\overset{(a)}{\leq} \text{Tr}[\mathcal{R}_{v,i}(\boldsymbol{w})] \\ &= \text{Tr}\big[\mathbb{E}\{\boldsymbol{v}_i(\boldsymbol{w})\boldsymbol{v}_i^T(\boldsymbol{w})|\mathcal{F}_{i-1}\}\big] \\ &= \mathbb{E}\{\text{Tr}[\boldsymbol{v}_i(\boldsymbol{w})\boldsymbol{v}_i^T(\boldsymbol{w})]|\mathcal{F}_{i-1}\} \\ &= \mathbb{E}\{\|\boldsymbol{v}_i(\boldsymbol{w})\|^2|\mathcal{F}_{i-1}\} \\ &\overset{(b)}{=} \sum_{k=1}^{N} \mathbb{E}\{\|\boldsymbol{v}_{k,i}(\boldsymbol{w}_k)\|^2|\mathcal{F}_{i-1}\} \\ &\overset{(c)}{\leq} \sum_{k=1}^{N} \{\alpha \cdot \|\boldsymbol{w}_k\|^2 + \sigma_v^2|\mathcal{F}_{i-1}\} \\ &= \sum_{k=1}^{N} \{\alpha \cdot \|\boldsymbol{w}_k - w^o + w^o\|^2 + \sigma_v^2|\mathcal{F}_{i-1}\} \\ &\leq \sum_{k=1}^{N} \{2\alpha\|\boldsymbol{w}_k - w^o\|^2 + 2\alpha\|w^o\|^2 + \sigma_v^2|\mathcal{F}_{i-1}\} \\ &= 2\alpha \cdot \|\boldsymbol{w} - \mathbb{1} \otimes w^o\|^2 + 2\alpha N\|w^o\|^2 + N\sigma_v^2 \end{aligned} \tag{6.132}$$

   where in step (a) we used $\|X\| \leq \text{Tr}(X)$ for any symmetric positive semi-definite matrix X, in step (b) we used the definition of $\boldsymbol{v}_i(\boldsymbol{w})$ in (6.23), and

in step (c) we used (6.16). Using (6.132) with $\boldsymbol{w} = \boldsymbol{\phi}_{i-1}$ and $\boldsymbol{w} = \mathbb{1} \otimes w^o$, respectively, for the two terms on the right-hand side of (6.131), we get

$$\begin{aligned}
&\big\|\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1}) - \mathcal{R}_{v,i}(\mathbb{1} \otimes w^o)\big\| \\
&\quad \le 2\alpha \cdot \|\tilde{\boldsymbol{\phi}}_{i-1}\|^2 + 4\alpha N\|w^o\|^2 + 2N\sigma_v^2 \\
&\quad \overset{(a)}{\le} 2\alpha \cdot \|\tilde{\boldsymbol{\phi}}_{i-1}\|^2 + \big(4\alpha N\|w^o\|^2 + 2N\sigma_v^2\big) \cdot \frac{\|\tilde{\boldsymbol{\phi}}_{i-1}\|^2}{r_V^2} \\
&\quad = \big(2\alpha + \frac{4\alpha N\|w^o\|^2 + 2N\sigma_v^2}{r_V^2}\big) \cdot \|\tilde{\boldsymbol{\phi}}_{i-1}\|^2
\end{aligned} \tag{6.133}$$

where in step (a) we used the fact that $\|\tilde{\boldsymbol{\phi}}_{i-1}\| > r_V$ in the current case.

In summary, from (6.130) and (6.133), we obtain the following bound that holds in general:

$$\begin{aligned}
&\big\|\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1}) - \mathcal{R}_{v,i}(\mathbb{1} \otimes w^o)\big\| \\
&\quad \le \max\left\{\lambda_v \cdot \|\tilde{\boldsymbol{\phi}}_{i-1}\|^\kappa, \left(2\alpha + \frac{4\alpha N\|w^o\|^2 + 2N\sigma_v^2}{r_V^2}\right) \cdot \|\tilde{\boldsymbol{\phi}}_{i-1}\|^2\right\} \\
&\quad \le \lambda_{VU} \cdot \max\left\{\|\tilde{\boldsymbol{\phi}}_{i-1}\|^2, \|\tilde{\boldsymbol{\phi}}_{i-1}\|^\kappa\right\} \\
&\quad \le \lambda_{VU} \cdot \left\{\|\tilde{\boldsymbol{\phi}}_{i-1}\|^2 + \|\tilde{\boldsymbol{\phi}}_{i-1}\|^\kappa\right\}
\end{aligned} \tag{6.134}$$

where

$$\lambda_{VU} \triangleq \max\big\{\lambda_v,\ 2\alpha + \frac{4\alpha N\|w^o\|^2 + 2N\sigma_v^2}{r_V^2}\big\} \tag{6.135}$$

Substituting (6.134) into (6.129), we arrive at

$$\begin{aligned}
&\limsup_{i\to\infty} \Big\|\mu_{\max}^2 (p^T \otimes I_M)\mathbb{E}\big[\mathcal{R}_{v,i}(\boldsymbol{\phi}_{i-1}) - \mathcal{R}_{v,i}(\mathbb{1} \otimes w^o)\big](p \otimes I_M)\Big\| \\
&\le \limsup_{i\to\infty} \mu_{\max}^2 \cdot \|p \otimes I_M\|^2 \cdot \lambda_{VU} \cdot \Big[\mathbb{E}\|\tilde{\boldsymbol{\phi}}_{i-1}\|^2 + \mathbb{E}\|\tilde{\boldsymbol{\phi}}_{i-1}\|^\kappa\Big]
\end{aligned}$$

$$
\begin{aligned}
&\overset{(a)}{=} \limsup_{i\to\infty} \mu_{\max}^2 \cdot \|p\otimes I_M\|^2 \cdot \lambda_{VU} \cdot \big[\mathbb{E}\|\mathcal{A}_1^T\tilde{\boldsymbol{w}}_{i-1}\|^2 + \mathbb{E}\|\mathcal{A}_1^T\tilde{\boldsymbol{w}}_{i-1}\|^\kappa\big] \\
&\le \limsup_{i\to\infty} \mu_{\max}^2 \cdot \|p\otimes I_M\|^2 \cdot \lambda_{VU} \cdot \big[\|\mathcal{A}_1^T\|^2 \cdot \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^2 + \|\mathcal{A}_1^T\|^\kappa \cdot \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^\kappa\big] \\
&= \limsup_{i\to\infty} \mu_{\max}^2 \cdot \|p\otimes I_M\|^2 \cdot \lambda_{VU} \cdot \Big[\|\mathcal{A}_1^T\|^2 \cdot \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^2 + \|\mathcal{A}_1^T\|^\kappa \cdot \mathbb{E}\big\{(\|\tilde{\boldsymbol{w}}_{i-1}\|^4)^{\kappa/4}\big\}\Big] \\
&\overset{(b)}{\le} \limsup_{i\to\infty} \mu_{\max}^2 \cdot \|p\otimes I_M\|^2 \cdot \lambda_{VU} \cdot \Big[\|\mathcal{A}_1^T\|^2 \cdot \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^2 + \|\mathcal{A}_1^T\|^\kappa \cdot (\mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^4)^{\kappa/4}\Big] \\
&\overset{(c)}{\le} \mu_{\max}^2 \cdot [O(\mu_{\max}) + O(\mu_{\max}^{\kappa/2})] \\
&= O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2})
\end{aligned} \tag{6.136}
$$

where in step (a) we used the relation $\tilde{\boldsymbol{\phi}}_{i-1} = \mathcal{A}_1^T\tilde{\boldsymbol{w}}_{i-1}$ from (5.81) in Chapter 5, in step (b) we applied Jensen's inequality $\mathbb{E}(\boldsymbol{x}^{\kappa/4}) \le (\mathbb{E}\boldsymbol{x})^{\kappa/4}$ since $x^{\kappa/4}$ is a concave function when $0 < \kappa \le 4$, and in step (c) we used the fact that $\mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^2$ is on the order of $O(\mu_{\max})$ in steady-state and that $\mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^4$ is on the order of $O(\mu_{\max}^2)$ in steady-state. Using (6.136) in recursion (6.128) leads to the following relation as $i \to \infty$:

$$
\begin{aligned}
\check{\Pi}_{a,i} = B_c\check{\Pi}_{a,i-1}B_c^T + \mu_{\max}^2(p^T\otimes I_M)\mathbb{E}\mathcal{R}_{v,i}(\mathbb{1}\otimes w^o)(p\otimes I_M) \\
+ O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2})
\end{aligned} \tag{6.137}
$$

which is a perturbed version of the following recursion

$$
\check{\Pi}_{a,i}^o = B_c\check{\Pi}_{a,i-1}^o B_c^T + \mu_{\max}^2(p^T\otimes I_M)\mathbb{E}\mathcal{R}_{v,i}(\mathbb{1}\otimes w^o)(p\otimes I_M) \tag{6.138}
$$

We now show that the covariance matrices obtained from these two recursions are close to each other in the sense that

$$
\limsup_{i\to\infty} \|\check{\Pi}_{a,i} - \check{\Pi}_{a,\infty}^o\| \le O\big(\mu_{\max}^{\min(2,1+\kappa/2)}\big) \tag{6.139}
$$

which also means that, in steady-state,

$$\check{\Pi}_{a,i} = \check{\Pi}^o_{a,i} + O\big(\mu_{\max}^{\min(2,1+\kappa/2)}\big) \tag{6.140}$$

Subtracting (6.138) from (6.137), we get

$$\begin{aligned}\check{\Pi}_{a,i} - \check{\Pi}^o_{a,i} = B_c(\check{\Pi}_{a,i-1} - \check{\Pi}^o_{a,i-1})B_c^T \\ + O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2})\end{aligned} \tag{6.141}$$

Taking the 2−induced norm of both sides, we get

$$\begin{aligned}
&\big\|\check{\Pi}_{a,i} - \check{\Pi}^o_{a,i}\big\| \\
&\leq \|B_c\|^2 \cdot \big\|\check{\Pi}_{a,i-1} - \check{\Pi}^o_{a,i-1}\big\| + O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2}) \\
&\overset{(a)}{\leq} \gamma_c^2 \cdot \big\|\check{\Pi}_{a,i-1} - \check{\Pi}^o_{a,i-1}\big\| + O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2}) \\
&\leq \gamma_c^{2i} \cdot \big\|\check{\Pi}_{a,0} - \check{\Pi}^o_{a,0}\big\| + \sum_{j=0}^{i-1} \gamma_c^{2j} \times [O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2})] \\
&\leq \gamma_c^{2i} \cdot \big\|\check{\Pi}_{a,0} - \check{\Pi}^o_{a,0}\big\| + \sum_{j=0}^{\infty} \gamma_c^{2j} \times [O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2})] \\
&\overset{(b)}{=} \gamma_c^{2i} \cdot \big\|\check{\Pi}_{a,0} - \check{\Pi}^o_{a,0}\big\| + \frac{O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2})}{1-\gamma_c^2} \\
&\leq \gamma_c^{2i} \cdot \big\|\check{\Pi}_{a,0} - \check{\Pi}^o_{a,0}\big\| + \frac{O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2})}{1-\gamma_c} \\
&\leq \gamma_c^{2i} \cdot \big\|\check{\Pi}_{a,0} - \check{\Pi}^o_{a,0}\big\| + \frac{O(\mu_{\max}^3) + O(\mu_{\max}^{\kappa/2+2})}{\mu_{\max}\lambda_L - \frac{1}{2}\mu_{\max}^2\|p\|_1\lambda_U^2} \\
&= \gamma_c^{2i} \cdot \big\|\check{\Pi}_{a,0} - \check{\Pi}^o_{a,0}\big\| + \frac{O(\mu_{\max}^2) + O(\mu_{\max}^{\kappa/2+1})}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1\lambda_U^2} \\
&= \gamma_c^{2i} \cdot \big\|\check{\Pi}_{a,0} - \check{\Pi}^o_{a,0}\big\| + \frac{O\big(\mu_{\max}^{\min(2,1+\kappa/2)}\big)}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1\lambda_U^2}
\end{aligned} \tag{6.142}$$

where in step (a) we are using (6.110) and in step (b) we are using the fact that $\gamma_c < 1$, which is already guaranteed by choosing $\mu_{\max}$ according to the stability condition (5.145) in Chapter 5. Taking lim sup of both sides of (6.142), we arrive at (6.139).

6.A.5 Final expression for Π_∞

Therefore, by (6.126) and (6.140), we have

$$\begin{aligned}\check{\Pi}_{c,\infty} &= \check{\Pi}^o_{a,\infty} + O(\mu_{\max}^{3/2}) + O\big(\mu_{\max}^{\min(2,1+\kappa/2)}\big) \\ &= \check{\Pi}^o_{a,\infty} + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \end{aligned} \tag{6.143}$$

Now we proceed to derive the expression for $\check{\Pi}^o_{a,\infty}$. As $i \to \infty$, the unperturbed recursion (6.138) converges to a unique solution $\check{\Pi}^o_{a,\infty}$ that satisfies the following discrete Lyapunov equation:

$$\check{\Pi}^o_{a,\infty} = B_c \check{\Pi}^o_{a,\infty} B_c^T + \mu_{\max}^2 (p^T \otimes I_M) \cdot \mathcal{R}_v \cdot (p \otimes I_M) \tag{6.144}$$

where we used (6.25) from Assumption 6.6. Vectorizing both sides of the above equation, we obtain

$$\begin{aligned}\mathrm{vec}(\check{\Pi}^o_{a,\infty}) &= \mu_{\max}^2 \cdot (I_{M^2} - B_c \otimes B_c)^{-1} \mathrm{vec}\left\{(p^T \otimes I_M) \cdot \mathcal{R}_v \cdot (p \otimes I_M)\right\} \\ &= \mu_{\max} \cdot (I_M \otimes H_c + H_c \otimes I_M - \mu_{\max} H_c \otimes H_c)^{-1} \\ &\quad \times \mathrm{vec}\left\{(p^T \otimes I_M) \cdot \mathcal{R}_v \cdot (p \otimes I_M)\right\} \\ &\overset{(a)}{=} \mu_{\max} \cdot (I_M \otimes H_c + H_c \otimes I_M)^{-1} \\ &\quad \times \big[I_{M^2} - \mu_{\max}(H_c \otimes H_c)(I_M \otimes H_c + H_c \otimes I_M)^{-1}\big]^{-1} \\ &\quad \times \mathrm{vec}\left\{(p^T \otimes I_M) \cdot \mathcal{R}_v \cdot (p \otimes I_M)\right\} \end{aligned} \tag{6.145}$$

where step (a) uses the fact that $(X+Y)^{-1} = X^{-1}(I+YX^{-1})^{-1}$ given X is invertible. Note that the existence of the inverse of $I_M \otimes H_c + H_c \otimes I_M$ is guaranteed by (6.21) for the following reason. First, condition (6.21) ensures that all the eigenvalues of H_c have positive real parts. To see this, let $\lambda(H_c)$ and x_0 ($x_0 \neq 0$) denote an eigenvalue of H_c and the corresponding eigenvector[1]. Then,

$$\begin{aligned} H_c x_0 = \lambda(H_c) \cdot x_0 &\Rightarrow x_0^* H_c x_0 = \lambda(H_c) \cdot \|x_0\|^2 \qquad (6.146) \\ &\Rightarrow (x_0^* H_c x_0)^* = \lambda^*(H_c) \cdot \|x_0\|^2 \\ &\Rightarrow x_0^* H_c^* x_0 = \lambda^*(H_c) \cdot \|x_0\|^2 \\ &\Rightarrow x_0^* H_c^T x_0 = \lambda^*(H_c) \cdot \|x_0\|^2 \qquad (6.147) \end{aligned}$$

where $(\cdot)^*$ denotes the conjugate transpose operator, and the last step uses the fact that H_c is real so that $H_c^* = H_c^T$. Summing (6.146) and (6.147) leads to

$$\begin{aligned} & x_0^*(H_c + H_c)^T x_0 = 2\mathrm{Re}\{\lambda(H_c)\} \cdot \|x_0\|^2 \\ \Rightarrow \quad & \mathrm{Re}\{\lambda(H_c)\} = \frac{x_0^*(H_c + H_c)^T x_0}{2\|x_0\|^2} \geq \lambda_L > 0 \qquad (6.148) \end{aligned}$$

where the last step uses (6.21). Furthermore, the M^2 eigenvalues of $I_M \otimes H_c + H_c \otimes I_M$ are $\lambda_{m_1}(H_c) + \lambda_{m_2}(H_c)$ for $m_1, m_2 = 1, \ldots, M$, where $\lambda_m(\cdot)$ denotes the mth eigenvalue of a matrix [82, p.143]. Therefore, the real parts of the eigenvalues of $I_M \otimes H_c + H_c \otimes I_M$ are $\mathrm{Re}\{\lambda_{m_1}(H_c)\} + \mathrm{Re}\{\lambda_{m_2}(H_c)\} > 0$ so that the matrix $I_M \otimes H_c + H_c \otimes I_M$ is not singular and is invertible. Observing that for any matrix

[1]Note that the matrix H_c need not be symmetric and hence its eigenvalues and eigenvectors need not be real.

