Robust Adaptation and Learning Over Networks

vom Fachbereich 18
Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung der Würde eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von
M.Sc. Sara Al-Sayed
geboren am 16.06.1986 in Kairo (Ägypten)

Referent:	Prof. Dr.-Ing. Abdelhak M. Zoubir
Korreferent:	Prof. Ali H. Sayed
Tag der Einreichung:	08.02.2016
Tag der mündlichen Prüfung:	28.04.2016

D 17
Darmstadt, 2016

Acknowledgments

I would like to thank my supervisor, Prof. Dr.-Ing. Abdelhak M. Zoubir, for granting me the space to develop both professionally and personally—I will continue to cherish his insight. I would also like to thank my co-supervisor, Prof. Ali H. Sayed, for his diligent criticism, which helped hone my skills as a researcher. I am especially grateful to him for hosting me on a three-month visit at the Adaptive Systems Laboratory at the University of California, Los Angeles (UCLA), where I got to experience his teaching and mentoring style firsthand. I would also like to extend my thanks to my doctoral examiners, Prof. Dr.-Ing. Anja Klein and Prof. Dr. Matthias Hollick.

My thanks also go to current and former members of and visitors to the Signal Processing Group for their perceptive feedback on countless occasions, as well as their camaraderie. I would especially like to single out Nevine Demitri for sharing her priceless knowledge of department workings inside out; Sahar Khawatmi for her reassuring, empathic visions; Stefan Leier for his reality checks; Michael Muma for engaging me in challenging collaborations; and Michael Fauss for stimulating discussions across our shared office, as our doctoral cycles synchronized. I also enjoyed interacting with Mouhammad Alhumaidi, Mark Ryan Balthasar, Patricia Binder, Raquel Fandos, Gökhan Gül, Jürgen Hahn, Lala Khadidja Hamaidi, Philipp Heidenreich, Roy Howard, Di Jin, Michael Lang, Michael Leigsnering, Zhihua Lu, Ahmed Moustafa, Ivana Perna, Simon Rosenkranz, Tim Schäck, Ann-Kathrin Seifert, Waqas Sharif, Adrian Šošić, Wassim Suleiman, Fiky Suratman, Gebremichael Teame, Freweyni Kidane Teklehaymanot, Christian Weiss, and Feng Yin. I am also grateful to Renate Koschella and Hauke Fath for their genuine support on matters administrative, technical, personal, and communal. At the Adaptive Systems Laboratory at UCLA, I sincerely benefited from group discussions with Hawraa Salami, Stefan Vlaski, Chengcheng Wang, Bicheng Ying, Chung-Kai Yu, and Kun Yuan.

I would also like to acknowledge the financial support of the projects Cocoon and HANDiCAMS, as well as MAKI and the department's equal opportunities office for contributing towards my travel expenses.

Last but not least, for nudging me along through the years, I am indebted to my parents, Hala and Mustapha, my extended family, SciMento team, and friends all over the world: Amira, Eman, Hala, Heba, Monika, Nadia, Omneya, Peter, Robin, Samira, and Yasmine. And, thank you, Michael, for your friendship and love!

Darmstadt, 01.06.2016

Kurzfassung

Im Zentrum dieser Dissertation stehen robuste adaptive Netzwerke. Es werden robuste Adaptionsstrategien entwickelt zur Lösung typischer Netzwerk-Inferenzprobleme, wie verteilte Schätzung und Detektion unter Impulsrauschen. Wie im Bereich der drahtlosen Kommunikation üblich, kann Impulsrauschen durch einen stochastischen Prozess beschrieben werden, dessen Realisierungen seltene, zufällige Samples enthalten, deren Amplitude deutlich größer ist, als unter nominalen Bedingungen zu erwarten wäre. Eine attraktive Eigenschaft derartiger robuster adaptiver Verfahren ist, dass weder für ihren Entwurf noch für ihren Betrieb eine exakte Kenntnis der Rauschverteilung nötig ist: Die robusten adaptiven Verfahren sind in der Lage letztere im laufenden Betrieb zu erlernen und ihre Parameter entsprechend anzupassen. Da die Verfahren nicht auf dem Einsatz einer zentralen Einheit (fusion center), sondern lediglich auf lokaler Interaktion der Knoten und einer verteilten Verarbeitung der Daten beruhen, erhöhen sie die Zuverlässigkeit des Netzwerks sowie dessen Ausfallsicherheit bei Knoten- und Verbindungsfehlern, Skalierbarkeit und Effizienz im Umgang mit Ressourcen. Verteilte, kooperative Datenverarbeitung findet Anwendung in vielen Bereichen, darunter drahtlose Sensornetzwerke zur Beobachtung von smart-homes, zur Umweltüberwachung, Qualitätssicherung und militärischen Aufklärung, sowie im Gesundheitswesen.

Da adaptive Systeme, die auf dem Prinzip der kleinsten mittleren quadratischen Abweichung beruhen, unter nicht gaußverteiltem Rauschen eine stark verminderte Leistung aufweisen, nutzen die in dieser Arbeit entwickelten robusten adaptiven Verfahren stattdessen nichtlineare Techniken der Datenverarbeitung und robuste Statistiken um die schädlichen Effekte des Impulsrauschens abzuschwächen. Zu diesem Zweck wird ein robuster adaptiver Filteralgorithmus entworfen, der eine adaptive, nichtlineare Fehlerkennlinie verwendet. Letztere wird dabei als konvexe Kombination zuvor festgelegter Basisfunktionen gewählt, wobei die Kombinationsgewichte zusammen mit der Schätzung der gesuchten Parameter so angepasst werden, dass in jeder Iteration die mittlere quadratische Abweichung von der optimalen Fehler-Nichtlinearität minimiert wird.

Anschließend wird ein robuster adaptiver Diffusionsalgorithmus vom Typ “adapt-then-combine” entwickelt, der eine Erweiterung seines allein operierenden Gegenstücks darstellt und sich zur Lösung von Schätzproblemen in Netzwerken mit von Impulsrauschen behafteten Beobachtungen eignet. Jeder Knoten des Netzwerks lässt dabei eine Kombination der Schätzungen seiner Nachbarn eine Iteration eines lokalen robusten adaptiven Filters durchlaufen, um so die Effekte der Datenverunreinigung

abzuschwächen. Dies führt zu einer besseren Gesamtleistung, die im stationären Zustand der von zentralisierten Systemen entspricht. Schließlich wird der robuste Diffusionsalgorithmus auf die Lösung verteilter Detektionsprobleme in Netzwerken mit Impulsrauschen erweitert. Die von dem robusten Algorithmus generierten Schätzungen werden dabei als Basis für den Entwurf robuster lokaler Detektoren verwendet, wobei die Form der Teststatistik und die Vorschriften zur Berechnung der Schwellenwerte durch eine Analyse der Dynamik des Algorithmus motiviert sind. Jeder Knoten im Netzwerk kooperiert mit seinen Nachbarn und nutzt deren Schätzungen zur Aktualisierung seines lokalen Detektors. Auf diese Weise verteilen sich die Informationen über das Ereignis von Interesse im Netzwerk, was zu einer höheren Detektionsleistung führt.

Mit Hilfe eines auf dem Prinzip der Energieerhaltung aufbauenden Verfahrens (energy conservation framework) wird das Verhalten der entwickelten Algorithmen im transienten und stationären Zustand analysiert. Zudem wird die Leistung des Algorithmus im Kontext der verteilten Detektion untersucht. Umfangreiche numerische Simulationen von Szenarien mit Impulsrauschen zeigen sowohl die Robustheit der vorgeschlagenen Verfahren im Vergleich zu aktuellen Algorithmen, als auch eine gute Übereinstimmung von Theorie und Praxis.

Abstract

This doctoral dissertation centers on robust adaptive networks. Robust adaptation strategies are devised to solve typical network inference tasks such as estimation and detection in a decentralized manner in the presence of impulsive contamination. Typical in wireless communication environments, an impulsive noise process can be described as one whose realizations contain sparse, random samples of amplitude much higher than nominally accounted for. An attractive feature that these robust adaptive strategies enjoy is that neither their development nor operation hinges on the availability of exact knowledge of the noise distribution: The robust adaptive strategies are capable of learning it on-the-fly and adapting their parameters accordingly. Forgoing data fusion centers, the network agents employing these strategies rely solely on local interactions and in-network processing to perform inference tasks, which renders networks more reliable, resilient to node and link failure, scalable, and resource efficient. Distributed cooperative processing finds applications in many areas including wireless sensor networks in smart-home, environmental, and industrial monitoring; healthcare; and military surveillance.

Since adaptive systems based on the mean-square-error criterion see their performance degrade in the presence of non-Gaussian noise, the robust adaptive strategies developed in this dissertation harness nonlinear data processing and robust statistics instead to mitigate the detrimental effects of impulsive noise. To this end, a robust adaptive filtering algorithm is developed that employs an adaptive error nonlinearity. The error nonlinearity is chosen to be a convex combination of preselected basis functions where the combination coefficients are adapted jointly with the estimate of the parameter of interest such that the mean-square-error relative to the optimal error nonlinearity is minimized in each iteration.

Then, a robust diffusion adaptation algorithm of the adapt-then-combine variety is developed as an extension of its stand-alone counterpart for distributed estimation over networks where the measurements may be corrupted by impulsive noise. Each node in the network runs a combination of its neighbors' estimates through one iteration of a local robust adaptive filter update to ameliorate the effects of contamination, leading to better overall network performance matching that of a centralized strategy at steady-state.

Finally, the robust diffusion adaptation algorithm is extended further to solve the problem of distributed detection over adaptive networks where the measurements may be corrupted by impulsive noise. The estimates generated by the robust algorithm

are used as basis for the design of robust local detectors, where the form of the test-statistics and the rule for the computation of the detection thresholds are motivated by the analysis of the algorithm dynamics. Each node in the network cooperates with its neighbors, utilizing their estimates, to update its local detector. Effectively, information pertaining to the event of interest percolates across the network, leading to enhanced detection performance.

The transient and steady-state behavior of the developed algorithms are analyzed in the mean and mean-square sense using the energy conservation framework. The performance of the algorithm is also examined in the context of distributed detection. Performance is validated extensively through numerical simulations in an impulsive noise scenario, revealing the robustness of the proposed strategies in comparison with state-of-the-art algorithms as well as good agreement between theory and practice.

Contents

1 Introduction **1**
1.1 Contributions . . . 2
1.2 Publications . . . 3
1.3 Dissertation Overview . . . 4

2 Preliminaries and State-of-the-Art **5**
2.1 Notation . . . 5
2.2 Adaptive Filtering With Error Nonlinearities . . . 5
2.2.1 Data Model and Estimation Problem . . . 5
2.2.2 Adaptive Filtering Algorithms . . . 7
2.2.3 Primer on Performance Analysis . . . 17
2.2.4 MSE-Optimal Error Nonlinearity . . . 23
2.3 Robust Estimation . . . 30
2.3.1 M-estimation . . . 31
2.3.2 Robust Adaptive Filtering . . . 35
2.4 Distributed Adaptation and Learning Over Networks . . . 36
2.4.1 Network Model . . . 36
2.4.2 Data Model and Problem Formulation . . . 37
2.4.3 Diffusion Adaptation Algorithms . . . 38

3 Robust Adaptation for Single Agents **43**
3.1 Robust Adaptive Filtering . . . 43
3.1.1 Data Model and Problem Formulation . . . 43
3.1.2 Joint Parameter Adaptation . . . 47
3.2 Performance Analysis . . . 49
3.2.1 Mean Behavior . . . 51
3.2.2 Variance Relation . . . 53
3.2.3 Steady-State Performance . . . 56
3.2.4 Mean-Square Behavior . . . 59
3.2.5 Mean-Square Stability . . . 61
3.2.6 Algorithm Complexity . . . 61
3.3 Simulation Results . . . 61
3.4 Conclusion . . . 66

4 Robust Adaptive Estimation Over Networks **77**
4.1 Distributed Estimation . . . 77
4.1.1 Data Model and Problem Formulation . . . 77

4.1.2 Robust Diffusion Adaptation 79
4.2 Performance of Robust Diffusion Estimation Algorithm 80
4.2.1 Error Recursions 81
4.2.2 Mean Performance 82
4.2.3 Mean-Square Performance 83
4.2.4 Comparison With the Diffusion LMS Algorithm 91
4.3 Simulation Results 92
4.4 Conclusion 95

5 Robust Adaptive Detection Over Networks 97
5.1 Distributed Detection 97
5.1.1 Data Model and Problem Formulation 97
5.1.2 Neyman–Pearson-Based Detection 98
5.1.3 Robust Diffusion Detection Algorithm 99
5.2 Performance of Robust Diffusion Detection Algorithm 102
5.2.1 Error Recursions 103
5.2.2 Mean Performance 103
5.2.3 Mean-Square Performance 104
5.2.4 Detection Performance 106
5.3 Simulation Results 108
5.4 Conclusion 108

6 Summary, Conclusions, and Future Research Directions 111
6.1 Summary and Conclusions 111
6.2 Future Research Directions 112

Appendix 115
A.1 Price's Theorem 115
A.2 Lower Bound on MSD in (2.62) 116
A.3 Derivation of Optimal Error Nonlinearity 117
A.4 Condition (3.13) 118
A.5 Proof of Lemma 1 119
A.6 Proof of Lemma 2 121
A.7 Derivation of $Q_{k,i}$ in (5.31) 122
A.8 Proof of Lemma 3 122
A.9 Derivation of (5.65)—Recursion for $\widehat{R}^{A=I}_{\widetilde{w}_{k,i}}$ 123

List of Abbreviations 125

List of Notation and Symbols 127

References **131**

Lebenslauf **139**

Chapter 1

Introduction

Many a modern-day data processing application is characterized by the data being spatially dispersed among networked agents and requiring decentralized processing, be it for storage or privacy constraints. Wireless sensor networks are a prime example, finding applications in many areas including smart-home, environmental, and industrial monitoring; healthcare; and military surveillance [CES04]. The nodes in the network are typically quite simple of design and have limited storage, communication, and computation capabilities. Networks of the sort are inundated with statistical data processing tasks, such as estimation, detection, filtering, smoothing, clustering, and classification [BSD13, VV11]. Distributed cooperative processing presents itself as a viable paradigm for the resolution of these tasks. Forgoing data fusion centers and relying solely on local interactions and in-network processing to perform these tasks renders networks more reliable, resilient to node and link failure, scalable, and resource efficient [STC+13]. Often, the data is of a time-varying nature. Hence, nodes need to learn and track underlying changes on-the-fly and adapt their collective behavior accordingly, in mimicry of biological systems [BS07], where simple local behavioral rules lead to the emergence of organized global behavior [CDF+03]. In this respect, adaptive networked engineering systems manifest dynamic cognitive behavior [Hay12].

Noise in engineering jargon refers to intrinsic or extrinsic random fluctuations disrupting the normal operation of the system[1]. Wireless environment surveys have shown the noise to be often impulsive in nature. An impulsive noise process can be described as one whose realizations contain sparse, random samples of amplitude much higher than nominally accounted for. Impulsive noise may be natural, due to atmospheric phenomena, or man-made, due to either electric machinery present in the operation environment, or multipath telecommunications signals [BKR97, Mid99, ZKCM12, ZB02].

Statistical signal processing techniques based on second-order statistics of the data are a relic of the era of linear systems, reflecting an excusable bias towards Gaussianity [EP06]. Nevertheless, general error criteria have been considered for a long time, albeit in the context of Gaussian processes in early works [She58, Bro62, Zak64, Ger69]. General error criteria were shown to be more fitting for non-Gaussian noise scenarios [SN93]. In practice, exact knowledge of the noise distribution is unavailable. Robust statistics offers an attractive framework to deal with noise uncertainty and departure

[1] See [Coh05] for a historical account of noise and the birth of the study of stochastic processes.

from nominal system design assumptions [HR09, HRRS86], paving the way for engineering robust signal processing and communications systems [KP85, KP83].

This dissertation focuses on the topic of robust adaptive networks, with the aim of devising robust adaptation strategies to solve typical network inference tasks in the presence of impulsive contamination.

1.1 Contributions

The following is a list of the original contributions presented in this dissertation:

- A robust adaptive filtering algorithm of the least-mean-squares (LMS) type was developed that employs an adaptive error nonlinearity. The error nonlinearity was chosen to be a convex combination of preselected basis functions where the combination coefficients are adapted jointly with the estimate of the parameter of interest such that the mean-square-error (MSE) relative to the optimal error nonlinearity is minimized in each iteration. While knowledge of the nature of the noise, impulsive or otherwise, serves to guide the choice of basis functions, exact distributional knowledge is not required since the robust algorithm is capable of learning it on-the-fly and adapting its parameters accordingly. The transient and steady-state behavior of the robust adaptive filtering algorithm were analyzed in the mean and mean-square sense using the energy conservation framework. The computational complexity was summarized. The performance of the algorithm was validated extensively in numerical simulations.

- A robust diffusion adaptation algorithm of the adapt-then-combine (ATC) variety was developed as a natural extension of its stand-alone counterpart for distributed estimation over networks where the measurements may be corrupted by impulsive noise. Each node in the network runs a combination of its neighbors' estimates through one iteration of a local robust adaptive filter update to ameliorate the effects of contamination. The robust adaptive update rule again employs an adaptive error nonlinearity that is a convex combination of preselected basis functions where the combination coefficients are adapted jointly with the estimate of the parameter of interest such that the MSE relative to the local optimal error nonlinearity is minimized in each iteration. The transient and steady-state behavior of the algorithm

were analyzed in the mean and mean-square sense using the energy conservation framework and performance was validated through numerical simulations.

- The robust diffusion adaptation algorithm developed was extended further to solve the problem of distributed detection over adaptive networks where the measurements may be corrupted by impulsive noise. The estimates generated by the robust algorithm are used as basis for the design of robust local detectors, where the form of the test-statistics and the rule for the computation of the detection thresholds were motivated by the analysis of the algorithm dynamics. The transient behavior of the algorithm was analyzed using the energy conservation framework. The detection performance was also established. The performance of the algorithm was validated through numerical simulations.

1.2 Publications

The period of doctoral candidacy has culminated in the following publications:

Internationally Refereed Journal Articles

- S. Al-Sayed, A. M. Zoubir, and A. H. Sayed, "Robust distributed estimation by networked agents," submitted to *IEEE Trans. Signal Process.*, 2016.
- S. Al-Sayed, A. M. Zoubir, and A. H. Sayed, "Robust adaptation in impulsive noise," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2851–2865, Jun. 2016.

Internationally Refereed Conference Papers

- S. Al-Sayed, A. M. Zoubir, and A. H. Sayed, "Robust distributed detection over adaptive diffusion networks," *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Florence, Italy, May 2014, pp. 7233–7237.
- S. Al-Sayed, A. M. Zoubir, and A. H. Sayed, "An optimal error nonlinearity for robust adaptation against impulsive noise," *Proc. IEEE Workshop Sig. Proc. Adv. Wireless Comm. (SPAWC)*, Darmstadt, Germany, Jun. 2013, pp. 415–419.

International Conference Papers

- M. R. Balthasar, S. Al-Sayed, S. Leier, and A. M. Zoubir, "Optimal area coverage in autonomous sensor networks," *Proc. Int. Conf. Underwater Acoust. (UA2014)* (invited paper), Rhodes, Greece, Jun. 2014, pp. 431–438.

1.3 Dissertation Overview

Ch. 2 fixes the notation used throughout the text and presents an overview of the ideas central to the dissertation: adaptive filtering with error nonlinearities, robust estimation, and distributed adaptation over networks. The exposition is rather lengthy, and serves to motivate and contextualize the contributions in this dissertation, string together the central ideas, and highlight the state-of-the-art in its subject area.

In Ch. 3, the first contribution is presented. The single-agent robust adaptive filtering algorithm is developed, analyzed, and simulated.

In Ch. 4, the second contribution is presented. The robust diffusion adaptation algorithm for distributed estimation over networks is developed, analyzed, and simulated.

In Ch. 5, the third contribution is presented. The robust diffusion adaptation algorithm for distributed detection over networks is developed, analyzed, and simulated.

A summary is presented and conclusions are drawn in Ch. 6. Directions for future research are also outlined.

Chapter 2
Preliminaries and State-of-the-Art

In addition to fixing the notation used throughout the text, this chapter presents an overview of the ideas central to the dissertation: adaptive filtering with error nonlinearities (Sec. 2.2), robust estimation (Sec. 2.3), and distributed adaptation over networks (Sec. 2.4). The exposition is rather lengthy, and serves to motivate and contextualize the contributions in this dissertation, string together the central ideas, and highlight the state-of-the-art in its subject area.

2.1 Notation

Lowercase letters are reserved for scalars and vectors, uppercase for matrices; boldface font is reserved for random variables, and normal font for deterministic variables. The time index appears in parenthesis for scalars, and in the subscript for vectors and matrices. All vectors are column vectors. The single exception to this rule is the row regression vector, for convenience of presentation. Transposition, inversion, pseudoinversion, and the trace and gradient operators are denoted by $(\cdot)^T$, $(\cdot)^{-1}$, $(\cdot)^\dagger$, $\mathrm{Tr}(\cdot)$, and ∇_x, respectively; and the Euclidean norm is denoted by $\|\cdot\|$. Expectation is denoted by $\mathbb{E}$. The notation $\mathbb{1}$ and I denotes the all-one vector and identity matrix of appropriate sizes, respectively; if the size is not clear from the context, it will appear explicitly as a subscript. The Kronecker product between two matrices is denoted by $\otimes$. The operator $\mathrm{col}\{\cdot\}$ stacks its arguments vertically; the operator $\mathrm{diag}\{\cdot\}$ is used bidirectionally to either form a diagonal matrix from its arguments, or recover the vector comprising the diagonal of its matrix argument; and the operator $\mathrm{vec}(\cdot)$ stacks the columns of its matrix argument on top of one another in a vector, $\mathrm{vec}^{-1}(\cdot)$ being the inverse operation. A list of notation and main symbols used throughout the text can be found at the end of the dissertation.

2.2 Adaptive Filtering With Error Nonlinearities

2.2.1 Data Model and Estimation Problem

At each time index $i \geq 0$, a noisy real-valued scalar measurement $d(i)$ is made of an unknown deterministic real-valued $M \times 1$ parameter vector w^o. The measurements are

related to the parameter via a stochastic linear regression model of the form:

$$\boldsymbol{d}(i) = \boldsymbol{u}_i w^o + \boldsymbol{v}(i), \quad i \geq 0. \tag{2.1}$$

The real-valued scalar measurements $\{d(i)\}$ are realizations of the random process $\{\boldsymbol{d}(i)\}$ specified in (2.1), where the $\{\boldsymbol{u}_i\}$ represent known real-valued row regression vectors, or regressors, of size M. The joint random process $\{\boldsymbol{d}(i), \boldsymbol{u}_i\}$ is assumed to be zero-mean wide-sense stationary. The second-order moments are denoted by:

$$\sigma_d^2 \triangleq \mathbb{E}\, \boldsymbol{d}^2(i) \quad \text{(scalar)} \tag{2.2}$$

$$R_u \triangleq \mathbb{E}\, \boldsymbol{u}_i^T \boldsymbol{u}_i \quad (M \times M) \tag{2.3}$$

$$r_{du} \triangleq \mathbb{E}\, \boldsymbol{d}(i) \boldsymbol{u}_i^T \quad (M \times 1) \tag{2.4}$$

For convenience, the covariance matrix R_u is assumed to be positive definite, i.e., $R_u > 0$. Hence, it is invertible. The sequence $\{\boldsymbol{v}(i)\}$ represents a real-valued scalar zero-mean independent and identically distributed (i.i.d.) process with variance:

$$\sigma_v^2 \triangleq \mathbb{E}\, \boldsymbol{v}^2(i). \tag{2.5}$$

The random variables $\boldsymbol{u}_i$ and $\boldsymbol{v}(j)$ are assumed to be independent for all i and j.

Given realizations $\{d(i), u_i\}$, the objective is to estimate the parameter vector w^o, subject to some error criterion. The parameter vector w^o is going to be referred to in the body of the text interchangeably as the weight vector since its entries weight those of the regressor u_i. Linear relations of the form (2.1) can be used to model both autoregressive (AR) and moving-average (MA) processes [Sch91]. In order to see this, consider realizations $\{d(i)\}$ of a zero-mean AR process of order M, $\{\boldsymbol{d}(i)\}$, which is represented by the following relation:

$$\boldsymbol{d}(i) = \sum_{m=1}^{M} a_m \boldsymbol{d}(i-m) + \boldsymbol{v}(i), \quad i \geq 0 \tag{2.6}$$

where $\{a_m\}$ are the model parameters and $\{\boldsymbol{v}(i)\}$ represents a zero-mean i.i.d. process with variance σ_v^2, with $\boldsymbol{v}(i)$ assumed independent of past outputs $\{\boldsymbol{d}(i-m) | m \geq 1\}$. By collecting the M most recently observed outputs up to time index $i-1$ into a $1 \times M$ regression vector $\boldsymbol{u}_i$:

$$\boldsymbol{u}_i \triangleq [\boldsymbol{d}(i-1) \quad \boldsymbol{d}(i-2) \quad \ldots \quad \boldsymbol{d}(i-M)] \tag{2.7}$$

and the model parameters $\{a_m\}$ into an $M \times 1$ weight vector w^o:

$$w^o \triangleq \operatorname{col}\{a_1, a_2, \ldots, a_M\} \tag{2.8}$$

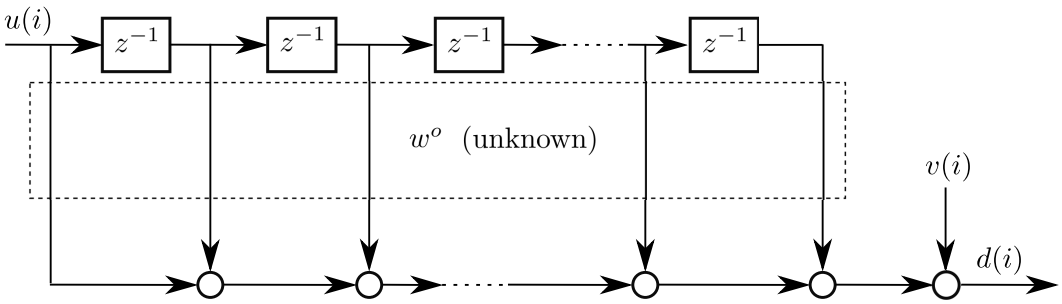


Figure 2.1: Finite-impulse-response (FIR) model identification problem.

it can be seen that (2.6) can be written equivalently as (2.1), where one wishes to estimate w^o given data $\{d(i), u_i\}$. Speech and audio signals can be modeled as AR processes in linear predictive coding applications [Hay01], for instance. On the other hand, an MA process is described as follows. Given an input sequence $\{u(i)\}$ that is a realization of a zero-mean wide-sense stationary random process $\{\boldsymbol{u}(i)\}$, the output of an MA model of order M in response to the input sequence is represented by the following relation:

$$\boldsymbol{d}(i) = \sum_{m=0}^{M-1} b_m \boldsymbol{u}(i-m) + \boldsymbol{v}(i), \quad i \geq 0 \tag{2.9}$$

where $\{b_m\}$ are the model parameters and $\{\boldsymbol{v}(i)\}$ represents a zero-mean i.i.d. process with variance σ_v^2, with $\boldsymbol{v}(i)$ assumed independent of the input $\boldsymbol{u}(j)$ for all i and j. By collecting the M most recent inputs up to time index i into a $1 \times M$ regression vector $\boldsymbol{u}_i$:

$$\boldsymbol{u}_i = [\boldsymbol{u}(i) \quad \boldsymbol{u}(i-1) \quad \ldots \quad \boldsymbol{u}(i-M+1)] \tag{2.10}$$

and the model parameters $\{b_m\}$ into an $M \times 1$ weight vector w^o:

$$w^o \triangleq \operatorname{col}\{b_0, b_1, \ldots, b_{M-1}\} \tag{2.11}$$

it can be seen that (2.9) can be written equivalently as (2.1), where one wishes to estimate w^o given data $\{d(i), u_i\}$. Data satisfying the model (2.9) arise essentially in finite-impulse-response (FIR) model identification problems (see Fig. 2.1) in diverse applications such as communication channel estimation [Hay01], line or acoustic echo cancellation [Kel70, BGM+01], and noise cancellation [WGM+75].

2.2.2 Adaptive Filtering Algorithms

In order to estimate the weight vector w^o, the data $\{d(i), u_i\}$ can be fed into an adaptive FIR filter, one with adjustable coefficients or tap weights that are updated recursively via a so-called *stochastic-gradient algorithm*. The algorithm processes the

data $\{d(i), u_i\}$ in real-time and outputs a sequence of weight estimates $\{w_i\}$, which constitutes the values assumed by the time-varying vector of filter coefficients. If the algorithm parameters are chosen appropriately, then, given sufficient time, the weight estimates $\{w_i\}$ eventually converge to w^o. Several stochastic-gradient algorithms have been developed to solve this estimation problem, to varying degrees of estimation accuracy and computational complexity [Say03, Hay13, WS85]. Of interest here, however, is the class of algorithms whose update equations take the following form. For $i \geq 0$, starting from some initial condition w_{-1}:

$$e(i) = d(i) - u_i w_{i-1} \tag{2.12a}$$

$$w_i = w_{i-1} + \mu u_i^T h\big(e(i)\big) \tag{2.12b}$$

where $h(\cdot)$ is an error nonlinearity whose interpretation is going to be elaborated upon shortly, and μ is a positive step-size parameter chosen in such a way as to ensure stability. Fig. 2.2 illustrates the structure suggested by the discussion, in the context of the FIR model identification problem from Sec. 2.2.1 and Fig. 2.1. The quantity $e(i)$, referred to as the output estimation error, represents the offset at time index i between the measured system output or reference signal $d(i)$ and the output of the adaptive filter $\hat{d}(i) = u_i w_{i-1}$. This error signal is then used to update the adaptive filter coefficients or weight estimate from w_{i-1} to w_i, through an error nonlinearity $h(\cdot)$. If the algorithm parameters are chosen appropriately, then as $i \to \infty$, the weight estimate w_i tends to w^o, the error signal $e(i)$ to the noise signal $v(i)$, and the adaptive filter output assumes values close to the system output. Effectively, the adaptive filter can be said to behave similarly to the system being probed, insofar as the adaptive filter has at least as many taps as the FIR model. Indeed, rigorous analysis of the class of algorithms described by (2.12) would establish the foregoing argument, as will subsequently be shown. First, however, the form (2.12) for adaptive filters with error nonlinearities will be motivated in an estimation-theoretic context [Say03].

Let $\boldsymbol{d}$ be a zero-mean real-valued scalar random variable with variance σ_d^2:

$$\mathbb{E}\,\boldsymbol{d} = 0, \quad \sigma_d^2 = \mathbb{E}\,\boldsymbol{d}^2 \tag{2.13}$$

and let $\boldsymbol{u}$ be a zero-mean real-valued random vector of size $1 \times M$ with a positive-definite covariance matrix R_u:

$$\mathbb{E}\,\boldsymbol{u} = 0, \quad R_u = \mathbb{E}\,\boldsymbol{u}^T \boldsymbol{u} \tag{2.14}$$

Moreover, let the $M \times 1$ cross-covariance vector of $\boldsymbol{d}$ and $\boldsymbol{u}$ be denoted by $r_{du} \triangleq \mathbb{E}\,\boldsymbol{d}\boldsymbol{u}^T$. Consider the problem of estimating $\boldsymbol{d}$ from $\boldsymbol{u}$ in a linear fashion, subject to some error criterion, by means of the following estimator:

$$\hat{\boldsymbol{d}} = \boldsymbol{u} w^o \tag{2.15}$$

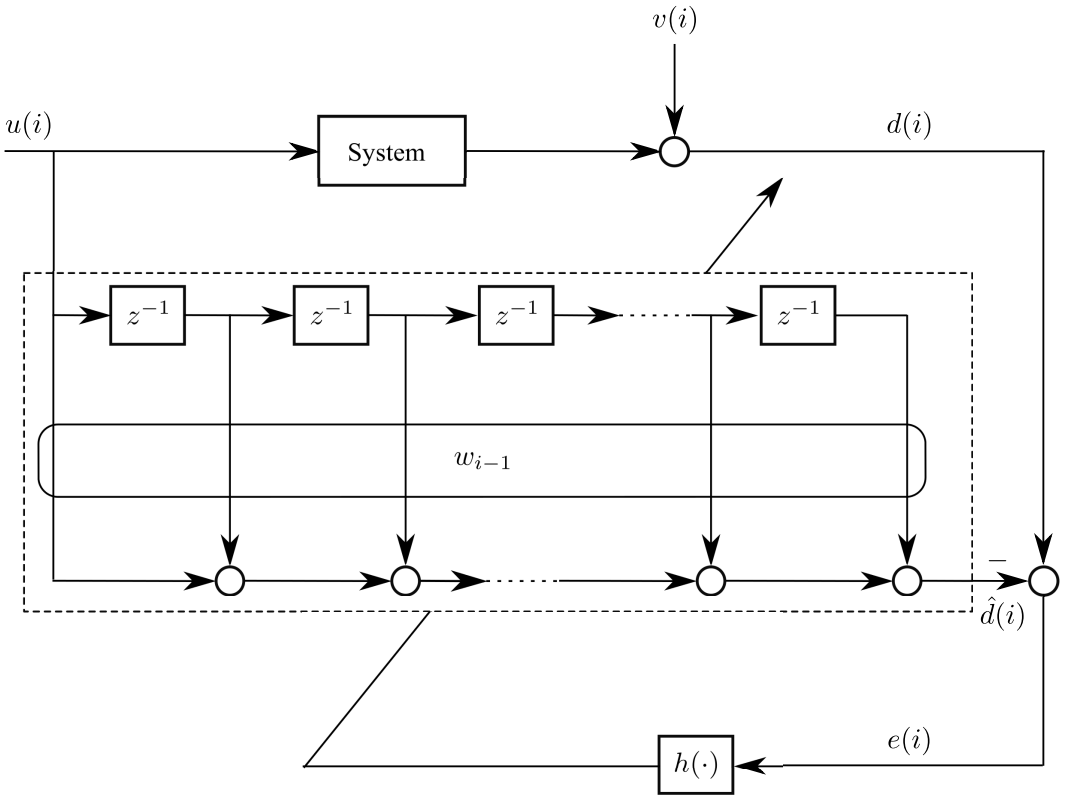


Figure 2.2: Structure for adaptive finite-impulse-response (FIR) model identification.

where w^o is the $M \times 1$ weight vector that minimizes the designated error criterion. To this end, define a differentiable, convex loss function $\rho\colon \mathbb{R} \to \mathbb{R}$. The convexity of ρ implies that it has global minima and no local minima. The weight vector w^o is chosen as the solution to the following optimization problem:

$$\min_{w} J(w) \triangleq \mathbb{E}\, \rho(\boldsymbol{d} - \boldsymbol{u}w) \tag{2.16}$$

where $J\colon \mathbb{R}^M \to \mathbb{R}$ denotes the cost function to be minimized, and the expectation is evaluated over the joint multivariate distribution of $\boldsymbol{d}$ and $\boldsymbol{u}$, with probability density function (pdf) $f_{\boldsymbol{d},\boldsymbol{u}}(d, u)$. The argument of ρ above, $\boldsymbol{d} - \boldsymbol{u}w$, is to be interpreted as the error in estimating $\boldsymbol{d}$ as $\hat{\boldsymbol{d}} = \boldsymbol{u}w$. Different choices of ρ lead to different error criteria with respect to w.