X where the necessary inverse holds, we have

$$\begin{aligned}
&(I-\mu_{\max}X)^{-1}(I-\mu_{\max}X)=I\\
\Leftrightarrow&(I-\mu_{\max}X)^{-1}-\mu_{\max}(I-\mu_{\max}X)^{-1}X=I\\
\Leftrightarrow&(I-\mu_{\max}X)^{-1}=I+\mu_{\max}(I-\mu_{\max}X)^{-1}X
\end{aligned}\tag{6.149}$$

and, hence,

$$\begin{aligned}
&\big[I_{M^2}-\mu_{\max}(H_c\otimes H_c)(I_M\otimes H_c+H_c\otimes I_M)^{-1}\big]^{-1}\\
&=I_{M^2}+\mu_{\max}\big[I-\mu_{\max}(H_c\otimes H_c)(I_M\otimes H_c+H_c\otimes I_M)^{-1}\big]^{-1}\\
&\quad\times(H_c\otimes H_c)(I_M\otimes H_c+H_c\otimes I_M)^{-1}\\
&\overset{(a)}{=}I_{M^2}+O(\mu_{\max})
\end{aligned}\tag{6.150}$$

where step (a) is because

$$\begin{aligned}
&\lim_{\mu_{\max}\to 0}\frac{1}{\mu_{\max}}\\
&\times\Big\{\mu_{\max}\big[I-\mu_{\max}(H_c\otimes H_c)(I_M\otimes H_c+H_c\otimes I_M)^{-1}\big]^{-1}\\
&\qquad(H_c\otimes H_c)(I_M\otimes H_c+H_c\otimes I_M)^{-1}\Big\}\\
&=\lim_{\mu_{\max}\to 0}\big[I-\mu_{\max}(H_c\otimes H_c)(I_M\otimes H_c+H_c\otimes I_M)^{-1}\big]^{-1}\\
&\qquad\times(H_c\otimes H_c)(I_M\otimes H_c+H_c\otimes I_M)^{-1}\\
&=(H_c\otimes H_c)(I_M\otimes H_c+H_c\otimes I_M)^{-1}\\
&=\text{constant}
\end{aligned}\tag{6.151}$$

Therefore, substituting (6.150) into (6.145) leads to

$$\mathrm{vec}(\check{\Pi}^o_{a,\infty})=\mu_{\max}\cdot\big[(I_M\otimes H_c+H_c\otimes I_M)^{-1}+O(\mu_{\max})\big]$$

$$
\begin{aligned}
&\times \operatorname{vec}\big\{(p^T \otimes I_M)\mathcal{R}_v(p \otimes I_M)\big\} \\
&= \mu_{\max} \cdot (I_M \otimes H_c + H_c \otimes I_M)^{-1} \\
&\quad \times \operatorname{vec}\big\{(p^T \otimes I_M)\mathcal{R}_v(p \otimes I_M)\big\} + O(\mu_{\max}^2)
\end{aligned} \tag{6.152}
$$

Combining (6.143) and (6.152), we get

$$
\begin{aligned}
\operatorname{vec}(\check{\Pi}_{c,\infty}) = \mu_{\max} \cdot (I_M \otimes H_c + H_c \otimes I_M)^{-1} &\operatorname{vec}\big\{(p^T \otimes I_M)\mathcal{R}_v(p \otimes I_M)\big\} \\
&+ O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big)
\end{aligned} \tag{6.153}
$$

By (6.75) and (6.97), the weighted MSE, $\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|_{\Sigma}^2$, is given by

$$
\begin{aligned}
\lim_{i\to\infty} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|_{\Sigma}^2 &= \operatorname{Tr}\big\{(\mathbb{1}\mathbb{1}^T \otimes \check{\Pi}_{c,\infty})(E_{kk} \otimes \Sigma)\big\} + O(\mu_{\max}^{3/2}) \\
&= \operatorname{Tr}\big\{(\mathbb{1}\mathbb{1}^T E_{kk}) \otimes (\check{\Pi}_{c,\infty}\Sigma)\big\} + O(\mu_{\max}^{3/2}) \\
&\overset{(a)}{=} \operatorname{Tr}\big\{\mathbb{1}\mathbb{1}^T E_{kk}\big\} \cdot \operatorname{Tr}\big\{\check{\Pi}_{c,\infty}\Sigma\big\} + O(\mu_{\max}^{3/2}) \\
&= \operatorname{Tr}\big\{\check{\Pi}_{c,\infty}\Sigma\big\} + O(\mu_{\max}^{3/2}) \\
&\overset{(b)}{=} \operatorname{Tr}(\Sigma\check{\Pi}_{c,\infty}) + O(\mu_{\max}^{3/2}) \\
&\overset{(c)}{=} \operatorname{Tr}(\Sigma^T\check{\Pi}_{c,\infty}) + O(\mu_{\max}^{3/2}) \\
&\overset{(d)}{=} (\operatorname{vec}(\Sigma))^T \operatorname{vec}(\check{\Pi}_{c,\infty}) + O(\mu_{\max}^{3/2}) \\
&\overset{(e)}{=} \mu_{\max} \cdot \big(\operatorname{vec}\big\{(p^T \otimes I_M) \cdot \mathcal{R}_v \cdot (p \otimes I_M)\big\}\big)^T \\
&\quad \times (I_M \otimes H_c^T + H_c^T \otimes I_M)^{-1}\operatorname{vec}(\Sigma) \\
&\quad + O(\mu_{\max}^{3/2}) + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big) \\
&= \mu_{\max} \cdot \big(\operatorname{vec}\big\{(p^T \otimes I_M) \cdot \mathcal{R}_v \cdot (p \otimes I_M)\big\}\big)^T \\
&\quad \times (I_M \otimes H_c^T + H_c^T \otimes I_M)^{-1}\operatorname{vec}(\Sigma) \\
&\quad + O\big(\mu_{\max}^{\min(3/2,1+\kappa/2)}\big)
\end{aligned} \tag{6.154}
$$

where step (a) uses the property $\mathrm{Tr}(X\otimes Y) = \mathrm{Tr}(X)\mathrm{Tr}(Y)$ for Kronecker products [82, p.142], step (b) uses the property $\mathrm{Tr}(XY) = \mathrm{Tr}(YX)$, step (c) uses the fact that Σ is symmetric, step (d) uses the property $\mathrm{Tr}(XY) = \big(\mathrm{vec}(X^T)\big)^T \mathrm{vec}(Y)$, and step (e) substitutes (6.143). Note that the term $(I_M \otimes H_c^T + H_c^T \otimes I_M)^{-1}\mathrm{vec}(\Sigma)$ is in fact the vectorized version of the solution matrix X to the Lyapunov equation (6.42) for any given positive semi-definite weighting matrix Σ. Using again the relation $\mathrm{Tr}(XY) = \big(\mathrm{vec}(X^T)\big)^T \mathrm{vec}(Y) = \Big(\big(\mathrm{vec}(X^T)\big)^T \mathrm{vec}(Y)\Big)^T = \mathrm{vec}(Y)^T\mathrm{vec}(X^T)$, the weighted MSE expression (6.154) becomes (6.41). As a final remark, since condition (6.21) ensures that all the eigenvalues of H_c have positive real parts, i.e., the matrix $-H_c$ is asymptotically stable, the following Lyapunov equation, which is equivalent to (6.42),

$$(-H_c^T)X + X(-H_c) = -\Sigma \tag{6.155}$$

will have a unique solution given by (6.43) [82, pp.145-146] and is positive semi-definite (strictly positive definite) if Σ is symmetric and positive semi-definite (strictly positive definite) (see [105, p.39] and [73, p.769]).

6.B Proof of Lemma 6.2

The arguments in the previous appendix relied on results (6.118) and (6.119) from Lemma 6.2. To establish these results, we first need to introduce a fourth-order version of the energy operator we dealt with in Appendices 5.B and 5.C in Chapter 5, and establish some of its properties.

Definition 6.1 (Fourth order moment operator)**.** *Let $x = \mathrm{col}\{x_1, \ldots, x_N\}$ with sub-vectors of size $M \times 1$ each. We define $P^{(4)}[x]$ to be an operator that maps*

from $\mathbb{R}^{MN}$ *to* $\mathbb{R}^N$*:*

$$P^{(4)}[x] \triangleq \mathrm{col}\{\|x_1\|^4, \|x_2\|^4, \ldots, \|x_N\|^4\} \tag{6.156}$$

□

By following the same line of reasoning as the one used for the energy operator $P[\cdot]$ in Appendices 5.B and 5.C in Chapter 5, we can establish the following properties for $P^{(4)}[\cdot]$.

Lemma 6.3 (Properties of the 4th order moment operator)**.** *The operator* $P^{(4)}[\cdot]$ *satisfies the following properties:*

1. ***(*Nonnegativity*):*** $P^{(4)}[x] \succeq 0$
2. ***(*Scaling*):*** $P^{(4)}[ax] = |a|^4 \cdot P^{(4)}[x]$
3. ***(*Convexity*):*** *Suppose* $x^{(1)}, \ldots, x^{(K)}$ *are* $N \times 1$ *block vectors formed in the same manner as* x*, and let* $a_1, \ldots, a_K$ *be non-negative real scalars that add up to one. Then,*

$$P^{(4)}\big[a_1 x^{(1)} + \cdots + a_K x^{(K)}\big] \preceq a_1 P^{(4)}\big[x^{(1)}\big] + \cdots + a_K P^{(4)}\big[x^{(K)}\big] \tag{6.157}$$

4. ***(*Super-additivity*):***

$$P^{(4)}[x+y] \preceq 8 \cdot P^{(4)}[x] + 8 \cdot P^{(4)}[y] \tag{6.158}$$

5. ***(*Linear transformation*):***

$$P^{(4)}[Qx] \preceq \|\bar{P}[Q]\|_\infty^3 \cdot \bar{P}[Q]\; P^{(4)}[x] \tag{6.159}$$

$$\preceq \|\bar{P}[Q]\|_\infty^4 \cdot \mathbb{1}\mathbb{1}^T \cdot P^{(4)}[x] \tag{6.160}$$

6. ***(*Update operation*):*** *The global update vector* $s(x) \triangleq \mathrm{col}\{s_1(x_1), \ldots, s_N(x_N)\}$ *satisfies the following relation on* $P^{(4)}[\cdot]$*:*

$$P^{(4)}[s(x) - s(y)] \preceq \lambda_U^4 \cdot P^{(4)}[x - y] \tag{6.161}$$

7. ***(*Centralizd operation*):***

$$P^{(4)}[T_c(x) - T_c(y)] \preceq \gamma_c^4 \cdot P^{(4)}[x - y] \tag{6.162}$$

with the same factor

$$\gamma_c \triangleq 1 - \mu_{\max}\lambda_L + \frac{1}{2}\mu_{\max}^2 \|p\|_1^2 \lambda_U^2 \tag{6.163}$$

8. ***(*Stable Kronecker Jordan operator*):*** *Suppose* $\mathcal{D}_L = D_L \otimes I_M$*, where* D_L *is the* $L \times L$ *Jordan matrix defined by* (5.122)–(5.123) *in Chapter 5. Then, for any* $LM \times 1$ *vectors* x_e *and* y_e*, we have*

$$P^{(4)}[\mathcal{D}_L x_e + y_e] \preceq \Gamma_{e,4} \cdot P^{(4)}[x_e] + \frac{8}{(1-|d_2|)^3} \cdot P^{(4)}[y_e] \tag{6.164}$$

where $\Gamma_{e,4}$ *is the* $L \times L$ *matrix defined as*

$$\Gamma_{e,4} \triangleq \begin{bmatrix} |d_2| & \frac{8}{(1-|d_2|)^3} & & \\ & \ddots & \ddots & \\ & & \ddots & \frac{8}{(1-|d_2|)^3} \\ & & & |d_2| \end{bmatrix} \tag{6.165}$$

□

To proceed, we recall the transformed recursions (5.97)–(5.98) from Chapter 5, namely,

$$\check{\boldsymbol{w}}_{c,i} = T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\,[\boldsymbol{z}_{i-1} + \boldsymbol{v}_i] \tag{6.166}$$

$$\boldsymbol{w}_{e,i} = \mathcal{D}_{N-1}\boldsymbol{w}_{e,i-1} - \mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\,[s(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1} + \boldsymbol{v}_i] \tag{6.167}$$

If we now apply the operator $P^{(4)}[\cdot]$ to recursions (6.166)–(6.167), and follow arguments similar to the those employed in Appendices 5.G and 5.H from Chapter 5, we arrive at the following result. The statement extends Lemma 5.6 in Chapter 5 to 4th order moments.

Lemma 6.4 (Recursion for the 4th order moments)**.** *The fourth order moments satisfy the following inequality recursion*

$$\check{\mathcal{W}}'_{4,i} \preceq F_4\ \check{\mathcal{W}}'_{4,i-1} + H_4\check{\mathcal{W}}'_{i-1} + \mu_{\max}^4 \cdot b_{v,4} \tag{6.168}$$

where

$$\check{\mathcal{W}}'_{4,i} \triangleq \mathrm{col}\big\{\mathbb{E}P^{(4)}[\check{\boldsymbol{w}}_{c,i}],\ \mathbb{E}P^{(4)}[\boldsymbol{w}_{e,i}]\big\} \tag{6.169}$$

$$\check{\mathcal{W}}'_{i} \triangleq \mathrm{col}\big\{\mathbb{E}P[\check{\boldsymbol{w}}_{c,i}],\ \mathbb{E}P[\boldsymbol{w}_{e,i}]\big\} \tag{6.170}$$

$$F_4 \triangleq \begin{bmatrix} f_{cc}(\mu_{\max}) & f_{ce}(\mu_{\max})\cdot \mathbb{1}^T \\ f_{ec}(\mu_{\max})\cdot \mathbb{1} & F_{ee}(\mu_{\max}) \end{bmatrix} \tag{6.171}$$

$$H_4 \triangleq \begin{bmatrix} h_{cc}(\mu_{\max}) & h_{ce}(\mu_{\max})\cdot \mathbb{1}^T \\ 0 & 0 \end{bmatrix} \tag{6.172}$$

$$b_{v,4} \triangleq \mathrm{col}\big\{b_{v_4,c},\ b_{v_4,e}\cdot \mathbb{1}\big\} \tag{6.173}$$

where γ_c *is defined in* (6.103)*, and* $\Gamma_{e,4}$ *is defined in* (6.165)*, Moreover, the entries in* (6.171)–(6.172) *are given by:*

$$\begin{aligned}
f_{cc}(\mu_{\max}) &\triangleq \gamma_c + \mu_{\max}^4 \cdot 432\alpha_4\|p\|_1^4 \\
&\quad + \mu_{\max}^2 \cdot 20\alpha\|p\|_1^2 \cdot \Big(2 + \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2 \cdot \frac{\lambda_L + \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\Big) \\
&= \gamma_c + O(\mu_{\max}^2) \qquad (6.174) \\
f_{ce}(\mu_{\max}) &\triangleq \mu_{\max} \cdot \frac{\|p\|_1^4 \cdot \lambda_U^4 \cdot \big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4}{(\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3} \\
&\quad + 432\mu_{\max}^4\alpha_4\|p\|_1^4 \cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4 \\
&\quad + 20\mu_{\max}^2\alpha\|p\|_1^2 \cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2 \\
&\quad \cdot \Big(\mu_{\max} \cdot \frac{2\|p\|_1^2 \cdot \lambda_U^2 \cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2} + \frac{\lambda_L + \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\Big) \\
&= O(\mu_{\max}) \qquad (6.175) \\
h_{cc}(\mu_{\max}) &\triangleq 10\mu_{\max}^2 \cdot \Big(4\alpha\|p\|_1^2 \cdot \|\tilde{w}_{c,0}\|^2 + 4\alpha\|p\|_1^2 \cdot \|w^o\|^2 + \sigma_v^2 \cdot \|p\|_1^2\Big) \\
&= O(\mu_{\max}^2) \qquad (6.176) \\
h_{ce}(\mu_{\max}) &\triangleq \frac{10\|p\|_1^4\lambda_U^2 \cdot \mu_{\max}^3}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2} \cdot \big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^2 \cdot \Big(4\alpha \cdot \|\tilde{w}_{c,0}\|^2 + 4\alpha \cdot \|w^o\|^2 + \sigma_v^2 \cdot\Big) \\
&= O(\mu_{\max}^3) \qquad (6.177) \\
b_{v_4,c} &\triangleq 2\|p\|_1^4 \cdot \big(27\alpha_4 \cdot (\|\tilde{w}_{c,0}\|^4 + \|w^o\|^4) + \sigma_{v4}^4\big) \\
&= \text{constant} \qquad (6.178) \\
F_{ee}(\mu_{\max}) &\triangleq \Gamma_{e,4} + \mu_{\max}^4 \cdot \frac{216N \cdot (\lambda_U^4 + 216\alpha_4)}{(1 - |\lambda_2(A)|)^3} \cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4 \cdot \|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\|_\infty^4 \cdot \mathbb{1}\mathbb{1}^T \\
&= \Gamma_{e,4} + O(\mu_{\max}^4) \qquad (6.179) \\
f_{ec}(\mu_{\max}) &\triangleq \mu_{\max}^4 \cdot \frac{5832N \cdot (\lambda_U^4 + 8\alpha_4)}{(1 - |\lambda_2(A)|)^3} \|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\|_\infty^4 \\
&= O(\mu_{\max}^4) \qquad (6.180)
\end{aligned}$$

$$b_{v_4,e} \triangleq \frac{216N \cdot \|\bar{P}[\mathcal{U}_R \mathcal{A}_2^T]\|_\infty^4}{(1-|\lambda_2(A)|)^3} \cdot \Big\{ 27\big[(\lambda_U^4 + \alpha_4) \cdot \|\tilde{w}_{c,0}\|^4 + \|g_4^o\|_\infty + \alpha_4 \cdot \|w^o\|^4\big] + \sigma_{v4}^4 \Big\}$$
$$= \text{constant} \tag{6.181}$$