Examples:

1. The choice $\rho(x) = x^2$ leads to the mean-square-error (MSE) criterion:

$$J(w) = \mathbb{E}\,(\boldsymbol{d} - \boldsymbol{u}w)^2\,. \tag{2.17}$$

2. The choice $\rho(x) = |x|$ leads to the mean-absolute-error criterion:

$$J(w) = \mathbb{E}\,|\boldsymbol{d} - \boldsymbol{u}w|\,. \tag{2.18}$$

Even though $\rho(x)$ in this case is not differentiable at $x = 0$, one can invoke instead the generalized derivative such that $\rho'(x) = \text{sign}(x)$ [Cla90], where

$$\text{sign}(x) \triangleq \begin{cases} +1, & \text{if} \quad x > 0 \\ 0, & \text{if} \quad x = 0 \\ -1, & \text{if} \quad x < 0 \end{cases}. \tag{2.19}$$

□

In the minimum mean-square-error (MMSE) estimation problem ($\rho(x) = x^2$), the optimal weight vector w^o that solves (2.16) can be obtained in closed form in terms of the moments R_u and r_{du}, namely, as the solution to the so-called normal equations:

$$R_u w^o = r_{du}. \tag{2.20}$$

Since the covariance matrix R_u is assumed to be positive definite, the solution w^o can be obtained as $w^o = R_u^{-1} r_{du}$. Matrix inversion is a costly operation computationally, however, and challenging for ill-conditioned matrices R_u. Alternatively, one can resort to iterative procedures to solve the MMSE estimation problem. As a matter of fact, it is generally not possible to solve (2.16) in closed form, making iterative procedures a viable solution alternative in the general case. Starting from an initial guess for w^o, w_{-1}, such procedures generate iterates $\{w_i, i \geq 0\}$ recursively until convergence to w^o. In particular, the update equation of a *steepest-descent method* has the following recursive form:

$$w_i = w_{i-1} - \mu\,[\nabla_w J(w_{i-1})]^T\,, \quad i \geq 0, \quad w_{-1} = \text{initial condition} \tag{2.21}$$

where $\nabla_w J(w)$, a row vector, denotes the gradient of the cost function $J(w)$ with respect to the weight vector w; and μ is a positive step-size parameter, chosen small enough to ensure stability. Since

$$[\nabla_w J(w_{i-1})]^T = \mathbb{E}\left(-\boldsymbol{u}^T h(\boldsymbol{d} - \boldsymbol{u}w_{i-1})\right) \tag{2.22}$$

where h denotes the derivative of ρ: $h = \rho'$, the steepest-descent method for solving (2.16) is ultimately given by

$$w_i = w_{i-1} + \mu\,\mathbb{E}\left(\boldsymbol{u}^T h(\boldsymbol{d} - \boldsymbol{u}w_{i-1})\right), \quad i \geq 0, \quad w_{-1} = \text{initial condition} \tag{2.23}$$

Some obvious difficulties arise when attempting to implement the steepest-descent method (2.23). Firstly, exact knowledge of the moment $\mathbb{E}\left(\boldsymbol{u}^T h(\boldsymbol{d} - \boldsymbol{u}w)\right)$ for $w =$

$w_{-1}, w_0, w_1, \ldots$ is required for implementation, necessitating knowledge of the joint pdf $f_{\boldsymbol{d},\boldsymbol{u}}(d,u)$. This knowledge is rarely available in practice, or the moment itself might be difficult to calculate for general nonlinearities h. The MMSE case is one notable exception that will be addressed shortly. Secondly, even when the required statistical knowledge is available, the statistics may vary with time, which implies that the optimal solution w^o will vary accordingly. It is necessary in this case to have a mechanism in place to learn those statistics and track them as they change. In order to overcome the aforementioned difficulties, the gradient vector in the steepest-descent update equation (2.23) can be replaced by some stochastic approximation of the moment in question based on streaming data $\{d(i), u_i\}$, satisfying the model (2.1), for example. The resulting methods are generically referred to as stochastic-gradient algorithms since the estimates employed for gradient approximation inevitably introduce random fluctuations into the procedure that are referred to as gradient noise [Say03]. Some approximations are surely better than others, and the resulting performance degradation can be quantified analytically as basis for comparison of the different stochastic-gradient algorithms with the original steepest-descent method, as well as with one another—this will be the subject of Sec. 2.2.3. In addition to providing estimates for the gradient vector, relying on streaming data equips the iterative procedure with learning and tracking capabilities so that it can truly adapt to drifts in the underlying statistics. A straightforward approximation of the transposed gradient vector (2.22) at time index i can be obtained by dropping the expectation operator and employing the instantaneous value in terms of the available data $d(i)$ and u_i:

$$\Big[\widehat{\nabla_w J}(w_{i-1})\Big]^T = -u_i^T h\big(d(i) - u_i w_{i-1}\big). \tag{2.24}$$

Substituting (2.24) into the steepest-descent method (2.23) results in the adaptive filtering algorithm (2.12), where the data $\{d(i), u_i\}$ satisfy the model (2.1).

Special choices of the error nonlinearity $h(\cdot)$ lead to well-known algorithms in the adaptive filtering literature. Table 2.1 lists some of those algorithms along with the cost functions used to motivate their form through *instantaneous* stochastic approximation of the corresponding gradient expressions. These are the

1. least-mean-squares (LMS),
2. sign-error LMS,
3. least-mean-fourth (LMF),
4. least-mean mixed-norm (LMMN), and
5. robust mixed-norm (RMN) algorithms

—the latter two utilizing some constant $0 \leq \delta \leq 1$.

Table 2.1: Adaptive Filtering Algorithms With Error Nonlinearities

Algorithm	**Error Nonlinearity** $h\big(e(i)\big)$	**Cost Function** $J(w)$
LMS	$e(i)$	$\mathbb{E}\left(\boldsymbol{d}-\boldsymbol{u}w\right)^2$
sign-error LMS	$\operatorname{sign}\big(e(i)\big)$	$\mathbb{E}\left\|\boldsymbol{d}-\boldsymbol{u}w\right\|$
LMF	$e^3(i)$	$\mathbb{E}\left(\boldsymbol{d}-\boldsymbol{u}w\right)^4$
LMMN	$\delta e(i)+(1-\delta)\,e^3(i)$	$\delta\,\mathbb{E}\left(\boldsymbol{d}-\boldsymbol{u}w\right)^2+\frac{1}{2}\,(1-\delta)\,\mathbb{E}\left(\boldsymbol{d}-\boldsymbol{u}w\right)^4$
RMN	$\delta\operatorname{sign}\big(e(i)\big)+(1-\delta)\,e(i)$	$\delta\,\mathbb{E}\left\|\boldsymbol{d}-\boldsymbol{u}w\right\|+\frac{1}{2}\,(1-\delta)\,\mathbb{E}\left(\boldsymbol{d}-\boldsymbol{u}w\right)^2$

The aforementioned algorithms approximate the optimal weight vector w^o iteratively subject to different error criteria. Hence, they are expected to behave differently under different signal conditions. Some of the properties of those algorithms are going to be touched upon briefly.

1. **LMS algorithm:** Based on the MSE criterion, the LMS algorithm was developed in the seminal work [WH60] by Widrow and Hoff in 1960. It has been popular ever since owing to its simplicity and low computational complexity.
2. **Sign-error LMS algorithm:** The algorithm was proposed in the early works [CM81, Dut82, Ger84] as a computationally simpler, albeit slower, alternative to the LMS algorithm in adaptive filtering applications since the number of multiplications in digital implementation are slashed by half through the use of shift registers. However, in addition to its computational simplicity, the sign-error LMS algorithm has been shown to be more robust than the LMS algorithm when the noise distribution is heavy-tailed [SN93]. This latter aspect will be clarified in Sec. 2.2.4.
3. **LMF algorithm:** A member of the family of least-mean $2p$-norm algorithms ($p \geq 1$), the algorithm was developed in [WW84], where it was also shown to outperform the LMS algorithm in the presence of uniform or Bernoulli noise or sinusoidal interference.
4. **LMMN algorithm:** Based on a convex combination of mean-square- and mean-fourth-error costs, the LMMN algorithm aims to trade off the performance of the LMS and LMF algorithms [CTC94, TC96]. It exhibits performance superior to both algorithms when the noise signal is a combination of Gaussian noise and shorter-tailed noise such as uniform or Bernoulli noise. An adaptive construction for the combination factor δ was proposed in [PC95, PC96].
5. **RMN algorithm:** Based on a convex combination of mean-absolute- and mean-square-error costs, the RMN algorithm aims to trade off the performance of the LMS and sign-error algorithms [CA97]. It exhibits performance superior to both algorithms when the noise signal is a combination of Gaussian noise and heavier-tailed noise. An adaptive construction for the combination factor δ based on robust statistics was proposed in the same work.

The LMS Algorithm

It is instructive to treat the MMSE problem, which culminates in the LMS algorithm according to the previous discussion, in some detail in order to highlight some of the aspects involved in the development and analysis of adaptive filtering algorithms. To this end, consider the MSE cost function

$$J(w) = \mathbb{E}\left(\boldsymbol{d} - \boldsymbol{u}w\right)^2. \tag{2.25}$$

The gradient of (2.25) is given by

$$\nabla_w J(w) = -2\left(r_{du} - R_u w\right)^T \tag{2.26}$$

where

$$r_{du} = \mathbb{E}\,\boldsymbol{d}\boldsymbol{u}^T, \qquad R_u = \mathbb{E}\,\boldsymbol{u}^T\boldsymbol{u} > 0. \tag{2.27}$$

Hence, minimizing (2.25) iteratively using the steepest-descent method entails the following update equation:

$$w_i = w_{i-1} + \mu\left(r_{du} - R_u w_{i-1}\right), \quad i \geq 0, \quad w_{-1} = \text{initial condition} \tag{2.28}$$

where the factor 2 was absorbed into the step-size μ. This is to say that at each time index i, the iterate w_i is obtained from w_{i-1} by updating the latter in a direction opposite to the transposed gradient evaluated at w_{i-1}, or equivalently, along the direction of steepest descent. Such a choice for update direction constitutes only a necessary condition for w_i to converge to w^o as $i \to \infty$. A sufficient condition for convergence is obtained by proper selection of the step-size μ. Namely, letting $\widetilde{w}_i \triangleq w^o - w_i$ denote the offset between the optimal weight vector w^o of the iterate at time i, and since the optimal weight vector w^o satisfies the normal equations

$$R_u w^o = r_{du} \tag{2.29}$$

then subtracting both sides of (2.28) from w^o and using (2.29) leads to the following weight-error recursion:

$$\widetilde{w}_i = \left(I - \mu R_u\right)\widetilde{w}_{i-1}. \tag{2.30}$$

From (2.30), it can be seen that the convergence of the iterate w_i to w^o (or the weight-error vector $\widetilde{w}_i$ to 0), irrespective of the initial condition w_{-1}, is ensured by the stability of the matrix $(I - \mu R_u)$, which requires selecting μ to satisfy

$$0 < \mu < \frac{2}{\lambda_{\max}} \tag{2.31}$$

where $\lambda_{\max}$ denotes the maximum eigenvalue of its symmetric matrix argument. The convergence behavior of the weight-error vector $\widetilde{w}_i$ evolving according to (2.30) can be easily shown to be exponential and controlled by the modes $\{1 - \mu\lambda_m\}$ or the time constants $\left\{-\frac{1}{2\ln|1-\mu\lambda_m|}\right\}$, where λ_m, $m = 1, \ldots, M$, are the eigenvalues of R_u, and $\ln(\cdot)$ is the natural logarithm—see [Say03] for more details. The performance of the steepest-descent method can also be characterized by its learning curve or MSE curve

$$\begin{aligned} J(i) &\triangleq J(w_{i-1}), \qquad i \geq 0 \\ &= \mathbb{E}\left(\boldsymbol{d} - \boldsymbol{u}w_{i-1}\right)^2 \\ &= \sigma_d^2 - r_{ud}R_u^{-1}r_{du} + \widetilde{w}_{i-1}^T R_u \widetilde{w}_{i-1} \end{aligned} \tag{2.32}$$

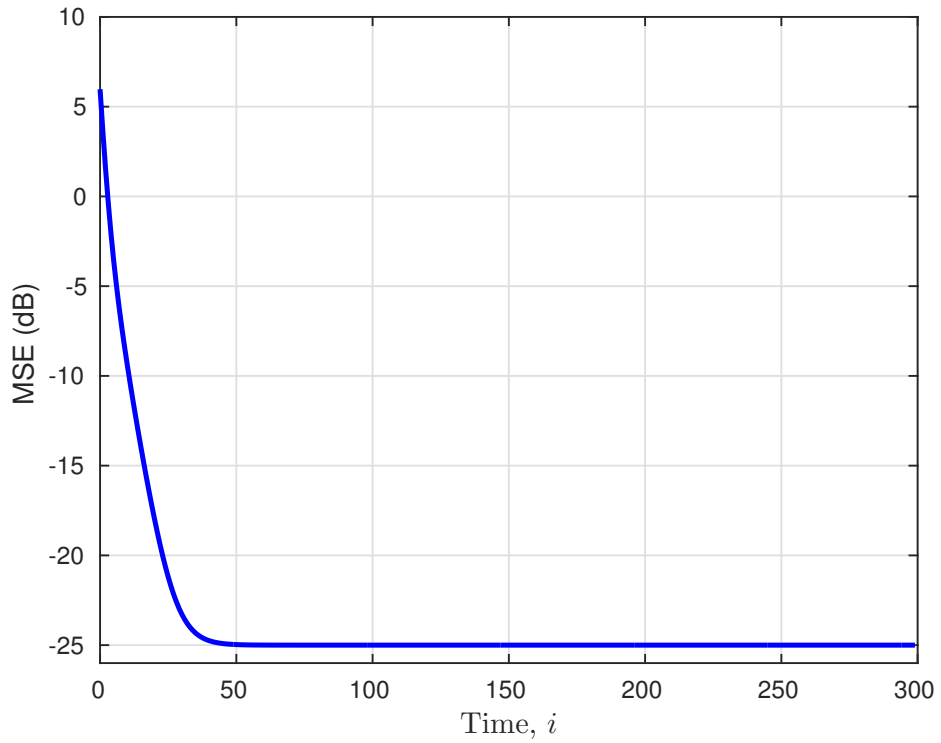


Figure 2.3: Typical mean-square-error (MSE) learning curve for the steepest-descent algorithm (2.28).

$$\triangleq J_{\min} + \widetilde{w}_{i-1}^T R_u \widetilde{w}_{i-1}$$

where $J_{\min}$ denotes the minimum cost:

$$J_{\min} = J(w^o) = \sigma_d^2 - r_{ud} R_u^{-1} r_{du}. \tag{2.33}$$

By choosing the step-size μ to satisfy $0 < \mu < \frac{2}{\lambda_{\max}}$, then $J(i) \to J_{\min}$ as $i \to \infty$. The convergence can be easily shown to be exponential and monotonic. A typical learning curve is shown in Fig. 2.3.

Running (2.28) to compute w^o requires exact knowledge of the moments R_u and r_{du}, however. In many applications, this information is either missing or time-varying. Available are rather realizations $\{d(i), u_i\}$ of the random variables $\{\boldsymbol{d}, \boldsymbol{u}\}$, satisfying the model (2.1), for example. In order to address these situations, the moments R_u and r_{du} may be replaced with stochastic approximations thereof. One possibility is to use an instantaneous approximation of the moments, i.e.,

$$r_{du} \approx d(i) u_i^T, \qquad R_u \approx u_i^T u_i. \tag{2.34}$$

Replacing the moments r_{du} and R_u in the steepest-descent method (2.28) by their

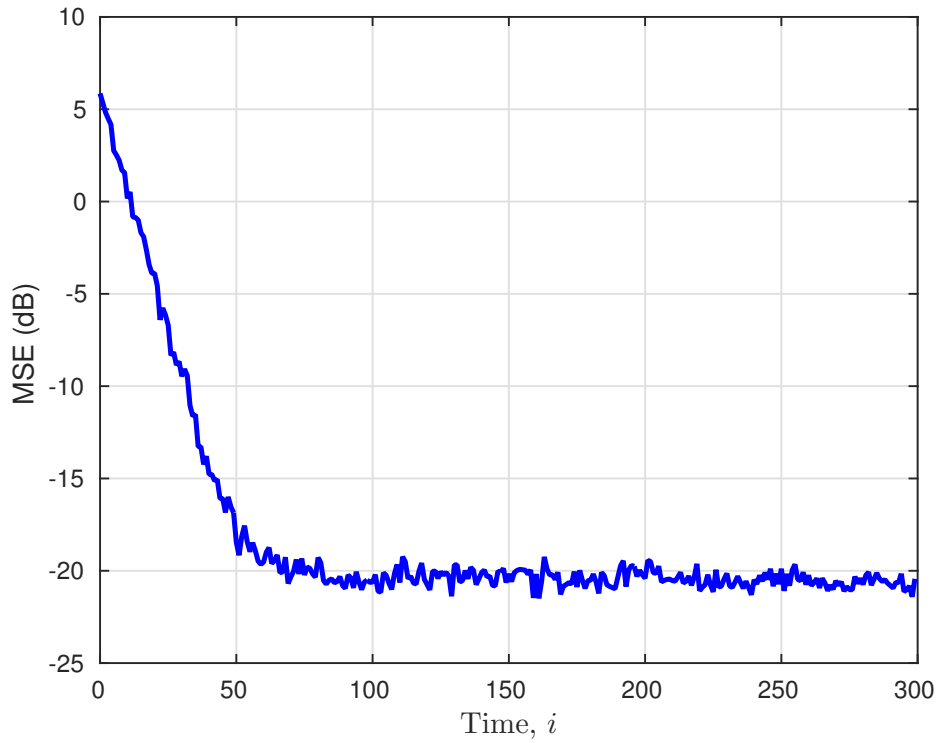


Figure 2.4: Typical ensemble-average mean-square-error (MSE) learning curve for the LMS algorithm (2.35). The data $\{d(i), u_i\}$ have the same underlying statistical profile used to generate the learning curve in Fig. 2.3.

approximations (2.34) leads to the LMS algorithm:

$$w_i = w_{i-1} + \mu u_i^T \left[d(i) - u_i w_{i-1}\right], \quad i \geq 0, \quad w_{-1} = \text{initial condition} \tag{2.35}$$

The resulting stochastic-gradient algorithm is effectively capable of learning the statistics of the process $\{\boldsymbol{d}(i), \boldsymbol{u}_i\}$ over time and tracking changes in the parameter w^o as well. Analogously to the MSE learning curve (2.32) of the underlying steepest-descent method, one can construct the MSE learning curve for the LMS algorithm. In the absence of the statistics $\{\sigma_d^2, r_{du}, R_u\}$, and recalling that the output estimation error is given by $e(i) = d(i) - u_i w_{i-1}$, the curve can be approximated for sufficiently small step-size μ by the sample-average over L experiments—see [Say03, Appendix 9.E]:

$$\widehat{J}(i) \triangleq \frac{1}{L} \sum_{\ell=1}^{L} \big(e^{(\ell)}(i)\big)^2, \qquad i \geq 0 \tag{2.36}$$

where the superscript (ℓ) denotes the realization by the ℓth experiment of the process $\{\boldsymbol{e}(i)\}$. A typical such ensemble-average learning curve is shown in Fig. 2.4.

□

Algorithms of the form (2.12) have also been devised in a different spirit as variations over the LMS algorithm that lend themselves to simple digital implementations. The sign-error LMS algorithm is a case in point. Other examples include the dual-sign LMS algorithm [SJ89, Mat91] and the power-of-two error LMS algorithm [XL86, Ewe92]. On the other hand, an LMS algorithm implementation where the error signal $e(i)$ undergoes a quantization operation can also be modeled by the form (2.12), with $h(\cdot)$ assuming a saturation-type nonlinearity [Ber88]. Irrespective of the myriad purposes they serve, the aforementioned constructions are interesting in their own right since their performance analyses have enriched the literature over the years, contributing invaluably to the repertoire of techniques towards the analysis of elaborate adaptive filtering algorithms.

Due to their stochastic, nonlinear, and time-varying nature, the analysis of adaptive filtering algorithms is not straightforward. Algorithms of the form (2.12) count among the most challenging to analyze due to the presence of the error nonlinearity. Performance analysis is the subject of the next section.

2.2.3 Primer on Performance Analysis

The goal of performance analysis is to address the following questions:

- Steady-state performance: How close is the limiting value of the sequence of the weight estimates $\{w_i\}$ to the optimal weight vector w^o?
- Stability: What are the conditions for convergence?
- Transient behavior: In what manner does convergence occur? How fast is convergence?

Due to their stochastic, nonlinear, and time-varying nature, exact analysis of adaptive filters is generally not possible [Say03]. Conventionally, assumptions are introduced to facilitate analysis. Several analysis frameworks have been developed in the literature, including

- independence analysis,
- averaging analysis, and
- the ordinary-differential-equation (ODE) method.

For a detailed treatment, see [Say03] for independence analysis, and [KY03, BMP87] for averaging analysis and the ODE method.

In this section, the steady-state mean-square performance analysis of adaptive filters with error nonlinearities is reproduced following [ANS01,ANS03,Say03]. Only as many independence assumptions as necessary are introduced to make analysis tractable such that the theoretical results yielded by the analysis maintain their validity for a broad range of applications. Subsequently, the optimal error nonlinearity that leads to best steady-state performance is re-derived. The reason why the derivation of steady-state performance expressions and subsequent optimization thereof are repeated here are fivefold:

- to bring together in one place all the assumptions made in [ANS01,ANS03,Say03] towards the derivations;
- to provide a clearer derivation of the lower bound on steady-state performance;
- to offer a glimpse into the analysis of adaptive filters, hopefully inciting an appreciation for the challenges involved in the analysis of the more intricate robust adaptive filtering algorithm that is to be developed in the next chapter;
- to survey and compare alternative techniques and assumptions towards the analysis of adaptive filters with error nonlinearities; and
- to establish a link to maximum-likelihood estimation and robust estimation.

Now we can proceed with the analysis. First, the adaptive filtering algorithm (2.12) is to be modeled as a stochastic difference equation where all quantities that appear are treated as random variables:

$$\boldsymbol{e}(i) = \boldsymbol{d}(i) - \boldsymbol{u}_i \boldsymbol{w}_{i-1} \tag{2.37a}$$

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} + \mu \boldsymbol{u}_i^T h\big(\boldsymbol{e}(i)\big) \tag{2.37b}$$

The initial condition $\boldsymbol{w}_{-1}$ is regarded as a random vector as well that is independent of all $\{\boldsymbol{d}(i), \boldsymbol{u}_i, \boldsymbol{v}(i)\}$.

Performance Measures

One performance measure is the *steady-state MSE*:

$$\text{MSE} \triangleq \lim_{i\to\infty} \mathbb{E}\, \boldsymbol{e}^2(i) \tag{2.38}$$

in terms of the output estimation error $\boldsymbol{e}(i) = \boldsymbol{d}(i) - \boldsymbol{u}_i \boldsymbol{w}_{i-1}$. Were the weight estimator $\boldsymbol{w}_{i-1}$ to converge to the optimal weight vector w^o, as would the original steepest-descent method, then the error signal $\boldsymbol{e}(i)$ would coincide with the noise signal $\boldsymbol{v}(i)$ at steady-state, and the steady-state MSE, with the noise variance σ_v^2. Since the stochastic-gradient algorithm introduces gradient noise into the recursion, however,

the steady-state MSE assumes a value larger than σ_v^2 by a quantity referred to as the *steady-state excess MSE* (EMSE). To see this, consider the weight-error vector:

$$\widetilde{\boldsymbol{w}}_i \triangleq w^o - \boldsymbol{w}_i \tag{2.39}$$

and the *a priori* estimation error:

$$\boldsymbol{e}_a(i) \triangleq \boldsymbol{u}_i \widetilde{\boldsymbol{w}}_{i-1} \tag{2.40}$$

which measures the offset between the term $\boldsymbol{u}_i w^o$ and its estimator $\boldsymbol{u}_i \boldsymbol{w}_{i-1}$ prior to adaptation. Then, it follows from the data model (2.1) that

$$\begin{aligned} \boldsymbol{e}(i) &= \boldsymbol{d}(i) - \boldsymbol{u}_i \boldsymbol{w}_{i-1} \\ &= \boldsymbol{u}_i w^o + \boldsymbol{v}(i) - \boldsymbol{u}_i \boldsymbol{w}_{i-1} \\ &= \boldsymbol{e}_a(i) + \boldsymbol{v}(i) \end{aligned} \tag{2.41}$$

Since by the data model assumptions in Sec. 2.2.1 $\boldsymbol{v}(i)$ is independent of $\boldsymbol{e}_a(i)$, and since it is zero-mean, then it holds that

$$\mathbb{E}\, \boldsymbol{e}^2(i) = \mathbb{E}\, \boldsymbol{e}_a^2(i) + \sigma_v^2. \tag{2.42}$$

Defining the steady-state EMSE as the steady-state mean-square value of the *a priori* estimation error:

$$\text{EMSE} \triangleq \lim_{i \to \infty} \mathbb{E}\, \boldsymbol{e}_a^2(i) \tag{2.43}$$

the steady-state MSE is then given by

$$\text{MSE} = \text{EMSE} + \sigma_v^2. \tag{2.44}$$

Another performance measure is the *steady-state mean-square deviation* (MSD), defined as

$$\text{MSD} \triangleq \lim_{i \to \infty} \mathbb{E}\, \|\widetilde{\boldsymbol{w}}_i\|^2 . \tag{2.45}$$

The steady-state MSD measures the offset between the optimal weight vector w^o and the estimator $\boldsymbol{w}_i$ in the mean-square sense at steady-state.

□

Variance Relation

In the course of adaptive filter analysis, substantial use is made of weighted squared Euclidean norms. For an $M \times 1$ vector x and $M \times M$ symmetric nonnegative-definite weighting matrix Σ, the weighted squared Euclidean norm of x is defined compactly as

$$\|x\|_\Sigma^2 \triangleq x^T \Sigma x. \tag{2.46}$$

The standard squared Euclidean norm of x is recovered by the choice $\Sigma = I$ and denoted simply as $\|x\|^2$. It will be shown later that the various performance measures, such as the steady-state EMSE and MSD, can be conveniently expressed as weighted squared Euclidean norms of the weight-error vector $\widetilde{\boldsymbol{w}}_i$.

Subtracting both sides of the adaptive update equation (2.37b) from the optimal weight vector w^o leads to

$$\widetilde{\boldsymbol{w}}_i = \widetilde{\boldsymbol{w}}_{i-1} - \mu \boldsymbol{u}_i^T h\big(\boldsymbol{e}(i)\big). \tag{2.47}$$

Equating the weighted squared Euclidean norms of either side of (2.47) and taking the expectation with respect to the distribution of the joint random process $\{\boldsymbol{d}(i), \boldsymbol{u}_i\}$ results in the following *weighted variance relation*:

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2 = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_\Sigma^2 - 2\mu\,\mathbb{E}\,\boldsymbol{e}_a^\Sigma(i) h\big(\boldsymbol{e}(i)\big) + \mu^2\,\mathbb{E}\,\|\boldsymbol{u}_i\|_\Sigma^2\, h^2\big(\boldsymbol{e}(i)\big) \tag{2.48}$$

where

$$\boldsymbol{e}_a^\Sigma(i) \triangleq \boldsymbol{u}_i \Sigma \widetilde{\boldsymbol{w}}_{i-1}, \tag{2.49}$$

denoting the weighted *a priori* estimation error.

An adaptive filter is said to be at steady-state when it holds that

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2 = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_\Sigma^2 = r < \infty, \quad \text{as } i \to \infty, \tag{2.50}$$

where r is a nonnegative constant. Evidently, transient analysis of the adaptive filter needs to be undertaken in order to determine the range of values of the step-size μ over which the variance $\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2$ remains bounded and converges to a finite value. Indeed, in [ANS01], one such mean-square stability condition was established. Hence, assuming the value of the step-size μ was chosen to ensure mean-square stability such that the adaptive filter eventually reaches steady-state, then, taking the limit as $i \to \infty$ of both sides of (2.48) results in the following steady-state weighted variance relation:

$$\mu\,\mathbb{E}\,\|\boldsymbol{u}_i\|_\Sigma^2\, h^2\big(\boldsymbol{e}(i)\big) = 2\,\mathbb{E}\,\boldsymbol{e}_a^\Sigma(i) h\big(\boldsymbol{e}(i)\big), \quad \text{as } i \to \infty. \tag{2.51}$$

□

Performance Evaluation

Since $\boldsymbol{e}(i) = \boldsymbol{e}_a(i) + \boldsymbol{v}(i)$, the relation (2.51) can be expressed in terms of $\boldsymbol{e}_a(i)$ and solved for the steady-state EMSE, $\lim_{i\to\infty} \mathbb{E}\,\boldsymbol{e}_a^2(i)$. In order to proceed with the solution, however, simplifying assumptions are necessary for the evaluation of the moments on either side of the relation (2.51). The first two assumptions below, AG and AU, were used in [ANS01, ANS03]:

- AG: At steady-state, the *a priori* estimation errors $\{\boldsymbol{e}_a(i), \boldsymbol{e}_a^{\Sigma}(i)\}$ are zero-mean, jointly Gaussian random variables.
- AU: At steady-state, the random variables $\|\boldsymbol{u}_i\|_{\Sigma}^2$ and $h^2\big(\boldsymbol{e}(i)\big)$ are uncorrelated, i.e.,
$$\lim_{i\to\infty} \mathbb{E}\,\|\boldsymbol{u}_i\|_{\Sigma}^2\, h^2\big(\boldsymbol{e}(i)\big) = \mathrm{Tr}(R_u\Sigma) \lim_{i\to\infty} \mathbb{E}\, h^2\big(\boldsymbol{e}(i)\big). \tag{2.52}$$
- The error nonlinearity h is differentiable, its derivative denoted by h'. Moreover, h is strictly monotonically increasing, i.e., $h' > 0$.

Although the third assumption on the differentiability and strict monotonicity of h was not mentioned explicitly in the cited works, it is necessary for the application of Price's theorem [Pri58], as can be appreciated from Appendix A.1. Assumption AG ("G" for Gaussian) is reasonable for long adaptive filters since the random variables $\{\boldsymbol{e}_a(i), \boldsymbol{e}_a^{\Sigma}(i)\}$ would thus amount to sums of a large number of random variables, so their distribution could be approximated as Gaussian by a central-limit argument—see [Say03, P. 485]. Assumption AU ("U" in reference to the regressors $\{\boldsymbol{u}_i\}$), on the other hand, is a weaker form of the independence assumption, where the regressor sequence $\{\boldsymbol{u}_i\}$ is assumed to be i.i.d. Assumption AU is also more realistic the longer the adaptive filter and the smaller the step-size μ.

For the evaluation of the moment on the right-hand side of the relation (2.51), Assumption AG is appealed to, which facilitates the application of a result of Price's theorem [Pri58], namely, Result 3 in Appendix A.1, summarized here. For scalar real-valued zero-mean jointly Gaussian random variables $\boldsymbol{x}$ and $\boldsymbol{y}$ that are independent of a third scalar real-valued zero-mean random variable $\boldsymbol{z}$, and for a function $f(\boldsymbol{y}+\boldsymbol{z})$ that is differentiable with respect to $\boldsymbol{y}$, it holds that

$$\mathbb{E}\,\boldsymbol{x} f(\boldsymbol{y}+\boldsymbol{z}) = \frac{\mathbb{E}\,\boldsymbol{x}\boldsymbol{y}}{\mathbb{E}\,\boldsymbol{y}^2} \cdot \mathbb{E}\,\boldsymbol{y} f(\boldsymbol{y}+\boldsymbol{z}). \tag{2.53}$$

Applying this result to the moment on the right-hand side of the relation (2.51) and recalling that $\boldsymbol{v}(i)$ is independent of $\boldsymbol{e}_a(i)$ leads to the following simplification:

$$\mathbb{E}\,\boldsymbol{e}_a^{\Sigma}(i) h\big(\boldsymbol{e}(i)\big) = \mathbb{E}\,\boldsymbol{e}_a^{\Sigma}(i)\boldsymbol{e}_a(i) \cdot \frac{\mathbb{E}\,\boldsymbol{e}_a(i) h\big(\boldsymbol{e}(i)\big)}{\mathbb{E}\,\boldsymbol{e}_a^2(i)}. \tag{2.54}$$

The expression on the right-hand side of (2.54) can be simplified further by invoking another result of Price's theorem, namely, Result 1 in Appendix A.1, summarized here. For scalar real-valued zero-mean jointly Gaussian random variables $\boldsymbol{x}$ and $\boldsymbol{y}$, and for a function $f(\boldsymbol{y})$ that is differentiable with respect to $\boldsymbol{y}$, it holds that

$$\mathbb{E}\,\boldsymbol{x} f(\boldsymbol{y}) = \mathbb{E}\,\boldsymbol{x}\boldsymbol{y} \cdot \mathbb{E}\,\frac{df}{d\boldsymbol{y}}. \tag{2.55}$$

Applying this result to the moment $\mathbb{E}\,\boldsymbol{e}_a(i)h\big(\boldsymbol{e}(i)\big)$ in (2.54), also invoking the fact that $\boldsymbol{v}(i)$ is independent of $\boldsymbol{e}_a(i)$, results in

$$\mathbb{E}\,\boldsymbol{e}_a(i)h\big(\boldsymbol{e}(i)\big) = \mathbb{E}\,\boldsymbol{e}_a^2(i)\,\mathbb{E}\,h'\big(\boldsymbol{e}(i)\big). \tag{2.56}$$

Finally, invoking Assumption AU to evaluate the moment on the left-hand side of the relation (2.51), and using (2.54) and (2.56), the steady-state weighted variance relation becomes

$$2\lim_{i\to\infty}\mathbb{E}\,\boldsymbol{e}_a^{\Sigma}(i)\boldsymbol{e}_a(i)\lim_{i\to\infty}\mathbb{E}\,h'\big(\boldsymbol{e}(i)\big) = \mu\,\mathrm{Tr}(R_u\Sigma)\lim_{i\to\infty}\mathbb{E}\,h^2\big(\boldsymbol{e}(i)\big) \tag{2.57}$$

or, equivalently,

$$\lim_{i\to\infty}\mathbb{E}\,\boldsymbol{e}_a^{\Sigma}(i)\boldsymbol{e}_a(i) = \frac{\mu}{2}\,\mathrm{Tr}(R_u\Sigma)\lim_{i\to\infty}\frac{\mathbb{E}\,h^2\big(\boldsymbol{e}(i)\big)}{\mathbb{E}\,h'\big(\boldsymbol{e}(i)\big)}. \tag{2.58}$$

Let the symbol ζ be shorthand for the steady-state EMSE of the adaptive filter. That is, $\zeta = \mathrm{EMSE} = \lim_{i\to\infty}\mathbb{E}\,\boldsymbol{e}_a^2(i)$. Hence, setting Σ in (2.58) to the identity matrix yields the following expression for the steady-state EMSE:

$$\zeta = \frac{\mu}{2}\,\mathrm{Tr}(R_u)\lim_{i\to\infty}\frac{\mathbb{E}\,h^2\big(\boldsymbol{e}(i)\big)}{\mathbb{E}\,h'\big(\boldsymbol{e}(i)\big)}. \tag{2.59}$$

In order to derive an expression for the steady-state MSD, (2.58) and (2.59) are first combined, giving

$$\lim_{i\to\infty}\mathbb{E}\,\boldsymbol{e}_a^{\Sigma}(i)\boldsymbol{e}_a(i) = \zeta\cdot\frac{\mathrm{Tr}(R_u\Sigma)}{\mathrm{Tr}(R_u)}. \tag{2.60}$$

For the simplification of the moment on the left-hand side of (2.60), another assumption is called for:

- At steady-state, $\widetilde{\boldsymbol{w}}_{i-1}$ is independent of $\boldsymbol{u}_i$.