Proof. See Appendix 6.C. □

Observe from (6.168) that the recursion of the fourth order moments are coupled with the second order moments contained in $\check{\mathcal{W}}'_{i-1}$. Therefore, we will augment recursion (6.168) together with the following recursion for the second-order moment developed in (6.182) of Chapter 5:

$$\check{\mathcal{W}}'_i \preceq \Gamma \check{\mathcal{W}}'_{i-1} + \mu_{\max}^2 b_v \tag{6.182}$$

to form the following joint recursion:

$$\begin{bmatrix} \check{\mathcal{W}}'_i \\ \check{\mathcal{W}}'_{4,i} \end{bmatrix} \preceq \begin{bmatrix} \Gamma & 0 \\ H_4 & F_4 \end{bmatrix} \begin{bmatrix} \check{\mathcal{W}}'_{i-1} \\ \check{\mathcal{W}}'_{4,i-1} \end{bmatrix} + \begin{bmatrix} \mu_{\max}^2 \cdot b_v \\ \mu_{\max}^4 \cdot b_{v,4} \end{bmatrix} \tag{6.183}$$

The stability of the above recursion is guaranteed by the stability of the matrices Γ and F_4, i.e.,

$$\rho(\Gamma) < 1 \quad \text{and} \quad \rho(F_4) < 1 \tag{6.184}$$

The stability of Γ has already been established in Appendix 5.I of Chapter 5. Now, we discuss the stability of F_4. Using (6.174)–(6.179) and the definition of

γ_c in (6.110), we can express F_4 as

$$F_4 = \begin{bmatrix} \gamma_c + O(\mu_{\max}^2) & O(\mu_{\max}) \cdot \mathbb{1}^T \\ O(\mu_{\max}^4) & \Gamma_{e,4} + O(\mu_{\max}^4) \end{bmatrix} \tag{6.185}$$

$$= \begin{bmatrix} 1 - \mu_{\max}\lambda_L & O(\mu_{\max}) \\ 0 & \Gamma_{e,4} \end{bmatrix} + O(\mu_{\max}^2) \tag{6.186}$$

which has a similar structure to Γ — see expressions (5.133)–(5.134) in Chapter 5, and where in the last step we absorb the factor $\mathbb{1}^T$ in the $(1,2)$-th block into $O(\mu_{\max})$. Therefore, following the same line of argument from (5.278) to (5.295) in Appendix 5.I of Chapter 5, we can show that F_4 is also stable when the step-size parameter $\mu_{\max}$ is sufficiently small. Iterating (6.183), we get

$$\begin{bmatrix} \check{\mathcal{W}}_i' \\ \check{\mathcal{W}}_{4,i}' \end{bmatrix} \preceq \begin{bmatrix} \Gamma & 0 \\ H_4 & F_4 \end{bmatrix}^i \begin{bmatrix} \check{\mathcal{W}}_0' \\ \check{\mathcal{W}}_{4,0}' \end{bmatrix} + \sum_{j=0}^{i-1} \begin{bmatrix} \Gamma & 0 \\ H_4 & F_4 \end{bmatrix}^j \cdot \begin{bmatrix} \mu_{\max}^2 \cdot b_v \\ \mu_{\max}^4 \cdot b_{v,4} \end{bmatrix} \tag{6.187}$$

When both Γ and F_4 are stable, we have

$$\begin{aligned} \limsup_{i\to\infty} \begin{bmatrix} \check{\mathcal{W}}_i' \\ \check{\mathcal{W}}_{4,i}' \end{bmatrix} &\preceq \left(I - \begin{bmatrix} \Gamma & 0 \\ H_4 & F_4 \end{bmatrix} \right)^{-1} \cdot \begin{bmatrix} \mu_{\max}^2 \cdot b_v \\ \mu_{\max}^4 \cdot b_{v,4} \end{bmatrix} \\ &= \begin{bmatrix} \mu_{\max}^2 \cdot (I-\Gamma)^{-1} b_v \\ (I-F_4)^{-1} H_4 \cdot \mu_{\max}^2 \cdot (I-\Gamma)^{-1} b_v + \mu_{\max}^4 \cdot (I-F_4)^{-1} b_{v,4} \end{bmatrix} \end{aligned} \tag{6.188}$$

which implies that, for the fourth-order moment, we get

$$\limsup_{i\to\infty} \check{\mathcal{W}}_{4,i}' \preceq (I - F_4)^{-1} H_4 \cdot \mu_{\max}^2 \cdot (I - \Gamma)^{-1} b_v + \mu_{\max}^4 \cdot (I - F_4)^{-1} b_{v,4} \tag{6.189}$$

To evaluate the right-hand side of the above expression, we derive expressions for

$(I - F_4)^{-1}$ and $(I - \Gamma)^{-1}$ using the following formula for inverting a 2×2 block matrix [82, p.48], [116, p.16]:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + A^{-1}BECA^{-1} & -A^{-1}BE \\ -ECA^{-1} & E \end{bmatrix} \tag{6.190}$$

where $E = (D - CA^{-1}B)^{-1}$. By (6.186), we have the following expression for $(I - F_4)^{-1}$:

$$\begin{aligned} (I - F_4)^{-1} &= \left(I - \begin{bmatrix} 1 - \mu_{\max}\lambda_L & O(\mu_{\max}) \\ 0 & \Gamma_{e,4} \end{bmatrix} - O(\mu_{\max}^2) \right)^{-1} \\ &= \begin{bmatrix} \mu_{\max}\lambda_L - O(\mu_{\max}^2) & -O(\mu_{\max}) - O(\mu_{\max}^2) \\ -O(\mu_{\max}^2) & I - \Gamma_{e,4} - O(\mu_{\max}^2) \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \mu_{\max}\lambda_L - O(\mu_{\max}^2) & O(\mu_{\max}) \\ O(\mu_{\max}^2) & I - \Gamma_{e,4} - O(\mu_{\max}^2) \end{bmatrix}^{-1} \end{aligned} \tag{6.191}$$

Applying relation (6.190) to (6.191), we have

$$\begin{aligned} E_4 &= \left(I - \Gamma_{e,4} - O(\mu_{\max}^2) - \frac{O(\mu_{\max}^2)O(\mu_{\max})}{\mu_{\max}\lambda_L - O(\mu_{\max}^2)} \right)^{-1} \\ &= \left(I - \Gamma_{e,4} + O(\mu_{\max}^2) \right)^{-1} \end{aligned} \tag{6.192}$$

$$(I - F_4)^{-1} = \begin{bmatrix} \frac{1}{\mu_{\max}\lambda_L - O(\mu_{\max}^2)} + O(\mu_{\max}) & -\frac{O(1) \cdot E_4}{\lambda_L - O(\mu_{\max})} \\ -\frac{E_4 \cdot O(\mu_{\max})}{\lambda_L - O(\mu_{\max})} & E_4 \end{bmatrix} \tag{6.193}$$

Furthermore, recall from (5.133)–(5.134) of Chapter 5 for the expression of Γ:

$$\Gamma = \begin{bmatrix} \gamma_c & \mu_{\max} h_c(\mu_{\max}) \cdot \mathbb{1}^T \\ 0 & \Gamma_e \end{bmatrix} + \mu_{\max}^2 \psi_0 \cdot \mathbb{1}\mathbb{1}^T$$

$$= \begin{bmatrix} 1-\mu_{\max}\lambda_L & O(\mu_{\max}) \\ 0 & \Gamma_e \end{bmatrix} + O(\mu_{\max}^2) \tag{6.194}$$

Observing that Γ and F_4 have a similar structure, we can similarly get the expression for $(I-\Gamma)^{-1}$ as

$$(I-\Gamma)^{-1} = \begin{bmatrix} \frac{1}{\mu_{\max}\lambda_L - O(\mu_{\max}^2)} + O(\mu_{\max}) & -\frac{O(1)\cdot E_2}{\lambda_L - O(\mu_{\max})} \\ -\frac{E_2 \cdot O(\mu_{\max})}{\lambda_L - O(\mu_{\max})} & E_2 \end{bmatrix} \tag{6.195}$$

$$\begin{aligned} E_2 &= \left[I - \Gamma_e - O(\mu_{\max}^2) - \frac{O(\mu_{\max}^2)O(\mu_{\max})}{\mu_{\max}\lambda_L - O(\mu_{\max}^2)} \right]^{-1} \\ &= \left(I - \Gamma_e + O(\mu_{\max}^2) \right)^{-1} \end{aligned} \tag{6.196}$$

In addition, by substituting (6.176)–(6.177) into (6.172), we note that

$$H_4 = \begin{bmatrix} O(\mu_{\max}^2) & O(\mu_{\max}^3) \\ 0 & 0 \end{bmatrix} \tag{6.197}$$

Substituting (6.193), (6.195) and (6.197) into the right-hand side of (6.189) and using we obtain

$$\begin{aligned} \limsup_{i\to\infty} \check{\mathcal{W}}'_{4,i} \preceq & \begin{bmatrix} \frac{1}{\mu_{\max}\lambda_L - O(\mu_{\max}^2)} + O(\mu_{\max}) & -\frac{O(1)\cdot E_4}{\lambda_L - O(\mu_{\max})} \\ -\frac{E_4 \cdot O(\mu_{\max})}{\lambda_L - O(\mu_{\max})} & E_4 \end{bmatrix} \times \begin{bmatrix} O(\mu_{\max}^2) & O(\mu_{\max}^3) \\ 0 & 0 \end{bmatrix} \\ & \times \mu_{\max}^2 \cdot \begin{bmatrix} \frac{1}{\mu_{\max}\lambda_L - O(\mu_{\max}^2)} + O(\mu_{\max}) & -\frac{O(1)\cdot E_2}{\lambda_L - O(\mu_{\max})} \\ \frac{E_2 O(\mu_{\max})}{\lambda_L - O(\mu_{\max})} & E_2 \end{bmatrix} \begin{bmatrix} b_{v,c} \\ b_{v,e}\cdot \mathbb{1} \end{bmatrix} \\ & + \mu_{\max}^4 \cdot \begin{bmatrix} \frac{1}{\mu_{\max}\lambda_L - O(\mu_{\max}^2)} + O(\mu_{\max}) & -\frac{O(1)\cdot E_4}{\lambda_L - O(\mu_{\max})} \\ -\frac{E_4 \cdot O(\mu_{\max})}{\lambda_L - O(\mu_{\max})} & E_4 \end{bmatrix} \begin{bmatrix} b_{v_4,c} \\ b_{v_4,e}\cdot \mathbb{1} \end{bmatrix} \\ = & \begin{bmatrix} O(\mu_{\max}^2) \\ O(\mu_{\max}^4) \end{bmatrix} \end{aligned} \tag{6.198}$$

where the last step follows from basic matrix algebra. Recalling the definition of $\check{\mathcal{W}}'_{4,i}$ in (6.169), we conclude (6.118)–(6.119) from (6.198).

6.C Proof of Lemma 6.4

6.C.1 Perturbation Bounds

Before pursuing the proof of Lemma 6.4, we first state a result that bounds the fourth-order moments of the perturbation terms that appear in (6.166), in a manner similar to the bounds we already have for the second-order moments in (6.84)–(6.87).

Lemma 6.5 (Fourth-order bounds on the perturbation terms)**.** *Referring to* (6.166)*, the following bounds hold for any* $i \geq 0$*.*

$$P^{(4)}[\boldsymbol{z}_{i-1}] \preceq \lambda_U^4 \cdot \left\|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\right\|_\infty^4 \cdot \mathbb{1}\mathbb{1}^T \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \tag{6.199}$$

$$P^{(4)}[s(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1})] \preceq 27\lambda_U^4 \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^4 \cdot \mathbb{1} + 27\lambda_U^4 \|\tilde{w}_{c,0}\|^4 \cdot \mathbb{1} + 27 \cdot g_4^o \tag{6.200}$$

$$\begin{aligned} \mathbb{E}\big\{P^{(4)}[\boldsymbol{v}_i]\big|\mathcal{F}_{i-1}\big\} \preceq\ & 216\alpha_4 \cdot \mathbb{1} \cdot P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}] \\ & + 216\alpha_4 \cdot \left\|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\right\|_\infty^4 \cdot \mathbb{1}\mathbb{1}^T \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\ & + 27\alpha_4 \cdot (\|\tilde{w}_{c,0}\|^4 + \|w^o\|^4) \cdot \mathbb{1} + \sigma_{v4}^4 \cdot \mathbb{1} \end{aligned} \tag{6.201}$$

where $g_4^o \triangleq P^{(4)}[s(\mathbb{1}\otimes w^o)]$*.*

Proof. See Appendix 6.D. □

6.C.2 Recursion for the 4th order moment of $\check{\boldsymbol{w}}_{c,i}$

To begin with, note that by evaluating the squared Euclidean norm of both sides of (6.166) we obtain:

$$\begin{aligned}\|\check{\boldsymbol{w}}_{c,i}\|^2 &= \big\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)(\boldsymbol{z}_{i-1}+\boldsymbol{v}_i)\big\|^2\\ &= \big\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^2+\mu_{\max}^2\cdot\big\|(p^T\otimes I_M)\boldsymbol{v}_i\big\|^2\\ &\quad -2\mu_{\max}\cdot\big[T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big]^T\cdot(p^T\otimes I_M)\boldsymbol{v}_i\end{aligned} \tag{6.202}$$

By further squaring both sides of the above expression, we get

$$\begin{aligned}\|\check{\boldsymbol{w}}_{c,i}\|^4 &= \big\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^4\\ &\quad+\big\{\mu_{\max}^2\cdot\|(p^T\otimes I_M)\boldsymbol{v}_i\|^2-2\mu_{\max}\cdot\big[T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})\\ &\qquad-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big](p^T\otimes I_M)\boldsymbol{v}_i\big\}^2\\ &\quad-4\mu_{\max}\cdot\big\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^2\\ &\qquad\cdot\big[T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big]^T\cdot(p^T\otimes I_M)\boldsymbol{v}_i\\ &\quad+2\mu_{\max}^2\cdot\big\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^2\cdot\|(p^T\otimes I_M)\boldsymbol{v}_i\|^2\end{aligned} \tag{6.203}$$

Taking the conditional expectation of both sides of the above expression given $\mathcal{F}_{i-1}$ and recalling that $\mathbb{E}[\boldsymbol{v}_i|\mathcal{F}_{i-1}]=0$ based on (6.15), we get

$$\begin{aligned}\mathbb{E}[\|\check{\boldsymbol{w}}_{c,i}\|^4|\mathcal{F}_{i-1}] &= \mathbb{E}\Big\{\big\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^4\Big|\mathcal{F}_{i-1}\Big\}\\ &\quad+\mathbb{E}\Big(\big\{\mu_{\max}^2\|(p^T\otimes I_M)\boldsymbol{v}_i\|^2-2\mu_{\max}\big[T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})\\ &\qquad-\mu_{\max}(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big](p^T\otimes I_M)\boldsymbol{v}_i\big\}^2\big|\mathcal{F}_{i-1}\Big)\\ &\quad+2\mu_{\max}^2\cdot\big\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^2\end{aligned}$$

$$
\begin{aligned}
&\cdot \mathbb{E}\big[\|(p^T \otimes I_M)\boldsymbol{v}_i\|^2 \big| \mathcal{F}_{i-1}\big] \\
&\overset{(a)}{\leq} \big\|T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\big\|^4 \\
&\quad + 2\mu_{\max}^4 \cdot \mathbb{E}\|(p^T \otimes I_M)\boldsymbol{v}_i\|^4 \\
&\quad + 8\mu_{\max}^2 \cdot \big\|T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\big\|^2 \\
&\qquad \cdot \mathbb{E}\big[\|(p^T \otimes I_M)\boldsymbol{v}_i\|^2 \big| \mathcal{F}_{i-1}\big] \\
&\quad + 2\mu_{\max}^2 \cdot \big\|T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\big\|^2 \\
&\qquad \cdot \mathbb{E}\big[\|(p^T \otimes I_M)\boldsymbol{v}_i\|^2 \big| \mathcal{F}_{i-1}\big] \\
&= \big\|T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\big\|^4 \\
&\quad + 2\mu_{\max}^4 \cdot \mathbb{E}\big[\|(p^T \otimes I_M)\boldsymbol{v}_i\|^4 \big| \mathcal{F}_{i-1}\big] \\
&\quad + 10\mu_{\max}^2 \cdot \big\|T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\big\|^2 \\
&\qquad \cdot \mathbb{E}\big[\|(p^T \otimes I_M)\boldsymbol{v}_i\|^2 \big| \mathcal{F}_{i-1}\big] \qquad (6.204)
\end{aligned}
$$

where step (a) uses the inequality $(x+y)^2 \leq 2x^2 + 2y^2$. To proceed, we call upon the following bounds.