This assumption is less restrictive than the independence assumption, where the regressor sequence $\{\boldsymbol{u}_i\}$ is assumed to be i.i.d. Referred to as Assumption AI in [ANS03], the independence assumption implies that $\widetilde{\boldsymbol{w}}_{i-1}$ is independent of $\boldsymbol{u}_i$ for all i, and not just at steady-state. Assumption AI was used in [ANS03] to derive an expression for the steady-state MSD. Although unrealistic in AR and MA process modeling applications (see Sec. 2.2.1), since successive regressors share common entries and cannot be statistically independent, Assumption AI significantly simplifies the transient analysis of adaptive filters and leads to results that match well with practice when the step-size μ is sufficiently small [Say03]. For the purposes of steady-state analysis, however,

the weaker assumption that $\widetilde{\boldsymbol{w}}_{i-1}$ and $\boldsymbol{u}_i$ are asymptotically independent may be used instead, from which follows that

$$\lim_{i\to\infty} \mathbb{E}\, \boldsymbol{e}_a^{\Sigma}(i)\boldsymbol{e}_a(i) = \lim_{i\to\infty} \mathbb{E}\, \|\widetilde{\boldsymbol{w}}_{i-1}\|^2_{\Sigma R_u}\,. \tag{2.61}$$

Recalling that the steady-state MSD is given by $\lim_{i\to\infty} \mathbb{E}\, \|\widetilde{\boldsymbol{w}}_i\|^2$, then, combining (2.59), (2.60), and (2.61), and setting $\Sigma = R_u^{-1}$ results in

$$\mathrm{MSD} = \frac{M\zeta}{\mathrm{Tr}(R_u)} = \frac{\mu}{2} M \lim_{i\to\infty} \frac{\mathbb{E}\, h^2\big(\boldsymbol{e}(i)\big)}{\mathbb{E}\, h'\big(\boldsymbol{e}(i)\big)}, \tag{2.62}$$

showing that the steady-state MSD and EMSE are related through appropriate scaling.

2.2.4 MSE-Optimal Error Nonlinearity

In [ANS01], the error nonlinearity h that minimizes the steady-state EMSE (2.59) or equivalently, the steady-state MSE (2.44) was derived. In this section, this optimal error nonlinearity is re-derived using the same approach as in [ANS01], albeit based on the minimization of the steady-state MSD (2.62). The two optimization problems are equivalent under the assumption that $\widetilde{\boldsymbol{w}}_{i-1}$ and $\boldsymbol{u}_i$ are asymptotically independent. The reason for preferring the steady-state MSD as the performance measure to be minimized here is because it lends itself to a more intuitive derivation of the optimal error nonlinearity.

First, it is to be noted that the steady-state MSD cannot be reduced beyond a lower bound in terms of the Cramér–Rao bound. The derivation of this lower bound, denoted as λ, can be found in Appendix A.2. It then follows that

$$\frac{\lim\limits_{i\to\infty} \mathbb{E}\, h^2\big(\boldsymbol{e}(i)\big)}{\lim\limits_{i\to\infty} \mathbb{E}\, h'\big(\boldsymbol{e}(i)\big)} \geq \frac{2}{\mu M}\lambda \triangleq \alpha. \tag{2.63}$$

Next, the ratio on the left-hand side of (2.63) is expressed in a less cluttered notation. Under Assumption AG on the Gaussianity of $\boldsymbol{e}_a(i)$, the moments

$$\mathbb{E}\, h^2\big(\boldsymbol{e}(i)\big) \quad \text{and} \quad \mathbb{E}\, h'\big(\boldsymbol{e}(i)\big) \tag{2.64}$$

derive their time variation from their dependence on the zero-mean *a priori* estimation error $\boldsymbol{e}_a(i)$ only through its time-varying variance $\mathbb{E}\, \boldsymbol{e}_a^2(i)$. For example, for any function $g\colon \mathbb{R} \to \mathbb{R}$, the moment $\mathbb{E}\, g\big(\boldsymbol{e}(i)\big)$ is given by

$$\mathbb{E}\, g\big(\boldsymbol{e}(i)\big) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(e_a + v) \frac{1}{\sqrt{2\pi\, \mathbb{E}\, \boldsymbol{e}_a^2(i)}} \exp\left[\frac{-e_a^2}{2\, \mathbb{E}\, \boldsymbol{e}_a^2(i)}\right] f_{\boldsymbol{v}}(v)\, de_a\, dv \tag{2.65}$$

where $f_{\boldsymbol{v}}(v)$ denotes the pdf of the noise random variable $\boldsymbol{v}(i)$ for all i, the noise random process being stationary. Since at steady-state, the variance $\mathbb{E}\,\boldsymbol{e}_a^2(i)$ would have converged to the steady-state EMSE, ζ, then, one can regard the output estimation error sequence $\{\boldsymbol{e}(i)\}$ as having converged in distribution to a random variable $\boldsymbol{e}^\star$ ($\boldsymbol{e}(i) \xrightarrow{d} \boldsymbol{e}^\star$) of mean zero and variance equal to the steady-state MSE, $\zeta + \sigma_v^2$, and with pdf $f_{\boldsymbol{e}^\star}\big(e^\star\big)$ [PP02]. Using this reasoning, it is justified to rewrite (2.63) as

$$\frac{\mathbb{E}\,h^2\big(\boldsymbol{e}^\star\big)}{\mathbb{E}\,h'\big(\boldsymbol{e}^\star\big)} \geq \alpha. \tag{2.66}$$

Akin to [ANS01], if the error nonlinearity $h(\cdot)$ is chosen as

$$\hat{h}(\cdot) \triangleq -\alpha \frac{f'_{\boldsymbol{e}^\star}(\cdot)}{f_{\boldsymbol{e}^\star}(\cdot)} \tag{2.67}$$

then the resulting ratio in (2.66) will achieve the lower bound α, and accordingly, the steady-state MSD will achieve the lower bound λ. The proof from [ANS01] is reproduced in Appendix A.3, where an additional assumption is imposed on the limiting error distribution, namely, that

$$\lim_{e^\star \to \pm\infty} f'_{\boldsymbol{e}^\star}(e^\star) = 0. \tag{2.68}$$

By using the expression for the optimal error nonlinearity $\hat{h}$ in (2.67) in the adaptive filtering algorithm (2.12), one can see that the constant factor α will appear multiplied with the step-size μ, in which case α can be absorbed into μ so that the optimal error nonlinearity effectively takes on the following form:

$$h^o\big(e(i)\big) \triangleq -\frac{f'_{\boldsymbol{e}(i)}\big(e(i)\big)}{f_{\boldsymbol{e}(i)}\big(e(i)\big)}. \tag{2.69}$$

It is essentially a time-varying nonlinearity, which converges to $-\frac{f'_{\boldsymbol{e}^\star}(\cdot)}{f_{\boldsymbol{e}^\star}(\cdot)}$ at steady-state, and results in the following form for the MSE-optimal adaptive filtering algorithm:

$$w_i = w_{i-1} + \mu u_i^T \left[\frac{f'_{\boldsymbol{e}(i)}\big(e(i)\big)}{f_{\boldsymbol{e}(i)}\big(e(i)\big)}\right]. \tag{2.70}$$

For future reference, all the assumptions that have been used to derive the MSE-optimal error nonlinearity h^o in this section are compiled here:

- The noise process $\{\boldsymbol{v}(i)\}$ is zero-mean i.i.d. and independent of the zero-mean regressor sequence $\{\boldsymbol{u}_i\}$.

- AG: At steady-state, the *a priori* estimation errors $\{\boldsymbol{e}_a(i), \boldsymbol{e}_a^{\Sigma}(i)\}$ are zero-mean, jointly Gaussian random variables.
- AU: At steady-state, the random variables $\|\boldsymbol{u}_i\|_{\Sigma}^2$ and $h^2\big(\boldsymbol{e}(i)\big)$ are uncorrelated.
- At steady-state, $\widetilde{\boldsymbol{w}}_{i-1}$ is independent of $\boldsymbol{u}_i$.
- $\lim_{e^\star \to \pm\infty} f'_{\boldsymbol{e}^\star}(e^\star) = 0$, where $f_{\boldsymbol{e}^\star}(e^\star)$ is the limiting pdf of the output estimation error sequence $\{\boldsymbol{e}(i)\}$ ($\boldsymbol{e}(i) \xrightarrow{d} \boldsymbol{e}^\star$).
- The error nonlinearity h is differentiable and strictly monotonically increasing.

Alternative Derivations

Other optimal error nonlinearities were derived in the literature under different sets of assumptions, analysis techniques, and optimization criteria. Table 2.2 lists some examples from the works [DM94, ANSK00], where

$$f'_{\boldsymbol{e}(i)|\widetilde{\boldsymbol{w}}_{i-1}}\big(e(i)\big) \tag{2.71}$$

denotes the pdf of the output estimation error $\boldsymbol{e}(i)$ conditioned on the weight-error vector $\widetilde{\boldsymbol{w}}_{i-1}$. The form of the conditional expectation terms in question is

$$\mathbb{E}\left[\Big(h^{(k)}\big(e(i)\big)\Big)^n \Big| \widetilde{\boldsymbol{w}}_{i-1}\right] \tag{2.72}$$

with $h^{(k)}$ denoting the kth derivative of h. Linearization analysis, used in the works [Dut82, WW84, Set92, DM94], for example, involves the expansion of the error nonlinearity $h^{(k)}\big(e(i)\big)$ in a Taylor series around $e(i) = v(i)$ (or $e_a(i) = 0$) for all i and retaining only the first few lower-order terms. Obviously, the results derived from such analysis are more accurate towards steady-state and for sufficiently small step-size μ such that the steady-state EMSE of the adaptive filter is negligible. In conditional analysis under a Gaussian assumption on the input $\boldsymbol{u}(i)$, the output estimation error $\boldsymbol{e}(i)$ conditioned on the weight-error vector $\widetilde{\boldsymbol{w}}_{i-1}$ is Gaussian. Conditional analysis was employed in the works [CM90, MC87, Ber88, Mat91, BB90, WKL91], revealing greater accuracy than linearization analysis over a wider range of adaptation conditions. As for the optimization criterion that is customarily adopted in the literature, steady-state performance is the objective to be optimized for a given convergence rate; variational calculus is the optimization method of choice.

Table 2.2: Alternative Derivations of Optimal Error Nonlinearity

	[DM94] (arbitrary input)	**[DM94] (i.i.d. Gaussian input)**	**[ANSK00]**
Form	$-\frac{f'_{\boldsymbol{v}}\big(e(i)\big)}{f_{\boldsymbol{v}}\big(e(i)\big)}$	$-\frac{f'_{\boldsymbol{e}(i)\vert\widetilde{\boldsymbol{w}}_{i-1}}\big(e(i)\big)}{f_{\boldsymbol{e}(i)\vert\widetilde{\boldsymbol{w}}_{i-1}}\big(e(i)\big)+\mu\sigma_u^2 f''_{\boldsymbol{e}(i)\vert\widetilde{\boldsymbol{w}}_{i-1}}\big(e(i)\big)}$	$-\frac{f'_{\boldsymbol{v}}\big(e(i)\big)}{f_{\boldsymbol{v}}\big(e(i)\big)+\frac{\mu}{2}\frac{\mathbb{E}\|\boldsymbol{u}_i\|^4}{\mathbb{E}\|\boldsymbol{u}_i\|^2}f''_{\boldsymbol{v}}\big(e(i)\big)}$
Assumptions	1) The noise process $\{\boldsymbol{v}(i)\}$ is zero-mean i.i.d. with symmetric pdf and independent of the zero-mean regressor sequence $\{\boldsymbol{u}_i\}$. 2) The weight-error vector $\widetilde{\boldsymbol{w}}_{i-1}$ is independent of the regressor $\boldsymbol{u}_i$. 3) The error nonlinearity h is sign-preserving, odd-symmetric, monotonically increasing, and twice differentiable. 4) The step-size μ is sufficiently small.	1) The noise process $\{\boldsymbol{v}(i)\}$ is zero-mean i.i.d. with symmetric pdf and independent of the regressor sequence $\{\boldsymbol{u}_i\}$. 2) The input sequence $\{\boldsymbol{u}\,(i)\}$ is zero-mean i.i.d. Gaussian with variance σ_u^2. 3) The error nonlinearity h is sign-preserving, odd-symmetric, monotonically increasing, and twice differentiable. 4) Conditional expectation terms of the form (2.72) are independent of the weight-error vector $\widetilde{\boldsymbol{w}}_{i-1}$.	1) The noise process $\{\boldsymbol{v}(i)\}$ is zero-mean i.i.d. with symmetric pdf and independent of the zero-mean regressor sequence $\{\boldsymbol{u}_i\}$. 2) The weight-error vector $\widetilde{\boldsymbol{w}}_{i-1}$ is independent of the regressor $\boldsymbol{u}_i$. 3) The error nonlinearity h is sign-preserving, odd-symmetric, monotonically increasing, and twice differentiable. 4) The step-size μ is sufficiently small.
Analysis	Linearization analysis	Conditional analysis	Linearization analysis
Optimization	Variational calculus	Variational calculus	Variational calculus

By inspecting the optimal error nonlinearities in Table 2.2, the following observations can be made:

- The nonlinearity derived in [ANSK00] is the same as the one derived in [DM94] for arbitrary input when the step-size μ is sufficiently small—the additional term that appears in the denominator of the first results from retaining additional terms from the Taylor series expansion.
- For sufficiently small step-size μ such that the steady-state EMSE of the adaptive filter is negligible compared to the noise variance, the nonlinearity derived in [ANSK00] is the same as the one derived in [DM94] for i.i.d. Gaussian input up to a proportionality constant. Both nonlinearities reduce to that derived in [DM94] for arbitrary input when the the step-size μ is even smaller.

One remark is in order concerning the properties of the error nonlinearity h under which the analysis in the works [DM94,ANSK00] was undertaken. When the noise pdf $f_{\boldsymbol{v}}(v)$ is symmetric, i.e., $\mathbb{E}\,\boldsymbol{v}^{2q-1}(i) = 0$, $q = 1, 2, \ldots$, usually the error nonlinearity h is chosen to be sign-preserving. The sign-preservation property then ensures that successive weight estimates $\{w_i\}$ descend the error surface $J(w)$ [Set92, TC96, ANSK00].

The assumptions adopted in the works [DM94, ANSK00] towards the derivation of the optimal error nonlinearity are generally more restrictive than those adopted in [ANS01] and summarized in this section. The less restrictive the assumptions, the more accurate the results for general input and noise properties as well as error nonlinearities. Furthermore, the analysis in [ANS01], reproduced in Sec. 2.2.3, does not rely on the linearization approach in [DM94, ANSK00] that has culminated in the optimal error nonlinearity

$$h^{\text{lin}}(x) \triangleq -\frac{f'_{\boldsymbol{v}}(x)}{f_{\boldsymbol{v}}(x)}, \quad x \in \mathbb{R} \tag{2.73}$$

for sufficiently small step-size μ. By avoiding linearization, the steady-state results in [ANS01], reproduced here, exhibit greater accuracy over a richer class of error nonlinearities while leading to an optimal error nonlinearity that is more attuned to the spirit of adaptive filtering and which reduces to (2.73) at steady-state. The particular error nonlinearity (2.73) arises in the related context of maximum-likelihood estimation. This connection will be expounded upon in a later note. Now, however, some light will be shed on aspects of optimal error nonlinearity implementation.

□

Implementation

The optimal error nonlinearity (2.69) does not lend itself to practical implementation in the absence of knowledge of the noise distribution. To see this, note that the pdf of the output estimation error $\boldsymbol{e}(i) = \boldsymbol{e}_a(i) + \boldsymbol{v}(i)$ under Assumption AG on the Gaussianity of the *a priori* estimation error $\boldsymbol{e}_a(i)$ and the independence of $\boldsymbol{e}_a(i)$ from $\boldsymbol{v}(i)$ can be expressed as

$$f_{\boldsymbol{e}(i)}\big(e(i)\big) = \frac{1}{\sqrt{2\pi\,\mathbb{E}\,\boldsymbol{e}_a^2(i)}} \exp\left[\frac{-e^2(i)}{2\,\mathbb{E}\,\boldsymbol{e}_a^2(i)}\right] * f_{\boldsymbol{v}}\big(e(i)\big) \tag{2.74}$$

where $*$ denotes convolution. It is then clear that knowledge of the moment $\mathbb{E}\,\boldsymbol{e}_a^2(i)$ for all i is required for the calculation of $f_{\boldsymbol{e}(i)}\big(e(i)\big)$ in addition to knowledge of the noise pdf $f_{\boldsymbol{v}}(v)$, both being, for all practical purposes, unattainable. In [DM94], adaptive estimation of the output estimation error pdf $f_{\boldsymbol{e}(i)}\big(e(i)\big)$ using a Middleton Class A semi-parametric model [Mid77], jointly with the adaptive estimation of the weight vector w^o, was proposed. However, the parameters of the model have a statistical–physical interpretation that renders their online estimation from streaming data challenging. In a similar vein, it was proposed in [ANS01] to approximate the output estimation error pdf $f_{\boldsymbol{e}(i)}\big(e(i)\big)$ by a truncated Edgeworth expansion [Nut85], which amounts to an approximation of the optimal error nonlinearity (2.69) via a polynomial of finite degree of only the odd powers of the output estimation error $\boldsymbol{e}(i)$, where the coefficients turn out to be defined in terms of its cumulants. While the latter approach offers a unifying view of some familiar adaptive filtering algorithms with error nonlinearities, such as the LMS, LMF, least-mean $2p$-norm, and LMMN algorithms (see Table 2.1), lending insight into their optimality, no guidelines are listed for the implementation of an adaptive procedure based on the approximation that would deliver good performance.

The approach in this dissertation is also semi-parametric, albeit grounded in a familiar concept in *robust* estimation, whose explanation is deferred to Sec. 2.3. First, however, we establish a connection to maximum-likelihood estimation.

□

Connection to Maximum-Likelihood Estimation

Consider a batch of N noisy measurements of the parameter w^o, $d(i)$, $i = 0, \ldots, N-1$, according to the data model (2.1). Collecting the N measurements, known regressors, and noise samples into vectors and matrices,

$$d_{N-1} \triangleq \begin{bmatrix} d(N-1) \\ d(N-2) \\ \vdots \\ d(0) \end{bmatrix}, \quad U_{N-1} \triangleq \begin{bmatrix} u_{N-1} \\ u_{N-2} \\ \vdots \\ u_0 \end{bmatrix}, \quad v_{N-1} \triangleq \begin{bmatrix} v(N-1) \\ v(N-2) \\ \vdots \\ v(0) \end{bmatrix}, \tag{2.75}$$

the batch data model can be written more compactly as

$$d_{N-1} = U_{N-1} w^o + v_{N-1}. \tag{2.76}$$

Following the data model assumptions from Sec. 2.2.1, the noise process $\{\boldsymbol{v}(i), 0 \leq i \leq N-1\}$ is zero-mean i.i.d. with variance σ_v^2, $f_{\boldsymbol{v}}(v)$ being the pdf of the noise random variable $\boldsymbol{v}(i)$. Additionally, it is assumed that the regressors u_i and u_j are drawn independently for all $i \neq j$, $i, j = 0, \ldots, N-1$. Let $f\big(d(0), \ldots, d(N-1); w^o\big)$ denote the likelihood function of the observations $d(0), \ldots, d(N-1)$. The likelihood function is parametrized by w^o. Given the independence of the observations, the log-likelihood function can be written out as

$$\begin{aligned} \ln f\big(d(0), \ldots, d(N-1); w^o\big) &= \ln \prod_{i=0}^{N-1} f_{\boldsymbol{v}}\big(d(i) - u_i w^o\big) \\ &= \sum_{i=0}^{N-1} \ln f_{\boldsymbol{v}}\big(d(i) - u_i w^o\big) \end{aligned} \tag{2.77}$$

The maximum-likelihood (ML) estimate of w^o from the observations is obtained by maximizing the log-likelihood function:

$$w^{\mathrm{ML}} \in \arg\min_{w} \sum_{i=0}^{N-1} -\ln f_{\boldsymbol{v}}\big(d(i) - u_i w\big). \tag{2.78}$$

Assuming $f_{\boldsymbol{v}}(\cdot)$ is differentiable, $f'_{\boldsymbol{v}}(\cdot)$ denoting the derivative, and letting $\psi_v(\cdot)$ denote the derivative of $-\ln f_{\boldsymbol{v}}(\cdot)$, known as the score function:

$$\psi_v(x) = \frac{d\{-\ln f_{\boldsymbol{v}}(x)\}}{dx} = -\frac{f'_{\boldsymbol{v}}(x)}{f_{\boldsymbol{v}}(x)}, \quad x \in \mathbb{R}, \tag{2.79}$$

then w^{ML} should satisfy

$$\sum_{i=0}^{N-1} -u_i \, \psi_v\big(d(i) - u_i w^{\mathrm{ML}}\big) = 0. \tag{2.80}$$

Note that the score function (2.79) is the same as the optimal adaptive filtering error nonlinearity (2.73) spewed by linearization analysis. In the following examples, the results are specialized to two noise distributions while highlighting the connection to adaptive filtering with optimal error nonlinearities.

Examples:

1. For Gaussian noise, the pdf is given by

$$f_{\boldsymbol{v}}(v) = \frac{1}{\sqrt{2\pi\sigma_v^2}} e^{-\frac{v^2}{2\sigma_v^2}} \tag{2.81}$$

and the score function by

$$\psi_v(x) = \frac{x}{\sigma_v^2}. \tag{2.82}$$

The maximum likelihood estimate w^{ML} minimizes the cost function

$$J^N(w) \triangleq \frac{1}{N}\sum_{i=0}^{N-1}\big(d(i) - u_i w\big)^2. \tag{2.83}$$

By the weak law of large numbers, as $N \to \infty$, the cost function approaches

$$J(w) \triangleq \mathbb{E}\,(\boldsymbol{d} - \boldsymbol{u}w)^2 \tag{2.84}$$

based on which the LMS algorithm is motivated. That is, the LMS algorithm constitutes an adaptive solution to the ML estimation problem under Gaussian noise.

2. For Laplace noise, the pdf is given by

$$f_{\boldsymbol{v}}\,(v) = \frac{1}{\sqrt{2}\sigma_v} e^{-\frac{\sqrt{2}|v|}{\sigma_v}} \tag{2.85}$$

and the score function by

$$\psi_v(x) = \frac{\sqrt{2}}{\sigma_v}\,\mathrm{sign}(x). \tag{2.86}$$

The maximum likelihood estimate w^{ML} minimizes the cost function

$$J^N(w) \triangleq \frac{1}{N}\sum_{i=0}^{N-1}\left|d(i) - u_i w\right|. \tag{2.87}$$

By the weak law of large numbers, as $N \to \infty$, the cost function approaches

$$J(w) \triangleq \mathbb{E}\,|\boldsymbol{d} - \boldsymbol{u}w| \tag{2.88}$$

based on which the sign-error LMS algorithm is motivated. That is, the sign-error LMS algorithm constitutes an adaptive solution to the ML estimation problem under Laplace noise.

2.3 Robust Estimation

In order to solve the ML estimation problem, the score function $\psi_v(\cdot)$ needs to be computed based on knowledge of the noise distribution. This knowledge is rarely available in practice, if at all. More often than not, the noise is assumed to be Gaussian, usually

for mathematical tractability. An example by Tukey [Tuk60] revealed how an estimator for a distribution parameter based on a Gaussian assumption can prove drastically inefficient when the sample is contaminated by outliers, or the distribution happens to be heavier tailed than the Gaussian, the two being practically synonymous [HR09]. In order to overcome this problem, one may consider a parametric approach, where some non-Gaussian distribution for the noise is assumed (t-distribution or generalized Gaussian distribution, for example), but whose parameters need to be estimated. The problem with this approach is that its range of validity is limited to scenarios where the true noise distribution is close to the one assumed; therefore, it is natural to expect the performance of the resulting ML estimator to be only as good as the underlying modeling assumption. Another approach is semi-parametric or non-parametric estimation of the noise distribution, allowing a rather general representation for the underlying noise model. Gaussian mixture modeling or Middleton Class A modeling, the latter being a form of statistical–physical modeling, are examples for the semi-parametric approach. Kernel density estimation is an example for the non-parametric approach. Yet another approach bridging the parametric on the one hand and the semi- or non-parametric on the other hand is that of robust statistics [ZKCM12]. Distributional robustness is the primary concern: A robust estimator is one that is relatively insensitive to deviations from an underlying nominal distribution, usually, but not necessarily, the Gaussian—in other words, resistant to the presence of outliers. A robust estimator should enjoy reasonably good efficiency at the nominal model and stability, such that small deviations do not degrade performance significantly; however, larger deviations should not result in breakdown [HR09]. The field of robust statistics was pioneered by Tukey [Tuk60], Huber [HR09], and Hampel [HRRS86], and is exemplified by the following two approaches [SV02]:

- the Huber minimax approach – quantitative robustness [Hub64, HR09]; and
- the Hampel approach based on influence functions – qualitative robustness [Ham68, HRRS86].

Huber's minimax approach is summarized here, since the approach developed in this dissertation draws on it. The concept of M-estimation is briefly introduced next as a generalization of ML estimation.

2.3.1 M-estimation

Consider the following batch data model for a general signal in additive noise:

$$d(i) = s_i(\theta) + v(i), \quad i = 0, \ldots, N-1 \tag{2.89}$$

where $\{\boldsymbol{v}(i), 0 \leq i \leq N-1\}$ is a zero-mean i.i.d. noise process with underlying pdf f (shorthand for $f_{\boldsymbol{v}}(v)$ to reduce notational clutter in this section only); and the signal $\{s_i(\theta), 0 \leq i \leq N-1\}$ is parametrized by the unknown vector θ. The parameter θ is to be estimated from the N observations $d(0), \ldots, d(N-1)$. The data model (2.89) can be specialized to the linear regression data model (2.76) by setting $\theta = w^o$ and $s_i(w^o) = u_i w^o$ given known regressors $\{u_i, 0 \leq i \leq N-1\}$.

The ML estimate of θ from the observations is obtained by maximizing the log-likelihood function:

$$\theta^{\mathrm{ML}} \in \arg\min_{\theta} \sum_{i=0}^{N-1} -\ln f\big(d(i) - s_i(\theta)\big). \tag{2.90}$$

Assuming f is differentiable, then θ^{ML} is the solution of the following system of equations

$$\sum_{i=0}^{N-1} \psi\big(d(i) - s_i(\theta)\big) \nabla_\theta s_i(\theta) = 0 \tag{2.91}$$

in terms of the score function $\psi = -\frac{f'}{f}$. In M-estimation, $-\ln f(\cdot)$ is replaced with a function $\rho(\cdot)$ that behaves similarly—see Fig. 2.5 for some symmetric examples. Assuming ρ is differentiable, $\varphi = \rho'$ denoting its derivative, then the M-estimate for θ is obtained by solving the following system of equations:

$$\sum_{i=0}^{N-1} \varphi\big(d(i) - s_i(\theta)\big) \nabla_\theta s_i(\theta) = 0. \tag{2.92}$$

Procedures such as modified residuals or iteratively re-weighted least-squares can be used to solve (2.92) [HR09]. Caution should be practiced when selecting an initial point for the procedures when using redescending M-estimators, for which the score function $\varphi(x)$ returns to zero away from the origin ($x = 0$), such as Tukey's biweight M-estimator or Hampel's redescending M-estimator, plotted in Figs. 2.5h and 2.5j, respectively. One can use a non-robust estimator for θ as initial point, for instance. The properties of M-estimators can be found in [HR09].

Let's focus in particular on the simple location estimation problem, where $s_i(\theta) = \theta \in \mathbb{R}$, $i = 0, \ldots, N-1$. Letting $\boldsymbol{d}$ denote the random variable whose realizations are the observations

$$d(i) = \theta + v(i), \quad i = 0, \ldots, N-1, \tag{2.93}$$

then θ represents the location parameter of the shifted pdf $f(d-\theta)$, which we would like to estimate. Denoted by $\hat{\theta}_N$, where the dependence on the number of samples is made explicit, the M-estimate for θ satisfies

$$\sum_{i=0}^{N-1} \varphi\big(d(i) - \hat{\theta}_N\big) = 0. \tag{2.94}$$

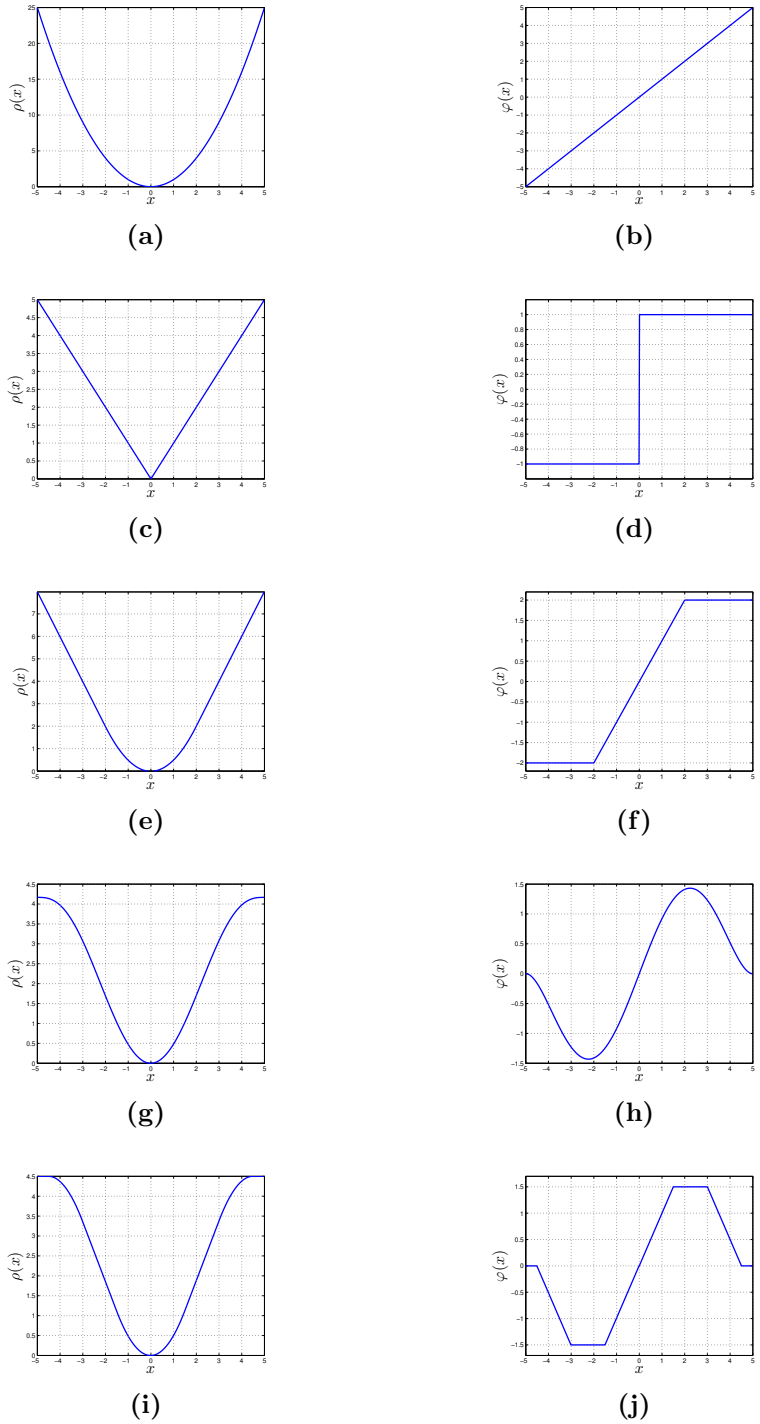


Figure 2.5: Exemplary ρ functions (left) and corresponding φ functions (right).

Under certain conditions on the pdf f and the selected score function φ, $\sqrt{N}\hat{\boldsymbol{\theta}}_N$ was shown to be asymptotically, as $N \to \infty$, normal with asymptotic variance

$$\gamma(f, \varphi) = \frac{\mathbb{E}\,\varphi^2}{(\mathbb{E}\,\varphi')^2}. \tag{2.95}$$

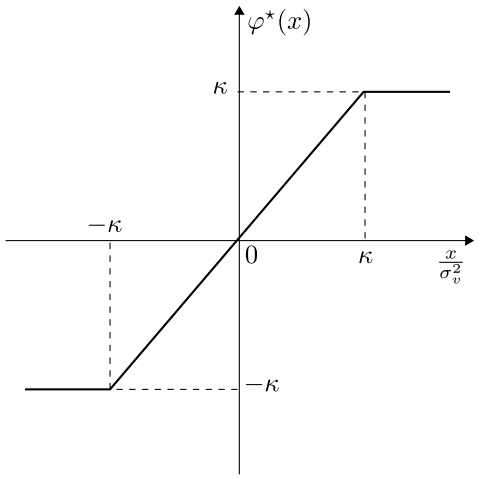


Figure 2.6: Minimax-optimal M-estimation nonlinearity over ε-contaminated Gaussian model.

The minimax approach to robust M-estimation consists in defining a neighborhood around the nominal, typically Gaussian, model, as well as a quantitative performance measure, such as the asymptotic variance of the M-estimator, and minimizing the worst value attained by the measure over the neighborhood. As a choice for neighborhood, consider the following class of ε-contaminated zero-mean Gaussian pdf:

$$\mathcal{F}_\varepsilon = \left\{ f | f = (1-\varepsilon)\,\mathcal{N}(0,\sigma_v^2) + \varepsilon f_C, f_C \in \mathcal{F}_C \right\} \tag{2.96}$$

where $\mathcal{F}_C$ is the class of symmetric contaminating pdfs and $\varepsilon \in [0,1]$ reflects the degree of contamination. The minimax problem is therefore formulated as

$$\min_{\varphi} \max_{f \in \mathcal{F}_\varepsilon} \gamma(f,\varphi) \tag{2.97}$$

and the solution, $(f^\star, \varphi^\star)$, was shown by Huber [Hub64, HR09] to be given by the least favorable density in $\mathcal{F}_\varepsilon$ (that which minimizes the Fisher information matrix [Kay98b]) and the clipping function

$$\varphi^\star(x) = \begin{cases} \frac{x}{\sigma_v^2}, & |x| \le \kappa\sigma_v^2 \\ \kappa\,\mathrm{sign}(x), & |x| > \kappa\sigma_v^2 \end{cases} \tag{2.98}$$

with the parameters κ and ε related by

$$2\frac{f_G(\kappa\sigma_v)}{\kappa\sigma_v} - 2F_G(-\kappa\sigma_v) = \frac{\varepsilon}{1-\varepsilon} \tag{2.99}$$

where f_G and F_G denote the zero-mean Gaussian pdf and distribution with variance σ_v^2, respectively. The clipping function (2.98), plotted in Fig. 2.6, basically bounds the effect of outlying observations.

Robust estimation procedures have enjoyed growing popularity in applications where the measurements are corrupted by impulsive noise. An impulsive noise process can be described as one whose realizations contain sparse, random samples of amplitude much higher than nominally accounted for and, hence, best modeled by heavy-tailed distributions. Impulsive noise may be natural, due to atmospheric phenomena, or man-made, due to either electric machinery present in the operation environment, or multipath telecommunications signals [BKR97, Mid99, ZKCM12, ZB02]. In the next subsection, the connection between M-estimation and robust adaptive filtering is highlighted.

2.3.2 Robust Adaptive Filtering

The approach detailed in the previous section can be carried over, with some variation, to regression—that is, to robustly estimate w^o in (2.76) [HR09]. As far as the popular LMS algorithm for the adaptive estimation of w^o is concerned, the presence of impulsive noise in the measurements degrades the adaptive filter's performance in terms of stability and steady-state behavior [Ber08]. Several LMS-type algorithms have been developed that are robust against impulsive noise, including mixed-norm algorithms [CA97, BPT$^+$03, BMC07], and algorithms that employ normalized LMS (NLMS)-type updates or adjustable step-sizes [VRBT08, GGSB00]. Apart from these algorithms, two other approaches have dominated the literature on robustness to impulsive noise: one based on robust statistics [HR09, HRRS86, ZKCM12], and another based on order statistics [DN03].

The approach based on robust statistics in [ZCN00] replaces the MSE cost function with another appropriately designed error function, resulting in an M-estimation approach. Minimizing the mean Huber's M-estimate error function via stochastic gradient descent methods results in an LMS-type algorithm with an error-clipping nonlinearity of the form (2.98). The error-clipping threshold was computed adaptively and concurrently with the algorithm in order to track the time-varying statistics of the non-stationary error signal, relying on past estimates [ZCN00].

In comparison, the approach based on order statistics applies, at each time index, a mean, median, or α-trimmed mean filter on a window of current and past data in order to compute the LMS-type update at that particular time index [HC92]. This amounts to data smoothing, which alleviates the impact of impulsive noise.

A recurrent feature of the LMS-type algorithms outlined so far is that their updates are generally nonlinear functions of the error signal. Nonlinear processing thus presents itself as critical in combating the effects of impulsive noise.

The choice of the nonlinearity in the cited works does not generally ascribe to optimality criteria. The problem of optimal nonlinearity design was addressed in Sec. 2.2.4. Optimal design techniques, however, are hampered by their prerequisite of exact knowledge of the noise probability density function, which is rarely available in practice. In Ch. 3, a robust adaptive filtering algorithm is developed that estimates semi-parametrically the optimal error nonlinearity *jointly* with the parameter of interest for improved stability and steady-state performance in impulsive noise environments. Ultimately, the algorithm is of the form:

$$e(i) = d(i) - u_i w_{i-1} \tag{2.100a}$$

$$w_i = w_{i-1} + \mu u_i^T h_i\big(e(i)\big) \tag{2.100b}$$

with the understanding that $h_i\big(e(i)\big)$ is a time-varying, adaptive error nonlinearity. It is important to mention here that the analysis of the resulting algorithm cannot be conducted in a similar fashion as in [ANS01, ANS03, Say03]. One complication that arises from the coupling of the two estimation problems is the difficulty in evaluating the two moments

$$\mathbb{E}\, h_i^2\big(\boldsymbol{e}(i)\big) \quad \text{and} \quad \mathbb{E}\, h_i'\big(\boldsymbol{e}(i)\big) \tag{2.101}$$

in closed form using the integral expression in (2.65). The success of the analysis of the original algorithm (2.12), reproduced in Sec. 2.2.3, actually hinges on the ability to evaluate the integral in closed form for a number of error nonlinearities and noise distributions. Therefore, in Ch. 3, linearization analysis will be conducted to derive stability bounds and steady-state performance expressions under certain conditions. The results will be verified numerically through simulations. In Chs. 4 and 5, the robust adaptive filtering algorithm is extended to solve the problem of robust distributed estimation and detection over adaptive networks. Adaptive networks are introduced in the next section.

2.4 Distributed Adaptation and Learning Over Networks

2.4.1 Network Model

The focus in this dissertation is on connected networks composed of N nodes. A connected network is one where each pair of nodes in the network are connected by at least one path. A path may possibly span multiple hops through intermediate nodes. A pair of nodes that are connected to one another directly through a single-hop path are referred to as neighbors. The neighborhood of node k, $k = 1, \ldots, N$, is the set of

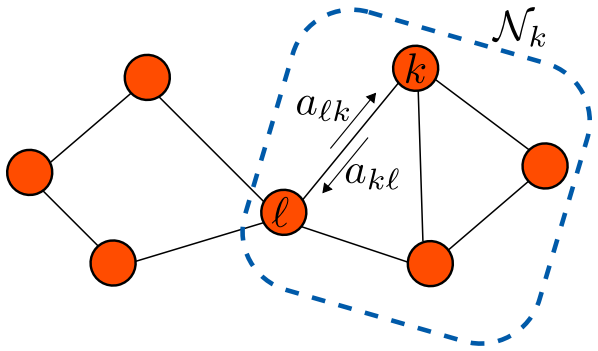


Figure 2.7: Exemplary connected network highlighting node k's neighborhood and the weights $\{a_{\ell k}, a_{k\ell}\}$ used to scale the data exchanged between nodes k and l.

all nodes, including itself, that it is connected to. It is denoted as $\mathcal{N}_k$. The degree of node k is the cardinality of its neighborhood, denoted by $n_k = |\mathcal{N}_k|$. The network topology is described by a graph with N vertices representing the nodes and a set of edges representing the links connecting neighbors with each another. The graph is assumed to be undirected so that if node k is a neighbor of node ℓ, then node ℓ is also a neighbor of node k, $k, \ell = 1, \ldots, N$.

Only neighbors are able to exchange data with each other over the link connecting them. The data exchange between neighboring nodes k and l is governed by a pair of nonnegative scalar weights $\{a_{\ell k}, a_{k\ell}\}$, where $a_{\ell k}$ designates the weight used by node k to scale the data it receives from node ℓ and $a_{k\ell}$ designates the weight used by node ℓ to scale the data it receives from node k. The weights $\{a_{\ell k}, a_{k\ell}\}$ may be different, and one or both may be zero. They can be interpreted as the levels of confidence nodes k and l attach to one another's data. Effectively, data exchange between neighbors may be bidirectional, unidirectional, or non-existent [STC+13, Say14b, Say14a].

Fig. 2.7 illustrates an exemplary connected network and highlights node k's neighborhood as well as the weights $\{a_{\ell k}, a_{k\ell}\}$ used to scale the data exchanged between nodes k and l. Despite the absence of self-loops, each node is its own neighbor.

2.4.2 Data Model and Problem Formulation

At each time index $i \geq 0$, each node k in the network has access to a noisy real-valued scalar measurement $d_k(i)$ relating to an unknown deterministic real-valued $M \times 1$ parameter vector w^o. The measurements are related to the parameter via a stochastic linear regression model of the form:

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^o + \boldsymbol{v}_k(i), \quad i \geq 0 \tag{2.102}$$

where $\boldsymbol{u}_{k,i}$ is a real-valued known row regression vector, or regressor, of size M; and $\boldsymbol{v}_k(i)$ is real-valued scalar measurement noise. The joint random process $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}, \boldsymbol{v}_k(i)\}$ is assumed to be zero-mean wide-sense stationary with the following properties:

- The regressors $\{\boldsymbol{u}_{k,i}\}$ are spatially and temporally independent. The covariance of the regressor $\boldsymbol{u}_{k,i}$ is denoted as $R_{u,k}$ and is positive definite, i.e., $R_{u,k} > 0$.
- The noise random variables $\{\boldsymbol{v}_k(i)\}$ are spatially and temporally independent. The variance of the noise random variable $\boldsymbol{v}_k(i)$ is denoted as $\sigma^2_{v,k}$.
- The random variables $\boldsymbol{u}_{k,i}$ and $\boldsymbol{v}_\ell(j)$ are independent for all k, ℓ, i, and j.

The aim is for each node k to adaptively estimate the weight vector w^o, availing itself of its own streaming data as well as its neighbors': $\left\{\{d_\ell(i), u_{\ell,i}\}, \ell \in \mathcal{N}_k\right\}$. While each node can individually run an instance of an adaptive filtering algorithm of the form (2.12), for example, it is to be expected that cooperation among the nodes can be beneficial, reducing the effects of gradient noise inherent to such an algorithm on the quality of each node's estimate. This aspect will become clear in the next section.

2.4.3 Diffusion Adaptation Algorithms

In order for each node to estimate the weight vector w^o in cooperation with its neighbors based on the adaptive filtering algorithm with error nonlinearity (2.12) studied in Sec. 2.2, one so-called *diffusion strategy* that the nodes can employ is the *Adapt-then-Combine* (ATC) diffusion strategy. First, consider the nonnegative scalar weights $\{a_{\ell k}\}$ introduced in (2.4.1) and assume they are chosen by the designer to satisfy the following properties, for each node $k = 1, \ldots, N$:

$$a_{\ell k} \geq 0, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \quad \sum_{\ell \in \mathcal{N}_k} a_{\ell k} = 1. \tag{2.103}$$

Collecting the weights $\{a_{\ell k}\}$ into an $N \times N$ matrix A such that the kth column of the matrix consists of the weights $\{a_{\ell k}, \ell = 1, \ldots, N\}$, then the last property in (2.103) translates into saying that the entries of each column of the matrix A add up to one:

$$\mathbb{1}^T A = \mathbb{1}^T. \tag{2.104}$$

Since its entries are also nonnegative, the matrix A is then left-stochastic. Now, the ATC diffusion strategy, illustrated in Fig. 2.8, is presented. For $i \geq 0$, starting from

Table 2.3: Combination Policy Examples

Name	**Rule** $(\ell \in \mathcal{N}_k,\ \ell \neq k)^*$
Uniform	$a_{\ell k} = 1/n_k$
Laplacian	$a_{\ell k} = 1/n_{\max}$
Maximum-Degree	$a_{\ell k} = 1/N$
Metropolis	$a_{\ell k} = 1/\max(n_k, n_\ell)$
Relative-Degree	$a_{\ell k} = n_\ell / \sum_{m\in\mathcal{N}_k} n_m$
Relative Degree-Variance	$a_{\ell k} = n_\ell \sigma_{v,\ell}^{-2} / \sum_{m\in\mathcal{N}_k} n_m \sigma_{v,m}^{-2}$
Relative-Variance	$a_{\ell k} = \gamma_\ell^{-2} / \sum_{m\in\mathcal{N}_k} \gamma_m^{-2}$
Adaptive Relative-Variance	$a_{\ell k}(i) = \widehat{\gamma}_{\ell k}^{-2}(i) / \sum_{m\in\mathcal{N}_k} \widehat{\gamma}_{mk}^{-2}(i)$

* For all rules, $\forall\, k$, $a_{\ell k} = 0$, if $\ell \notin \mathcal{N}_k$; $a_{kk} = 1 - \sum_{\ell=1}^{N} a_{\ell k}$.

some initial condition $w_{k,-1}$, each node k updates its previous estimate $w_{k,i-1}$ for the weight vector w^o using the following update equations:

$$\textbf{ATC diffusion :} \left\{ \begin{array}{rcl} e_k(i) & = & d_k(i) - u_{k,i} w_{k,i-1} \\ \psi_{k,i} & = & w_{k,i-1} + \mu_k u_{k,i}^T h_k\big(e_k(i)\big) \\ w_{k,i} & = & \sum\limits_{\ell\in\mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{array} \right. \tag{2.105}$$

where $h_k(\cdot)$ is node k's error nonlinearity, and μ_k is its step-size parameter. Basically, at each time index i, each node k updates its current estimate $w_{k,i-1}$ in an adaptive filtering fashion through the output error $e_k(i)$ in terms of its own data $u_{k,i}$ and $d_k(i)$, forming an intermediate estimate, $\psi_{k,i}$. Each node k then collects the intermediate estimates from its neighbors in $\mathcal{N}_k$, and weights them according to some *combination policy* satisfying (2.103), hence forming the final estimate $w_{k,i}$. One property of a left-stochastic combination policy that is crucial to the behavior of the diffusion adaptation algorithm (2.105), as will be appreciated in Chs. 4 and 5, is that its spectral radius (maximum-magnitude eigenvalue) is one. Table 2.3 lists some examples of combination policies. In the second row from the bottom of the table, $\gamma_k \triangleq \mu_k \sigma_{v,k}^2 \operatorname{Tr}(R_{u,k})$ for the relative-variance rule, and in the last low, this quantity is estimated adaptively by each node for each of its neighbors, giving rise to the adaptive relative-variance rule—see [Say14a] for implementation details. Note that setting $A = I$ entails that each node revert to non-cooperative, stand-alone adaptation.

The algorithm (2.105) can be seen as a special case of an algorithm employing a generic

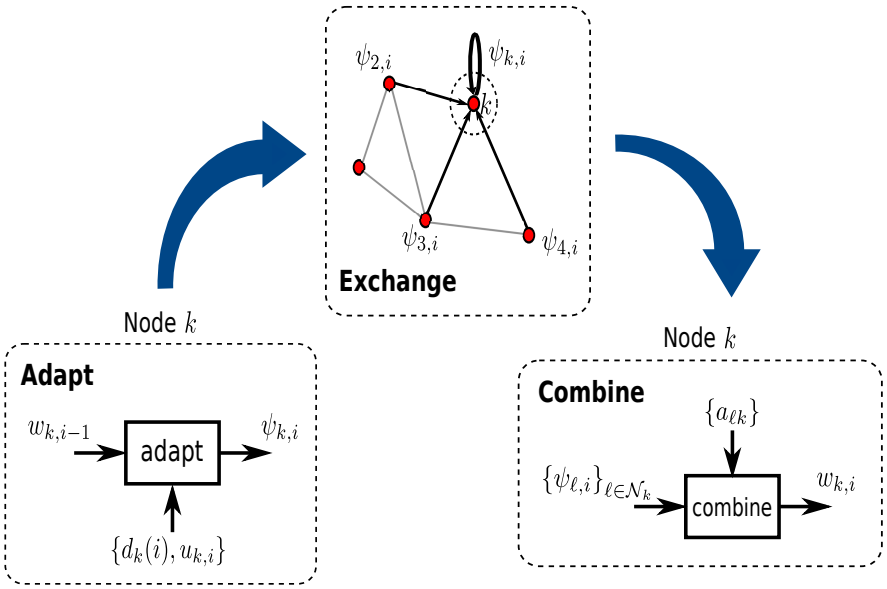


Figure 2.8: Adapt-then-combine (ATC) diffusion strategy.

update vector in the adaptation step:

$$\left\{\begin{array}{rcl}\psi_{k,i} &=& w_{k,i-1} - \mu_k \hat{s}_{k,i}(w_{k,i-1}) \\ w_{k,i} &=& \displaystyle\sum_{\ell\in\mathcal{N}_k} a_{\ell k}\psi_{\ell,i}\end{array}\right. \tag{2.106}$$

The generic update vector $\hat{s}_{k,i}(w_{k,i-1})$ may represent the stochastic gradient of a local, agent-specific cost function in a multi-agent network, for example. The algorithm (2.106) was studied extensively in [CS15a, CS15b]. According to these works, one of the prerequisites for the estimates $\{\boldsymbol{w}_{k,i}\}$ to converge to the desired weight vector w^o in the mean-square sense is that there exist an $M \times 1$ deterministic vector function $s_k(w)$ such that, for all $M \times 1$ vectors $\boldsymbol{w}$ in the filtration $\boldsymbol{\mathcal{F}}_{i-1}$ generated by the past history of iterates $\{\boldsymbol{w}_{k,j}\}$ for $j \leq i-1$ and all k, the following holds:

$$\mathbb{E}\,\{\hat{\boldsymbol{s}}_{k,i}(\boldsymbol{w})|\boldsymbol{\mathcal{F}}_{i-1}\} = s_k(\boldsymbol{w}) \tag{2.107a}$$

$$s_k(w^o) = 0 \tag{2.107b}$$

The works [CS15a, CS15b] are far richer, however, in that they study the behavior of the generic algorithm (2.106) when it does not necessarily hold that a solution $w^\star$ exists such that $s_k(w^\star) = 0$ for all k. In particular, the limit point of the estimates $\{\boldsymbol{w}_{k,i}\}$ is characterized in [CS15a,CS15b], as well as the stability, learning behavior and rate, and steady-state performance of the algorithm. In addition to that, two other distributed, cooperative strategies are considered, namely, the adapt-then-combine (CTA) diffusion strategy and the consensus strategy. Interestingly, it is shown that these strategies enable each node in the network, through local interactions and in-network processing, to achieve the same level of performance as that of a centralized strategy corresponding to a fully connected network. The focus in this dissertation, however, is on the ATC diffusion strategy since it has been shown to outperform the others [TS12, Say14a].

A special case of both (2.105) and (2.106), by setting $h_k\big(e_k(i)\big) = e_k(i)$ and $\hat{s}_{k,i}(w_{k,i-1}) = -u_{k,i}^T\,[d_k(i) - u_{k,i}w_{k,i-1}]$, is the *ATC diffusion LMS* algorithm, with the following update equations:

$$\left\{\begin{array}{rcl} \psi_{k,i} &=& w_{k,i-1} + \mu_k u_{k,i}^T\,[d_k(i) - u_{k,i}w_{k,i-1}] \\ w_{k,i} &=& \sum\limits_{\ell\in\mathcal{N}_k} a_{\ell k}\psi_{\ell,i} \end{array}\right. \tag{2.108}$$

Naturally, there are also the CTA diffusion and consensus variants of this algorithm. All three strategies were studied extensively in [Say14b,Say14a]. The benefit of cooperation among the nodes of the network manifests itself clearly in LMS adaptation using (2.108), for example. Let's assume the data $\{d_k(i), u_{k,i}\}$ satisfy the model (2.102) but that the covariance matrices $\{R_{u,k}\}$ are not positive definite for any of the nodes, i.e., $R_{u,k} > 0$ does not hold for any $k = 1, \ldots, N$. Then, a node running the LMS algorithm individually may not be able to uniquely estimate the desired weight vector w^o since there are infinite solutions to the normal equations

$$R_{u,k}w^o = r_{du,k} \tag{2.109}$$

in this case and only partial information available locally at each node k, where $r_{du,k} \triangleq \mathbb{E}\,\boldsymbol{d}_k(i)\boldsymbol{u}_{k,i}^T$. However, it was shown in [Say14a], for example, that it need only hold that $\sum_{k=1}^N R_{u,k} > 0$, a global observability condition, for all the nodes to be able to recover w^o uniquely if they run a distributed, cooperative strategy such as (2.108).

The choice $\hat{s}_{k,i}(w_{k,i-1}) = -u_{k,i}^T\,[d_k(i) - u_{k,i}w_{k,i-1}]$ in the algorithm (2.106) can be viewed as setting the update vector to the transposed stochastic gradient of a local MSE cost function:

$$J_k(w) \triangleq \mathbb{E}\big(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}w\big)^2. \tag{2.110}$$

As a matter of fact, as shown in [Say14b, CS12, Say14a], for data $\{d_k(i), u_{k,i}\}$ satisfying the model (2.102), the estimates $\{\boldsymbol{w}_{k,i}\}$ arising from (2.108) converge in the mean-square sense to the unique minimizer of the global MSE cost

$$J^{\rm glob}(w) \triangleq \sum_{k=1}^N \mathbb{E}\big(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}w\big)^2, \tag{2.111}$$

which is the desired weight vector w^o. As hinted at before, costs more general than the MSE cost can be accommodated as well by setting

$$\hat{s}_{k,i}(w_{k,i-1}) = \Big[\widehat{\nabla_w J_k}(w_{k,i-1})\Big]^T \tag{2.112}$$

where $\widehat{\nabla_w J_k}(w)$ is an approximate stochastic gradient of some general local cost function $J_k(w)$. If each of the local cost functions is minimized by the weight vector w^o, and if the global cost

$$J^{\rm glob}(w) \triangleq \sum_{k=1}^N J_k(w) \tag{2.113}$$

is strongly convex, such that the weight vector w^o is its unique minimizer, then the algorithm

$$\left\{\begin{array}{rcl} \psi_{k,i} & = & w_{k,i-1} - \mu_k \widehat{\nabla_w J_k}(w_{k,i-1}) \\ w_{k,i} & = & \sum\limits_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{array}\right. \tag{2.114}$$

generates estimates $\{\boldsymbol{w}_{k,i}\}$ that converge in the mean-square sense to w^o [CS15a, CS15b].

LMS-based diffusion strategies for distributed adaptation can also see their performance degrade in the presence of impulsive noise. In [CST11], a robust diffusion estimation algorithm was realized by using the adaptive projected subgradient method [YO05, TSY11], culminating in a combine-project-adapt protocol, where the output errors at each node are projected onto halfspaces defined by Huber's M-estimate error function (2.98). For the method's implementation, however, some parameters need to be tuned in accordance with the practitioner's knowledge of the noise distribution. In the course of this dissertation, this knowledge is assumed to be lacking, prompting a more robust construction.

The robust algorithm where the optimal nonlinearity is estimated from streaming data, jointly with the parameter of interest w^o, of the form (2.100), can also be embedded into an ATC diffusion strategy of the form

$$\left\{\begin{array}{rcl} e_k(i) & = & d_k(i) - u_{k,i} w_{k,i-1} \\ \psi_{k,i} & = & w_{k,i-1} + \mu_k u_{k,i}^T \hat{h}_{k,i}\big(e_k(i)\big) \\ w_{k,i} & = & \sum\limits_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{array}\right. \tag{2.115}$$

Such an algorithm is developed and analyzed in Ch. 4, harnessing the powerful techniques devised throughout the cited literature. Though of the generic algorithmic form (2.106), the algorithm (2.115) cannot be analyzed in as similar a fashion as in [CS15a, CS15b]. One complication that arises from the coupling of the two estimation problems is the difficulty in identifying the aforementioned deterministic vector function $s_k(w)$ and its properties, a feature that has facilitated the analysis of the algorithm (2.106) considerably. As with the stand-alone robust counterpart developed and analyzed in Ch. 3, linearization analysis will be conducted to derive stability bounds and steady-state performance expressions under certain conditions. The results will be verified numerically through simulations.

Chapter 3
Robust Adaptation for Single Agents

In this chapter, a robust adaptive filtering algorithm is developed that estimates semi-parametrically the mean-square-error–optimal error nonlinearity (2.69) jointly with the parameter of interest for improved stability and steady-state performance in impulsive noise environments. Comprehensive theoretical performance analysis as well as numerical simulation of the resulting robust adaptive rule are conducted.

In Sec. 3.1, the robust adaptive filtering algorithm is developed. In Sec. 3.2, mean and mean-square analysis of the algorithm's performance is conducted using the energy conservation framework [ANS03, Say03]. In Sec. 3.3, simulation results are presented. Conclusions are drawn in Sec. 3.4.[1]

3.1 Robust Adaptive Filtering

3.1.1 Data Model and Problem Formulation

The goal is to adaptively estimate an unknown deterministic real-valued $M \times 1$ parameter w^o from available data $\big\{\{d(i), u_i\}, i \geq 0\big\}$. The data are related to w^o via the linear regression model:

$$d(i) = u_i w^o + v(i) \tag{3.1}$$

where the $\{d(i)\}$ are real-valued scalar measurements, and the $\{u_i\}$ are real-valued row regression vectors of size M. The data $\{d(i), u_i\}$ arise from realizations of jointly wide-sense stationary zero-mean random processes $\{\boldsymbol{d}(i), \boldsymbol{u}_i\}$. The regressors have covariance matrix $R_u = \mathbb{E}\,\boldsymbol{u}_i^T \boldsymbol{u}_i > 0$, while the noise process $\{\boldsymbol{v}(i)\}$ is a real-valued zero-mean impulsive white process with variance σ_v^2. It is assumed that the noise probability density function (pdf), $f_{\boldsymbol{v}}(v)$, is symmetric, i.e., $\mathbb{E}\,\boldsymbol{v}^{2p-1}(i) = 0$, $p = 1, 2, \ldots$. The random variables $\boldsymbol{u}_i$ and $\boldsymbol{v}(j)$ are assumed to be independent for all i and j.

The least-mean-squares (LMS) filter is a stochastic gradient algorithm based on minimizing the mean-square-error (MSE) cost function:

$$J(w) \triangleq \mathbb{E}\big(\boldsymbol{d}(i) - \boldsymbol{u}_i w\big)^2. \tag{3.2}$$

[1] This chapter is based on the journal article:
S. Al-Sayed, A. M. Zoubir, and A. H. Sayed, "Robust adaptation in impulsive noise," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2851–2865, Jun. 2016.

With the output error defined as $e(i) \triangleq d(i) - u_i w_{i-1}$, the LMS recursion is given by

$$w_i = w_{i-1} + \mu u_i^T e(i), \quad i \geq 0 \tag{3.3}$$

where μ is a small positive step-size parameter.

In robust adaptive filtering [ZCN00,GGSB00,VRBT08], the cost function (3.2) is modified to

$$J^\rho(w) \triangleq \mathbb{E}\,\rho\big(\boldsymbol{d}(i) - \boldsymbol{u}_i w\big) \tag{3.4}$$

where $\rho\colon \mathbb{R} \to \mathbb{R}$ is some M-estimate (maximum-likelihood–type) function [HR09]. Assuming $\rho(x)$ is differentiable, the steepest-descent recursion that attempts to minimize (3.4), subject to a suitable choice of the initial condition, takes the form:

$$w_i = w_{i-1} - \mu\big(\nabla_w J^\rho(w_{i-1})\big)^T. \tag{3.5}$$

Let $h(x) \triangleq \frac{d\rho(x)}{dx}$, referred to as the score function. Qualitative robustness is ensured if the score function $h(x)$ is bounded and continuous [HR09]. This means that small changes in x do not lead to big changes in $h(x)$. By forgoing the expectation in (3.5), the resulting stochastic instantaneous approximation of (3.5) is

$$w_i = w_{i-1} + \mu u_i^T h\big(e(i)\big). \tag{3.6}$$

The LMS recursion (3.3) is recovered when $\rho(x) = \frac{x^2}{2}$. It was shown in [DM94] that the optimal score function that minimizes the steady-state MSE is

$$h_1^{\mathrm{opt}}(x) = -\frac{f'_{\boldsymbol{v}}(x)}{f_{\boldsymbol{v}}(x)} \tag{3.7}$$

where the notation $g'(x)$ stands for $\frac{dg(x)}{dx}$. In this case, the LMS algorithm is MSE-optimal when $\{\boldsymbol{v}(i)\}$ is Gaussian, with the $\frac{1}{\sigma_v^2}$ proportionality constant absorbed into the step-size parameter μ. However, the LMS algorithm is suboptimal when the noise is non-Gaussian [Say03,SV02]. Yet, in order to design the filter optimally using (3.7), the noise pdf must be known exactly, which rarely holds in practice. Under less restrictive assumptions, the authors in [ANS01] derived an optimal score function that holds over a wider range of adaptation and not only at steady-state, leading to the choice:

$$h_{2,i}^{\mathrm{opt}}(x) = -\frac{f'_{\boldsymbol{e}(i)}(x)}{f_{\boldsymbol{e}(i)}(x)} \tag{3.8}$$

in terms of the pdf of the error signal, $\boldsymbol{e}(i)$. This function is more intuitive in an adaptive setting, and reduces to $h_1^{\mathrm{opt}}(x)$ at steady-state.

In [TBG00] and [BZ02], in the context of *offline* robust estimation, where the practitioner has access to a batch of data, the optimal score function $h_{2,i}^{\text{opt}}\big(e(i)\big)$ was approximated by an iteration-dependent function, $h_i\big(e(i)\big)$, that is a linear combination of B preselected basis functions:

$$h_i\big(e(i)\big) = \alpha_i^T \varphi_i \tag{3.9}$$

where

$$\alpha_i \triangleq [\alpha_i(1), \ldots, \alpha_i(B)]^T \tag{3.10}$$

is the vector of combination weights, and

$$\varphi_i \triangleq \big[\phi_1\big(e(i)\big), \ldots, \phi_B\big(e(i)\big)\big]^T \tag{3.11}$$

is the vector of basis functions evaluated at the residual error. The vector α_i is chosen to minimize the MSE between the true and approximate score functions:

$$\alpha_i^{\text{opt}} \triangleq \arg\min_{\alpha_i} \; \mathbb{E}\left(h_{2,i}^{\text{opt}}\big(\boldsymbol{e}(i)\big) - h_i\big(\boldsymbol{e}(i)\big)\right)^2. \tag{3.12}$$

In the online adaptive context pertinent to this work, it is imperative to compute α_i adaptively and jointly with w_i. This is treated in Sec. 3.1.2, where in the process of deriving the adaptive update for α_i the condition

$$\mathbb{E}\,\phi_b(x) h_{2,i}^{\text{opt}}(x) = \mathbb{E}\,\phi_b'(x) \tag{3.13}$$

for any b will be exploited. Condition (3.13) follows from integration by parts of the left-hand side of (3.13) and using (3.8), under the assumption (see Appendix A.4 for the derivation):

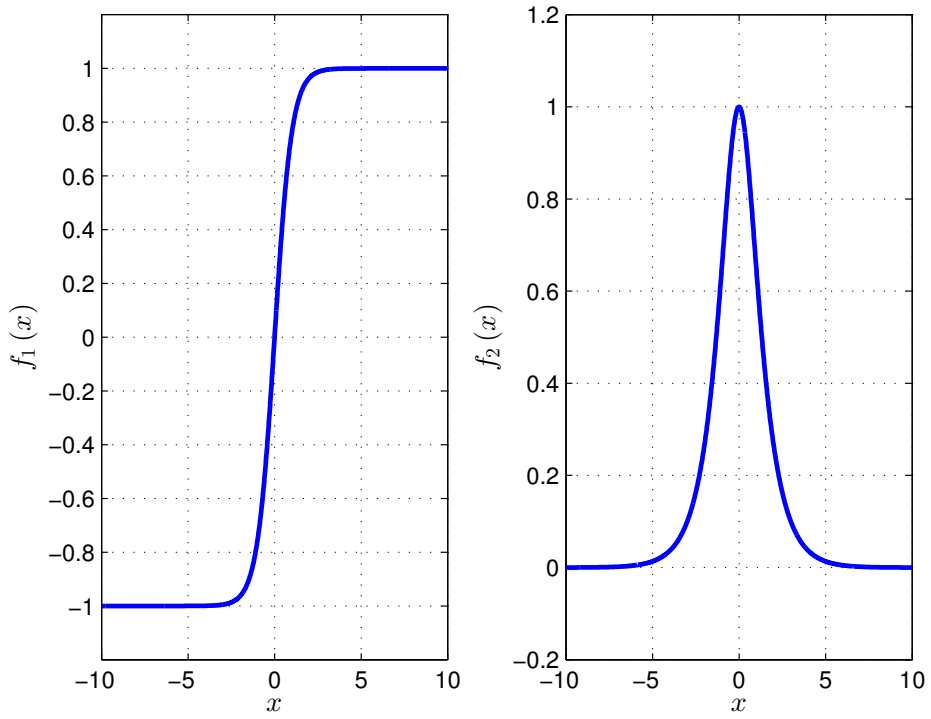
$$\lim_{x \to \pm\infty} \phi_b(x) f_{\boldsymbol{e}(i)}(x) = 0. \tag{3.14}$$

The choice of basis functions should conform to prior knowledge about the nature of the noise in the data model (3.1) [TBG00, BZ02, HR09, HRRS86], if available. Since it is known the noise can be of an impulsive nature, a sensible choice that scales down impulsive samples and trades off robustness with LMS performance under Gaussian noise would be $\phi_1(x) = x$ and $\phi_b(x)$, $b = 2, \ldots, B$, some bounded nonlinear functions. One example would be the hyperbolic tangent basis:

$$\phi_b(x) = \tanh\big((b-1)x\big), \quad b = 2, \ldots, B. \tag{3.15}$$

where $\tanh(\cdot)$ above and $\operatorname{sech}(\cdot)$ for future reference denote the hyperbolic tangent and secant functions, respectively. Both functions are plotted in Fig. 3.1. Replacing $h(x)$ in (3.6) by the approximation of the optimal score function in (3.9), the recursion now becomes

$$w_i = w_{i-1} + \mu u_i^T \sum_{b=1}^{B} \alpha_i(b) \phi_b\big(e(i)\big). \tag{3.16}$$



(a) (b)

Figure 3.1: Function plots: (a) hyperbolic tangent: $f_1(x) = \tanh(x)$ and (b) hyperbolic secant: $f_2(x) = \mathrm{sech}(x)$.