Lemma 6.6 (Useful bounds)**.** *The following bounds hold for arbitrary i:*

$$
\begin{aligned}
&\big\|T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\big\|^4 \\
&\quad \leq \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^4 \\
&\qquad + \frac{\mu_{\max}}{(\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3} \cdot \|p\|_1^4 \cdot \lambda_U^4 \cdot \big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4 \cdot \mathbb{1}^T P^{(4)}[\boldsymbol{w}_{e,i-1}] \qquad (6.205) \\
&\mathbb{E}\big[\,\big\|(p^T \otimes I_M)\boldsymbol{v}_i\big\|^4 \,\big|\, \mathcal{F}_{i-1}\big] \\
&\quad \leq 216\alpha_4\|p\|_1^4 \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^4 + 216\alpha_4\|p\|_1^4 \cdot \big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4 \cdot \mathbb{1}^T \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\qquad + 27\alpha_4\|p\|_1^4 \cdot \|\tilde{w}_{c,0}\|^4 + 27\alpha_4 \cdot \|p\|_1^4 \cdot \|w^o\|^4 + \sigma_{v4}^4 \cdot \|p\|_1^4 \qquad (6.206) \\
&\big\|T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max} \cdot (p^T \otimes I_M)\boldsymbol{z}_{i-1}\big\|^2
\end{aligned}
$$

$$\leq \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2} \cdot \|p\|_1^2 \cdot \lambda_U^2 \cdot \left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2 \cdot \mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \tag{6.207}$$

$$\begin{aligned}
&\mathbb{E}\big[\left\|(p^T \otimes I_M)\boldsymbol{v}_i\right\|^2 \big| \mathcal{F}_{i-1}\big] \\
&\quad \leq 4\alpha\|p\|_1^2 \cdot P[\check{\boldsymbol{w}}_{c,i-1}] + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2 \cdot \|p\|_1^2 \cdot \mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \\
&\qquad + 4\alpha\|\tilde{w}_{c,0}\|^2 \cdot \|p\|_1^2 + 4\alpha\|p\|_1^2 \cdot \|w^o\|^2 + \sigma_v^2 \cdot \|p\|_1^2
\end{aligned} \tag{6.208}$$

Proof. See Appendix 6.E. □

Substituting (6.205)–(6.208) into (6.204), we obtain

$$\begin{aligned}
\mathbb{E}[\|\check{\boldsymbol{w}}_{c,i}\|^4 | \mathcal{F}_{i-1}] \preceq\ & \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^4 \\
& + \frac{\mu_{\max}}{(\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3} \cdot \|p\|_1^4 \cdot \lambda_U^4 \cdot \left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^4 \\
& \quad \cdot \mathbb{1}^T P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
& + 2\mu_{\max}^4 \cdot \Big\{ 216\alpha_4\|p\|_1^4 \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^4 \\
& \quad + 216\alpha_4\|p\|_1^4 \cdot \left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^4 \cdot \mathbb{1}^T \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
& \quad + 27\alpha_4\|p\|_1^4 \cdot \|\tilde{w}_{c,0}\|^4 + 27\alpha_4\|p\|_1^4 \cdot \|w^o\|^4 + \sigma_{v4}^4 \cdot \|p\|_1^4 \Big\} \\
& + 10\mu_{\max}^2 \cdot \Big\{ \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 \\
& + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2} \cdot \|p\|_1^2 \cdot \lambda_U^2 \cdot \left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2 \\
& \quad \cdot \mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \Big\} \\
& \cdot \Big\{ 4\alpha\|p\|_1^2 \cdot P[\check{\boldsymbol{w}}_{c,i-1}] \\
& + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2 \cdot \|p\|_1^2 \cdot \mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \\
& + 4\alpha\|p\|_1^2(\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2 \cdot \|p\|_1^2 \Big\}
\end{aligned} \tag{6.209}$$

We further call upon the following inequality to bound the last term in (6.209):

$$
\begin{aligned}
&(a\cdot x+b\cdot y)(c\cdot x+d\cdot y+e)\\
&\quad=ac\cdot x^2+bd\cdot y^2+(ad+bc)xy+ae\cdot x+be\cdot y\\
&\quad\le ac\cdot x^2+bd\cdot y^2+(ad+bc)\frac{1}{2}(x^2+y^2)+ae\cdot x+be\cdot y\\
&\quad=\left(ac+\frac{ad+bc}{2}\right)x^2+\left(bd+\frac{ad+bc}{2}\right)y^2+ae\cdot x+be\cdot y
\end{aligned}
\tag{6.210}
$$

Applying the above inequality to the last term in (6.209) with

$$
\begin{aligned}
a&=\gamma_c\\
b&=\frac{\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|p\|_1^2\cdot\lambda_U^2\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2\\
c&=4\alpha\|p\|_1^2\\
d&=4\alpha\|p\|_1^2\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2\\
e&=4\alpha\|p\|_1^2\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\|p\|_1^2\cdot\|w^o\|^2+\sigma_v^2\cdot\|p\|_1^2\\
x&=\|\check{\boldsymbol{w}}_{c,i-1}\|^2\\
y&=\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}]=\|\boldsymbol{w}_{e,i-1}\|^4
\end{aligned}
$$

we get

$$
\begin{aligned}
&\left\{\gamma_c\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^2+\frac{\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|p\|_1^2\cdot\lambda_U^2\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2\cdot\mathbb{1}^TP[\boldsymbol{w}_{e,i-1}]\right\}\\
&\times\Big\{4\alpha\|p\|_1^2\cdot P[\check{\boldsymbol{w}}_{c,i-1}]+4\alpha\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\cdot\|p\|_1^2\cdot\mathbb{1}^TP[\boldsymbol{w}_{e,i-1}]\\
&\qquad+4\alpha\|p\|_1^2(\|\tilde{w}_{c,0}\|^2+\|w^o\|^2)+\sigma_v^2\cdot\|p\|_1^2\Big\}\\
&\le\left(ac+\frac{ad+bc}{2}\right)\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4+\left(bd+\frac{ad+bc}{2}\right)\cdot\|\boldsymbol{w}_{e,i-1}\|^4\\
&\qquad+ae\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^2+be\cdot\|\boldsymbol{w}_{e,i-1}\|^2
\end{aligned}
$$

$$
\begin{aligned}
&\overset{(a)}{\preceq} \left(c+\frac{d+bc}{2}\right)\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4+\left(bd+\frac{d+bc}{2}\right)\cdot\|\boldsymbol{w}_{e,i-1}\|^4\\
&\quad+e\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^2+be\cdot\|\boldsymbol{w}_{e,i-1}\|^2\\
&=\Big(4\alpha\|p\|_1^2+2\alpha\|p\|_1^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\\
&\quad+\frac{2\alpha\|p\|_1^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\|p\|_1^2\cdot\lambda_U^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\Big)\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\\
&\quad+\Big(\frac{4\alpha\|p\|_1^4\lambda_U^2\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4+2\alpha\|p\|_1^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\\
&\qquad+\frac{2\alpha\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\Big)\cdot\|\boldsymbol{w}_{e,i-1}\|^4\\
&\quad+\Big(4\alpha\|p\|_1^2\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\|p\|_1^2\cdot\|w^o\|^2+\sigma_v^2\cdot\|p\|_1^2\Big)\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^2\\
&\quad+\frac{\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^2\cdot\Big(4\alpha\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\cdot\|w^o\|^2+\sigma_v^2\cdot\Big)\cdot\|\boldsymbol{w}_{e,i-1}\|^2
\end{aligned}
\tag{6.211}
$$

where inequality (a) is using $a=\gamma_c<1$, which is guaranteed for sufficiently small step-sizes. Substituting (6.211) into (6.209), we get

$$
\begin{aligned}
&\mathbb{E}[\|\check{\boldsymbol{w}}_{c,i}\|^4|\mathcal{F}_{i-1}]\\
&\quad\preceq\gamma_c\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\\
&\qquad+\frac{\mu_{\max}}{(\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3}\cdot\|p\|_1^4\cdot\lambda_U^4\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4\\
&\qquad\quad\cdot\mathbb{1}^TP^{(4)}[\boldsymbol{w}_{e,i-1}]\\
&\qquad+2\mu_{\max}^4\|p\|_1^4\cdot\Big\{216\alpha_4\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\\
&\qquad\quad+216\alpha_4\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4\cdot\mathbb{1}^TP^{(4)}[\boldsymbol{w}_{e,i-1}]+27\alpha_4\cdot\|\tilde{w}_{c,0}\|^4\\
&\qquad\quad+27\alpha_4\cdot\|w^o\|^4+\sigma_{v4}^4\Big\}\\
&\qquad+10\mu_{\max}^2\cdot\Big\{\Big(4\alpha\|p\|_1^2+2\alpha\|p\|_1^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\\
&\qquad\quad+\frac{2\alpha\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\Big)\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4
\end{aligned}
$$

$$
\begin{aligned}
&+\Big(\frac{4\alpha\|p\|_1^4\cdot\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4\\
&\quad+2\alpha\|p\|_1^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\\
&\quad+\frac{2\alpha\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\Big)\cdot\|\boldsymbol{w}_{e,i-1}\|^4\\
&+\Big(4\alpha\|p\|_1^2\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\|p\|_1^2\cdot\|w^o\|^2+\sigma_v^2\cdot\|p\|_1^2\Big)\\
&\quad\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^2\\
&+\frac{\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^2\\
&\quad\cdot\Big(4\alpha\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\cdot\|w^o\|^2+\sigma_v^2\cdot\Big)\cdot\|\boldsymbol{w}_{e,i-1}\|^2\Big\}\\
&\overset{(a)}{=}\gamma_c\cdot P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}]\\
&+\frac{\mu_{\max}}{(\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3}\cdot\|p\|_1^4\cdot\lambda_U^4\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4\\
&\quad\cdot\mathbb{1}^TP^{(4)}[\boldsymbol{w}_{e,i-1}]\\
&+2\mu_{\max}^4\|p\|_1^4\cdot\Big\{216\alpha_4\cdot P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}]\\
&\quad+216\alpha_4\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4\cdot\mathbb{1}^TP^{(4)}[\boldsymbol{w}_{e,i-1}]\\
&\quad+27\alpha_4\cdot\|\tilde{w}_{c,0}\|^4+27\alpha_4\cdot\|w^o\|^4+\sigma_{v4}^4\Big\}\\
&+10\mu_{\max}^2\cdot\Big\{\Big(4\alpha\|p\|_1^2+2\alpha\|p\|_1^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\\
&\quad+\frac{2\alpha\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\Big)\cdot P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}]\\
&+\Big(\frac{4\alpha\|p\|_1^4\cdot\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4\\
&\quad+2\alpha\|p\|_1^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\\
&\quad+\frac{2\alpha\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\Big)\\
&\quad\cdot\mathbb{1}^TP^{(4)}[\boldsymbol{w}_{e,i-1}]\\
&+\Big(4\alpha\|p\|_1^2\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\|p\|_1^2\cdot\|w^o\|^2+\sigma_v^2\cdot\|p\|_1^2\Big)
\end{aligned}
$$

$$
\begin{aligned}
&\cdot P[\check{\boldsymbol{w}}_{c,i-1}] \\
&+\frac{\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2 \\
&\cdot\Big(4\alpha\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\cdot\|w^o\|^2+\sigma_v^2\cdot\Big)\cdot\mathbb{1}^T P[\boldsymbol{w}_{e,i-1}]\Big\} \\
=&\Big\{\gamma_c+432\mu_{\max}^4\alpha_4\|p\|_1^4 \\
&+10\mu_{\max}^2\cdot\Big(4\alpha\|p\|_1^2+2\alpha\|p\|_1^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2 \\
&+\frac{2\alpha\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\Big)\Big\}\cdot P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}] \\
&+\Big\{\mu_{\max}\cdot\frac{\|p\|_1^4\cdot\lambda_U^4\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^4}{(\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3} \\
&+432\mu_{\max}^4\alpha_4\|p\|_1^4\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4 \\
&+10\mu_{\max}^2\cdot\Big(\frac{4\alpha\|p\|_1^4\cdot\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4 \\
&\quad+2\alpha\|p\|_1^2\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2 \\
&\quad+\frac{2\alpha\|p\|_1^4\lambda_U^2\cdot\mu_{\max}}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2\Big)\Big\} \\
&\cdot\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&+10\mu_{\max}^2\cdot\Big(4\alpha\|p\|_1^2\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\|p\|_1^2\cdot\|w^o\|^2+\sigma_v^2\cdot\|p\|_1^2\Big) \\
&\cdot P[\check{\boldsymbol{w}}_{c,i-1}] \\
&+\frac{10\|p\|_1^4\lambda_U^2\cdot\mu_{\max}^3}{\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^2 \\
&\cdot\Big(4\alpha\cdot\|\tilde{w}_{c,0}\|^2+4\alpha\cdot\|w^o\|^2+\sigma_v^2\cdot\Big)\cdot\mathbb{1}^T\cdot P[\boldsymbol{w}_{e,i-1}] \\
&+2\mu_{\max}^4\cdot\|p\|_1^4\cdot\left(27\alpha_4\cdot(\|\tilde{w}_{c,0}\|^4+\|w^o\|^4)+\sigma_{v4}^4\right) \\
=&\Big\{\gamma_c+\mu_{\max}^4\cdot 432\alpha_4\|p\|_1^4 \\
&+\mu_{\max}^2\cdot 20\alpha\|p\|_1^2
\end{aligned}
$$

$$
\begin{aligned}
&\quad \cdot \Big(2 + \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \cdot \frac{\lambda_L + \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\Big)\Big\} \\
&\quad \cdot P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}] \\
&+ \Big\{\mu_{\max} \cdot \frac{\|p\|_1^4 \cdot \lambda_U^4 \cdot \big\|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\big\|_\infty^4}{(\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3} \\
&\quad + 432\mu_{\max}^4 \alpha_4 \|p\|_1^4 \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^4 \\
&\quad + 20\mu_{\max}^2 \alpha \|p\|_1^2 \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \\
&\qquad \cdot \Big(\mu_{\max} \cdot \frac{2\|p\|_1^2 \cdot \lambda_U^2 \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2} \\
&\qquad\quad + \frac{\lambda_L + \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2}\Big)\Big\} \cdot \mathbb{1}^T \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&+ 10\mu_{\max}^2 \cdot \Big(4\alpha\|p\|_1^2 \cdot \|\tilde{w}_{c,0}\|^2 + 4\alpha\|p\|_1^2 \cdot \|w^o\|^2 \\
&\quad + \sigma_v^2 \cdot \|p\|_1^2\Big) \cdot P[\check{\boldsymbol{w}}_{c,i-1}] \\
&+ \frac{10\|p\|_1^4\lambda_U^2 \cdot \mu_{\max}^3}{\lambda_L - \frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2} \cdot \big\|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\big\|_\infty^2 \\
&\quad \cdot \Big(4\alpha \cdot \|\tilde{w}_{c,0}\|^2 + 4\alpha \cdot \|w^o\|^2 + \sigma_v^2 \cdot\Big) \cdot \mathbb{1}^T \cdot P[\boldsymbol{w}_{e,i-1}] \\
&+ 2\mu_{\max}^4 \cdot \|p\|_1^4 \cdot \big(27\alpha_4 \cdot (\|\tilde{w}_{c,0}\|^4 + \|w^o\|^4) + \sigma_{v4}^4\big)
\end{aligned} \tag{6.212}
$$

where step (a) is using the following relations:

$$\|\boldsymbol{w}_{e,i-1}\|^4 = \mathbb{1}^T \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \tag{6.213}$$

$$\|\boldsymbol{w}_{e,i-1}\|^2 = \mathbb{1}^T \cdot P[\boldsymbol{w}_{e,i-1}] \tag{6.214}$$

$$\|\check{\boldsymbol{w}}_{c,i-1}\|^4 = P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}] \tag{6.215}$$

$$\|\check{\boldsymbol{w}}_{c,i-1}\|^2 = P[\check{\boldsymbol{w}}_{c,i-1}] \tag{6.216}$$

Using the notation defined in (6.174)–(6.178) and taking expectations of both

sides of (6.212) with respect to $\mathcal{F}_{i-1}$, we obtain

$$\begin{aligned}\mathbb{E}P^{(4)}[\check{\boldsymbol{w}}_{c,i}] \preceq\ & f_{cc}(\mu_{\max}) \cdot \mathbb{E}P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}] \\ & + f_{ce}(\mu_{\max}) \cdot \mathbb{1}^T \cdot \mathbb{E}P^{(4)}[\boldsymbol{w}_{e,i-1}] \\ & + h_{cc}(\mu_{\max}) \cdot \mathbb{E}P[\check{\boldsymbol{w}}_{c,i-1}] \\ & + h_{ce}(\mu_{\max}) \cdot \mathbb{1}^T \cdot \mathbb{E}P[\boldsymbol{w}_{e,i-1}] \\ & + \mu_{\max}^4 \cdot b_{v_4,c} \end{aligned} \tag{6.217}$$

6.C.3 Recursion for the 4th order moment of $\boldsymbol{w}_{e,i}$

We now derive an inequality recursion for $\mathbb{E}\|\boldsymbol{w}_{e,i}\|^4$. First, applying $P^{(4)}[\cdot]$ operator to both sides of (6.167), we get

$$\begin{aligned}
&P^{(4)}[\boldsymbol{w}_{e,i}] \\
&= P^{(4)}\Big[\mathcal{D}_{N-1}\boldsymbol{w}_{e,i-1} - \mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\big(s(\mathbb{1}\otimes\boldsymbol{w}_{e,i-1}) + \boldsymbol{z}_{i-1} + \boldsymbol{v}_i\big)\Big] \\
&\overset{(a)}{\preceq} \Gamma_{e,4} \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] + \frac{8}{(1-|\lambda_2(A)|)^3} \cdot P^{(4)}\big[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}\big(s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1} + \boldsymbol{v}_i\big)\big] \\
&\overset{(b)}{\preceq} \Gamma_{e,4} \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\quad + \frac{8}{(1-|\lambda_2(A)|)^3} \cdot \big\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}]\big\|_\infty^4 \cdot \mathbb{1}\mathbb{1}^T \cdot P^{(4)}\big[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1} + \boldsymbol{v}_i\big] \\
&\overset{(c)}{\preceq} \Gamma_{e,4} \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\quad + \mu_{\max}^4 \cdot \frac{8}{(1-|\lambda_2(A)|)^3} \cdot \big\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\big\|_\infty^4 \cdot \mathbb{1}\mathbb{1}^T \cdot P^{(4)}\big[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}) + \boldsymbol{z}_{i-1} + \boldsymbol{v}_i\big] \\
&= \Gamma_{e,4} \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\quad + \frac{8\mu_{\max}^4\big\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\big\|_\infty^4}{(1-|\lambda_2(A)|)^3} \cdot \mathbb{1}\mathbb{1}^T \cdot P^{(4)}\Big[\frac{1}{3}\cdot 3s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}) + \frac{1}{3}\cdot 3\boldsymbol{z}_{i-1} + \frac{1}{3}\cdot 3\boldsymbol{v}_i\Big] \\
&\overset{(d)}{\preceq} \Gamma_{e,4} \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}]
\end{aligned}$$

$$
\begin{aligned}
&+\frac{8\mu_{\max}^4\left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^4}{(1-|\lambda_2(A)|)^3}\cdot\mathbb{1}\mathbb{1}^T\times\Big\{\frac{1}{3}\cdot P^{(4)}\big[3s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})\big]+\frac{1}{3}\cdot P^{(4)}\big[3\boldsymbol{z}_{i-1}\big]\\
&\qquad+\frac{1}{3}\cdot P^{(4)}\big[3\boldsymbol{v}_i\big]\Big\}\\
&\overset{(e)}{=}\Gamma_{e,4}\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}]\\
&\quad+\mu_{\max}^4\cdot\frac{8}{(1-|\lambda_2(A)|)^3}\cdot\left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\\
&\qquad\times\big\{27\cdot P^{(4)}[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})]+27\cdot P^{(4)}[\boldsymbol{z}_{i-1}]+27\cdot P^{(4)}[\boldsymbol{v}_i]\big\}
\end{aligned}
\tag{6.218}
$$

where step (a) uses (6.164), step (b) uses (6.160), step (c) uses the sub-multiplicative property (5.106) from Chapter 5 and the sub-multiplicative property of matrix norms:

$$
\begin{aligned}
&\bar{P}[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}]\preceq\bar{P}[\mathcal{U}_R]\cdot\bar{P}[\mathcal{A}_2^T]\cdot\bar{P}[\mathcal{M}]\\
&\Rightarrow\left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}]\right\|_\infty\le\left\|\bar{P}[\mathcal{U}_R]\right\|_\infty\cdot\left\|\bar{P}[\mathcal{A}_2^T]\right\|_\infty\cdot\left\|\bar{P}[\mathcal{M}]\right\|_\infty\\
&\Rightarrow\left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T\mathcal{M}]\right\|_\infty\le\mu_{\max}\cdot\left\|\bar{P}[\mathcal{U}_R]\right\|_\infty\cdot\left\|\bar{P}[\mathcal{A}_2^T]\right\|_\infty
\end{aligned}
\tag{6.219}
$$

step (d) uses the convex property (6.157), and step (e) uses the scaling property in Lemma 6.3. Applying the expectation operator to both sides of the above inequality conditioned on $\mathcal{F}_{i-1}$, we obtain

$$
\begin{aligned}
\mathbb{E}\big[P^{(4)}[\boldsymbol{w}_{e,i}]\big|\mathcal{F}_{i-1}\big]\preceq\Gamma_{e,4}\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}]&\\
+\frac{8\mu_{\max}^4\left\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\right\|_\infty^4}{(1-|\lambda_2(A)|)^3}\cdot\mathbb{1}\mathbb{1}^T\cdot\Big\{27\cdot P^{(4)}[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})]&\\
+27\cdot P^{(4)}[\boldsymbol{z}_{i-1}]+27\cdot\mathbb{E}\{P^{(4)}[\boldsymbol{v}_i]|\mathcal{F}_{i-1}\}\Big\}&
\end{aligned}
\tag{6.220}
$$

In the above expression, we are using the fact that $\boldsymbol{w}_{c,i-1}$ and $\boldsymbol{z}_{i-1}$ are determined by the history up to time $i-1$. Therefore, given $\mathcal{F}_{i-1}$, these two quantities are

deterministic and known so that

$$\mathbb{E}\big\{P^{(4)}[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})]\big|\mathcal{F}_{i-1}\big\} = P^{(4)}[s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})] \tag{6.221}$$

$$\mathbb{E}\big\{P^{(4)}[\boldsymbol{z}_{i-1}]\big|\mathcal{F}_{i-1}\big\} = P^{(4)}[\boldsymbol{z}_{i-1}] \tag{6.222}$$

Substituting (6.199)–(6.201) into the right-hand side of the above inequality, we get

$$\begin{aligned}
&\mathbb{E}\big[P^{(4)}[\boldsymbol{w}_{e,i}]\big|\mathcal{F}_{i-1}\big] \\
&\quad\preceq \Gamma_{e,4}\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\qquad+\mu_{\max}^4\cdot\frac{8}{(1-|\lambda_2(A)|)^3}\cdot\big\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\big\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T \\
&\qquad\quad\cdot\Big\{27\cdot\Big[27\lambda_U^4\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\cdot\mathbb{1}+27\lambda_U^4\|\tilde{w}_{c,0}\|^4\cdot\mathbb{1}+27\cdot g_4^o\Big] \\
&\qquad\qquad+27\cdot\Big[\lambda_U^4\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}]\Big] \\
&\qquad\qquad+27\cdot\Big[216\alpha_4\cdot\mathbb{1}\cdot P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}]+216\alpha_4\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\qquad\qquad\qquad+27\alpha_4\cdot(\|\tilde{w}_{c,0}\|^4+\|w^o\|^4)\cdot\mathbb{1}+\sigma_{v4}^4\cdot\mathbb{1}\Big]\Big\} \\
&\quad=\Big[\Gamma_{e,4}+\mu_{\max}^4\cdot\frac{216N\cdot(\lambda_U^4+216\alpha_4)}{(1-|\lambda_2(A)|)^3}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4 \\
&\qquad\qquad\cdot\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\Big]\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\qquad+\mu_{\max}^4\cdot\frac{5832N\cdot(\lambda_U^4+8\alpha_4)}{(1-|\lambda_2(A)|)^3}\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\|_\infty^4\cdot P^{(4)}[\boldsymbol{w}_{c,i-1}]\cdot\mathbb{1} \\
&\qquad+\mu_{\max}^4\cdot\frac{216\cdot\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\|_\infty^4}{(1-|\lambda_2(A)|)^3}\cdot\Big\{27\big[(\lambda_U^4+\alpha_4)\cdot\|\tilde{w}_{c,0}\|^4\cdot N \\
&\qquad\quad+\mathbb{1}^T g_4^o+\alpha_4\cdot N\|w^o\|^4\big]+\sigma_{v4}^4\cdot N\Big\}\cdot\mathbb{1} \\
&\quad\preceq\Big[\Gamma_{e,4}+\mu_{\max}^4\cdot\frac{216N\cdot(\lambda_U^4+216\alpha_4)}{(1-|\lambda_2(A)|)^3}\cdot\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^4 \\
&\qquad\qquad\cdot\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\Big]\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}]
\end{aligned}$$

$$
\begin{aligned}
&+\mu_{\max}^4\cdot\frac{5832N\cdot(\lambda_U^4+8\alpha_4)}{(1-|\lambda_2(A)|)^3}\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\|_\infty^4\cdot P^{(4)}[\boldsymbol{w}_{c,i-1}]\cdot\mathbb{1}\\
&+\mu_{\max}^4\cdot\frac{216N\cdot\|\bar{P}[\mathcal{U}_R\mathcal{A}_2^T]\|_\infty^4}{(1-|\lambda_2(A)|)^3}\cdot\Big\{27\big[(\lambda_U^4+\alpha_4)\cdot\|\tilde{w}_{c,0}\|^4\\
&\quad+\|g_4^o\|_\infty+\alpha_4\cdot\|w^o\|^4\big]+\sigma_{v4}^4\Big\}\cdot\mathbb{1}
\end{aligned}
\tag{6.223}
$$

where the last step uses $\mathbb{1}^T g_4^o\le|\mathbb{1}^T g_4^o|\le N\|g_4^o\|_\infty$. Using the notation defined in (6.179)–(6.180) and applying the expectation operator to both sides of (6.223) with respect to $\mathcal{F}_{i-1}$, we arrive at

$$
\begin{aligned}
\mathbb{E}P^{(4)}[\boldsymbol{w}_{e,i}]\preceq F_{ee}(\mu_{\max})\cdot\mathbb{E}P^{(4)}[\boldsymbol{w}_{e,i-1}]+f_{ec}(\mu_{\max})\cdot\mathbb{1}\cdot\mathbb{E}P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}]\\
+\mu_{\max}^4\cdot b_{v_4,e}\cdot\mathbb{1}
\end{aligned}
\tag{6.224}
$$

6.D Proof of Lemma 6.5

First, we establish the bound for $P[\boldsymbol{z}_{i-1}]$ in (6.199). To begin with, recall the following two relations from (5.81) and (5.62) in Chapter 5:

$$
\boldsymbol{\phi}_i=\mathcal{A}_1^T\boldsymbol{w}_i \tag{6.225}
$$

$$
\boldsymbol{w}_i=\mathbb{1}\otimes\boldsymbol{w}_{c,,i}+(U_L\otimes I_M)\boldsymbol{w}_{e,i} \tag{6.226}
$$

By the definition of $\boldsymbol{z}_{i-1}$ in (6.83), we get:

$$
\begin{aligned}
P^{(4)}[\boldsymbol{z}_{i-1}]&=P^{(4)}[s(\boldsymbol{\phi}_{i-1})-s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})]\\
&\overset{(a)}{=}P^{(4)}[s(\mathcal{A}_1^T\boldsymbol{w}_{i-1})-s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})]\\
&\overset{(b)}{=}P^{(4)}\big[s\left(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1}+(A_1^TU_L\otimes I_M)\boldsymbol{w}_{e,i-1}\right)-s(\mathbb{1}\otimes\boldsymbol{w}_{c,i-1})\big]\\
&\overset{(c)}{\preceq}\lambda_U^4\cdot P^{(4)}\left[(A_1^TU_L\otimes I_M)\boldsymbol{w}_{e,i-1}\right]
\end{aligned}
$$

$$\overset{(d)}{\preceq} \lambda_U^4 \cdot \big\|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\big\|_\infty^4 \cdot \mathbb{1}\mathbb{1}^T \cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \tag{6.227}$$

where step (a) substitutes (6.225), step (b) substitutes (6.226), step (c) uses the variance relation (6.161), and step (d) uses property (6.160).

Next, we prove the bound for $P^{(4)}[s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1})]$. It holds that

$$\begin{aligned}
&P^{(4)}[s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1})] \\
&\quad = P^{(4)}\Big[\frac{1}{3}\cdot 3\big(s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1}) - s(\mathbb{1} \otimes \bar{w}_{c,i-1})\big) \\
&\qquad + \frac{1}{3}\cdot 3\big(s(\mathbb{1} \otimes \bar{w}_{c,i-1}) - s(\mathbb{1} \otimes w^o)\big) + \frac{1}{3}\cdot 3 \cdot s(\mathbb{1} \otimes w^o)\Big] \\
&\quad \overset{(a)}{\preceq} \frac{1}{3} \cdot P^{(4)}\big[3\big(s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1}) - s(\mathbb{1} \otimes \bar{w}_{c,i-1})\big)\big] \\
&\qquad + \frac{1}{3} \cdot P^{(4)}\big[3\big(s(\mathbb{1} \otimes \bar{w}_{c,i-1}) - s(\mathbb{1} \otimes w^o)\big)\big] + \frac{1}{3} \cdot P^{(4)}\big[3 \cdot s(\mathbb{1} \otimes w^o)\big] \\
&\quad \overset{(b)}{=} 3^3 \cdot P^{(4)}\big[s(\mathbb{1} \otimes \boldsymbol{w}_{c,i-1}) - s(\mathbb{1} \otimes \bar{w}_{c,i-1})\big] \\
&\qquad + 3^3 \cdot P^{(4)}\big[s(\mathbb{1} \otimes \bar{w}_{c,i-1}) - s(\mathbb{1} \otimes w^o)\big] + 3^3 \cdot P^{(4)}\big[s(\mathbb{1} \otimes w^o)\big] \\
&\quad \overset{(c)}{\preceq} 27\lambda_U^4 \cdot P^{(4)}\big[\mathbb{1} \otimes (\boldsymbol{w}_{c,i-1} - \bar{w}_{c,i-1})\big] \\
&\qquad + 27\lambda_U^4 \cdot P^{(4)}[\mathbb{1} \otimes (\bar{w}_{c,i-1} - w^o)] + 27 \cdot P^{(4)}[s(\mathbb{1} \otimes w^o)] \\
&\quad \overset{(d)}{=} 27\lambda_U^4 \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^4 \cdot \mathbb{1} + 27\lambda_U^4 \cdot \|\bar{w}_{c,i-1} - w^o\|^4 \cdot \mathbb{1} + 27 \cdot P^{(4)}[s(\mathbb{1} \otimes w^o)] \\
&\quad \overset{(e)}{\preceq} 27\lambda_U^4 \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^4 \cdot \mathbb{1} + 27\lambda_U^4 \|\tilde{w}_{c,0}\|^4 \cdot \mathbb{1} + 27 \cdot P^{(4)}[s(\mathbb{1} \otimes w^o)]
\end{aligned} \tag{6.228}$$

where step (a) uses the convexity property (6.157), step (b) uses the scaling property in Lemma 6.3, step (c) uses the variance relation (6.161), step (d) uses the definition of the operator $P^{(4)}[\cdot]$, and step (e) uses the bound $\|\tilde{w}_{c,i}\|^2 \leq \gamma_c^{2i} \cdot \|\tilde{w}_{c,0}\|^2$ from (5.129) of Chapter 5 and the fact that $\gamma_c < 1$.