Some remarks are in order before proceeding with the development of the robust algorithm. First, the entries of α_i will be required to be nonnegative. This is because the error nonlinearity in (3.6) or (3.9) should be, generally, sign-preserving so that successive iterates descend the error surface [Set92, TC96, ANSK00]. Moreover, the authors of [BZ02] imposed a convexity constraint on the entries of α_i for $B > 1$ in the offline estimation context, illustrating performance gains. A similar convexity constraint on the entries of α_i will be considered, i.e., the entries of α_i will be nonnegative and add up to 1.

Remark 1. In the work [ASZS13], a more restrictive choice was considered for the basis functions other than $\phi_1(x) = x$ and (3.15). Specifically, B basis functions that arise from zero-mean Gaussian pdfs with distinct variances were selected:

$$\phi_b(x; \sigma_b^2) = \frac{x}{\sigma_b^2}, \quad b = 1, \ldots, B. \tag{3.17}$$

Let

$$s \triangleq \varphi_i' = \left[\frac{1}{\sigma_1^2}, \ldots, \frac{1}{\sigma_B^2}\right]^T. \tag{3.18}$$

Replacing $h(x)$ in (3.6) by the approximation (3.9), and using (3.17) and (3.18), the recursion would instead become

$$w_i = w_{i-1} + \mu\left(\alpha_i^T s\right) u_i^T e(i). \tag{3.19}$$

Observe that the resulting recursion amounts to an LMS implementation with a variable step-size (VSS-LMS), where $\mu(i) \triangleq \mu\left(\alpha_i^T s\right)$. This algorithm is referred to it as RVSS-LMS, with "R" standing for robust. Several VSS-LMS variants have been developed in the literature in order to improve the tradeoff between misadjustment and convergence rate compared to LMS [KJ92, AM97, PC99, SSS04, ZLCH08]. In [ASZS13], on the other hand, the variable step-size was designed with the intent of enhancing the robustness of the LMS filter against impulsive noise. The selection of $\{\sigma_b^2\}$ to aptly model the impulsive noise necessitates knowledge of the impulsive noise variance. If this knowledge is not available, then prior to running RVSS-LMS, the noise variance has to be estimated and $\{\sigma_b^2\}$ chosen over some appropriate range.

3.1.2 Joint Parameter Adaptation

In this section, a technique from [TYS10] is applied to solve (3.12) adaptively, subject to the aforementioned constraints on α_i. Let $\Omega_+ \triangleq \left\{\alpha \in \mathbb{R}_+^B | \alpha^T \mathbb{1} = 1\right\}$, where $\mathbb{R}_+^B$ is the set of $B \times 1$ vectors on the set of nonnegative real numbers $\mathbb{R}_+$ and $B > 1$. The case $B = 1$ will be addressed later. We seek the solution to the following convex optimization problem:

$$\min_{\alpha \in \Omega_+} \mathbb{E}\Big(h_{2,i}^{\text{opt}}\big(\boldsymbol{e}(i)\big) - \boldsymbol{\varphi}_i^T \alpha\Big)^2. \tag{3.20}$$

We would like to transform (3.20) into a more tractable form that eliminates the constraints. First, we ignore the nonnegativity constraint on the entries of α and later adjust the solution to accommodate this requirement. Hence, let $\Omega \triangleq \left\{\alpha \in \mathbb{R}^B | \alpha^T \mathbb{1} = 1\right\}$, and introduce the projection operator $\mathcal{P}_\Omega$ from $\mathbb{R}^B$ onto Ω:

$$\mathcal{P}_\Omega(\beta) = \left(I - \frac{\mathbb{1}\mathbb{1}^T}{B}\right)\beta + \frac{\mathbb{1}}{B} \quad \forall\, \beta \in \mathbb{R}^B. \tag{3.21}$$

Every $\alpha \in \Omega$ can be represented as $\alpha = \mathcal{P}_\Omega(\beta)$ for some $\beta \in \mathbb{R}^B$. We are therefore motivated to introduce the unconstrained optimization problem:

$$\min_{\beta \in \mathbb{R}^B} J(\beta) \triangleq \mathbb{E}\Big(h_{2,i}^{\text{opt}}\big(\boldsymbol{e}(i)\big) - \boldsymbol{\varphi}_i^T \mathcal{P}_\Omega(\beta)\Big)^2. \tag{3.22}$$

Let

$$\Pi \triangleq I - \frac{\mathbb{1}\mathbb{1}^T}{B}. \tag{3.23}$$

The gradient of the cost function $J(\beta)$ is given by

$$\begin{aligned}\nabla_\beta J(\beta) &= -2\mathbb{E}\left[\left(h_{2,i}^{\text{opt}}\big(\boldsymbol{e}(i)\big) - \boldsymbol{\varphi}_i^T \mathcal{P}_\Omega(\beta)\boldsymbol{\varphi}_i^T\Pi\right]\right. \\ &= -2\left\{\mathbb{E}\left[h_{2,i}^{\text{opt}}\big(\boldsymbol{e}(i)\big)\boldsymbol{\varphi}_i^T\right] - \mathcal{P}_\Omega^T(\beta)\left(\mathbb{E}\,\boldsymbol{\varphi}_i\boldsymbol{\varphi}_i^T\right)\right\}\Pi \\ &= 2\left[\mathcal{P}_\Omega^T(\beta)R_{\varphi_i} - \mathbb{E}\,\boldsymbol{\varphi}_i'^T\right]\Pi \end{aligned} \tag{3.24}$$

where we have appealed to (3.13), $R_{\varphi_i} \triangleq \mathbb{E}\,\boldsymbol{\varphi}_i\boldsymbol{\varphi}_i^T$, and the vector φ_i' is given by

$$\varphi_i' = \text{col}\left\{\phi_1'(i), \ldots, \phi_B'(i)\right\}. \tag{3.25}$$

The steepest-descent recursion that solves (3.22) is therefore of the form:

$$\left\{\begin{array}{rcl}\beta_i &=& \beta_{i-1} - 2\tau(i)\Pi\left[R_{\varphi_i}\mathcal{P}_\Omega(\beta_{i-1}) - \mathbb{E}\,\boldsymbol{\varphi}_i'\right] \\ \alpha_i &=& \mathcal{P}_\Omega(\beta_i)\end{array}\right. \tag{3.26}$$

where $\tau(i)$ is a nonnegative step-size sequence, the computation of which is discussed further ahead. Note that if β_{-1} is chosen from Ω, then it is ensured that $\beta_i \in \Omega$ for all i. This follows from the fact that, for any vector x of size B, it holds that $\mathbb{1}^T\Pi x = 0$. Therefore, the recursion in (3.26) becomes

$$\alpha_i = \alpha_{i-1} - 2\tau(i)\Pi\left(R_{\varphi_i}\alpha_{i-1} - \mathbb{E}\,\boldsymbol{\varphi}_i'\right), \quad \alpha_{-1} \in \Omega. \tag{3.27}$$

The moments R_{φ_i} and $\mathbb{E}\,\boldsymbol{\varphi}_i'$ in (3.27) may be estimated concurrently by means of the following smoothing recursions:

$$\widehat{R}_{\varphi_i} = \nu\widehat{R}_{\varphi_{i-1}} + (1-\nu)\,\varphi_i\varphi_i^T \tag{3.28}$$

$$\widehat{\varphi}_i' = \nu\widehat{\varphi}_{i-1}' + (1-\nu)\,\varphi_i' \tag{3.29}$$

with $\nu \in (0,1)$ and usually close to one. In this case, we replace (3.27) by

$$\alpha_i = \alpha_{i-1} - 2\tau(i)\Pi\left(\widehat{R}_{\varphi_i}\alpha_{i-1} - \widehat{\varphi}_i'\right), \quad \alpha_{-1} \in \Omega. \tag{3.30}$$

We are now in a position to incorporate the nonnegativity constraint on the entries of α_i at each iteration. One way to accomplish this task *approximately* is to start from an initial condition $\alpha_{-1} \in \Omega_{++} \triangleq \left\{\alpha \in \mathbb{R}_{++}^B | \alpha^T\mathbb{1} = 1\right\}$, where $\mathbb{R}_{++}^B$ is the set of $B \times 1$ vectors on the set of positive real numbers $\mathbb{R}_{++}$, and construct the step-size sequence in (3.30) as follows:

$$\tau(i) \triangleq \tau \cdot \left(\frac{\min\left\{\alpha_{i-1}(b) | 1 \le b \le B\right\}}{\left\|2\Pi\left(\widehat{R}_{\varphi_i}\alpha_{i-1} - \widehat{\varphi}_i'\right)\right\|_\infty + \epsilon}\right) \tag{3.31}$$

where $\tau \in (0,1)$ and $\epsilon > 0$ are constants, with the latter chosen very small to prevent division by zero; and $\|.\|_\infty$ denotes the maximum absolute entry of its vector argument.

Remark 2. The case $B = 1$ for whichever choice of $\phi_1(x) \equiv \phi(x)$ amounts to $\alpha(i) = 1$ for all i and can still be addressed by relations (3.30) and (3.32)–(3.34). The analysis of the resulting algorithm was treated in [ANS01, ANS03, Say03]. The LMS algorithm is recovered when $\phi(x) = x$.

Remark 3. The constant factor τ in (3.31) helps control the convergence rate of the basis function weights α_i. One can also consider adjusting this factor so that α_i converges in step with the weight vector w_i. Doing so helps ensure that the error nonlinearity $h(i)$ is given ample time, yet not exceedingly long, to learn the noise distribution. Since according to the mean weight-error recursion (3.49) given further ahead, the convergence rate of the adaptive algorithm is dictated by the matrix $I - \mu p(i) R_u$, with $p(i)$ defined in (3.46)–(3.47), it is deduced that, for sufficiently small step-size μ, the slowest adaptation mode is $1 - \mu p(i)\lambda_{\min}(R_u)$. Hence, introducing a time-varying factor, $\breve{\tau}(i)$, in place of τ, the former may be set to the sigmoid function evaluated at a stochastic approximation for $p(i)\lambda_{\min}(R_u)$. This construction would suggest the following alternative to (3.31)—simulations in the last section of this chapter illustrate this implementation:

$$\widehat{\lambda}(i) = \nu\widehat{\lambda}(i-1) + (1-\nu)\,\frac{\|u_i\|^2}{M} \tag{3.32}$$

$$\breve{\tau}(i) = \mathrm{sgm}\left[\left(\alpha_{i-1}^T \widehat{\varphi}_i'\right)\widehat{\lambda}(i)\right] \tag{3.33}$$

$$\tau(i) = \breve{\tau}(i)\cdot\left(\frac{\min\{\alpha_{i-1}(b) | 1 \le b \le B\}}{\left\|2\Pi\left(\widehat{R}_{\varphi_i}\alpha_{i-1} - \widehat{\varphi}_i'\right)\right\|_\infty + \epsilon}\right) \tag{3.34}$$

where $\mathrm{sgm}(x) \triangleq \frac{1}{1+e^{-x}} \in (0,1)$. Since $p(i) > 0$ and $R_u > 0$ by the model assumptions and *(A4)*—see Sec. 3.2, $\breve{\tau}(i)$ effectively takes on values in the range $(0.5, 1)$. The estimate for $\widehat{\lambda}(i)$ in (3.32) is reasonable for regressor covariance matrices R_u with relatively small eigenvalue spread. The resulting algorithm is listed in Table 3.1.

3.2 Performance Analysis

The stability and steady-state performance of the robust algorithm under the data model introduced in Sec. 3.1.1 will now be analyzed. Let $\widetilde{\boldsymbol{w}}_i \triangleq w^o - \boldsymbol{w}_i$ denote the weight-error vector, which is a random quantity. In order to make the analysis tractable, some simplifying assumptions need to be introduced; similar assumptions are typical in analyses of adaptive implementations due to the nonlinear and stochastic nature of the update relations:

- *(A1)* The regressors $\{\boldsymbol{u}_i\}$ are independently and identically distributed (i.i.d.), which implies that $\boldsymbol{u}_i$ and $\widetilde{\boldsymbol{w}}_{i-1}$ are independent of each other for all i.

Table 3.1: Robust Adaptive Filtering Algorithm

Initializations: B, $\{\phi_b(x)\}$, Π, $\alpha_{-1} \in \Omega_{++}$, $\widehat{R}_{\varphi_{-1}}$, $\widehat{\varphi}'_{-1}$, ν, $\widehat{\lambda}(-1)$, ϵ, μ. Start with $w_{-1} = 0$. For every time index $i \geq 0$, repeat

Error nonlinearity update:

$$e(i) = d(i) - u_i w_{i-1} \tag{3.35a}$$
$$\phi_b(i) \equiv \phi\big(e(i)\big),\ b = 1, \ldots, B \tag{3.35b}$$
$$\varphi_i = \operatorname{col}\{\phi_1(i), \ldots, \phi_B(i)\} \tag{3.35c}$$
$$\widehat{R}_{\varphi_i} = \nu \widehat{R}_{\varphi_{i-1}} + (1-\nu)\,\varphi_i \varphi_i^T \tag{3.35d}$$
$$\phi'_b(i) \equiv \phi'_b\big(e(i)\big),\ b = 1, \ldots, B \tag{3.35e}$$
$$\varphi'_i = \operatorname{col}\{\phi'_1(i), \ldots, \phi'_B(i)\} \tag{3.35f}$$
$$\widehat{\varphi}'_i = \nu \widehat{\varphi}'_{i-1} + (1-\nu)\,\varphi'_i \tag{3.35g}$$
$$\delta_i = 2\Pi(\widehat{R}_{\varphi_i}\alpha_{i-1} - \widehat{\varphi}'_i) \tag{3.35h}$$
$$\widehat{\lambda}(i) = \nu \widehat{\lambda}(i-1) + (1-\nu)\,\frac{\|u_i\|^2}{M} \tag{3.35i}$$
$$\breve{\tau}(i) = \operatorname{sgm}\Big[\big(\alpha_{i-1}^T \widehat{\varphi}'_i\big)\,\widehat{\lambda}(i)\Big] \tag{3.35j}$$
$$\tau(i) = \breve{\tau}(i) \cdot \left(\frac{\min\{\alpha_{i-1}(b), 1 \leq b \leq B\}}{\|\delta_i\|_\infty + \epsilon}\right) \tag{3.35k}$$
$$\alpha_i = \alpha_{i-1} - \tau(i)\delta_i \tag{3.35l}$$
$$h(i) = \alpha_i^T \varphi_i \tag{3.35m}$$

Adaptive update:

$$w_i = w_{i-1} + \mu u_i^T h(i) \tag{3.36}$$

- $(A2)$ $\boldsymbol{\alpha}_i$ is independent of $\boldsymbol{u}_i$, $\boldsymbol{v}(i)$, and $\widetilde{\boldsymbol{w}}_{i-1}$ for all i.
- $(A3)$ The step-size μ is sufficiently small.
- $(A4)$ The basis functions $\{\phi_b(x)\}$ are sign-preserving, odd-symmetric, monotonically increasing, and twice differentiable.

The first assumption is reasonable under small step-size μ [Say03]. The second assumption is reasonable under small step-size μ, more so when ν is close to 1, and asymptotically, as $i \to \infty$ [KJ92]. Note that, as the filter progresses towards steady-state and the estimator $\boldsymbol{w}_i$ approaches w^o, the error signal $\boldsymbol{e}(i)$ approaches $\boldsymbol{v}(i)$. Under such conditions, it is reasonable to expect the second condition to hold since $\boldsymbol{\alpha}_i$ will be largely determined by the process $\boldsymbol{v}(i)$, which is independent of $\boldsymbol{u}_i$ and $\widetilde{\boldsymbol{w}}_{i-1}$. Moreover, since $\boldsymbol{\alpha}_i$ varies slowly by virtue of its convexity as well as the boundedness of $\breve{\tau}(i)$ on $(0.5, 1)$, we can assume $\boldsymbol{\alpha}_i$ to be independent of $\boldsymbol{v}(i)$ towards steady-state. Clearly, the accuracy of the performance expressions that are derived in the sequel under these assumptions will be dependent on how well these conditions hold. Some differences between actual performance in simulations and predicted theoretical performance are expected due to the approximations. The differences will tend to be smaller for small

step-sizes and fewer basis functions.

From model (3.1), it holds that $\boldsymbol{e}(i) = \boldsymbol{e}_a(i) + \boldsymbol{v}(i)$, where $\boldsymbol{e}_a(i) = \boldsymbol{u}_i\widetilde{\boldsymbol{w}}_{i-1}$ is the *a priori* estimation error. We recall the stochastic recursion corresponding to (3.16):

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} + \mu\boldsymbol{u}_i^T\boldsymbol{h}(i) \tag{3.37}$$

where

$$\boldsymbol{h}(i) = \sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\phi_b\big(\boldsymbol{e}(i)\big). \tag{3.38}$$

Subtracting both sides of (3.37) from w^o leads to

$$\widetilde{\boldsymbol{w}}_i = \widetilde{\boldsymbol{w}}_{i-1} - \mu\boldsymbol{u}_i^T\boldsymbol{h}(i). \tag{3.39}$$

In the sequel, we approximate $\boldsymbol{h}(i)$ using a second-order Taylor series approximation of the basis functions $\{\phi_b(x)\}$ around $e_a(i) = 0$ for all $i \geq 0$ as

$$\begin{aligned}\boldsymbol{h}(i) &= \sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\phi_b\big(\boldsymbol{e}(i)\big)\\ &\approx \sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\boldsymbol{\phi}_{v,b}(i) + \boldsymbol{e}_a(i)\sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\boldsymbol{\phi}'_{v,b}(i)\\ &\quad + \frac{1}{2}\boldsymbol{e}_a^2(i)\sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\boldsymbol{\phi}''_{v,b}(i)\end{aligned} \tag{3.40}$$

where

$$\boldsymbol{\phi}_{v,b}(i) \equiv \phi_b\big(\boldsymbol{v}(i)\big) \tag{3.41}$$

$$\boldsymbol{\phi}'_{v,b}(i) \equiv \phi'_b\big(\boldsymbol{v}(i)\big) \tag{3.42}$$

$$\boldsymbol{\phi}''_{v,b}(i) \equiv \phi''_b\big(\boldsymbol{v}(i)\big) \tag{3.43}$$

3.2.1 Mean Behavior

Taking the expectation of both sides of (3.39),

$$\mathbb{E}\,\widetilde{\boldsymbol{w}}_i = \mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1} - \mu\mathbb{E}\,\boldsymbol{u}_i^T\boldsymbol{h}(i). \tag{3.44}$$

Evaluating the last expectation in (3.44) using (3.40),

$$\mathbb{E}\,\boldsymbol{u}_i^T\boldsymbol{h}(i) = \left(\sum_{b=1}^{B}\mathbb{E}\,\boldsymbol{\alpha}_i(b)\,\mathbb{E}\,\boldsymbol{\phi}'_{v,b}(i)\right)R_u\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}. \tag{3.45}$$

In (3.45), in addition to (*A1*) and (*A2*), (*A4*) on the odd-symmetry of the basis functions $\{\phi_b(x)\}$ was invoked, along with the assumptions on the noise samples $\boldsymbol{v}(i)$ being independent with zero odd moments, and independent of $\boldsymbol{u}_j$ for all i and j. Using the following definitions:

$$\overline{\varphi_v'} \triangleq \operatorname{col}\left\{\mathbb{E}\,\boldsymbol{\phi}_{v,1}'(i), \ldots, \mathbb{E}\,\boldsymbol{\phi}_{v,B}'(i)\right\} \tag{3.46}$$

$$p(i) \triangleq \mathbb{E}\,\boldsymbol{\alpha}_i^T \overline{\varphi_v'} \tag{3.47}$$

where the subscript i has been dropped from $\overline{\varphi_v'}$ since the moment is time-invariant for wide-sense stationary noise processes, we can rewrite (3.45) as

$$\mathbb{E}\,\boldsymbol{u}_i^T \boldsymbol{h}(i) = p(i) R_u \mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}. \tag{3.48}$$

The mean weight-error recursion (3.44) then becomes

$$\mathbb{E}\,\widetilde{\boldsymbol{w}}_i = [I - \mu p(i) R_u]\,\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}. \tag{3.49}$$

Let $\{\lambda_m(R_u)\}$, $m = 1, \ldots, M$, denote the eigenvalues of R_u. Note that by (*A4*), $p(i)$ is positive for all i. From [Say03] and [CS11], one sufficient condition for the asymptotic unbiasedness of (3.49), i.e., $\lim_{i\to\infty} \mathbb{E}\,\widetilde{\boldsymbol{w}}_i = 0$ irrespective of the initial condition, is for there to exist a time index i_1^* and a number $0 < \theta_1 < 1$, such that $|1 - \mu p(i)\lambda_m(R_u)| \leq \theta_1 < 1$ for all $i > i_1^*$ and all $m = 1, \ldots, M$. This translates into the requirement:

$$\frac{1-\theta_1}{\lambda_{\min}(R_u)} \leq \mu p(i) \leq \frac{1+\theta_1}{\lambda_{\max}(R_u)} \tag{3.50}$$

where $\lambda_{\min}(\cdot)$ and $\lambda_{\max}(\cdot)$ denote the minimum and maximum eigenvalues of their matrix arguments, respectively. For example, if $R_u = \sigma_u^2 I$, then this condition requires selecting μ small enough to ensure

$$\frac{1-\theta_1}{\sigma_u^2} \leq \mu p(i) \leq \frac{1+\theta_1}{\sigma_u^2}. \tag{3.51}$$

More generally, to ensure that the lower bound in (3.50) is smaller than the upper bound, we need to require the condition number of R_u, denoted by $\varrho \triangleq \lambda_{\max}(R_u)/\lambda_{\min}(R_u)$, to satisfy

$$\frac{\varrho - 1}{\varrho + 1} < \theta_1 < 1. \tag{3.52}$$

Note that the ratio on the left is always strictly smaller than one for finite ϱ. For the conditions in (3.51) or (3.50) to be realized, we need to ensure that $p(i)$ is bounded for all i. This follows from the boundedness of the two moments $\mathbb{E}\,\boldsymbol{\alpha}_i$ and $\overline{\varphi_v'}$. While the boundedness of $\mathbb{E}\,\boldsymbol{\alpha}_i$ is warranted by convexity, the boundedness of the latter can be guaranteed through appropriate choice of the basis functions $\{\phi_b(x)\}$. For example, for the choice $\phi_1(x) = x$ and $\phi_b(x) = \tanh\big((b-1)x\big)$, $b = 2, \ldots, B$,

$$\overline{\varphi_v'} = \left[1, \mathbb{E}\,\operatorname{sech}^2\big(\boldsymbol{v}(i)\big), \ldots, \mathbb{E}\,(B-1)\operatorname{sech}^2\big((B-1)\boldsymbol{v}(i)\big)\right]^T$$

which is bounded since $f(x) = \operatorname{sech}(x)$ is bounded in x—see Fig. 3.1.

The sufficient stability condition in (3.50) or (3.51) has little practical relevance given that the moment $p(i)$ is time-varying. A tighter sufficient condition can be motivated as follows. First, let $\overline{\phi'_{v,b}} \triangleq \mathbb{E}\,\boldsymbol{\phi}'_{v,b}(i)$ for all b, where the time index has been dropped from the time-invariant moment. Then, define

$$b_{\min} \triangleq \arg\min_b \overline{\phi'_{v,b}}, \quad b_{\max} \triangleq \arg\max_b \overline{\phi'_{v,b}}. \tag{3.53}$$

Note that

$$\min_{\alpha\in\Omega_+} \alpha^T\overline{\varphi'_v} = \overline{\phi'_{v,b_{\min}}}, \quad \max_{\alpha\in\Omega_+} \alpha^T\overline{\varphi'_v} = \overline{\phi'_{v,b_{\max}}}, \tag{3.54}$$

since the solutions of either linear program are the vertices of the $(B-1)$-dimensional polytope $\alpha \in \Omega_+$ where $\alpha^T\overline{\varphi'_v}$ is minimized or maximized, respectively. By employing the minimum and maximum values in (3.54) as lower and upper bounds on $p(i)$, it follows that, for all i and m,

$$|1 - \mu p(i)\lambda_m(R_u)| \leq \max\left\{|1-\mu\omega_{\min}|, |1-\mu\omega_{\max}|\right\} \tag{3.55}$$

where

$$\omega_{\min} \triangleq \overline{\phi'_{v,b_{\min}}}\lambda_{\min}(R_u), \quad \omega_{\max} \triangleq \overline{\phi'_{v,b_{\max}}}\lambda_{\max}(R_u). \tag{3.56}$$

Hence, a sufficient condition for the asymptotic unbiasedness of (3.49) is

$$|1-\mu\omega_{\min}| < 1, \qquad |1-\mu\omega_{\max}| < 1. \tag{3.57}$$

These two conditions are satisfied if μ is chosen such that

$$0 < \mu < \frac{2}{\omega_{\max}}. \tag{3.58}$$

The mean stability condition for the LMS algorithm can be recovered from (3.58) when $B = 1$ and $\phi(x) = x$ such that $\omega_{\max} = \lambda_{\max}(R_u)$ [Say03]:

$$0 < \mu < \frac{2}{\lambda_{\max}(R_u)}. \tag{3.59}$$

3.2.2 Variance Relation

The weighted energy-conservation relation [ANS03,Say03] corresponding to the weight-error recursion (3.39) is given by

$$\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2 = \|\widetilde{\boldsymbol{w}}_{i-1}\|_\Sigma^2 - 2\mu\boldsymbol{u}_i\Sigma\widetilde{\boldsymbol{w}}_{i-1}\boldsymbol{h}(i) + \mu^2\|\boldsymbol{u}_i\|_\Sigma^2\,\boldsymbol{h}^2(i) \tag{3.60}$$

where Σ is a symmetric nonnegative-definite weighting matrix that we are free to choose. Taking the expectation of (3.60) yields the weighted variance relation:

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2 = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_\Sigma^2 - 2\mu\underbrace{\mathbb{E}\,\boldsymbol{u}_i\Sigma\widetilde{\boldsymbol{w}}_{i-1}\boldsymbol{h}(i)}_{①} + \mu^2\underbrace{\mathbb{E}\,\|\boldsymbol{u}_i\|_\Sigma^2\,\boldsymbol{h}^2(i)}_{②}. \tag{3.61}$$

The moments ① and ② need to be evaluated. As for ①, referring to (3.40),

$$\begin{aligned}\mathbb{E}\,\boldsymbol{u}_i\Sigma\widetilde{\boldsymbol{w}}_{i-1}\boldsymbol{h}(i) &= \sum_{b=1}^{B}\mathbb{E}\,\boldsymbol{\alpha}_i(b)\,\mathbb{E}\,\boldsymbol{\phi}'_{v,b}(i)\cdot\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}^T\Sigma\boldsymbol{u}_i^T\boldsymbol{u}_i\widetilde{\boldsymbol{w}}_{i-1}\\ &= p(i)\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_{R_u\Sigma}^2\end{aligned} \tag{3.62}$$

where we have appealed to *(A1)*, *(A2)*, and *(A4)*, in addition to the model assumptions; and used (3.47). As for ②, by squaring both sides of (3.40), discarding powers of $\boldsymbol{e}_a(i)$ higher than 2, multiplying with $\|\boldsymbol{u}_i\|_\Sigma^2$, taking the expectation and invoking *(A1)*, *(A2)*, and *(A4)*, in addition to the model assumptions, it follows that

$$\begin{aligned}\mathbb{E}\,\|\boldsymbol{u}_i\|_\Sigma^2\,\boldsymbol{h}^2(i) &= \mathbb{E}\,\|\boldsymbol{u}_i\|_\Sigma^2\cdot\mathbb{E}\left(\sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\boldsymbol{\phi}_{v,b}(i)\right)^2\\ &\quad+\mathbb{E}\,\|\boldsymbol{u}_i\|_\Sigma^2\,\boldsymbol{e}_a^2(i)\cdot\left\{\mathbb{E}\left(\sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\boldsymbol{\phi}'_{v,b}(i)\right)^2\right.\\ &\quad\left.+\,\mathbb{E}\left(\sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\boldsymbol{\phi}_{v,b}(i)\right)\cdot\left(\sum_{b=1}^{B}\boldsymbol{\alpha}_i(b)\boldsymbol{\phi}''_{v,b}(i)\right)\right\} &\text{(3.63a)}\\ &= s(i)\,\mathrm{Tr}(R_u\Sigma)+t(i)\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|^2_{\mathbb{E}\|\boldsymbol{u}_i\|_\Sigma^2\boldsymbol{u}_i^T\boldsymbol{u}_i} &\text{(3.63b)}\end{aligned}$$

where

$$s(i) = \mathbb{E}\left(\boldsymbol{\alpha}_i^T\boldsymbol{\varphi}_{v,i}\right)^2 = \mathrm{Tr}(R_{\alpha_i}R_{\varphi_v}) \tag{3.64}$$

$$\begin{aligned}t(i) &= \mathbb{E}\left(\boldsymbol{\alpha}_i^T\boldsymbol{\varphi}'_{v,i}\right)^2+\mathbb{E}\left(\boldsymbol{\alpha}_i^T\boldsymbol{\varphi}_{v,i}\right)\left(\boldsymbol{\alpha}_i^T\boldsymbol{\varphi}''_{v,i}\right)\\ &= \mathrm{Tr}(R_{\alpha_i}R_{\varphi'_v}+R_{\alpha_i}R_{\varphi_v\varphi''_v})\end{aligned} \tag{3.65}$$

with the vector $\boldsymbol{\varphi}_{v,i}$ given by

$$\boldsymbol{\varphi}_{v,i} = \mathrm{col}\left\{\boldsymbol{\phi}_{v,1}(i),\ldots,\boldsymbol{\phi}_{v,B}(i)\right\} \tag{3.66}$$

and the vector $\boldsymbol{\varphi}''_{v,i}$ by

$$\boldsymbol{\varphi}''_{v,i} = \mathrm{col}\left\{\boldsymbol{\phi}''_{v,1}(i),\ldots,\boldsymbol{\phi}''_{v,B}(i)\right\} \tag{3.67}$$

and with

$$R_{\alpha_i} \triangleq \mathbb{E}\,\boldsymbol{\alpha}_i\boldsymbol{\alpha}_i^T \tag{3.68}$$

$$R_{\varphi_v} \triangleq \mathbb{E}\,\boldsymbol{\varphi}_{v,i}\boldsymbol{\varphi}_{v,i}^T \tag{3.69}$$

$$R_{\varphi'_v} \triangleq \mathbb{E}\,\boldsymbol{\varphi}'_{v,i}\boldsymbol{\varphi}'^{T}_{v,i} \tag{3.70}$$

$$R_{\varphi_v\varphi''_v} \triangleq \mathbb{E}\,\boldsymbol{\varphi}_{v,i}\boldsymbol{\varphi}''^{T}_{v,i} \tag{3.71}$$

where the subscript i has been dropped from the latter three time-invariant moments. The boundedness of $s(i)$ and $t(i)$ follows from the boundedness of $\boldsymbol{\alpha}_i$ as well as the noise moments in question. Using (3.62) and (3.63b), the weighted variance relation (3.61) can be written as

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2_\Sigma = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|^2_{\Sigma'} + \mu^2 s(i)\,\mathrm{Tr}(R_u\Sigma) \tag{3.72}$$

$$\Sigma' = \Sigma - 2\mu p(i) R_u\Sigma + \mu^2 t(i)\mathbb{E}\,\|\boldsymbol{u}_i\|^2_\Sigma\,\boldsymbol{u}_i^T\boldsymbol{u}_i \tag{3.73}$$

Two cases can be outlined.

3.2.2.1 Special Case—Gaussian Regressors

The case of Gaussian regressors simplifies the analysis since the fourth-order moment $\mathbb{E}\,\|\boldsymbol{u}_i\|^2_\Sigma\,\boldsymbol{u}_i^T\boldsymbol{u}_i$ in (3.73) can then be evaluated in closed form. A change of coordinates will prove handy. Let $R_u = U\Lambda U^T$ be the eigendecomposition of R_u, where Λ is diagonal with the eigenvalues $\lambda_m(R_u)$, $m = 1,\ldots,M$, and U is an orthogonal matrix whose columns are the corresponding eigenvectors. Furthermore, let $\bar{\boldsymbol{w}}_i \triangleq U^T\widetilde{\boldsymbol{w}}_i$, $\bar{\boldsymbol{u}}_i \triangleq \boldsymbol{u}_i U$, and $\bar{\Sigma} \triangleq U^T\Sigma U$. The weighted variance relation (3.72)–(3.73) can then be transformed into

$$\mathbb{E}\,\|\bar{\boldsymbol{w}}_i\|^2_{\bar{\Sigma}} = \mathbb{E}\,\|\bar{\boldsymbol{w}}_{i-1}\|^2_{\bar{\Sigma}'} + \mu^2 s(i)\,\mathrm{Tr}(\Lambda\bar{\Sigma}) \tag{3.74}$$

$$\bar{\Sigma}' = \bar{\Sigma} - 2\mu p(i)\Lambda\bar{\Sigma} + \mu^2 t(i)\big(\Lambda\,\mathrm{Tr}(\Lambda\bar{\Sigma}) + 2\Lambda\bar{\Sigma}\Lambda\big) \tag{3.75}$$

where, in (3.75), we exploited the property that $\mathbb{E}\,\|\bar{\boldsymbol{u}}_i\|^2_\Sigma\,\bar{\boldsymbol{u}}_i^T\bar{\boldsymbol{u}}_i$ evaluates to $\Lambda\,\mathrm{Tr}(\Lambda\bar{\Sigma}) + 2\Lambda\bar{\Sigma}\Lambda$ for Gaussian regressors [Say03]. Note that if we choose $\bar{\Sigma}$ in (3.75) to be a diagonal matrix, then $\bar{\Sigma}'$ will be a diagonal matrix as well. The equation can therefore be expressed more compactly in terms of the diagonal entries of the matrices on either side. Let $\bar{\sigma} \triangleq \mathrm{diag}\{\bar{\Sigma}\}$ and $\lambda \triangleq \mathrm{diag}\{\Lambda\}$. Diagonalizing (3.75), the weighted variance relation (3.74) for Gaussian regressors can be expressed as

$$\mathbb{E}\,\|\bar{\boldsymbol{w}}_i\|^2_{\bar{\sigma}} = \mathbb{E}\,\|\bar{\boldsymbol{w}}_{i-1}\|^2_{\bar{F}_i\bar{\sigma}} + \mu^2 s(i)\left(\lambda^T\bar{\sigma}\right) \tag{3.76}$$

$$\bar{F}_i = I - 2\mu p(i)\Lambda + 2\mu^2 t(i)\Lambda^2 + \mu^2 t(i)\lambda\lambda^T \tag{3.77}$$

where the notation $\|X\|^2_y$ is used as shorthand for $\|X\|^2_{\mathrm{diag}\{y\}}$, with X and y being a matrix and a vector of appropriate dimensions, respectively.