Finally, we establish the bound on $P^{(4)}[\boldsymbol{v}_i]$ in (6.201). Introduce the $MN \times 1$

vector $\boldsymbol{x}$ according to (6.225)–(6.226):

$$\boldsymbol{x} \triangleq \mathbb{1} \otimes \boldsymbol{w}_{c,i-1} + \mathcal{A}_1^T \mathcal{U}_L \boldsymbol{w}_{e,i-1} \equiv \mathcal{A}_1^T \boldsymbol{w}_{i-1} = \boldsymbol{\phi}_{i-1} \tag{6.229}$$

We partition $\boldsymbol{x}$ into block form as $\boldsymbol{x} = \mathrm{col}\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, where each $\boldsymbol{x}_k$ is $M \times 1$. Then, by the definition of $\boldsymbol{v}_i$ from (6.82), we have

$$\begin{aligned}
&\mathbb{E}\left\{P^{(4)}[\boldsymbol{v}_i]\big|\mathcal{F}_{i-1}\right\} \\
&\quad = \mathbb{E}\left\{P^{(4)}\big[\hat{\boldsymbol{s}}_i(\boldsymbol{x}) - s(\boldsymbol{x})\big]\big|\mathcal{F}_{i-1}\right\} \\
&\quad = \mathrm{col}\big\{\mathbb{E}\big[\|\hat{\boldsymbol{s}}_{1,i}(\boldsymbol{x}_1) - s_1(\boldsymbol{x}_1)\|^4|\mathcal{F}_{i-1}\big], \ldots, \mathbb{E}\big[\|\hat{\boldsymbol{s}}_{N,i}(\boldsymbol{x}_N) - s_N(\boldsymbol{x}_N)\|^4|\mathcal{F}_{i-1}\big]\big\} \\
&\quad \overset{(a)}{\preceq} \begin{bmatrix} \alpha_4 \cdot \|\boldsymbol{x}_1\|^4 + \sigma_{v4}^4 \\ \vdots \\ \alpha_4 \cdot \|\boldsymbol{x}_N\|^4 + \sigma_{v4}^4 \end{bmatrix} \\
&\quad = \alpha_4 \cdot P^{(4)}[\boldsymbol{x}] + \sigma_{v4}^4 \cdot \mathbb{1}
\end{aligned} \tag{6.230}$$

where step (a) uses (6.38). Now we bound $\mathbb{E}P^{(4)}[\boldsymbol{x}]$ to complete the proof:

$$\begin{aligned}
P^{(4)}[\boldsymbol{x}] &= P^{(4)}\left[\mathbb{1} \otimes \boldsymbol{w}_{c,i-1} + \mathcal{A}_1^T \mathcal{U}_L \boldsymbol{w}_{e,i-1}\right] \\
&= P^{(4)}\Big[\frac{1}{3}\cdot 3\big(\mathbb{1}\otimes \boldsymbol{w}_{c,i-1} + \mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1} - \mathbb{1}\otimes\bar{w}_{c,i-1}\big) \\
&\quad + \frac{1}{3}\cdot 3\big(\mathbb{1}\otimes\bar{w}_{c,i-1} - \mathbb{1}\otimes w^o\big) + \frac{1}{3}\cdot 3\cdot\mathbb{1}\otimes w^o\Big] \\
&\overset{(a)}{\preceq} 27\cdot P^{(4)}\left[\mathbb{1}\otimes\boldsymbol{w}_{c,i-1} + \mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1} - \mathbb{1}\otimes\bar{w}_{c,i-1}\right] \\
&\quad + 27\cdot P^{(4)}\left[\mathbb{1}\otimes\bar{w}_{c,i-1} - \mathbb{1}\otimes w^o\right] + 27\cdot P^{(4)}\left[\mathbb{1}\otimes w^o\right] \\
&\overset{(b)}{=} 27\cdot P^{(4)}\left[\mathbb{1}\otimes\check{\boldsymbol{w}}_{c,i-1} + \mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1}\right] \\
&\quad + 27\cdot P^{(4)}\left[\mathbb{1}\otimes\tilde{w}_{c,i-1}\right] + 27\cdot P^{(4)}\left[\mathbb{1}\otimes w^o\right] \\
&\overset{(c)}{\preceq} 27\cdot\big(8\cdot P^{(4)}\left[\mathbb{1}\otimes\check{\boldsymbol{w}}_{c,i-1}\right] + 8\cdot P^{(4)}\left[\mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1}\right]\big)
\end{aligned}$$

$$
\begin{aligned}
&\quad +27\cdot P^{(4)}\big[\mathbb{1}\otimes\tilde{w}_{c,i-1}\big]+27\cdot P^{(4)}\big[\mathbb{1}\otimes w^o\big] \\
&\overset{(d)}{=} 216\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\cdot\mathbb{1}+216\cdot P^{(4)}\big[\mathcal{A}_1^T\mathcal{U}_L\boldsymbol{w}_{e,i-1}\big] \\
&\quad + 27\cdot\|\tilde{w}_{c,i-1}\|^4\cdot\mathbb{1}+27\cdot P^{(4)}\big[\mathbb{1}\otimes w^o\big] \\
&\overset{(e)}{\preceq} 216\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\cdot\mathbb{1}+216\cdot\big\|\bar{P}\big[\mathcal{A}_1^T\mathcal{U}_L\big]\big\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\quad +27\cdot\|\tilde{w}_{c,i-1}\|^4\cdot\mathbb{1}+27\cdot P^{(4)}\big[\mathbb{1}\otimes w^o\big] \\
&\overset{(f)}{\preceq} 216\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\cdot\mathbb{1}+216\cdot\big\|\bar{P}\big[\mathcal{A}_1^T\mathcal{U}_L\big]\big\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\quad +27\cdot\|\tilde{w}_{c,0}\|^4\cdot\mathbb{1}+27\cdot P^{(4)}\big[\mathbb{1}\otimes w^o\big] \\
&= 216\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\cdot\mathbb{1}+216\cdot\big\|\bar{P}\big[\mathcal{A}_1^T\mathcal{U}_L\big]\big\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}] \\
&\quad +27\cdot(\|\tilde{w}_{c,0}\|^4+\|w^o\|^4)\cdot\mathbb{1}
\end{aligned}
\tag{6.231}
$$

where step (a) uses the convexity property (6.157) and the scaling property in Lemma 6.3, step (b) uses the variance relation (6.161), step (c) uses the convexity property (6.157), step (d) uses the definition of the operator $P^{(4)}[\cdot]$, step (e) uses the variance relation (6.159), and step (f) uses the bound $\|\tilde{w}_{c,i}\|^2 \le \gamma_c^{2i}\cdot\|\tilde{w}_{c,0}\|^2$ from (5.129) of Chapter 5 and $\gamma_c < 1$. Substituting (6.231) into (6.230), we obtain (6.201).

6.E Proof of Lemma 6.6

First, we prove (6.205). It holds that

$$
\begin{aligned}
&\big\|T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^4 \\
&\quad = P^{(4)}\big[T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1}) - \mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big] \\
&\quad = P^{(4)}\Big[\gamma_c\cdot\frac{1}{\gamma_c}\big(T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1})\big) + (1-\gamma_c)\cdot\frac{-\mu_{\max}}{1-\gamma_c}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\Big]
\end{aligned}
$$

$$
\begin{aligned}
&\overset{(a)}{\preceq} \gamma_c \cdot P^{(4)}\big[\frac{1}{\gamma_c}\big(T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})\big)\big]+(1-\gamma_c)\cdot P^{(4)}\big[\frac{-\mu_{\max}}{1-\gamma_c}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big]\\
&\overset{(b)}{=} \gamma_c \cdot \frac{1}{\gamma_c^4}\cdot P^{(4)}\big[\big(T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})\big)\big]\\
&\quad+(1-\gamma_c)\cdot\frac{\mu_{\max}^4}{(1-\gamma_c)^4}\cdot P^{(4)}\big[(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big]\\
&\overset{(c)}{\preceq} \gamma_c \cdot P^{(4)}[\boldsymbol{w}_{c,i-1}-\bar{w}_{c,i-1}]+\frac{\mu_{\max}^4}{(1-\gamma_c)^3}\cdot P^{(4)}\left[(p^T\otimes I_M)\boldsymbol{z}_{i-1}\right]\\
&\overset{(d)}{=} \gamma_c \cdot P^{(4)}[\tilde{\boldsymbol{w}}_{c,i-1}]+\frac{\mu_{\max}^4}{(1-\gamma_c)^3}\cdot\left\|\sum_{k=1}^N p_k\boldsymbol{z}_{k,i-1}\right\|^4\\
&= \gamma_c \cdot P^{(4)}[\tilde{\boldsymbol{w}}_{c,i-1}]+\frac{\mu_{\max}^4}{(1-\gamma_c)^3}\cdot\left(\sum_{l=1}^N p_l\right)^4\cdot\left\|\sum_{k=1}^N\frac{p_k}{\sum_{l=1}^N p_l}\boldsymbol{z}_{k,i-1}\right\|^4\\
&\overset{(e)}{\preceq} \gamma_c \cdot P^{(4)}[\tilde{\boldsymbol{w}}_{c,i-1}]+\frac{\mu_{\max}^4}{(1-\gamma_c)^3}\cdot\left(\sum_{l=1}^N p_l\right)^4\cdot\sum_{k=1}^N\frac{p_k}{\sum_{l=1}^N p_l}\|\boldsymbol{z}_{k,i-1}\|^4\\
&= \gamma_c\cdot P^{(4)}[\tilde{\boldsymbol{w}}_{c,i-1}]+\frac{\mu_{\max}^4}{(1-\gamma_c)^3}\cdot\|p\|_1^3\cdot\sum_{k=1}^N p_k\|\boldsymbol{z}_{k,i-1}\|^4\\
&\overset{(f)}{=} \gamma_c\cdot P^{(4)}[\tilde{\boldsymbol{w}}_{c,i-1}]+\frac{\mu_{\max}\|p\|_1^3}{(\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3}\cdot p^T\cdot P^{(4)}[\boldsymbol{z}_{i-1}]\\
&\overset{(g)}{\preceq} \gamma_c\cdot P^{(4)}[\tilde{\boldsymbol{w}}_{c,i-1}]\\
&\quad+\frac{\mu_{\max}\|p\|_1^3}{(\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3}\cdot p^T\cdot\lambda_U^4\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T P^{(4)}[\boldsymbol{w}_{e,i-1}]\\
&= \gamma_c\cdot\|\tilde{\boldsymbol{w}}_{c,i-1}\|^4+\frac{\mu_{\max}\lambda_U^4\cdot\|p\|_1^4\cdot\left\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\right\|_\infty^4}{(\lambda_L-\frac{1}{2}\mu_{\max}\|p\|_1^2\lambda_U^2)^3}\cdot\mathbb{1}^T P^{(4)}[\boldsymbol{w}_{e,i-1}] \qquad (6.232)
\end{aligned}
$$

where step (a) uses property (6.157), step (b) uses the scaling property in Lemma 6.3, step (c) uses property (6.162), step (d) introduces $\boldsymbol{z}_{k,i-1}$ as the kth $M\times 1$ sub-vector of $\boldsymbol{z}_{1-1}=\text{col}\{\boldsymbol{z}_{1,i-1},\ldots,\boldsymbol{z}_{N,i-1}\}$, step (e) applies Jensen's inequality to the convex function $\|\cdot\|^4$, step (f) uses the definition of the operator $P^{(4)}[\cdot]$, and step (g) uses bound (6.199).

Second, we prove (6.206). Let $\boldsymbol{v}_{k,i}$ denote the kth $M\times 1$ sub-vector of $\boldsymbol{v}_i=$

$\text{col}\{\boldsymbol{v}_{1,i},\ldots,\boldsymbol{v}_{N,i}\}$. Then,

$$
\begin{aligned}
&\mathbb{E}\big[\,\big\|(p^T\otimes I_M)\boldsymbol{v}_i\big\|^4\,\big|\mathcal{F}_{i-1}\big]\\
&\quad=\mathbb{E}\Big[\Big\|\sum_{k=1}^{N}p_k\boldsymbol{v}_{k,i}\Big\|^4\Big|\mathcal{F}_{i-1}\Big]\\
&\quad=\left(\sum_{l=1}^{N}p_l\right)^4\cdot\mathbb{E}\Big[\,\Big\|\sum_{k=1}^{N}\frac{p_k}{\sum_{l=1}^{N}p_l}\boldsymbol{v}_{k,i}\Big\|^4\,\Big|\mathcal{F}_{i-1}\Big]\\
&\quad\overset{(a)}{\leq}\left(\sum_{l=1}^{N}p_l\right)^4\cdot\sum_{k=1}^{N}\frac{p_k}{\sum_{l=1}^{N}p_l}\mathbb{E}\left[\|\boldsymbol{v}_{k,i}\|^4\,|\mathcal{F}_{i-1}\right]\\
&\quad\overset{(b)}{=}\|p\|_1^3\cdot p^T\cdot\mathbb{E}\big\{P^{(4)}[\boldsymbol{v}_i]\big|\mathcal{F}_{i-1}\big\}\\
&\quad\overset{(c)}{\preceq}\|p\|_1^3\cdot p^T\cdot\Big\{216\alpha_4\cdot\mathbb{1}\cdot P^{(4)}[\check{\boldsymbol{w}}_{c,i-1}]\\
&\qquad\quad+216\alpha_4\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4\cdot\mathbb{1}\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}]\\
&\qquad\quad+27\alpha_4\cdot(\|\tilde{w}_{c,0}\|^4+\|w^o\|^4)\cdot\mathbb{1}+\sigma_{v4}^4\cdot\mathbb{1}\Big\}\\
&\quad=216\alpha_4\|p\|_1^4\cdot\|\check{\boldsymbol{w}}_{c,i-1}\|^4\\
&\qquad+216\alpha_4\|p\|_1^4\cdot\big\|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\big\|_\infty^4\cdot\mathbb{1}^T\cdot P^{(4)}[\boldsymbol{w}_{e,i-1}]\\
&\qquad+27\alpha_4\|p\|_1^4\cdot\|\tilde{w}_{c,0}\|^4+27\alpha_4\cdot\|p\|_1^4\cdot\|w^o\|^4+\sigma_{v4}^4\cdot\|p\|_1^4
\end{aligned}
\tag{6.233}
$$

where step (a) applies Jensen's inequality to the convex function $\|\cdot\|^4$, step (b) uses the definition of the operator $P^{(4)}[\cdot]$, and step (c) substitutes (6.201).

Third, we prove (6.207):

$$
\begin{aligned}
&\big\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})-\mu_{\max}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^2\\
&\quad=\Big\|\gamma_c\cdot\frac{1}{\gamma_c}\big(T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})\big)+(1-\gamma_c)\cdot\frac{-\mu_{\max}}{1-\gamma_c}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\Big\|^2\\
&\quad\overset{(a)}{\leq}\gamma_c\cdot\Big\|\frac{1}{\gamma_c}\big(T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})\big)\Big\|^2+(1-\gamma_c)\cdot\Big\|\frac{-\mu_{\max}}{1-\gamma_c}\cdot(p^T\otimes I_M)\boldsymbol{z}_{i-1}\Big\|^2\\
&\quad=\gamma_c\cdot\frac{1}{\gamma_c^2}\cdot\|T_c(\boldsymbol{w}_{c,i-1})-T_c(\bar{w}_{c,i-1})\|^2+(1-\gamma_c)\cdot\frac{\mu_{\max}^2}{(1-\gamma_c)^2}\cdot\big\|(p^T\otimes I_M)\boldsymbol{z}_{i-1}\big\|^2
\end{aligned}
$$

$$
\begin{aligned}
&= \gamma_c \cdot \frac{1}{\gamma_c^2} \cdot P\left[T_c(\boldsymbol{w}_{c,i-1}) - T_c(\bar{w}_{c,i-1})\right] + (1-\gamma_c) \cdot \frac{\mu_{\max}^2}{(1-\gamma_c)^2} \cdot \left\|(p^T \otimes I_M)\boldsymbol{z}_{i-1}\right\|^2 \\
&\overset{(b)}{\preceq} \gamma_c \cdot P\left[\boldsymbol{w}_{c,i-1} - \bar{w}_{c,i-1}\right] + \frac{\mu_{\max}^2}{1-\gamma_c} \cdot \left\|(p^T \otimes I_M)\boldsymbol{z}_{i-1}\right\|^2 \\
&= \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\|p\|_1^2\lambda_U^2} \cdot \left\|(p^T \otimes I_M)\boldsymbol{z}_{i-1}\right\|^2 \\
&= \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\|p\|_1^2\lambda_U^2} \cdot \left\|\sum_{k=1}^N p_k \boldsymbol{z}_{k,i-1}\right\|^2 \\
&= \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\|p\|_1^2\lambda_U^2} \cdot \Big(\sum_{l=1}^N p_l\Big)^2 \cdot \left\|\sum_{k=1}^N \frac{p_k}{\sum_{l=1}^N p_l} \boldsymbol{z}_{k,i-1}\right\|^2 \\
&\overset{(c)}{\leq} \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\|p\|_1^2\lambda_U^2} \cdot \Big(\sum_{l=1}^N p_l\Big)^2 \cdot \sum_{k=1}^N \frac{p_k}{\sum_{l=1}^N p_l} \|\boldsymbol{z}_{k,i-1}\|^2 \\
&= \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\|p\|_1^2\lambda_U^2} \cdot \|p\|_1 \cdot p^T \cdot P[\boldsymbol{z}_{i-1}] \\
&\overset{(d)}{\leq} \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\|p\|_1^2\lambda_U^2} \cdot \|p\|_1 \cdot p^T \cdot \lambda_U^2 \cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T \cdot P[\boldsymbol{w}_{e,i-1}] \\
&= \gamma_c \cdot \|\check{\boldsymbol{w}}_{c,i-1}\|^2 + \frac{\mu_{\max}}{\lambda_L - \frac{1}{2}\|p\|_1^2\lambda_U^2} \cdot \|p\|_1^2 \cdot \lambda_U^2 \cdot \|\bar{P}[\mathcal{A}_1^T\mathcal{U}_L]\|_\infty^2 \cdot \mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \qquad (6.234)
\end{aligned}
$$

where steps (a) and (c) apply Jensen's inequality to the convex function $\|\cdot\|^2$, step (b) uses property $P[T_c(x) - T_c(y)] \preceq \gamma_c^2 \cdot P[x-y]$ from (5.118) in Chapter 5, and step (d) substitutes the bound in (6.84).

Finally, we prove (6.208). With the block structure $\boldsymbol{v}_i = \text{col}\{\boldsymbol{v}_{1,i}, \ldots, \boldsymbol{v}_{N,i}\}$ defined previously, we have

$$
\begin{aligned}
&\mathbb{E}\big[\, \left\|(p^T \otimes I_M)\boldsymbol{v}_i\right\|^2 \big| \mathcal{F}_{i-1}\big] \\
&\quad = \mathbb{E}\Big[\Big\|\sum_{k=1}^N p_k \boldsymbol{v}_{k,i}\Big\|^2 \Big| \mathcal{F}_{i-1}\Big] \\
&\quad = \Big(\sum_{l=1}^N p_l\Big)^2 \cdot \mathbb{E}\Big[\Big\|\sum_{k=1}^N \frac{p_k}{\sum_{l=1}^N p_l} \boldsymbol{v}_{k,i}\Big\|^2 \Big| \mathcal{F}_{i-1}\Big]
\end{aligned}
$$

$$
\begin{aligned}
&\overset{(a)}{\leq} \Big(\sum_{l=1}^{N} p_l\Big)^2 \cdot \sum_{k=1}^{N} \frac{p_k}{\sum_{l=1}^{N} p_l} \mathbb{E}\Big[\Big\|\boldsymbol{v}_{k,i}\Big\|^2 |\mathcal{F}_{i-1}\Big] \\
&= \Big(\sum_{l=1}^{N} p_l\Big) \cdot \sum_{k=1}^{N} p_k \mathbb{E}\Big[\Big\|\boldsymbol{v}_{k,i}\Big\|^2 |\mathcal{F}_{i-1}\Big] \\
&= \|p\|_1 \cdot p^T \cdot \mathbb{E}\big\{P[\boldsymbol{v}_i]\big|\mathcal{F}_{i-1}\big\} \\
&\overset{(b)}{\leq} \|p\|_1 \cdot p^T \cdot \Big\{4\alpha \cdot \mathbb{1} \cdot P[\check{\boldsymbol{w}}_{c,i-1}] + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \cdot \mathbb{1}\mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \\
&\quad + \big[4\alpha \cdot (\|\tilde{w}_{c,0}\|^2 + \|w^o\|^2) + \sigma_v^2\big] \cdot \mathbb{1}\Big\} \\
&= 4\alpha\|p\|_1^2 \cdot P[\check{\boldsymbol{w}}_{c,i-1}] + 4\alpha \cdot \|\bar{P}[\mathcal{A}_1^T \mathcal{U}_L]\|_\infty^2 \cdot \|p\|_1^2 \cdot \mathbb{1}^T P[\boldsymbol{w}_{e,i-1}] \\
&\quad + 4\alpha\|\tilde{w}_{c,0}\|^2 \cdot \|p\|_1^2 + 4\alpha\|p\|_1^2 \cdot \|w^o\|^2 + \sigma_v^2 \cdot \|p\|_1^2
\end{aligned} \tag{6.235}
$$

where step (a) applies Jensen's inequality to the convex function $\|\cdot\|^2$, and step (b) substituting (6.86).

CHAPTER 7

Future Issues

In this dissertation, we addressed several important aspects of adaptation and learning over large-scale multi-agent systems. Based on the "data-distributed" and "model-distributed" nature of multi-agent systems, we formulated two forms of global cost functions: "sum-of-costs" and "cost-of-sum". Both diffusion and consensus strategies can be used to solve these problems. We addressed a critical question in multi-agent adaptation: whether the distributed strategies can approach the performance of a centralized strategy? The answer to the question was in the affirmative in the small step-size regime. That is, when the step-size is small enough, the learning behavior of each agent approaches that of the centralized strategy. This is an important conclusion, especially in the design and application of multi-agent system to big data problems, where "data-distributed" and "model-distributed" features are prevalent. In the following, we list three potential topics for future exploration:

- We have not investigated the information flow over the network when diffusive learning approaches are performed. That is, how much information should be shared in order to guarantee a certain performance level? It is useful to understand what kind of information is important and should be shared with neighbors. Viewing such problems from an information theoretic perspective might be a promising direction to gain deeper insights.

- One of the important motivations for distributed processing is that it allows the agents to maintain their private data and sub-models while sharing only the necessary information. Studying the the multi-agent system from the perspective of information security and analyzing the tradeoff between data privacy/security and learning performance are important future directions.

- We discussed the application of "cost-of-sum" to large-scale dictionary learning problems. Applying it to other machine learning and distributed decision making problems can be a useful direction. Besides, it is also interesting to explore other forms of global cost functions other than "sum-of-costs" and "cost-of-sum" in multi-agent adaptation and learning.

References

[1] A. Agarwal and J. Duchi. Distributed delayed stochastic optimization. In *Proc. Neural Information Processing Systems (NIPS)*, pages 873–881, Granada, Spain, Dec. 2011.

[2] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.*, 54(11):4311–4322, Nov. 2006.

[3] L. M. Aiello, G. Petkos, C. Martin, D. Corney, S. Papadopoulos, R. Skraba, A. Goker, I. Kompatsiaris, and A. Jaimes. Sensing trending topics in twitter. *IEEE Trans. Multimedia*, 15(6):1268–1282, Oct. 2013.

[4] T. M. Apostol. *Mathematical Analysis: A Modern Approach to Advanced Calculus*. Addison-Wesley Publishing Company, Inc., 1957.

[5] J. Arenas-Garcia, M. Martinez-Ramon, A. Navia-Vazquez, and A. R. Figueiras-Vidal. Plant identification via adaptive combination of transversal filters. *Signal Processing*, 86(9):2430–2438, 2006.

[6] F. Bach and E. Moulines. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In *Proc. of the 25th Annual Conf. Neural Inf. Process. Syst. (NIPS)*, pages 451–459, Granada, Spain, Dec. 2011.

[7] S. Barbarossa and G. Scutari. Bio-inspired sensor network design. *IEEE Signal Process. Mag.*, 24(3):26–35, May 2007.

[8] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[9] D. P. Bertsekas. A new class of incremental gradient methods for least squares problems. *SIAM J. Optim.*, 7(4):913–926, 1997.

[10] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 2nd edition, 1999.

[11] D. P. Bertsekas and J. N. Tsitsiklis. Gradient convergence in gradient methods with errors. *SIAM J. Optim.*, 10(3):627–642, 2000.

[12] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, Belmont, 1997.

[13] P. Bianchi, G. Fort, and W. Hachem. Performance of a distributed stochastic approximation algorithm. *IEEE Trans. Inf. Theory*, 59(11):7405–7418, Nov. 2013.

[14] P. Bianchi, G. Fort, W. Hachem, and J. Jakubowicz. Convergence of a distributed parameter estimator for sensor networks with local averaging of the estimates. In *Proc. IEEE ICASSP*, pages 3764–3767, Prague, Czech, May 2011.

[15] L. Bottou. Online learning and stochastic approximations. *On-line learning in neural networks*, 17(9), 1998.

[16] L. Bottou and Y. LeCun. Large scale online learning. In *Proc. Neural Information Processing Systems (NIPS)*, pages 1–8, Lake Tahoe, Nevada, Dec. 2003.

[17] L. Bottou and Y. LeCun. On-line learning for very large data sets. *Applied Stochastic Models in Business and Industry*, 21(2):137–151, 2005.

[18] S. Boyd, P. Diaconis, and L. Xiao. Fastest mixing Markov chain on a graph. *SIAM Rev.*, 46(4):667–689, Dec. 2004.

[19] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Trans. Inf. Theory*, 52(6):2508–2530, Jun. 2006.

[20] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[21] P. Braca, S. Marano, and V. Matta. Running consensus in wireless sensor networks. In *Proc. 11th IEEE Int. Conf. on Information Fusion*, pages 1–6, Cologne, Germany, June 2008.

[22] D. Cai, X. Wang, and X. He. Probabilistic dyadic data analysis with local and global consistency. In *Proceedings of the 26th International Conference on Machine Learning*, pages 105–112, Montreal, Canada, Jun. 2009.

[23] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed. A diffusion RLS scheme for distributed estimation over adaptive networks. In *Proc. IEEE Workshop on Signal Process. Advances Wireless Comm. (SPAWC)*, pages 1–5, Helsinki, Finland, June 2007.

[24] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed. Diffusion recursive least-squares for distributed estimation over adaptive networks. *IEEE Trans. Signal Process.*, 56(5):1865–1877, May 2008.

[25] F. S. Cattivelli and A. H. Sayed. Diffusion LMS algorithms with information exchange. In *Proc. Asilomar Conf. Signals, Syst. Comput.*, pages 251–255, Pacific Grove, CA, Nov. 2008.

[26] F. S. Cattivelli and A. H. Sayed. Diffusion LMS strategies for distributed estimation. *IEEE Trans. Signal Process.*, 58(3):1035–1048, Mar. 2010.

[27] F. S. Cattivelli and A. H. Sayed. Diffusion strategies for distributed Kalman filtering and smoothing. *IEEE Trans. Autom. Control*, 55(9):2069–2084, Sep. 2010.

[28] F. S. Cattivelli and A. H. Sayed. Modeling bird flight formations using diffusion adaptation. *IEEE Trans. Signal Process.*, 59(5):2038–2051, May 2011.

[29] F. S. Cattivelli and A. H. Sayed. Self-organization in bird flight formations using diffusion adaptation. In *Proc. 3rd International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, pages 49–52, Aruba, Dutch Antilles, Dec. 2009.

[30] N. Cesa-Bianchi, A. Conconi, and C. Gentile. On the generalization ability of on-line learning algorithms. *IEEE Trans. Inf. Theory*, 50(9):2050–2057, Sep. 2004.

[31] P. Chainais and C. Richard. Distributed dictionary learning over a sensor network. *arXiv:1304.3568*, Apr. 2013.

[32] P. Chainais and C. Richard. Learning a common dictionary over a sensor network. In *Proc. IEEE CAMSAP*, St Martin, French West Indies, Dec. 2013.

[33] T.-H. Chang, A. Nedic, and A. Scaglione. Distributed constrained optimization by consensus-based primal-dual perturbation method. *available as arXiv:1304.5590*, Apr. 2013.

[34] J. Chen and A. H. Sayed. Diffusion adaptation strategies for distributed optimization and learning over networks. *IEEE Trans. Signal Process.*, 60(8):4289–4305, Aug. 2012.

[35] J. Chen and A. H. Sayed. On the limiting behavior of distributed optimization strategies. In *Proc. Allerton Conf.*, pages 1535–1542, Monticello, IL, Oct. 2012.

[36] J. Chen and A. H. Sayed. Distributed Pareto optimization via diffusion adaptation. *IEEE J. Sel. Topics Signal Process.*, 7(2):205–220, Apr. 2013.

[37] J. Chen and A. H. Sayed. On the learning behavior of adaptive networks — Part I: Transient analysis. *Submitted for publication* [also available as arXiv:1312.7581], Dec. 2013.

[38] J. Chen and A. H. Sayed. On the learning behavior of adaptive networks — Part II: Performance analysis. *Submitted for publication* [also available as arXiv:1312.7580], Dec. 2013.

[39] J. Chen, Z. J. Towfic, and A. H. Sayed. Dictionary learning over distributed models. *Submitted for publication,* [also available as arXiv: 1402.1515], Feb. 2014.

[40] J. Chen, Z. J. Towfic, and A. H. Sayed. Online dictionary learning over distributed models. In *Proc. IEEE ICASSP*, pages 1–5, Florence, Italy, May 2014.

[41] Y. Chi, Y. Eldar, and R. Calderbank. PETRELS: Parallel subspace estimation and tracking by recursive least squares from partial observations. *IEEE Trans. Signal Process.*, 61(23):5947–5959, Dec. 2013.

[42] S. Chouvardas, K. Slavakis, and S. Theodoridis. Adaptive robust distributed learning in diffusion sensor networks. *IEEE Trans. Signal Process.*, 59(10):4692–4707, Oct. 2011.

[43] J. Dean, G. S. Corrado, R. Monga, K. Chen, M. Devin, Q. V. Le, M. Z. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, and A. Y. Ng. Large scale distributed deep networks. In *Proc. NIPS*, pages 1–9, Lake Tahoe, NV, Dec. 2012.

[44] O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao. Optimal distributed online prediction. In *Proc. International Conference on Machine Learning (ICML)*, pages 713–720, Bellevue, WA, Jun. 2011.

[45] O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao. Optimal distributed online prediction using mini-batches. *The Journal of Machine Learning Research*, 13:165–202, Jan. 2012.

[46] P. Di Lorenzo and S. Barbarossa. A bio-inspired swarming algorithm for decentralized access in cognitive radio. *IEEE Trans. Signal Process.*, 59(12):6160–6174, Dec. 2011.

[47] P. Di Lorenzo and A. H. Sayed. Sparse distributed learning based on diffusion adaptation. *IEEE Trans. Signal Process.*, 61(6):1419–1433, Mar. 2013.

[48] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione. Gossip algorithms for distributed signal processing. *Proc. IEEE*, 98(11):1847–1864, Nov. 2010.

[49] D. H. Dini and D. P. Mandic. Cooperative adaptive estimation of distributed noncircular complex signals. In *Proc. Asilomar Conf. Signals, Syst. and Comput.*, pages 1518–1522, Pacific Grove, CA, Nov. 2012.

[50] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inf. Theory Theory*, 41(3):613–627, May 1995.

[51] C. Eksin, P. Molavi, A. Ribeiro, and A. Jadbabaie. Learning in network games with incomplete information: Asymptotic analysis and tractable implementation of rational behavior. *IEEE Signal Process. Mag.*, 30(3):30–42, May 2013.