3.2.2.2 General Regressors

More generally, when the regressors are not necessarily Gaussian, we let $\sigma \triangleq \mathrm{vec}(\Sigma)$ and $r_u \triangleq \mathrm{vec}(R_u)$. Vectorizing (3.73) and exploiting the property $\mathrm{vec}(X\Sigma Y) =$

$\left(Y^T \otimes X\right)\sigma$, with $\otimes$ denoting the Kronecker product, the weighted variance relation (3.72)–(3.73) for general regressors can be expressed as

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\sigma^2 = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_{F_i\sigma}^2 + \mu^2 s(i)\left(r_u^T\sigma\right) \tag{3.78}$$

$$F_i = I - 2\mu p(i)\left(I \otimes R_u\right) + 2\mu^2 t(i)\mathbb{E}\left(\left[\boldsymbol{u}_i^T\boldsymbol{u}_i\right]^T \otimes \left[\boldsymbol{u}_i^T\boldsymbol{u}_i\right]\right) \tag{3.79}$$

where the notation $\|X\|_y^2$ is now being used as shorthand for $\|X\|_{\mathrm{vec}^{-1}(y)}^2$.

3.2.3 Steady-State Performance

Let

$$\mathrm{MSE} \triangleq \lim_{i\to\infty} \mathbb{E}\,\boldsymbol{e}^2(i) \tag{3.80}$$

and

$$\mathrm{EMSE} \triangleq \lim_{i\to\infty} \mathbb{E}\,\boldsymbol{e}_a^2(i) \tag{3.81}$$

where EMSE stands for excess mean-square error. It holds that $\mathrm{MSE} = \mathrm{EMSE} + \sigma_v^2$. Let

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|_\Sigma^2 \triangleq \lim_{i\to\infty} \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2\,. \tag{3.82}$$

Then, one can also define the mean-square deviation (MSD):

$$\mathrm{MSD} \triangleq \lim_{i\to\infty} \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2 = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|^2 \tag{3.83}$$

and the EMSE may be expressed, under *(A1)*, as

$$\mathrm{EMSE} = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|_{R_u}^2\,. \tag{3.84}$$

3.2.3.1 Special Case—Gaussian Regressors

Referring to the weighted variance relation (3.76)–(3.77) and taking the limit as $i \to \infty$, under the assumption that the moments $\mathbb{E}\,\boldsymbol{\alpha}_i$ and R_{α_i} approach some constant values $\mathbb{E}\,\boldsymbol{\alpha}_\infty$ and R_{α_∞}, respectively, it holds that

$$\mathbb{E}\,\|\bar{\boldsymbol{w}}_\infty\|_{\left(I-\bar{F}_\infty\right)\bar{\sigma}}^2 = \mu^2 s(\infty)\left(\lambda^T\bar{\sigma}\right) \tag{3.85}$$

where

$$\begin{aligned}\bar{F}_\infty &\triangleq \lim_{i\to\infty} \bar{F}_i \\ &= I - 2\mu p(\infty)\Lambda + 2\mu^2 t(\infty)\Lambda^2 + \mu^2 t(\infty)\lambda\lambda^T\end{aligned} \tag{3.86}$$

and

$$p(\infty) \triangleq \lim_{i\to\infty} p(i) = \mathbb{E}\,\boldsymbol{\alpha}_\infty^T \bar{\varphi}_v' \tag{3.87}$$

$$s(\infty) \triangleq \lim_{i\to\infty} s(i) = \mathrm{Tr}(R_{\alpha_\infty} R_{\varphi_v}) \tag{3.88}$$

$$t(\infty) \triangleq \lim_{i\to\infty} t(i) = \mathrm{Tr}(R_{\alpha_\infty} R_{\varphi_v'} + R_{\alpha_\infty} R_{\varphi_v \varphi_v''}) \tag{3.89}$$

In the following, the matrix $\bar{F}_\infty$ is assumed to be stable. Since $\mathrm{MSD} = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|^2 = \mathbb{E}\,\|\bar{\boldsymbol{w}}_\infty\|^2$, then, substituting $\bar{\sigma} = \left(I - \bar{F}_\infty\right)^{-1} \mathbb{1}$ in (3.85) results in

$$\mathrm{MSD} = \mu^2 s(\infty) \lambda^T \left(I - \bar{F}_\infty\right)^{-1} \mathbb{1} \tag{3.90}$$

which evaluates to

$$\mathrm{MSD} = \mu \frac{s(\infty)}{p(\infty)} \cdot \frac{\sum_{m=1}^{M} \frac{1}{1-\mu\frac{t(\infty)}{p(\infty)}\lambda_m(R_u)}}{2 - \mu\frac{t(\infty)}{p(\infty)} \cdot \sum_{m=1}^{M} \frac{\lambda_m(R_u)}{1-\mu\frac{t(\infty)}{p(\infty)}\lambda_m(R_u)}}. \tag{3.91}$$

Similarly, for the EMSE, by substituting $\bar{\sigma} = \left(I - \bar{F}_\infty\right)^{-1} \lambda$ in (3.85), we obtain

$$\mathrm{EMSE} = \mu \frac{s(\infty)}{p(\infty)} \cdot \frac{\sum_{m=1}^{M} \frac{\lambda_m(R_u)}{1-\mu\frac{t(\infty)}{p(\infty)}\lambda_m(R_u)}}{2 - \mu\frac{t(\infty)}{p(\infty)} \cdot \sum_{m=1}^{M} \frac{\lambda_m(R_u)}{1-\mu\frac{t(\infty)}{p(\infty)}\lambda_m(R_u)}}. \tag{3.92}$$

The MSD and EMSE expressions for the LMS algorithm are given by [Say03]:

$$\mathrm{MSD}^{\mathrm{LMS}} = \mu\sigma_v^2 \frac{\sum_{m=1}^{M} \frac{1}{1-\mu\lambda_m(R_u)}}{2 - \mu \sum_{m=1}^{M} \frac{\lambda_m(R_u)}{1-\mu\lambda_m(R_u)}} \tag{3.93}$$

$$\mathrm{EMSE}^{\mathrm{LMS}} = \mu\sigma_v^2 \frac{\sum_{m=1}^{M} \frac{\lambda_m(R_u)}{1-\mu\lambda_m(R_u)}}{2 - \mu \sum_{m=1}^{M} \frac{\lambda_m(R_u)}{1-\mu\lambda_m(R_u)}} \tag{3.94}$$

For small step-size μ such that $\mu\frac{t(\infty)}{p(\infty)}\lambda_m(R_u) \ll 1$ for all $m = 1, \ldots, M$, expressions (3.91) and (3.92) simplify to

$$\mathrm{MSD} \approx \frac{\mu M \frac{s(\infty)}{p(\infty)}}{2 - \mu\frac{t(\infty)}{p(\infty)}\,\mathrm{Tr}(R_u)} \tag{3.95}$$

$$\mathrm{EMSE} \approx \frac{\mu\frac{s(\infty)}{p(\infty)}\,\mathrm{Tr}(R_u)}{2 - \mu\frac{t(\infty)}{p(\infty)}\,\mathrm{Tr}(R_u)} \tag{3.96}$$

Remark 4. Consider the following *exact* steady-state variance relation [Say03, Theorem 6.4.1], adapted to our robust algorithm:

$$\mu\mathbb{E}\,\|\boldsymbol{u}_i\|^2\,\boldsymbol{h}^2(i) = 2\,\mathbb{E}\,\boldsymbol{e}_a(i)\boldsymbol{h}(i), \quad i\to\infty. \tag{3.97}$$

Consider as well the following assumption in addition to $(A1)$–$(A4)$:

- $(A5)$ The quantities $\|\boldsymbol{u}_i\|^2$ and $\boldsymbol{h}^2(i)$ are asymptotically uncorrelated.

Then, evaluating (3.97) leads to the same expression for the EMSE as in (3.96), without the need for a Gaussian assumption on the regressors. □

For even smaller step-size μ such that the $\mathcal{O}(\mu^2)$ terms in (3.86) may be ignored,

$$\bar{F}_\infty \approx I - 2\mu p(\infty)\Lambda, \tag{3.98}$$

which is stable if

$$0 < \mu p(\infty) < \frac{1}{\lambda_{\max}(R_u)}, \tag{3.99}$$

and (3.95) and (3.96) become

$$\text{MSD} \approx \frac{\mu M\frac{s(\infty)}{p(\infty)}}{2}, \quad \text{EMSE} \approx \frac{\mu\frac{s(\infty)}{p(\infty)}\,\text{Tr}(R_u)}{2}. \tag{3.100}$$

3.2.3.2 General Regressors

Similarly, we obtain

$$\text{MSD} = \mu^2 s(\infty) r_u^T\,(I-F_\infty)^{-1}\,\mathbb{1} \tag{3.101}$$

$$\text{EMSE} = \mu^2 s(\infty) r_u^T\,(I-F_\infty)^{-1}\,r_u \tag{3.102}$$

where

$$\begin{aligned} F_\infty &\triangleq \lim_{i\to\infty} F_i \\ &= I - 2\mu p(\infty)\,(I\otimes R_u) + 2\mu^2 t(\infty)\mathbb{E}\left(\left[\boldsymbol{u}_i^T\boldsymbol{u}_i\right]^T\otimes\left[\boldsymbol{u}_i^T\boldsymbol{u}_i\right]\right) \end{aligned} \tag{3.103}$$

which is assumed to be stable, provided that the matrix $\mathbb{E}\left(\left[\boldsymbol{u}_i^T\boldsymbol{u}_i\right]^T\otimes\left[\boldsymbol{u}_i^T\boldsymbol{u}_i\right]\right)$ is finite.

For small step-size μ such that the $\mathcal{O}(\mu^2)$ term in (3.103) may be ignored,

$$F_\infty \approx I - 2\mu p(\infty)\,(I\otimes R_u)\,, \tag{3.104}$$

where the same stability condition (3.99) applies, and (3.101) and (3.102) become identical to those in (3.100).

Remark 5. One must clarify in what sense the algorithm developed here is robust to impulsive noise. This property follows from the fact that the algorithm is designed to approximate the MSE-optimal error nonlinearity (3.8) from [ANS01]. It is noteworthy that the steady-state mean-square performance expressions in (3.100) under small step-size match those in [ANS01] for the class of smooth nonlinearities when $e(i) \approx v(i)$ (see Eq. (65) in [ANS01]). The latter were shown in [ANS01] to be minimized by the choice of nonlinearity (3.8).

Remark 6. As previously mentioned in Remark 2, the LMS algorithm is recovered when $B = 1$ and $\phi(x) = x$. It can be verified that in this case, $p(i) = t(i) = 1$ and $s(i) = \sigma_v^2$ for all i. By substituting these values into (3.91)–(3.92), (3.95)–(3.96), and (3.100), the well-known steady-state mean-square performance expressions for the LMS algorithm are recovered [Say03].

Remark 7. For the verification of the steady-state mean-square performance expressions derived in this subsection, $p(\infty)$, $s(\infty)$, and $t(\infty)$ need to be evaluated according to (3.87)–(3.89). The moments pertaining to the noise process $\boldsymbol{v}(i)$ can be evaluated subject to knowledge of its distribution. The steady-state first- and second-order moments of the vector of basis function weights $\boldsymbol{\alpha}_i$, $\mathbb{E}\,\boldsymbol{\alpha}_\infty$ and R_{α_∞}, however, need to be approximated. One way to do so is through Monte Carlo simulation. Another way is to approximate the limiting value α_∞ by the constrained solution to the normal equations:

$$\alpha_\infty \approx \arg\min_{\alpha \in \Omega_+} \; \alpha^T R_{\varphi_v} \alpha - 2\alpha^T \overline{\varphi'_v}. \tag{3.105}$$

Note that the optimization problem in (3.105) is the same as in (3.20) with the output error signal $\boldsymbol{e}(i)$ substituted with the noise signal $\boldsymbol{v}(i)$ and subject to (3.13). The steady-state moments $\mathbb{E}\,\boldsymbol{\alpha}_\infty$ and R_{α_∞} may then be approximated by the instantaneous values α_∞ and $\alpha_\infty \alpha_\infty^T$, respectively.

3.2.4 Mean-Square Behavior

The recursion for the weighted variance relation (3.72) is not self-contained. For small step-size μ, the $\mathcal{O}(\mu^2)$ term in (3.73) may be ignored, and the weighted variance relation (3.72) becomes

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2 = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_\Sigma^2 - 2\mu p(i) \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_{R_u \Sigma}^2 + \mu^2 s(i)\,\mathrm{Tr}(R_u \Sigma). \tag{3.106}$$

According to the Cayley–Hamilton Theorem [ANS03, Say03], it holds that $R_u^M = -c_0 I - c_1 R_u - \ldots - c_{M-1} R_u^{M-1}$, where $c(x) \triangleq \det(xI - R_u) = c_o + c_1 x + \ldots + x^M$,

$\det(\cdot)$ denoting the determinant of its matrix argument, so that

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|^2_{R_u^M} = -\sum_{m=0}^{M-1} c_m \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|^2_{R_u^m}\,. \tag{3.107}$$

Writing out (3.106) for $\Sigma = I, R_u, \ldots, R_u^{M-1}$ results in the following state-space model:

$$\mathcal{W}_i = \mathcal{B}_i \mathcal{W}_{i-1} + \mu^2 s(i)\mathcal{Y} \tag{3.108}$$

where the state-vector $\mathcal{W}_i$ and the vector $\mathcal{Y}$ are given by

$$\mathcal{W}_i \triangleq \begin{bmatrix} \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2 \\ \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2_{R_u} \\ \vdots \\ \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2_{R_u^{M-1}} \end{bmatrix}, \quad \mathcal{Y} \triangleq \begin{bmatrix} \mathrm{Tr}(R_u) \\ \mathrm{Tr}(R_u^2) \\ \vdots \\ \mathrm{Tr}(R_u^M) \end{bmatrix}, \tag{3.109}$$

and $\mathcal{B}_i$ by (3.110) at the bottom of the page. Eq. (3.108) represents a nonlinear time-invariant state-space model, where the first and second entries in the state-vector $\mathcal{W}_i$ represent the transient MSD and EMSE, respectively.

Remark 8. Referring again to (3.61), if we invoke *(A5)* in addition to *(A1)*–*(A4)* to evaluate $\mathbb{E}\,\|\boldsymbol{u}_i\|^2_\Sigma\, \boldsymbol{h}^2(i)$, we get

$$\mathbb{E}\,\|\boldsymbol{u}_i\|^2_\Sigma\, \boldsymbol{h}^2(i) = s(i)\,\mathrm{Tr}(R_u\Sigma) + t(i)\,\mathrm{Tr}(R_u\Sigma)\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|^2_{R_u} \tag{3.111}$$

—see (3.63b) for comparison—where $s(i)$ and $t(i)$ are still given by (3.64) and (3.65). An alternative weighted variance relation to (3.72)–(3.73) is therefore given by

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2_\Sigma = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|^2_{\Sigma'} + \mu^2 s(i)\,\mathrm{Tr}(R_u\Sigma) \tag{3.112}$$

$$\Sigma' = \Sigma - 2\mu p(i) R_u\Sigma + \mu^2 t(i)\,\mathrm{Tr}(R_u\Sigma) R_u \tag{3.113}$$

and the resulting state-space model by

$$\mathcal{W}_i = \left(\mathcal{B}_i + \mu^2 t(i)\mathcal{Y} e_2^T\right)\mathcal{W}_{i-1} + \mu^2 s(i)\mathcal{Y} \tag{3.114}$$

where e_2 is the all-zero vector of length M and second entry equal to 1. Note that the state-space model (3.108) can be recovered from (3.114) by ignoring the $\mathcal{O}(\mu^2)$ term multiplying the state-vector $\mathcal{W}_{i-1}$.

$$\mathcal{B}_i \triangleq \begin{bmatrix} 1 & -2\mu p(i) & 0 & \ldots & 0 & 0 \\ 0 & 1 & -2\mu p(i) & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & 1 & -2\mu p(i) \\ 2\mu c_0 p(i) & 2\mu c_1 p(i) & 2\mu c_2 p(i) & \ldots & 2\mu c_{M-2} p(i) & 1 + 2\mu c_{M-1} p(i) \end{bmatrix} \tag{3.110}$$

3.2.5 Mean-Square Stability

It is observed from (3.108) that one sufficient condition for mean-square stability, i.e., $\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2$ and $\mathbb{E}\,\boldsymbol{e}_a^2(i)$ tend to some finite constant values irrespective of the initial condition, w_{-1}, is for there to exist a time index i_2^* and a number $0 < \theta_2 < 1$, such that $|\lambda_{\max}(\mathcal{B}_i)| \leq \theta_2 < 1$ for all $i > i_2^*$. Let $\mathcal{C}$ be the companion matrix of R_u. It follows that $\mathcal{B}_i = I - 2\mu p(i)\mathcal{C}$, with eigenvalues $1 - 2\mu p(i)\lambda_m(R_u)$, $m = 1, \ldots, M$. Hence, mean-square stability conditions can be deduced that are a factor 2 tighter than the mean-stability conditions in (3.50) and (3.58). Combining with the assumption of sufficiently small step-size that led to the recursion (3.106) and hence, model (3.108), it is concluded that sufficiently small μ helps ensure mean-square stability.

3.2.6 Algorithm Complexity

The robust adaptive filtering algorithm in Table 3.1 involves the following operations per iteration:

- $3M + B^3 + 4B^2 + 5B + 5$ multiplications
- $3M + B^3 + 2B^2 + 3B - 1$ additions
- 1 division
- 2 sorting operations ($\mathcal{O}(B \log_2 B)$)
- $2B + 1$ function evaluations or table lookups

Excluding the function evaluations or table lookups, the total number of operations is $\mathcal{O}(6M + B^3)$. For comparison, the LMS algorithm involves $2M + 1$ multiplications and $2M$ additions per iteration, implying that the robust algorithm introduces $\mathcal{O}(2M + B^3)$ additional complexity per iteration.

3.3 Simulation Results

A system identification setup is considered, where the aim is to estimate a randomly initialized unit-norm w^o of size $M = 10$. The regressors $\{\boldsymbol{u}_i\}$ are i.i.d. zero-mean Gaussian vectors with covariance matrix $R_u = \text{diag}\left\{\sigma_{u,1}^2, \ldots, \sigma_{u,M}^2\right\}$, where $\left\{\sigma_{u,m}^2\right\}$ are drawn independently and uniformly over $[5, 15)$ dB. The noise samples $\{\boldsymbol{v}(i)\}$ are drawn independently of the regressors and are i.i.d. Considered is an ε-contaminated

Gaussian mixture model for the noise, which is a typical one for impulsive noise. The pdf is given by

$$f_{\boldsymbol{v}}(v) = (1-\varepsilon)\,\mathcal{N}(0,\bar{\sigma}_v^2) + \varepsilon\mathcal{N}(0,\kappa\bar{\sigma}_v^2),$$

where $\bar{\sigma}_v^2$ is the nominal noise variance, ε is the contamination ratio, and $\kappa \gg 1$. The case $\varepsilon = 0$ signifies uncontaminated Gaussian noise. The effective noise variance is given by $\sigma_v^2 = (1-\varepsilon)\,\bar{\sigma}_v^2 + \varepsilon\kappa\bar{\sigma}_v^2$. The initial estimate for w^o is set to $w_{-1} = 0$. For the robust algorithm in Table 3.1, the initial estimate α_{-1} is set to $\frac{1}{B}\mathbb{1}$ for whichever choice of B, the number of basis functions. For the smoothing operations, ν is set to 0.9 and initial conditions to zero. The parameter ϵ is set to 10^{-6}. All simulation results in this section are obtained by averaging over 1000 experiments. The following set of basis functions is employed, unless mentioned otherwise: $\phi_1(x) = x$ and $\phi_b(x) = \tanh\big((b-1)x\big)$, $b = 2,\ldots,B$.

First, the effect on the learning and steady-state behavior of the number of basis functions used is investigated. The nominal noise variance is $\bar{\sigma}_v^2 = -10$ dB, $\varepsilon = 0.1$, and $\kappa = 100$. First, the performance of the robust algorithm for different number of basis functions B is compared with that of the LMS algorithm at the same adaptation rate, i.e., equal step-size $\mu = 1 \times 10^{-3}$. The resulting learning performance is depicted in Fig. 3.2a in terms of the transient MSD. The corresponding steady-state MSDs, obtained by averaging the last 100 samples of the respective MSD curves, are plotted in Fig. 3.2b, along with their theoretical counterparts according to (3.93), (3.91), and (3.87)–(3.89) for verification. The moments $\mathbb{E}\,\boldsymbol{\alpha}_\infty$ and R_{α_∞} are approximated in two ways—see Remark 7 in Sec. 3.2.3: the first approximation is where CVX is used to solve (3.105) [GB14]; and the second is based on Monte Carlo (MC) simulation. The robust algorithm is seen to converge to steady-state slower than the LMS algorithm for the same step-size. On the other hand, the robust algorithm outperforms the LMS algorithm in terms of steady-state performance. Moreover, increasing the number of basis functions appears to speed up convergence and worsen steady-state performance. The first approximation of the moments $\mathbb{E}\,\boldsymbol{\alpha}_\infty$ and R_{α_∞} does not lead to a tight fit between theory and simulation. This is to have been expected since the approximation does not account for the particular manner in which the basis function weights α_i are adapted according to (3.30) and (3.32)–(3.34) to solve (3.20). The second, Monte Carlo-based approximation of the moments, on the other hand, produces a tighter fit between theory and simulation, albeit exhibiting a discrepancy towards bigger values of B. Increasing the number of basis functions improves the ability of the algorithm to approximate the optimal nonlinearity (3.8) with better accuracy, which reflects positively on the performance of the algorithm. On the other hand, adding basis functions increases the number of parameters that need to be adapted, which degrades performance. Therefore, there is a compromise between convergence and performance,

as is typical for such scenarios.

Now, the performance of the robust algorithm for different number of basis functions B is compared with that of the LMS algorithm at the same initial convergence rate, for which $\mu^{\text{rob}} = 1 \times 10^{-3}$ and $\mu^{\text{LMS}} = 2.7 \times 10^{-4}$. The resulting learning performance is depicted in Fig. 3.3a in terms of the transient MSD. The corresponding steady-state MSDs, obtained by averaging the last 100 samples of the respective MSD curves, are plotted in Fig. 3.3b, along with their theoretical counterparts. The LMS algorithm is seen to converge slower than the robust algorithm to worse steady-state performance. Moreover, increasing the number of basis functions appears to speed up convergence and worsen steady-state performance.

Depicted in Figs. 3.4a and 3.4b is the transient MSD performance of the LMS and robust algorithms with $B = 2$ basis functions when the measurements are corrupted by contaminated Gaussian noise, at $\bar{\sigma}_v^2 = -10$ dB, with different contamination ratios ε, and $\kappa = 100$. The adaptation rate of both algorithms is kept constant for all ε and is chosen in such a way so that the convergence time of the robust algorithm is the same as that of the LMS algorithm in Fig. 3.4a at $\varepsilon = 0$ (no contamination): $\mu^{\text{LMS}} = 8 \times 10^{-4}$ and $\mu^{\text{rob}} = 1 \times 10^{-3}$. The corresponding steady-state MSDs, obtained by averaging the last 100 samples of the respective MSD curves, are plotted in Fig. 3.5a, along with their theoretical counterparts. The corresponding ratio $\frac{s(\infty)}{p(\infty)}$ is plotted in Fig. 3.5b: While for the LMS algorithm, this ratio is simply equal to the effective noise variance, σ_v^2 (see Remark 6); for the robust algorithm, the ratio is computed using the second, Monte Carlo-based approximation of the moments $\mathbb{E}\,\boldsymbol{\alpha}_\infty$ and R_{α_∞}. It is seen that an increase in the contamination ratio slows down the convergence of the robust algorithm, compared with the LMS algorithm, where convergence is seen to speed up. On the other hand, it is evident that the algorithm is less sensitive than the LMS algorithm to increasingly impulsive noise.

We now investigate the performance of the algorithm under a colored regression sequence $\boldsymbol{u}(i)$ in view of the deployment of the time-varying factor $\breve{\tau}(i)$ for the adaptation of the step-size $\tau(i)$ to update the basis function weights α_i—see (3.32)–(3.34). The colored regression sequence is obtained by filtering an i.i.d. zero-mean Gaussian random process with variance σ_u^2 through a first-order autoregressive model with transfer function $\frac{\sqrt{1-a^2}}{1-az^{-1}}$, where a is the coefficient of the AR(1) process, reflecting the degree of correlation. We consider $a = 0$ (white regression sequence), 0.4, and 0.8 and two signal-to-noise (SNR) ratio scenarios under Gaussian noise ($\varepsilon = 0$) and $B = 5$ basis functions: $\sigma_u^2 = 10$ dB and $\bar{\sigma}_v^2 = -10$ dB (high SNR); $\sigma_u^2 = 0$ dB and $\bar{\sigma}_v^2 = 0$ dB (low SNR). The step-sizes in the two scenarios are $\mu = 1 \times 10^{-3}$ and $\mu = 5 \times 10^{-3}$, respectively. The results for the two scenarios are illustrated in Figs. 3.6 and 3.7. While Figs. 3.6a

and 3.7a depict the resulting simulated MSD learning curves as well as the theoretical MSD learning curves according to (3.114) and theoretical steady-state MSDs according to (3.91), Figs. 3.6b and 3.7b depict the mean transient behavior of the factor $\breve{\tau}(i)$. Since the difference between the theoretical MSD learning curves (3.114) and (3.108) was observed to be negligible, only the former is plotted. The moments $\mathbb{E}\,\boldsymbol{\alpha}_i$ and R_{α_i} involved in the calculation of the moments $p(i)$ and $s(i)$ according to (3.47) and (3.64), respectively, are approximated via Monte Carlo simulations. It is seen from Figs. 3.6a and 3.7a that despite the slower convergence of the algorithm with increasing correlation level, the steady-state performance remains unaffected. However, the factor $\breve{\tau}(i)$ is seen from Figs. 3.6b and 3.7b to converge at roughly the same time irrespective of the correlation level, and in synchrony with the algorithm under the white regression sequence ($a = 0$). This is consistent with the implementation under consideration in (3.32)–(3.33), which for sufficiently small step-size μ drives $\breve{\tau}(i)$ to adapt at the same rate as the average convergence mode of the algorithm. Regarding the agreement between the simulated and theoretical transient MSD learning curves in Figs. 3.6a and 3.7a, it is seen that in the high-SNR scenario, where the nonlinearity of the algorithm is prominent in the first stages of adaptation, the theoretical curves, based on the second-order Taylor approximation of the nonlinearity (3.40), deviate from their simulated counterparts in transience. This is not the case, however, in the low-SNR scenario. It is also noteworthy that the simulated and theoretical steady-state MSDs are in agreement across the correlation levels under consideration, despite the analysis having been conducted under the independence assumption (*A1*) on the regressors $\{\boldsymbol{u}_i\}$. Now, in order to appreciate the robustness of the proposed $\breve{\tau}(i)$ construction (3.33), the performance of the robust algorithm with $B = 5$ basis functions is examined under contaminated Gaussian noise, with $\varepsilon = 0.1$ and $\kappa = 1000$, at $\bar{\sigma}_v^2 = 0$ dB, and using the colored regression sequence $\boldsymbol{u}(i)$ with correlation level $a = 0.4$. For comparison, three other constructions for $\breve{\tau}(i)$ are considered: one employing (3.33), where $\widehat{\lambda}(i)$ is set instead to $\lambda_{\min}(R_u)$, assuming it is known; and another two constructions where $\breve{\tau}(i)$ is set to constant values, 0.05 and 0.99. The step-size is chosen to be $\mu = 5 \times 10^{-3}$. The results are plotted in Fig. 3.8, where Fig. 3.8a depicts the resulting MSD learning curves and Fig. 3.8b, the mean transient behavior of the factor $\breve{\tau}(i)$. While the adaptive construction (3.33) incurs no performance loss despite the approximation used for the slowest convergence mode, it also enjoys a stabilizing effect, which comes at the price of complexity: Setting the $\breve{\tau}(i)$ factor, for reasons of simplicity, to a constant value that might be too high for the impulsive noise environment curbs the ability of the robust algorithm to learn the underlying distribution, leading to unpredictable performance.

Finally, in Figs. 3.9a and 3.9b, the performance of the robust algorithm of Table 3.1 is compared with others in the literature in severe noisy environments, similar to the

one proposed in [CA97], for an i.i.d. unit-variance regression sequence, i.e., $R_u = I_M$. In addition to the LMS algorithm, considered are the sign-LMS [Say03], robust mixed-norm (RMN) [CA97], and least-mean M-estimate (LMM) [ZCN00] algorithms. While the sign-LMS algorithm is of the same order of complexity as the LMS algorithm, the RMN and LMM algorithms both introduce $\mathcal{O}(N_w \log_2 N_w)$ additional complexity per iteration relative to the LMS algorithm, excluding function evaluation or table lookup, where N_w is the window-length parameter of the respective algorithm. In Fig. 3.9a, the MSD learning curves are plotted for measurements corrupted by uncontaminated Gaussian noise ($\varepsilon = 0$) at $\bar{\sigma}_v^2 = 0$ dB. The steady-state MSDs for these algorithms have been equalized: $\mu^{\text{rob}} = \mu^{\text{LMS}} = \mu^{\text{LMM}} = 1\times10^{-2}$, $\mu^{\text{sign-LMS}} = 8\times10^{-3}$, $\mu^{\text{RMN}} = 3\times10^{-3}$. The window length for the RMN and LMM algorithms is set to 10. For the LMM algorithm, the clipping threshold is adapted such that the outlier probability does not exceed 0.01; and the smoothing parameter for the estimation of the output error variance based on the normalized median absolute deviation [RC93] is set to 0.9. Also plotted is the learning curve given the optimal nonlinearity, calculated according to (3.7), using the same step-size as the robust algorithm. In Fig. 3.9b, the same curves are plotted for measurements corrupted by contaminated Gaussian noise, with $\varepsilon = 0.1$ and $\kappa = 1000$. The step-sizes for these algorithms are the same as those in Fig. 3.9a at $\varepsilon = 0$ (no contamination). According to Fig. 3.9a, at $\varepsilon = 0$, the learning curve of the LMM algorithm coincides with that of the LMS algorithm. As a matter of fact, given this scenario and choice of LMM algorithm parameters, the LMM algorithm behaves like the LMS algorithm. The parameters actually act in its favor in this scenario: The LMM algorithm selects between the LMS algorithm and an error clipping function at each time index. The selection is based on comparing the error magnitude with a threshold. The threshold is the magnitude of a Gaussian random variable, as a model for the error signal, the probability of exceeding which should not exceed a certain limit preset by the practitioner, here 10 %. This implies that the threshold might be high for Gaussian noise, but low for a heavy-tailed pdf. Hence, if the error signal is almost never clipped, the LMM algorithm behaves like the LMS algorithm. Note that in order to calculate the threshold, the LMM algorithm requires an estimate of the error variance, which is estimated over a window whose length is also chosen by the practitioner. In the absence of prior knowledge about the noise distribution in order to guide the selection of these tuning parameters, the performance of the algorithm is rendered sensitive to changing conditions. Indeed, the performance of the LMM algorithm is inferior to all but the LMS algorithm when the noise is contaminated while all parameters remain fixed. On the other hand, the RMN algorithm employs an adaptive convex combination of LMS and sign-LMS updates, steered by an assessment of the instantaneous reference signal, rather than the error signal. This construction renders the algorithm performance sensitive to the statistics of the regression signal and

the system to be identified. It can therefore be appreciated from Figs. 3.9a and 3.9b that for slower convergence in nominal noise conditions, the robust algorithm developed here achieves the best steady-state performance in the severe noisy environment under consideration.

In conclusion, we remind the reader that robustness in the offline batch setting is associated with a tradeoff: good performance in the presence of contamination at the expense of some mean-square performance loss under Gaussian noise. Given that in adaptive filtering convergence rate is yet another performance measure to contend with, then it should be expected that for the same target steady-state mean-square performance under Gaussian noise, the convergence rate will have to be compromised relative to the LMS algorithm.

As a final illustrative example for the behavior of the robust algorithm in comparison with the other algorithms considered here, the convergence rates under equalized steady-state MSDs are compared given Laplace noise, which is heavier tailed than Gaussian noise and whose pdf is given by

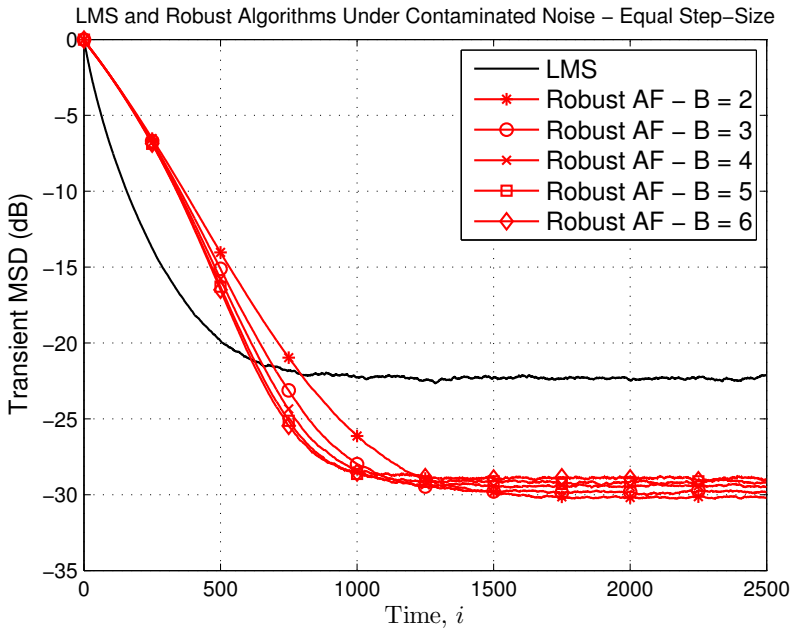
$$f_{\boldsymbol{v}}(v) = \frac{1}{\sqrt{2}\sigma_v} e^{-\frac{\sqrt{2}|v|}{\sigma_v}}.$$

We consider $\sigma_v^2 = 10$ dB. Noting that $h_1^{\text{opt}}(x)$ for Laplace noise is given by $\frac{\sqrt{2}}{\sigma_v}\text{sign}(x)$, the following set of basis functions is considered: $\phi_1(x) = x$ and $\phi_2(x) = \text{sign}(x)$. The step-sizes are chosen as $\mu^{\text{rob}} = \mu^{\text{sign-LMS}} = 5 \times 10^{-3}$, $\mu^{\text{LMS}} = 1.25 \times 10^{-3}$, and $\mu^{\text{LMM}} = 1.55 \times 10^{-3}$. All other parameters are set to the same values as before. The resulting MSD learning curves are plotted in Fig. 3.10. Since the RMN algorithm failed to converge within the same time frame as the other algorithms, its learning curve is not displayed. Also plotted is the learning curve given the optimal nonlinearity, $h_1^{\text{opt}}(x) = \frac{\sqrt{2}}{\sigma_v}\text{sign}(x)$, using the same step-size as the robust algorithm, which is also the same as that of the sign-LMS algorithm in this example; so the effective step-size is $\mu^{\text{opt}} = \frac{1}{\sqrt{5}}\mu^{\text{rob}}$, leading to slower convergence. It can be seen that the robust and sign-LMS algorithms behave similarly and converge at the same time, while the LMS algorithm converges slower.

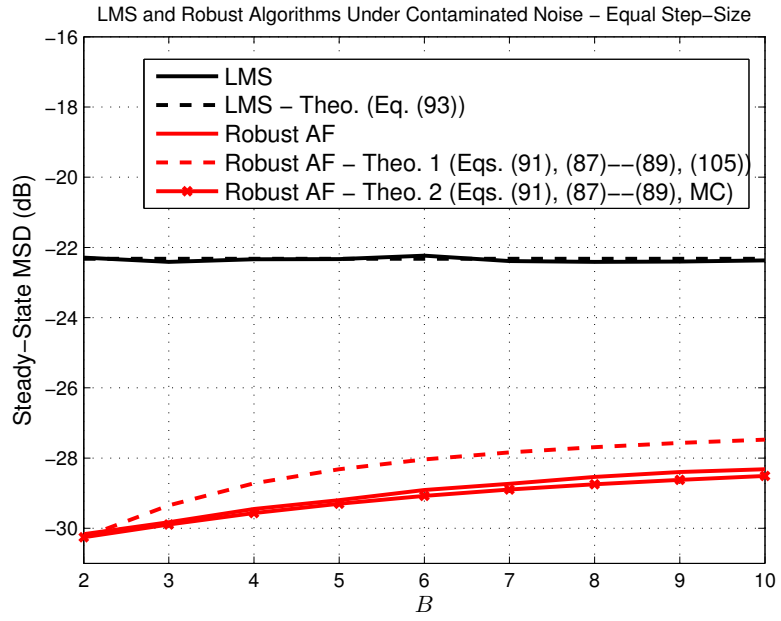
3.4 Conclusion

A robust LMS-type adaptive algorithm was developed that employs an adaptive error nonlinearity. The error nonlinearity was chosen to be a convex combination of preselected basis functions where the combination coefficients are adapted jointly with the

weight vector such that the MSE relative to the optimal error nonlinearity is minimized in each iteration. While knowledge of the nature of the noise, impulsive or otherwise, serves to guide the choice of basis functions, exact distributional knowledge is not required, which endows the algorithm with robustness and flexibility. The transient and steady-state behavior of the algorithm were analyzed in the mean and mean-square sense using the energy conservation framework subject to a set of reasonable assumptions given the nonlinear and stochastic nature of the algorithm. The performance of the algorithm was illustrated in simulation in an impulsive noise scenario. The computational complexity of the algorithm was summarized, revealing that the robust algorithm introduces $\mathcal{O}(2M + B^3)$ additional complexity per iteration compared to the LMS algorithm.

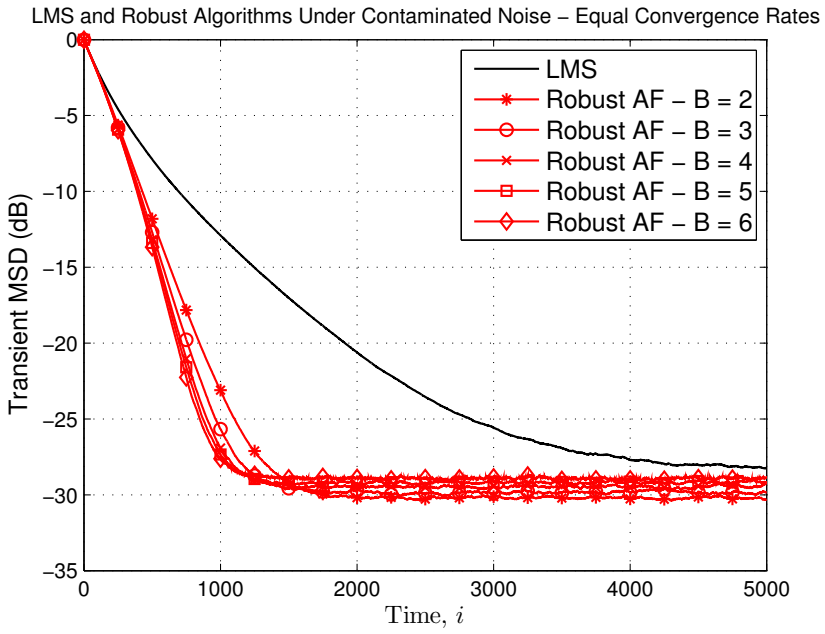


(a)

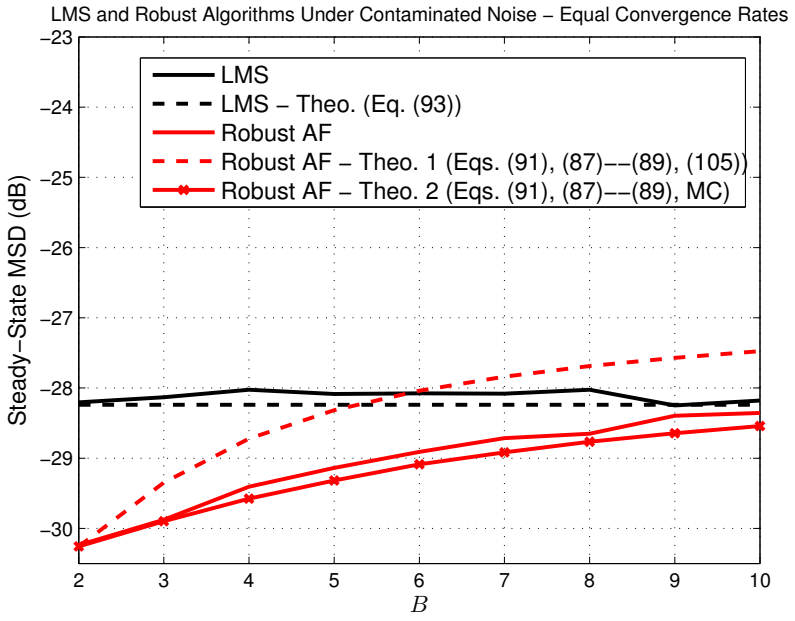


(b)

Figure 3.2: (a) Transient and (b) steady-state MSD performance of the LMS algorithm (black) compared with that of the robust adaptive algorithm (red) using the same step-size μ and for an increasing number of basis functions B. The measurements are corrupted by contaminated Gaussian noise, with $\varepsilon = 0.1$ and $\kappa = 100$, at $\bar{\sigma}_v^2 = -10$ dB. The dashed lines in Fig. 3.2b represent the theoretical steady-state MSD according to (3.93), (3.91), and (3.87)–(3.89). The moments $\mathbb{E}\,\boldsymbol{\alpha}_\infty$ and R_{α_∞} are approximated in two ways: the first approximation is based on solving (3.105) and the second is based on Monte Carlo (MC) simulation.

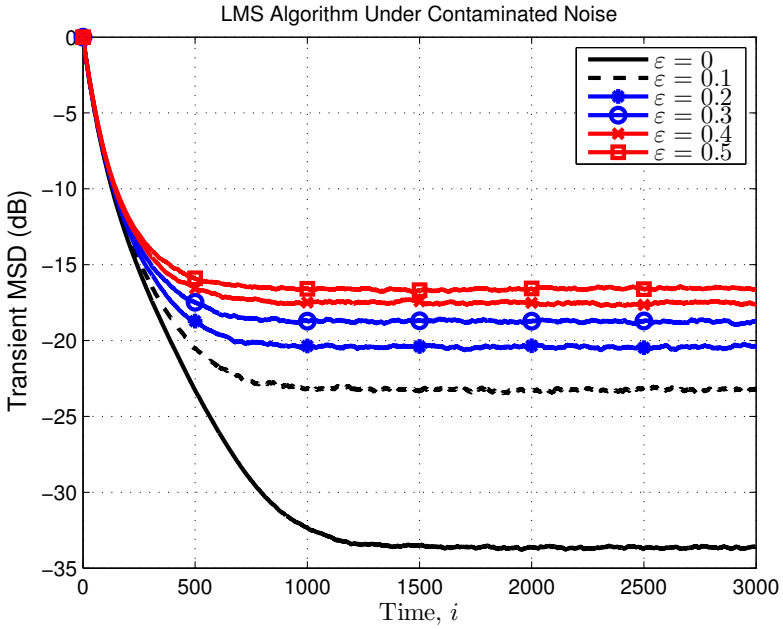


(a)

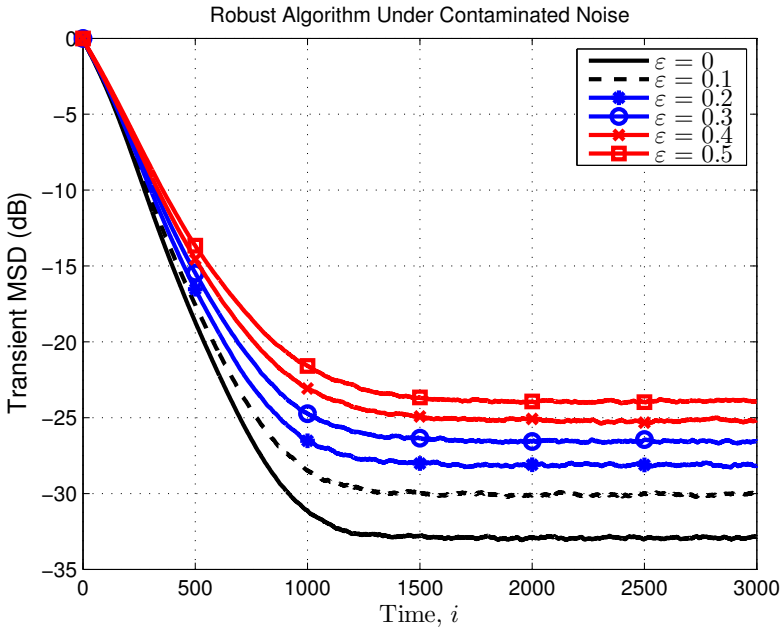


(b)

Figure 3.3: (a) Transient and (b) steady-state MSD performance of the LMS algorithm (black) compared with that of the robust adaptive algorithm (red) at equal initial convergence rates and for an increasing number of basis functions B. The measurements are corrupted by contaminated Gaussian noise, with $\varepsilon = 0.1$ and $\kappa = 100$, at $\bar{\sigma}_v^2 = -10$ dB. The dashed lines in Fig. 3.3b represent the theoretical steady-state MSD.

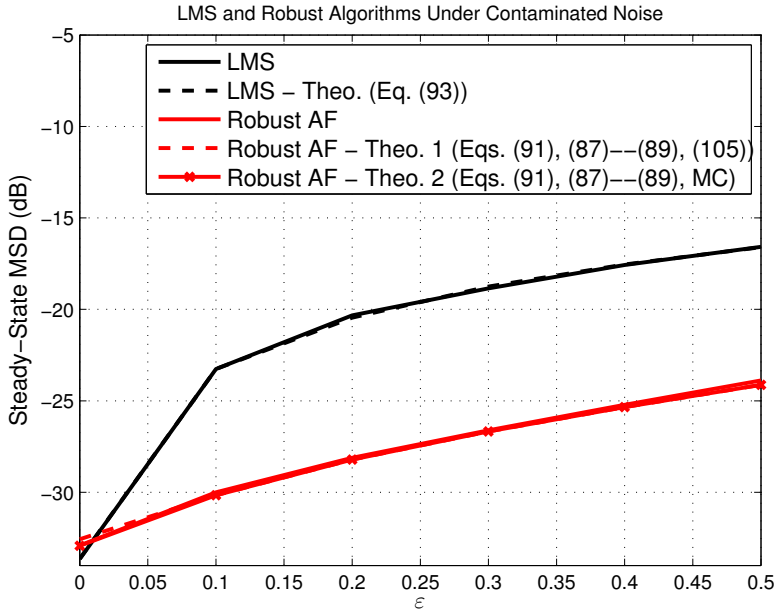


(a)

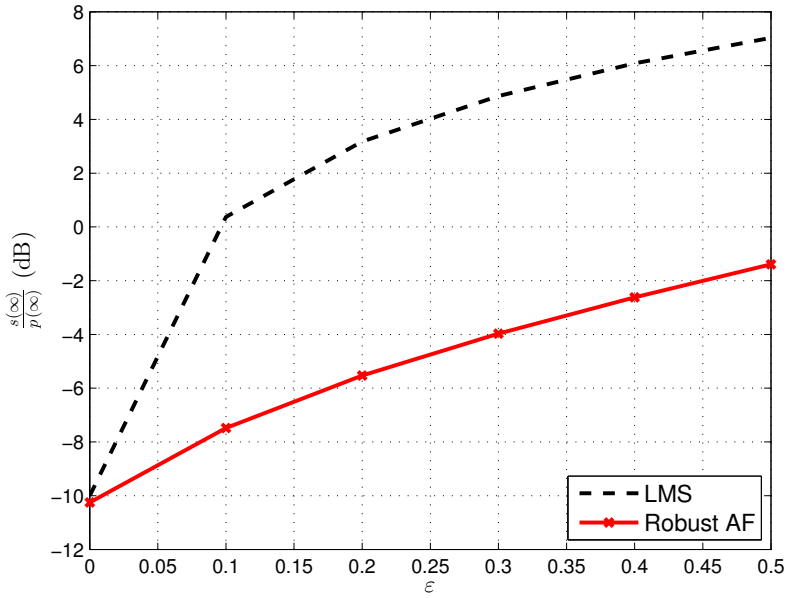


(b)

Figure 3.4: Transient MSD performance of the (a) LMS algorithm and (b) robust adaptive algorithm with $B = 2$ basis functions where the measurements are corrupted by contaminated Gaussian noise, at $\bar{\sigma}_v^2 = -10$ dB, with different contamination ratios ε, and $\kappa = 100$. The adaptation rate of both algorithms is kept constant for all ε and is chosen in such a way so that the convergence times of the robust algorithm and the LMS algorithm are the same at $\varepsilon = 0$ (no contamination).

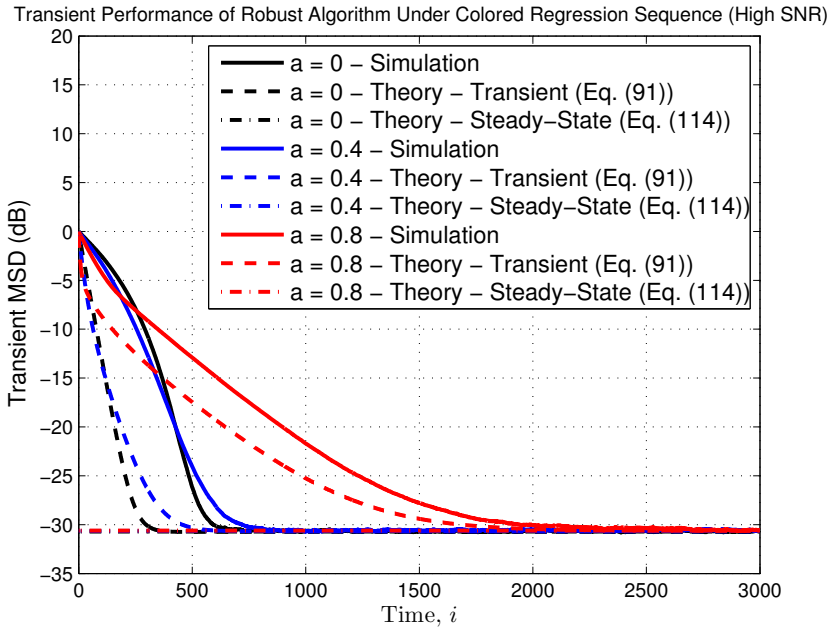


(a)

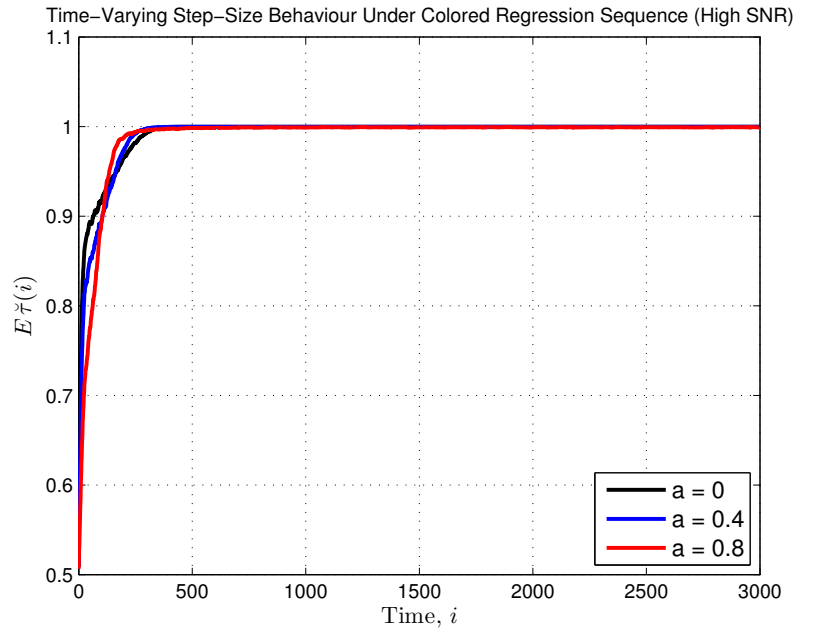


(b)

Figure 3.5: (a) Steady-state MSD performance of the LMS algorithm (black) compared with that of the robust adaptive algorithm (red) with $B = 2$ basis functions. The measurements are corrupted by contaminated Gaussian noise, at $\bar{\sigma}_v^2 = -10$ dB, with increasing contamination ratio ε, and $\kappa = 100$. The adaptation rate of both algorithms is kept constant for all ε and is chosen in such a way so that the convergence times of the robust algorithm in Fig. 3.4b and LMS algorithm in Fig. 3.4a are the same at $\varepsilon = 0$ (no contamination). The dashed lines represent the theoretical steady-state MSD. (b) The corresponding value of the ratio $\frac{s(\infty)}{p(\infty)}$ for increasing contamination ratio ε.

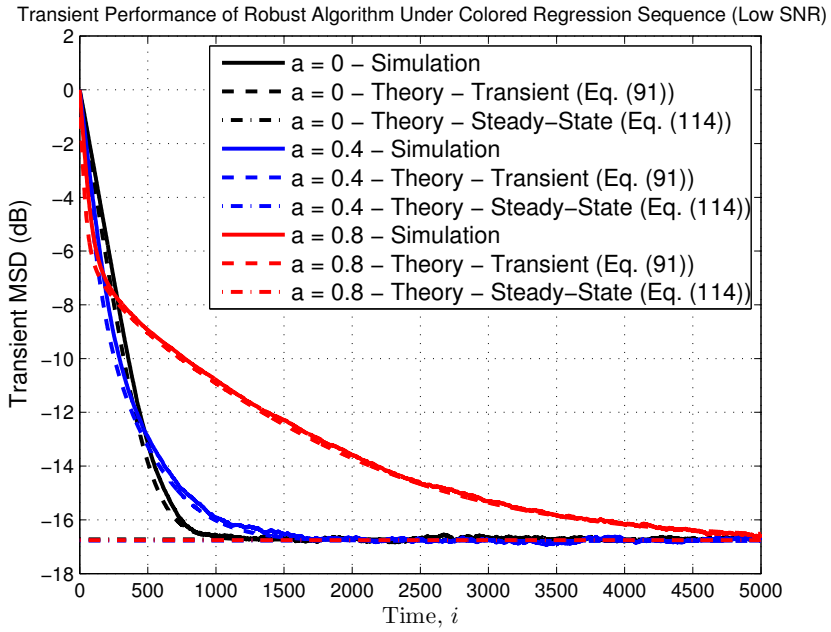


(a)

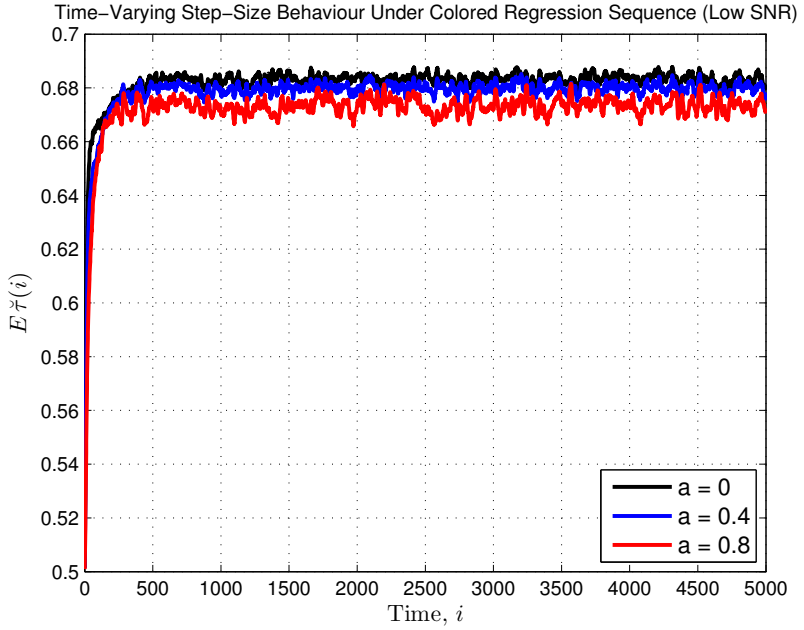


(b)

Figure 3.6: (a) Transient MSD performance of the robust algorithm with $B = 5$ basis functions under colored regression sequence with different degrees of correlation a in a high-SNR scenario: $\sigma_u^2 = 10$ dB and Gaussian noise with $\bar{\sigma}_v^2 = -10$ dB. The dashed lines represent the theoretical transient and steady-state MSD performance. (b) Temporal evolution of $\mathbb{E}\,\breve{\tau}(i)$.

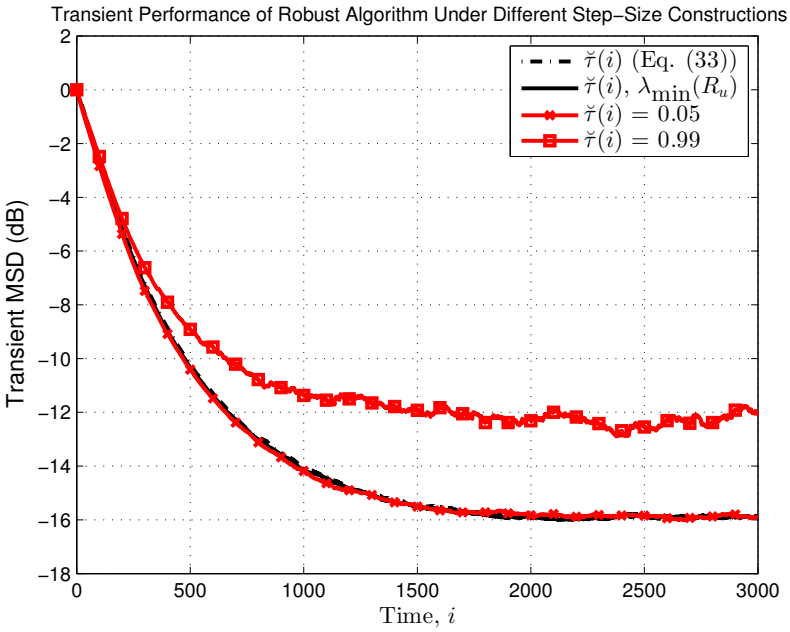


(a)

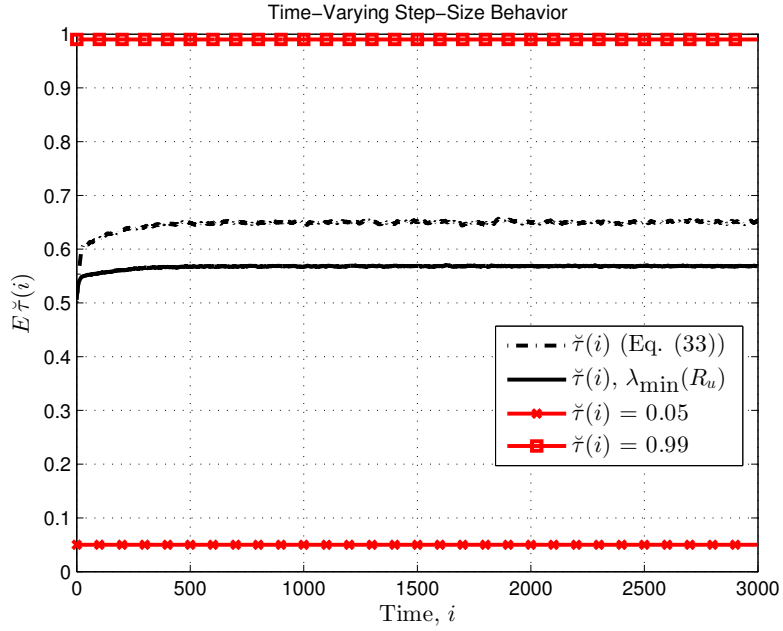


(b)

Figure 3.7: (a) Transient MSD performance of the robust algorithm with $B = 5$ basis functions under colored regression sequence with different degrees of correlation a in a low-SNR scenario: $\sigma_u^2 = 0$ dB and Gaussian noise with $\bar{\sigma}_v^2 = 0$ dB. The dashed lines represent the theoretical transient and steady-state MSD performance. (b) Temporal evolution of $\mathbb{E}\,\breve{\tau}(i)$.

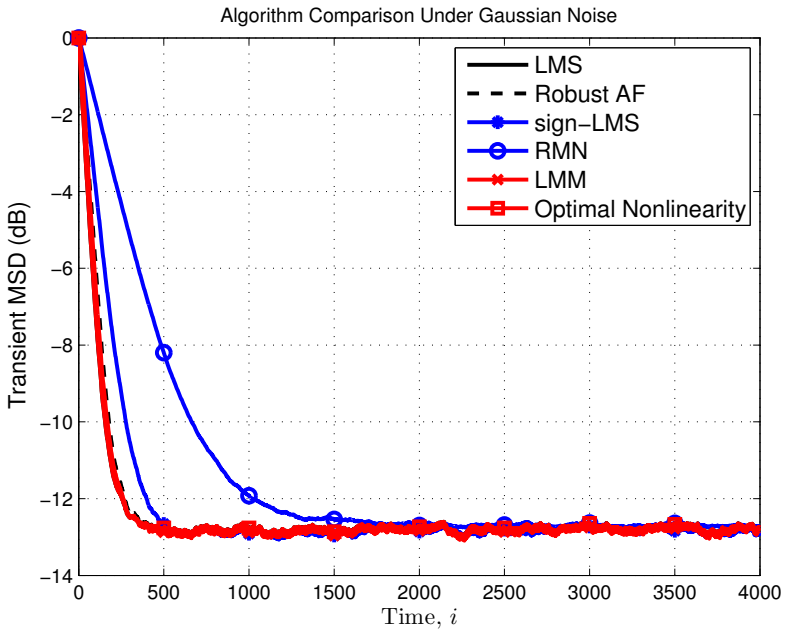


(a)

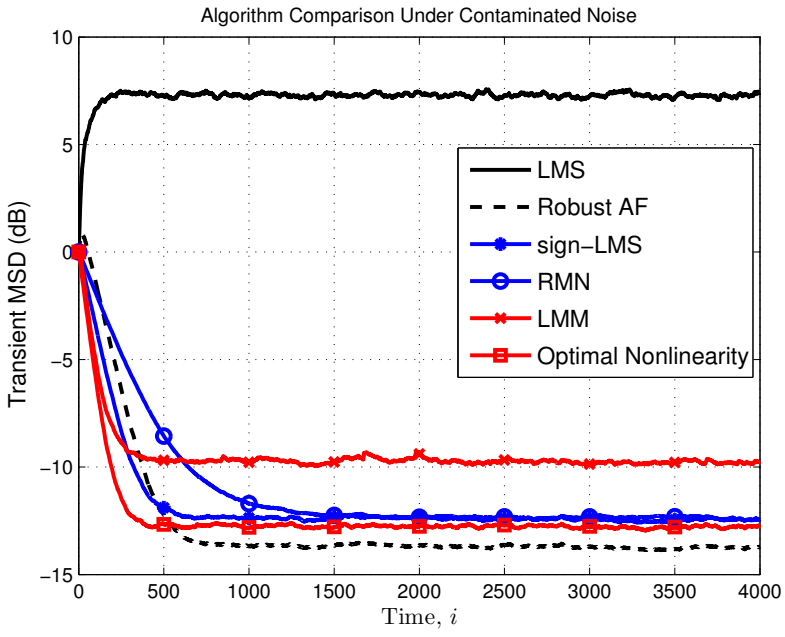


(b)

Figure 3.8: (a) Transient MSD performance of the robust algorithm with $B = 5$ basis functions under colored regression sequence ($a = 0.4$) with $\sigma_u^2 = 0$ dB and different constructions for the time-varying factor $\breve{\tau}(i)$. The measurements are corrupted by contaminated Gaussian noise, with $\varepsilon = 0.1$ and $\kappa = 1000$, at $\bar{\sigma}_v^2 = 0$ dB. (b) Temporal evolution of $\mathbb{E}\,\breve{\tau}(i)$.



(a)



(b)

Figure 3.9: MSD learning curves for the LMS, sign-LMS, robust mixed-norm (RMN), least-mean M-estimate (LMM) algorithms, and the robust algorithm developed here where the measurements are corrupted by (a) uncontaminated Gaussian noise and (b) contaminated Gaussian noise, with $\varepsilon = 0.1$ and $\kappa = 1000$, at $\bar{\sigma}_v^2 = 0$ dB. The step-sizes for these algorithms are the same as those in Fig. 3.9a at $\varepsilon = 0$ (no contamination). Also plotted is the learning curve given the optimal nonlinearity, using the same step-size as the robust algorithm.

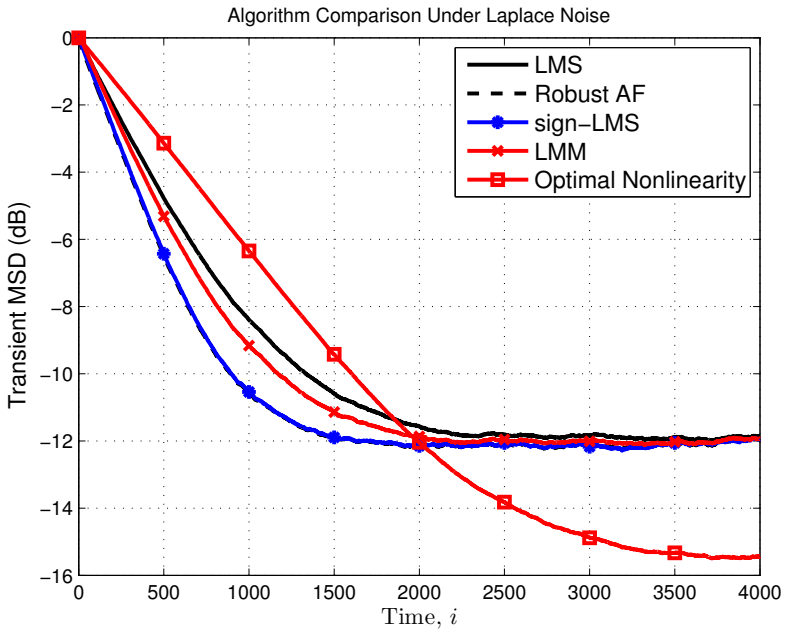


Figure 3.10: MSD learning curves under equalized steady-state MSDs for the LMS, sign-LMS, least-mean M-estimate (LMM) algorithms, and the robust algorithm of this work, with basis functions $\phi_1(x) = x$ and $\phi_2(x) = \text{sign}(x)$, where the measurements are corrupted by Laplace noise with $\sigma_v^2 = 10$ dB. Also plotted is the learning curve given the optimal nonlinearity, using the same step-size as the robust algorithm.

Chapter 4

Robust Adaptive Estimation Over Networks

In this chapter, the problem of distributed estimation over adaptive networks in the presence of impulsive noise is considered. To this end, each node in a graph topology cooperates with its neighbors, diffusing information through the network, in order to estimate parameters of interest using local neighborhood measurements that are corrupted by impulsive noise.

In Ch. 3, a robust adaptation strategy for stand-alone agents in the presence of impulsive noise was developed. In this chapter, the more challenging multi-agent scenario is studied, where a collection of agents are now coupled by the topology and work together to solve the estimation task in the presence of impulsive contamination across the network. Thus, a robust distributed adaptation strategy is called for.