[52] C. Eksin and A. Ribeiro. Distributed network optimization with heuristic rational agents. *IEEE Trans. Signal Proc.*, 60(10):5396–5411, Oct. 2012.

[53] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.*, 15(12):3736–3745, Dec. 2006.

[54] A. Feuer and E. Weinstein. Convergence analysis of LMS filters with uncorrelated gaussian data. *IEEE Trans. Acoust., Speech, Signal Process.*, 33(1):222–230, Feb. 1985.

[55] M. Figueiredo and R. D. Nowak. A bound optimization approach to wavelet-based image deconvolution. In *Proc. IEEE ICIP*, volume 2, pages 779–782, Genoa, Italy, Sep. 2005.

[56] M. A. T. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak. Majorization–minimization algorithms for wavelet-based image restoration. *IEEE Trans. Image Process.*,, 16(12):2980–2991, Nov. 2007.

[57] W. A. Gardner. Learning characteristics of stochastic-gradient-descent algorithms: A general study, analysis, and critique. *Signal Process.*, 6(2):113–133, Apr. 1984.

[58] O. N. Gharehshiran, V. Krishnamurthy, and G. Yin. Distributed energy-aware diffusion least mean squares: Game-theoretic learning. *IEEE Journal Sel. Topics Signal Process.*, 7(5):821–836, Jun. 2013.

[59] G. H. Golub and C. F. Van Loan. *Matrix Computations (3rd Edition)*. Johns Hopkins University Press, 1996.

[60] G. J. Gordon. No-regret algorithms for online convex programs. *Advances in Neural Information Processing Systems*, 19:489, 2007.

[61] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.0 beta. `http://cvxr.com/cvx`, September 2013.

[62] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, Apr. 1970.

[63] J. H. van Hateren and A. van der Schaaf. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc. Biological Sciences*, 265(1394):359–366, Mar. 1998.

[64] S. Haykin. *Adaptive Filter Theory, 2nd Edition*. Prentice Hall, 2002.

[65] E. Hazan, A. Kalai, S. Kale, and A. Agarwal. Logarithmic regret algorithms for online convex optimization. In *Learning Theory*, pages 499–513. Springer, 2006.

[66] J.-B. Hiriart-Urruty and C. Lemaréchal. *Fundamentals of Convex Analysis*. Springer, 2001.

[67] Y.-W. Hong and A. Scaglione. A scalable synchronization protocol for large scale sensor networks and its applications. *IEEE J. Sel. Areas Comm.*, 23(5):1085–1099, May 2005.

[68] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1990.

[69] D. Hunter and K. Lange. A tutorial on MM algorithms. *The American Statistician*, 58(1):30–37, Feb. 2004.

[70] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. Autom. Control*, 48(6):988–1001, 2003.

[71] B. Johansson, T. Keviczky, M. Johansson, and K.H. Johansson. Subgradient methods and consensus algorithms for solving convex optimization problems. In *Proc. IEEE Conf. Decision and Control (CDC)*, pages 4185–4190, Cancun, Mexico, Dec. 2008. IEEE.

[72] S. Jones, R. Cavin III, and W. Reed. Analysis of error-gradient adaptive linear estimators for a class of stationary dependent processes. *IEEE Trans. Inf. Theory*, 28(2):318–329, Mar. 1982.

[73] T. Kailath, A. H. Sayed, and B. Hassibi. *Linear Estimation*. Prentice-Hall, Inc., 2000.

[74] S. M. Kakade and A. Tewari. On the generalization ability of online strongly convex programming algorithms. In *NIPS*, pages 801–808, Lake Tahoe, NV, Dec. 2008.

[75] S. Kar and J. M. F. Moura. Convergence rate analysis of distributed gossip (linear parameter) estimation: Fundamental limits and tradeoffs. *IEEE J. Sel. Topics. Signal Process.*, 5(4):674–690, Aug. 2011.

[76] S. Kar, J. M. F. Moura, and H. V. Poor. Distributed linear parameter estimation: Asymptotically efficient adaptive strategies. *SIAM Journal on Control and Optimization*, 51(3):2200–2229, 2013.

[77] S. Kar, J. M. F. Moura, and K. Ramanan. Distributed parameter estimation in sensor networks: Nonlinear observation models and imperfect communication. *IEEE Trans. Inf. Theory*, 58(6):3575–3605, Jun. 2012.

[78] S. P. Kasiviswanathan, H. Wangy, A. Banerjeey, and P. Melville. Online ℓ_1-dictionary learning with application to novel document detection. In *Proc. NIPS*, pages 2267–2275, Lake Tahoe, Nevada, Dec. 2012.

[79] S.M. Kay. *Fundamentals of Statistical Signal Processing, Volume 2: Detection Theory*. Prentice Hall PTR, 1998.

[80] E. Kreyszig. *Introductory Functional Analysis with Applications*. Wiley, NY, 1989.

[81] H. J. Kushner and G. Yin. *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, 2003.

[82] A. J. Laub. *Matrix Analysis for Scientists and Engineers*. SIAM, PA, 2005.

[83] M. Lee, H. Shen, J. Z. Huang, and J. S. Marron. Biclustering via sparse singular value decomposition. *Biometrics*, 66(4):1087–1095, Dec. 2010.

[84] S. Lee and A. Nedic. Distributed random projection algorithm for convex optimization. *IEEE Journal Sel. Topics Signal Process.*, 7(2):221–229, Apr. 2013.

[85] L. Li and J. A. Chambers. A new incremental affine projection-based adaptive algorithm for distributed networks. *Signal Processing*, 88(10):2599–2603, Oct. 2008.

[86] J. Liu, X.-C. Tai, H. Huang, and Z. Huan. A weighted dictionary learning model for denoising images corrupted by mixed noise. *IEEE Trans. Image Process.*, 22(3):1108–1120, Mar. 2013.

[87] C. G. Lopes and A. H. Sayed. Distributed processing over adaptive networks. In *Proc. Adaptive Sensor Array Processing Workshop*, MIT Lincoln Laboratory, MA, June 2006.

[88] C. G. Lopes and A. H. Sayed. Incremental adaptive strategies over distributed networks. *IEEE Trans. Signal Process.*, 55(8):4064–4077, Aug. 2007.

[89] C. G. Lopes and A. H. Sayed. Diffusion least-mean squares over adaptive networks: Formulation and performance analysis. *IEEE Trans. Signal Process.*, 56(7):3122–3136, Jul. 2008.

[90] C.G. Lopes and A.H. Sayed. Diffusion least-mean squares over adaptive networks. In *Proc. IEEE ICASSP*, volume 3, pages 917–920, Honolulu, HI, Apr. 2007.

[91] S. V. Macua, P. Belanovic, and S. Zazo. Diffusion gradient temporal difference for cooperative reinforcement learning with linear function approximation. In *Proc. IEEE International Workshop on Cognitive Information Process. (CIP)*, pages 1–6, Parador de Baiona, Spain, May 2012.

[92] J. Mairal. SPAMS: SPAMS (SPArse Modeling Software), version 2.4. `http://spams-devel.gforge.inria.fr/`, December 2013.

[93] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *The Journal of Machine Learning Research*, 11:19–60, Mar. 2010.

[94] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. In *Proc. NIPS*, pages 1033–1040, Lake Tahoe, Nevada, Dec. 2008.

[95] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Trans. Image Process.*, 17(1):53–69, Jan. 2008.

[96] A. Nedic and D. P. Bertsekas. Incremental subgradient methods for nondifferentiable optimization. *SIAM J. Optim.*, 12(1):109–138, 2001.

[97] A. Nedic and A. Ozdaglar. Distributed subgradient methods for multiagent optimization. *IEEE Trans. Autom. Control*, 54(1):48–61, 2009.

[98] A. Nedic and A. Ozdaglar. Cooperative distributed multi-agent optimization. *Convex Optimization in Signal Processing and Communications,* Y. Eldar and D. Palomar (Eds.), Cambridge University Press, pages 340–386, 2010.

[99] M. B. Nevelson and R. Z. Hasminskii. *Stochastic Approximation and Recursive Estimation*. American Mathematical Society, 1976.

[100] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proc. IEEE*, 95(1):215–233, Jan. 2007.

[101] H. Ouyang and A. Gray. Data-distributed weighted majority and online mirror descent. *Arxiv preprint arXiv:1105.2274*, 2011.

[102] D.P. Palomar and M. Chiang. A tutorial on decomposition methods for network utility maximization. *IEEE J. Sel. Areas Commun.*, 24(8):1439–1451, Aug. 2006.

[103] N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):123–231, 2013.

[104] G. Peyré. The numerical tours of signal processing - advanced computational signal and image processing. *IEEE Computing in Science and Engineering*, 13(4):94–97, Jul. 2011.

[105] B. Polyak. *Introduction to Optimization*. Optimization Software, NY, 1987.

[106] B. T. Polyak and A. B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

[107] J. B. Predd, S. R. Kulkarni, and H. V. Poor. A collaborative training algorithm for distributed learning. *IEEE Trans. Inf. Theory*, 55(4):1856–1871, Apr. 2009.

[108] M. G. Rabbat and R. D. Nowak. Quantized incremental algorithms for distributed optimization. *IEEE J. Sel. Areas Commun.*, 23(4):798–808, 2005.

[109] S. S. Ram, A. Nedic, and V. V. Veeravalli. Distributed stochastic subgradient projection algorithms for convex optimization. *J. Optim. Theory Appl.*, 147(3):516–545, 2010.

[110] W. Ren and R. W. Beard. Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Trans. Autom. Control*, 50(5):655–661, May 2005.

[111] D. Ruppert. Efficient estimations from a slowly convergent robbins-monro process. *Technical Report 781, Cornell University Operations Research and Industrial Engineering*, 1988.

[112] J. Sacks. Asymptotic distribution of stochastic approximation procedures. *The Annals of Mathematical Statistics*, 29(2):373–405, Jun. 1958.

[113] V. Saligrama, M. Alanyali, and O. Savas. Distributed detection in sensor networks with packet losses and finite capacity links. *IEEE Trans. Signal Proc.*, 54(11):4118–4132, Oct. 2006.

[114] S. Sardellitti, M. Giona, and S. Barbarossa. Fast distributed average consensus algorithms based on advection-diffusion processes. *IEEE Trans. Signal Process.*, 58(2):826–842, Feb. 2010.

[115] A. H. Sayed. Diffusion adaptation over networks. *in* Academic Press Library in Signal Processing, vol. 3, R. Chellapa and S. Theodoridis, *editors,* pp. 323–454, Elsevier, 2014 [also available online as arXiv:1205.4220v2, May 2012].

[116] A. H. Sayed. *Adaptive Filters*. Wiley, NJ, 2008.

[117] A. H. Sayed. Adaptive networks. *Proc. IEEE*, 102(4):460–497, Apr. 2014.

[118] A. H. Sayed and C. G. Lopes. Adaptive processing over distributed networks. *IEICE Trans. Fund. Electron., Commun. Comput. Sci.*, E90-A(8):1504–1510, Aug. 2007.

[119] A. H. Sayed, A. Tarighat, and N. Khajehnouri. Network-based wireless location: challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Process. Mag.*, 22(4):24–40, 2005.

[120] A. H. Sayed, S.-Y. Tu, J. Chen, X. Zhao, and Z. J. Towfic. Diffusion strategies for adaptation and learning over networks. *IEEE Signal Process. Mag.*, 30(3):155–171, May 2013.

[121] S. Shalev-Shwartz. *Online learning: Theory, algorithms, and applications*. PhD thesis, Hebrew University. Hebrew University. Hebrew University, 2007.

[122] S. Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

[123] H. Shen and J. Z. Huang. Sparse principal component analysis via regularized low rank matrix approximation. *Journal of Multivariate Analysis*, 99(6):1015–1034, Jul. 2008.

[124] M. T. M. Silva and V. H. Nascimento. Improving the tracking capability of adaptive filters via convex combination. *IEEE Trans. Signal Process.*, 56(7):3137–3149, 2008.

[125] K. Srivastava and A. Nedic. Distributed asynchronous constrained stochastic optimization. *IEEE J. Sel. Topics Signal Process.*, 5(4):772–790, Aug. 2011.

[126] S. S. Stankovic, M. S. Stankovic, and D. M. Stipanovic. Decentralized parameter estimation by consensus based stochastic approximation. *IEEE Trans. Autom. Control*, 56(3):531–543, Mar. 2011.

[127] N. Takahashi, I. Yamada, and A. H. Sayed. Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis. *IEEE Trans. Signal Process.*, 58(9):4795–4810, Sep. 2010.

[128] T. Takahashi, R. Tomioka, and K. Yamanishi. Discovering emerging topics in social streams via link-anomaly detection. *IEEE Trans. Knowl. Data Eng.*, 26(1):120–130, Jan. 2014.

[129] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Academic Press, 4th edition, 2008.

[130] S. Theodoridis, K. Slavakis, and I. Yamada. Adaptive learning in a world of projections. *IEEE Signal Process. Mag.*, 28(1):97–123, Jan. 2011.

[131] I. Tosic and P. Frossard. Dictionary learning. *IEEE Signal Processing Magazine*, 28(2):27–38, Mar. 2011.

[132] B. Touri, A. Nedic, and S. S. Ram. Asynchronous stochastic convex optimization over random networks: Error bounds. In *Proc. Inf. Theory and Appl. Workshop (ITA)*, pages 1–10, San Diego, CA, Jan. 2010. IEEE.

[133] Z. J. Towfic, J. Chen, and A. H. Sayed. Collaborative learning of mixture models using diffusion adaptation. In *Proc. IEEE Workshop on Mach. Learning Signal Process. (MLSP)*, pages 1–6, Beijing, China, Sept. 2011.

[134] Z. J. Towfic and A. H. Sayed. Adaptive penalty-based distributed stochastic convex optimization. *accepted for publication, IEEE Trans. Signal Process. [also available as arXiv:1312.4415]*, Dec. 2013.

[135] Z. J. Towfic and A. H. Sayed. Adaptive stochastic convex optimization over networks. In *Proc. Allerton Conf.*, pages 1–6, Monticello, IL, Oct. 2013.

[136] K. I. Tsianos, S. Lawlor, and M. G. Rabbat. Consensus-based distributed optimization: Practical issues and applications in large-scale machine learning. In *Proc. Annual Allerton Conference on Commun. Control and Comput.*, pages 1543–1550, Monticello, IL, Oct. 2012.

[137] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Trans. Autom. Control*, 31(9):803–812, 1986.

[138] S.-Y. Tu and A. H. Sayed. Mobile adaptive networks. *IEEE J. Sel. Topics. Signal Process.*, 5(4):649–664, Aug. 2011.

[139] S.-Y. Tu and A. H. Sayed. Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks. *IEEE Trans. Signal Process.*, 60(12):6217–6234, Dec. 2012.

[140] S-Y. Tu and A. H. Sayed. Mobile adaptive networks with self-organization abilities. In *Proc. 7th International Symposium on Wireless Communication Systems*, pages 379–383, York, United Kingdom, Sep. 2010.

[141] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson Jr. Stationary and nonstationary learning characterisitcs of the LMS adaptive filter. *Proc. IEEE*, 64(8):1151—1162, Aug. 1976.

[142] K. Yuan, Q. Ling, and W. Yin. On the convergence of decentralized gradient descent. *to appear in SIAM Journal on Optimization, [also avaiable as arXiv:1310.7063]*, 2014.

[143] C. Zach and M. Pollefeys. Practical methods for convex multi-view reconstruction. In *Proc. ECCV*, pages 354–367. Heraklion, Greece, Sep. 2010.

[144] X. Zhao and A. H. Sayed. Performance limits of LMS-based adaptive networks. In *Proc. IEEE ICASSP*, pages 3768–3771, Prague, Czech, May 2011.

[145] X. Zhao and A. H. Sayed. Probability distribution of steady-state errors and adaptation over networks. In *Proc. IEEE Statistical Signal Processing Workshop (SSP)*, pages 253–256, Nice, France, Jun. 2011.

[146] X. Zhao and A. H. Sayed. Performance limits for distributed estimation over LMS adaptive networks. *IEEE Trans. Signal Process.*, 60(10):5107–5124, Oct. 2012.

[147] X. Zhao, S.-Y. Tu, and A. H. Sayed. Diffusion adaptation over networks under imperfect information exchange and non-stationary data. *IEEE Trans. Signal Process.*, 60(7):3460–3475, July 2012.

[148] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proc. International Conference on Machine Learning (ICML)*, pages 928–936, Washington, DC, USA, Aug. 2003. School of Computer Science, Carnegie Mellon University.

[149] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286, Jan. 2006.