By extending the framework of Ch. 3, a robust diffusion algorithm is developed in this chapter with automatic tuning and adaptation abilities; one that seeks the unknown parameter while at the same time, by means of an embedded step, identifying the optimal error nonlinearity for enhanced robustness. The performance of the resulting algorithm is examined and supporting simulations are provided.

In Sec. 4.1, the robust algorithm developed in Ch. 3 is extended to solve the problem of robust distributed estimation over adaptive networks, and subsequently analyzed in Sec. 4.2 using the energy conservation analysis framework [Say03, Say14b, Say14a]. In Sec. 4.3, simulation results are presented. Conclusions are drawn in Sec. 4.4.[1]

4.1 Distributed Estimation

4.1.1 Data Model and Problem Formulation

Considered here is a network composed of N nodes distributed over some region in space. Two nodes that can exchange data are said to be connected. The set of nodes

[1] This chapter has served as basis for the journal article:
S. Al-Sayed, A. M. Zoubir, and A. H. Sayed, "Robust distributed estimation by networked agents," submitted to *IEEE Trans. Signal Process.*, 2016.

connected to node k, including itself, is referred to as its neighborhood, and is denoted by $\mathcal{N}_k$. The degree of node k, denoted by n_k, is the number of its neighbors. At each time index $i \geq 0$, each node k has access to a real-valued scalar measurement $d_k(i)$ relating to an unknown real-valued vector parameter w^o of size M according to

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^o + \boldsymbol{v}_k(i) \tag{4.1}$$

where $\boldsymbol{u}_{k,i}$ is a real-valued row regression vector of size M; and $\{\boldsymbol{v}_k(i)\}$ is a real-valued scalar wide-sense stationary zero-mean impulsive noise process with variance $\sigma_{v,k}^2$. The random processes $\{\boldsymbol{d}_k(i)\}$ and $\{\boldsymbol{u}_{k,i}\}$ are zero-mean and jointly stationary. The regressors $\boldsymbol{u}_{k,i}$ and $\boldsymbol{u}_{\ell,j}$ are spatially and temporally independent for $k \neq \ell$ or $i \neq j$, where the covariance matrix of $\boldsymbol{u}_{k,i}$ is denoted as $R_{u,k} = \mathbb{E}\, \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i}$ and assumed to be positive definite. The random variables $\boldsymbol{v}_k(i)$ and $\boldsymbol{v}_\ell(j)$ are also spatially and temporally independent for $k \neq \ell$ or $i \neq j$. It is assumed that the noise probability density functions (pdfs), $f_{\boldsymbol{v}_k}(v)$, are symmetric for all k, i.e., $\mathbb{E}\, \boldsymbol{v}_k^{2p-1}(i) = 0$, $p = 1, 2, \ldots$ Moreover, the random variables $\boldsymbol{u}_{k,i}$ and $\boldsymbol{v}_\ell(j)$ are independent for all k, ℓ, i, and j.

The aim is for the nodes to adaptively estimate w^o, availing themselves of the data collected from their neighbors in order to minimize the following *global* mean-square-error (MSE) cost function:

$$J(w) \triangleq \sum_{k=1}^{N} \mathbb{E}\big(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w\big)^2. \tag{4.2}$$

In [CS10, STC+13, Say14b, Say14a], the following adapt-then-combine (ATC) least-mean-squares (LMS) diffusion estimation algorithm was developed to minimize (4.2). Consider an $N \times N$ matrix A with nonnegative real entries $a_{\ell k}$ satisfying

$$a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \quad \mathbb{1}^T A = \mathbb{1}^T. \tag{4.3}$$

Let $e_k(i) \triangleq d_k(i) - u_{k,i} w_{k,i-1}$ denote the output error of the kth node at time index i. The update equations of the algorithm for each node k for $i \geq 0$ are given by [CS10, STC+13, Say14b, Say14a]:

$$\left\{ \begin{array}{rcl} \psi_{k,i} & = & w_{k,i-1} + \mu_k u_{k,i}^T e_k(i) \\ w_{k,i} & = & \sum\limits_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{array} \right. \tag{4.4}$$

where μ_k is a small positive step-size parameter; and the initial condition $w_{k,-1}$ may be chosen arbitrarily. At each time index i, each node k updates its current estimate $w_{k,i-1}$ in an LMS fashion using its own data $u_{k,i}$ and $d_k(i)$, forming an intermediate estimate, $\psi_{k,i}$. Each node k then collects the intermediate estimates from its neighbors in $\mathcal{N}_k$, and weights them according to some combination policy satisfying (4.3), hence

forming the final estimate $w_{k,i}$. In a manner similar to the single-agent case discussed in Ch. 3, the performance of algorithm (4.4) may degrade in the presence of impulsive noise. The purpose here is to devise a distributed version that is robust to such noise processes.

4.1.2 Robust Diffusion Adaptation

Motivated by the discussion in Ch. 3, one may introduce an agent-dependent and time-varying error nonlinearity, $h_{k,i}\big(e_k(i)\big)$, into the adaptation step:

$$\psi_{k,i} = w_{k,i-1} + \mu_k u_{k,i}^T h_{k,i}\big(e_k(i)\big) \tag{4.5}$$

and select it to be a linear combination of $B_k \geq 1$ preselected sign-preserving basis functions, i.e.,

$$h_{k,i}\big(e_k(i)\big) = \alpha_{k,i}^T \varphi_{k,i}. \tag{4.6}$$

Here, the vectors $\alpha_{k,i}$ and $\varphi_{k,i}$ have length B_k, and they consist of nonnegative combination weights at time index i and the values of the preselected basis functions evaluated at $e_k(i)$, namely,

$$\alpha_{k,i} \triangleq [\alpha_{k,i}(1), \ldots, \alpha_{k,i}(B_k)]^T \tag{4.7}$$

$$\varphi_{k,i} \triangleq \big[\phi_{k,1}\big(e_k(i)\big), \ldots, \phi_{k,B_k}\big(e_k(i)\big)\big]^T \tag{4.8}$$

If node k were to run the stand-alone counterpart of the adaptive filter in (4.5), by setting $w_{k,i}$ to $\psi_{k,i}$, then the optimal nonlinearity that minimizes node k's MSE is given by [ANS01]:

$$h_{k,i}^{\text{opt}}(x) = -\frac{f'_{\boldsymbol{e}_k(i)}(x)}{f_{\boldsymbol{e}_k(i)}(x)} \tag{4.9}$$

in terms of the pdf of the error signal, where the notation $g'(x)$ stands for $\frac{dg(x)}{dx}$. Since the pdf is not available in practice, the nonlinearity is chosen instead according to (4.6), and the vector $\alpha_{k,i}$ is found by minimizing the MSE between $h_{k,i}\big(e_k(i)\big)$ and the optimal nonlinearity:

$$\alpha_{k,i}^{\text{opt}} \triangleq \arg\min_{\alpha_{k,i}} \mathbb{E}\Big(h_{k,i}^{\text{opt}}\big(\boldsymbol{e}_k(i)\big) - h_{k,i}\big(\boldsymbol{e}_k(i)\big)\Big)^2. \tag{4.10}$$

For online adaptation purposes, each node k estimates $\alpha_{k,i}^{\text{opt}}$ adaptively and jointly with w^o, by recourse to a stochastic-gradient recursion for (4.10) and subject to a convexity constraint on the entries of $\alpha_{k,i}$, i.e., they are constrained to be nonnegative and add up to one, to ensure boundedness. Following the same derivation as in Ch. 3 in the single-agent case, we arrive at the multi-agent robust version of the diffusion strategy (4.4)

listed in Table 4.1, where now $h_k(i)$ is being used instead of the more explicit notation $h_{k,i}\big(e_k(i)\big)$ used in (4.5). The ensuing moments, $R_{\varphi_{k,i}} \triangleq \mathbb{E}\,\boldsymbol{\varphi}_{k,i}\boldsymbol{\varphi}_{k,i}^T$ and $\mathbb{E}\,\boldsymbol{\varphi}'_{k,i}$, where a primed vector denotes entry-wise differentiation, are estimated in (4.13d) and (4.13g) by means of smoothing recursions, where $\nu_k \in (0,1)$ is a constant, usually chosen close to one; $\epsilon > 0$ is a very small constant to prevent division by zero; $\mathrm{sgm}(x) \triangleq \frac{1}{1+e^{-x}} \in (0,1)$ is the sigmoid function; and $\|.\|_\infty$ denotes the maximum absolute entry of its vector argument. Moreover, we have $\Pi_k \triangleq I - \frac{\mathbb{1}\mathbb{1}^T}{B_k}$, of size B_k, and

$$\Omega_{++,k} \triangleq \Big\{\alpha \in \mathbb{R}_{++}^{B_k} | \alpha^T \mathbb{1} = 1\Big\} \tag{4.11}$$

where $\mathbb{R}_{++}^{B_k}$ is the set of $B_k \times 1$ vectors on the set of positive real numbers $\mathbb{R}_{++}$.

For impulsive noise scenarios, a sensible choice of basis that scales down impulsive samples and trades off robustness with LMS performance under Gaussian noise is $\phi_{k,1}(x) = x$ for all k and $\phi_{k,b}(x)$, $b = 2, \ldots, B_k$, some bounded nonlinear functions, e.g., the hyperbolic tangent basis:

$$\phi_{k,b}(x) = \tanh\big((b-1)x\big), \quad b = 2, \ldots, B_k. \tag{4.12}$$

Remark 9. The case $B_k = 1$ for all k for whichever choices of $\{\phi_{k,1}(x)\}$ amounts to $\alpha_k(i) = 1$ for all k and i. The analysis of the resulting algorithm was treated in [CS15a, CS15b]. The diffusion LMS algorithm is recovered when $\phi_{k,1}(x) = x$ for all k.

4.2 Performance of Robust Diffusion Estimation Algorithm

In this section, the performance of the robust diffusion estimation algorithm is analyzed subject to the data model described in Sec. 4.1.1. Let $\widetilde{\boldsymbol{w}}_{k,i} \triangleq w^o - \boldsymbol{w}_{k,i}$ denote node k's weight-error vector at time index i. The following assumptions are introduced, which are analogous to *(A2)*–*(A4)* from Ch. 3:

- *(D-A1)* $\boldsymbol{\alpha}_{k,i}$ is independent of $\boldsymbol{u}_{\ell,i}$, $\boldsymbol{v}_\ell(i)$, and $\widetilde{\boldsymbol{w}}_{\ell,i-1}$ for all k, ℓ, and i.
- *(D-A2)* The step-sizes $\{\mu_k\}$ are sufficiently small.
- *(D-A3)* The basis functions $\{\phi_{k,b}(x)\}$ are sign-preserving, odd-symmetric, monotonically increasing, and differentiable.

Assumption *(D-A1)* is reasonable under small step-sizes $\{\mu_k\}$, more so when the $\{\nu_k\}$ are close to one, and asymptotically, as $i \to \infty$ [KJ92]—see Ch. 3 for further discussion.

Table 4.1: Robust Diffusion Estimation Algorithm

Initializations: A, ϵ, B_k, $\{\phi_{k,b}(x)\}$, Π_k, $\alpha_{k,-1} \in \Omega_{++,k}$, $\widehat{R}_{\varphi_{k,-1}}$, $\widehat{\varphi}'_{k,-1}$, ν_k, $\widehat{\lambda}_k(-1)$, μ_k for every node k. Start with $w_{k,-1} = 0$ for every node k. For every time index $i \geq 0$, repeat

Error nonlinearity update: for every node k, repeat

$$e_k(i) = d_k(i) - u_{k,i} w_{k,i-1} \tag{4.13a}$$
$$\phi_{k,b}(i) \equiv \phi_{k,b}\big(e_k(i)\big),\ b = 1, \ldots, B_k \tag{4.13b}$$
$$\varphi_{k,i} = \operatorname{col}\{\phi_{k,1}(i), \ldots, \phi_{k,B_k}(i)\} \tag{4.13c}$$
$$\widehat{R}_{\varphi_{k,i}} = \nu_k \widehat{R}_{\varphi_{k,i-1}} + (1-\nu_k)\,\varphi_{k,i}\varphi_{k,i}^T \tag{4.13d}$$
$$\phi'_{k,b}(i) \equiv \phi'_{k,b}\big(e_k(i)\big),\ b = 1, \ldots, B_k \tag{4.13e}$$
$$\varphi'_{k,i} = \operatorname{col}\big\{\phi'_{k,1}(i), \ldots, \phi'_{k,B_k}(i)\big\} \tag{4.13f}$$
$$\widehat{\varphi}'_{k,i} = \nu_k \widehat{\varphi}'_{k,i-1} + (1-\nu_k)\,\varphi'_{k,i} \tag{4.13g}$$
$$\delta_{k,i} = 2\Pi_k(\widehat{R}_{\varphi_{k,i}}\alpha_{k,i-1} - \widehat{\varphi}'_{k,i}) \tag{4.13h}$$
$$\widehat{\lambda}_k(i) = \nu_k \widehat{\lambda}_k(i-1) + (1-\nu_k)\,\frac{\|u_{k,i}\|^2}{M} \tag{4.13i}$$
$$\breve{\tau}_k(i) = \operatorname{sgm}\Big[\big(\alpha_{k,i-1}^T \widehat{\varphi}'_{k,i}\big)\,\widehat{\lambda}_k(i)\Big] \tag{4.13j}$$
$$\tau_k(i) = \breve{\tau}_k(i)\frac{\min\{\alpha_{k,i-1}(b), 1 \leq b \leq B_k\}}{\|\delta_{k,i}\|_\infty + \epsilon} \tag{4.13k}$$
$$\alpha_{k,i} = \alpha_{k,i-1} - \tau_k(i)\delta_{k,i} \tag{4.13l}$$
$$h_k(i) = \alpha_{k,i}^T \varphi_{k,i} \tag{4.13m}$$

Adaptation step: for every node k, repeat

$$\psi_{k,i} = w_{k,i-1} + \mu_k u_{k,i}^T h_k(i) \tag{4.14}$$

Combination step: for every node k, repeat

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k}\psi_{\ell,i} \tag{4.15}$$

4.2.1 Error Recursions

We recall the update equations for each node k:

$$\begin{aligned} \boldsymbol{\psi}_{k,i} &= \boldsymbol{w}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i}^T \boldsymbol{h}_k(i) \\ \boldsymbol{w}_{k,i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k}\boldsymbol{\psi}_{\ell,i} \end{aligned} \tag{4.16}$$

where $\boldsymbol{h}_k(i) = \boldsymbol{\alpha}_{k,i}^T \boldsymbol{\varphi}_{k,i}$. We introduce the error quantity

$$\widetilde{\boldsymbol{\psi}}_{k,i} \triangleq w^o - \boldsymbol{\psi}_{k,i}. \tag{4.17}$$

We further introduce the following quantities, which collect variables from across the network:

$$\boldsymbol{h}_i \triangleq \operatorname{col}\{\boldsymbol{h}_1(i), \ldots, \boldsymbol{h}_N(i)\} \tag{4.18}$$

$$\widetilde{\boldsymbol{w}}_i \triangleq \operatorname{col}\{\widetilde{\boldsymbol{w}}_{1,i}, \ldots, \widetilde{\boldsymbol{w}}_{N,i}\} \tag{4.19}$$

$$\widetilde{\boldsymbol{\psi}}_i \triangleq \operatorname{col}\left\{\widetilde{\boldsymbol{\psi}}_{1,i}, \ldots, \widetilde{\boldsymbol{\psi}}_{N,i}\right\} \tag{4.20}$$

$$\boldsymbol{\mathcal{U}}_i \triangleq \operatorname{diag}\{\boldsymbol{u}_{1,i}, \ldots, \boldsymbol{u}_{N,i}\} \tag{4.21}$$

$$\mathcal{R}_u \triangleq \operatorname{diag}\{R_{u,1}, \ldots, R_{u,N}\} \tag{4.22}$$

$$\mathcal{M} \triangleq \operatorname{diag}\{\mu_1 I_M, \ldots, \mu_N I_M\} \tag{4.23}$$

$$\mathcal{A} \triangleq A \otimes I_M \tag{4.24}$$

Then, exploiting (4.3), the recursions in (4.16) lead to

$$\widetilde{\boldsymbol{w}}_i = \mathcal{A}^T \widetilde{\boldsymbol{w}}_{i-1} - \mathcal{A}^T \mathcal{M} \boldsymbol{\mathcal{U}}_i^T \boldsymbol{h}_i. \tag{4.25}$$

From model (4.1), it holds that $\boldsymbol{e}_k(i) = \boldsymbol{e}_{a,k}(i) + \boldsymbol{v}_k(i)$, where $\boldsymbol{e}_{a,k}(i) = \boldsymbol{u}_{k,i}\widetilde{\boldsymbol{w}}_{k,i-1}$ is the *a priori* error of node k. In the sequel, each $\boldsymbol{h}_k(i)$ is approximated, similarly as in Ch. 3, using a first-order Taylor series approximation of the basis functions $\{\phi_{k,b}(x)\}$ around $e_{a,k}(i) = 0$ for all $i \geq 0$ as follows:

$$\begin{aligned}
\boldsymbol{h}_k(i) &= \sum_{b=1}^{B} \boldsymbol{\alpha}_{k,i}(b)\phi_{k,b}\big(\boldsymbol{e}_k(i)\big) \\
&\approx \sum_{b=1}^{B} \boldsymbol{\alpha}_{k,i}(b)\boldsymbol{\phi}_{v,k,b}(i) + \boldsymbol{e}_{a,k}(i)\sum_{b=1}^{B} \boldsymbol{\alpha}_{k,i}(b)\boldsymbol{\phi}'_{v,k,b}(i) \\
&= \boldsymbol{\alpha}_{k,i}^T \boldsymbol{\varphi}_{v,k,i} + \boldsymbol{e}_{a,k}(i)\boldsymbol{\alpha}_{k,i}^T \boldsymbol{\varphi}'_{v,k,i}
\end{aligned} \tag{4.26}$$

where

$$\boldsymbol{\phi}_{v,k,b}(i) \equiv \phi_{k,b}\big(\boldsymbol{v}_k(i)\big), \quad b = 1, \ldots, B_k \tag{4.27}$$

$$\boldsymbol{\varphi}_{v,k,i} = \operatorname{col}\{\boldsymbol{\phi}_{v,k,1}(i), \ldots, \boldsymbol{\phi}_{v,k,B_k}(i)\} \tag{4.28}$$

4.2.2 Mean Performance

Taking the expectation of both sides of (4.25),

$$\mathbb{E}\,\widetilde{\boldsymbol{w}}_i = \mathcal{A}^T \mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1} - \mathcal{A}^T \mathcal{M} \mathbb{E}\,\boldsymbol{\mathcal{U}}_i^T \boldsymbol{h}_i. \tag{4.29}$$

We now evaluate $\mathbb{E}\,\boldsymbol{\mathcal{U}}_i^T \boldsymbol{h}_i$ using (4.26) under (*D-A1*)–(*D-A3*). First, recall the model assumptions on the regressors $\{\boldsymbol{u}_{k,i}\}$, namely, that they are spatially and temporally independent. Likewise, the noise samples $\{\boldsymbol{v}_k(i)\}$ are spatially and temporally independent with symmetric pdfs and independent of the regressors. It follows that $\boldsymbol{u}_{k,i}$ and $\widetilde{\boldsymbol{w}}_{k,i-1}$ are independent for all k and i so that

$$\mathbb{E}\,\boldsymbol{u}_{k,i}^T \boldsymbol{h}_k(i) = p_k(i) R_{u,k} \mathbb{E}\,\widetilde{\boldsymbol{w}}_{k,i-1} \tag{4.30}$$

where

$$p_k(i) = (\mathbb{E}\,\boldsymbol{\alpha}_{k,i})^T \left(\mathbb{E}\,\boldsymbol{\varphi}'_{v,k,i}\right) \triangleq (\mathbb{E}\,\boldsymbol{\alpha}_{k,i})^T\, \overline{\varphi'_{v,k}}. \tag{4.31}$$

The time subscript i has been dropped from $\overline{\varphi'_{v,k}} \triangleq \mathbb{E}\,\boldsymbol{\varphi}'_{v,k,i}$ since the moment is time-invariant for wide-sense stationary noise processes. Introducing the following matrices:

$$P_i \triangleq \operatorname{diag}\{p_1(i), \ldots, p_N(i)\} \tag{4.32}$$

$$\mathcal{P}_i \triangleq P_i \otimes I_M \tag{4.33}$$

it follows that

$$\mathbb{E}\,\boldsymbol{\mathcal{U}}_i^T \boldsymbol{h}_i = \mathcal{R}_u \mathcal{P}_i \mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}. \tag{4.34}$$

The mean weight-error recursion (4.29) then becomes

$$\mathbb{E}\,\widetilde{\boldsymbol{w}}_i = \mathcal{A}^T \left[I - \mathcal{M}\mathcal{R}_u \mathcal{P}_i\right] \mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}. \tag{4.35}$$

In [CS11, Lemma 2], the authors derived sufficient conditions for the asymptotic unbiasedness of the weight estimates $\{\boldsymbol{w}_{k,i}\}$ given weight-error recursions of a general form similar to (4.35). In particular, the weight estimates $\{\boldsymbol{w}_{k,i}\}$ are asymptotically unbiased for all nodes $k = 1, \ldots, N$, if there exists a time index i^*, a number $0 < \theta < 1$, and a submultiplicative norm $\|\cdot\|$ such that $\left\|\mathcal{A}^T \left[I - \mathcal{M}\mathcal{R}_u \mathcal{P}_i\right]\right\| \leq \theta < 1$ for all $i > i^*$. We can outline two special cases where this condition is satisfied in terms of well-known norms. First, since the matrix A has nonnegative entries and satisfies $\mathbb{1}^T A = \mathbb{1}$, then the maximum absolute row sum, or ∞-norm, of A^T is $\left\|A^T\right\|_\infty = 1$. The same holds for the matrix $\mathcal{A}^T$. A sufficient condition for asymptotic stability can then be derived as $\|I - \mu_k p_k(i) R_{u,k}\|_\infty \leq \theta < 1$, for some θ and for all k and $i > i^*$. Another sufficient condition can be derived in terms of the 2-norm. Since $\|A\|_1 = 1$ (maximum absolute column sum), it follows that the spectral radius of A is equal to one. Then, if A is symmetric, its spectral radius and 2-norm coincide, i.e., $\|A\|_2 = 1$. Let $\{\lambda_m(R_{u,k})\}$, $m = 1, \ldots, M$, denote the eigenvalues of $R_{u,k}$. A sufficient condition for asymptotic stability can then be derived by requiring $|1 - \mu_k p_k(i) \lambda_m(R_{u,k})| \leq \theta < 1$, for some θ and for all m, k, and $i > i^*$.

4.2.3 Mean-Square Performance

The following additional assumption is made:

- *(D-A4)* For sufficiently large i, the process $\{\boldsymbol{\alpha}_{k,i}\}$ is i.i.d. for all k, where the first- and second-order moments $\mathbb{E}\,\boldsymbol{\alpha}_{k,i}$ and $R_{\alpha_{k,i}} \triangleq \mathbb{E}\,\boldsymbol{\alpha}_{k,i}\boldsymbol{\alpha}_{k,i}^T$ have reached finite constant values denoted by

$$\mathbb{E}\,\boldsymbol{\alpha}_{k,\infty} \triangleq \lim_{i\to\infty} \mathbb{E}\,\boldsymbol{\alpha}_{k,i}, \qquad R_{\alpha_{k,\infty}} \triangleq \lim_{i\to\infty} R_{\alpha_{k,i}}. \tag{4.36}$$

This assumption is for mathematical tractability. Simulations indicate that with the $\tau_k(i)$ construction we have devised, $\boldsymbol{\alpha}_{k,i}$ converges in step with $\boldsymbol{w}_{k,i}$.

We rewrite the adaptation step in (4.16) as

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu_k \hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}_{k,i-1}) \tag{4.37}$$

in terms of the update vector

$$\begin{aligned} \hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}_{k,i-1}) &= -\boldsymbol{u}_{k,i}^T \boldsymbol{h}_k(i) \\ &\approx -\boldsymbol{u}_{k,i}^T \left[\boldsymbol{\alpha}_{k,i}^T \boldsymbol{\varphi}_{v,k,i} + \boldsymbol{e}_{a,k}(i) \boldsymbol{\alpha}_{k,i}^T \boldsymbol{\varphi}'_{v,k,i}\right] \end{aligned} \tag{4.38}$$

The update vector $\hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}_{k,i-1})$ is written explicitly in terms of $\boldsymbol{w}_{k,i-1}$, since it depends on $\boldsymbol{h}_k(i) \equiv h_{k,i}\big(\boldsymbol{e}_k(i)\big)$ and $\boldsymbol{e}_k(i) = \boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}$. In the following lemma, it will be shown that $\hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}_{k,i-1})$ satisfies certain properties.

Lemma 1 (Update Vector Properties). Under the model assumptions, (*D-A1*)–(*D-A4*), and the Taylor series approximation (4.26), the approximate update vector $\hat{\boldsymbol{g}}_{k,i}$ (4.38) for each k and sufficiently large i satisfies the following properties:

Randomness: There exists an $M \times 1$ deterministic vector function $g_k(\boldsymbol{w})$ such that, for all $M \times 1$ vectors $\boldsymbol{w}$ in the filtration $\boldsymbol{\mathcal{F}}_{i-1}$ generated by the past history of iterates $\{\boldsymbol{w}_{k,j}\}$ for $j \leq i-1$ and all k, it holds that

$$\mathbb{E}\left\{\hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}) | \boldsymbol{\mathcal{F}}_{i-1}\right\} = g_k(\boldsymbol{w}). \tag{4.39}$$

Moreover, for all k, there exist $\beta_k \geq 0$ and $\sigma_{g,k}^2 \geq 0$ such that for all $\boldsymbol{w} \in \boldsymbol{\mathcal{F}}_{i-1}$, it holds that

$$\mathbb{E}\left\{\|\hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}) - g_k(\boldsymbol{w})\|^2 \big| \boldsymbol{\mathcal{F}}_{i-1}\right\} \leq \beta_k \|w^o - \boldsymbol{w}\|^2 + \sigma_{g,k}^2. \tag{4.40}$$

Lipschitz: There exist $\lambda_{U,k} \geq 0$ for all k such that for all x, $y \in \mathbb{R}^M$ and all k, it holds that

$$\|g_k(x) - g_k(y)\| \leq \lambda_{U,k} \|x - y\| \tag{4.41}$$

where the subscript "U" in $\lambda_{U,k}$ refers to the latter determining the upper bound.

Strong monotonicity: There exist $\lambda_{L,k} > 0$ for all k such that for all x, $y \in \mathbb{R}^M$ and all k, it holds that

$$(x-y)^T [g_k(x) - g_k(y)] \geq \lambda_{L,k} \|x - y\|^2 \tag{4.42}$$

where the subscript "L" in $\lambda_{L,k}$ refers to the latter determining the lower bound. □

Proof. See Appendix A.5. ■

The arguments in the appendix show that

$$g_k(\boldsymbol{w}) = -p_k(\infty)R_{u,k}\,(w^o - \boldsymbol{w}) \tag{4.43}$$

for each k, where

$$p_k(\infty) \triangleq \lim_{i\to\infty} p_k(i) = (\mathbb{E}\,\boldsymbol{\alpha}_{k,\infty})^T\,\overline{\varphi'_{v,k}}. \tag{4.44}$$

Since expression (4.43) shows that $g_k(w)$ is differentiable for all k, the following property, equivalent to the Lipschitz and strong monotonicity properties, can be established. Let the $M \times M$ matrix D_k denote the gradient of the vector function $g_k(w)$, i.e.,

$$D_k \triangleq \nabla_w g_k(w) = p_k(\infty)R_{u,k}. \tag{4.45}$$

Then, the matrix D_k is bounded for all k as

$$\lambda_{L,k} I_M \le D_k \le \lambda_{U,k} I_M \tag{4.46}$$

where $\lambda_{L,k}$ and $\lambda_{U,k}$ were shown in Appendix A.5 to be given by

$$\lambda_{L,k} = p_k(\infty)\lambda_{\min}(R_{u,k}) \tag{4.47}$$

$$\lambda_{U,k} = p_k(\infty)\lambda_{\max}(R_{u,k}) \tag{4.48}$$

where $\lambda_{\min}(\cdot)$ and $\lambda_{\min}(\cdot)$ denote the minimum and maximum eigenvalues of their square matrix arguments, respectively. Let $\overline{\phi'_{v,k,b}} \triangleq \mathbb{E}\,\boldsymbol{\phi}'_{v,k,b}(i)$ for all k and b. Then, it holds by virtue of the convexity of the entries of $\boldsymbol{\alpha}_{k,i}$ for all k and i that

$$\min_{1\le b\le B_k} \overline{\phi'_{v,k,b}} \le p_k(\infty) \le \max_{1\le b\le B_k} \overline{\phi'_{v,k,b}}. \tag{4.49}$$

Hence, $\lambda_{L,k}$ and $\lambda_{U,k}$ in (4.46) can be replaced with

$$\bar{\lambda}_{L,k} \triangleq \left(\min_{1\le b\le B_k} \overline{\phi'_{v,k,b}}\right)\lambda_{\min}(R_{u,k}) \tag{4.50}$$

$$\bar{\lambda}_{U,k} \triangleq \left(\max_{1\le b\le B_k} \overline{\phi'_{v,k,b}}\right)\lambda_{\max}(R_{u,k}) \tag{4.51}$$

for all k since

$$0 < \bar{\lambda}_{L,k} \le \lambda_{L,k}, \qquad \bar{\lambda}_{U,k} \ge \lambda_{U,k}. \tag{4.52}$$

Now, let

$$\boldsymbol{v}^g_{k,i}(\boldsymbol{w}) \triangleq \hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}) - g_k(\boldsymbol{w}) \tag{4.53}$$

represent the noise incurred by stochastic approximation for each node k and any $\boldsymbol{w} \in \boldsymbol{\mathcal{F}}_{i-1}$. We refer to it as the update noise vector. Then, the update equations (4.16) with the adaptation step reformulated as in (4.37) and using (4.53) become

$$\begin{aligned}\boldsymbol{\psi}_{k,i} &= \boldsymbol{w}_{k,i-1} - \mu_k\left[g_k(\boldsymbol{w}_{k,i-1}) + \boldsymbol{v}^g_{k,i}(\boldsymbol{w}_{k,i-1})\right] \\ \boldsymbol{w}_{k,i} &= \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\boldsymbol{\psi}_{\ell,i}\end{aligned} \tag{4.54}$$

Subtracting from w^o and exploiting (4.3) results in the following form for the error recursions:

$$\begin{aligned}\widetilde{\boldsymbol{\psi}}_{k,i} &= \widetilde{\boldsymbol{w}}_{k,i-1} + \mu_k \left[g_k(\boldsymbol{w}_{k,i-1}) + \boldsymbol{v}_{k,i}^g(\boldsymbol{w}_{k,i-1})\right] \\ \widetilde{\boldsymbol{w}}_{k,i} &= \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\widetilde{\boldsymbol{\psi}}_{\ell,i}\end{aligned} \tag{4.55}$$

We recall the mean-value theorem for any real-valued M-dimensional differentiable vector function $f(x)$, with M-dimensional vector argument x, and any M-dimensional vectors x_1 and x_2 [Pol87]:

$$f(x_2) = f(x_1) + \left[\int_0^1 \nabla_x f(x_1 + t\Delta x)dt\right]\Delta x \tag{4.56}$$

where $t \in [0,1]$ is a scalar variable and $\Delta x \triangleq x_2 - x_1$. We invoke the result on $g_k(w)$ with $x = w$, $x_1 = w^o$, and $x_2 = \boldsymbol{w}_{k,i-1}$ to obtain:

$$\begin{aligned}g_k(\boldsymbol{w}_{k,i-1}) &= g_k(w^o) \\ &\quad - \left[\int_0^1 \nabla_w g_k(w^o - t\widetilde{\boldsymbol{w}}_{k,i-1})dt\right]\widetilde{\boldsymbol{w}}_{k,i-1} \\ &= -D_k\widetilde{\boldsymbol{w}}_{k,i-1}\end{aligned} \tag{4.57}$$

where we used the fact from (4.43) that $g_k(w^o) = 0$, as well as (4.45). Using (4.57), the error recursions (4.55) become

$$\widetilde{\boldsymbol{\psi}}_{k,i} = [I - \mu_k D_k]\,\widetilde{\boldsymbol{w}}_{k,i-1} + \mu_k \boldsymbol{v}_{k,i}^g(\boldsymbol{w}_{k,i-1}) \tag{4.58}$$

$$\widetilde{\boldsymbol{w}}_{k,i} = \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\widetilde{\boldsymbol{\psi}}_{\ell,i} \tag{4.59}$$

Using the following definitions:

$$\boldsymbol{v}_i^g(\boldsymbol{w}_{i-1}) \triangleq \operatorname{col}\left\{\boldsymbol{v}_{1,i}^g(\boldsymbol{w}_{1,i-1}), \ldots, \boldsymbol{v}_{N,i}^g(\boldsymbol{w}_{N,i-1})\right\} \tag{4.60}$$

$$\mathcal{D} \triangleq \operatorname{diag}\{D_1, \ldots, D_N\} \tag{4.61}$$

the recursions (4.58)–(4.59) lead to the network weight-error recursion

$$\widetilde{\boldsymbol{w}}_i = \mathcal{A}^T\left[I - \mathcal{M}\mathcal{D}\right]\widetilde{\boldsymbol{w}}_{i-1} + \mathcal{A}^T\mathcal{M}\boldsymbol{v}_i^g(\boldsymbol{w}_{i-1}). \tag{4.62}$$

Traditionally in the energy conservation framework [Say08,CS10,Say14b], one evaluates the weighted variance relation associated with (4.62) by equating the squared weighted Euclidean norms of both sides with respect to a symmetric nonnegative-definite weighting matrix Σ that we are free to choose, taking the expectation, and using (4.53) and (4.39) to write

$$\begin{aligned}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2 &= \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_{\Sigma'}^2 + \mathbb{E}\left\|\mathcal{A}^T\mathcal{M}\boldsymbol{v}_i^g(\boldsymbol{w}_{i-1})\right\|_\Sigma^2 \\ \Sigma' &= \left[I - \mathcal{M}\mathcal{D}\right]\mathcal{A}\Sigma\mathcal{A}^T\left[I - \mathcal{M}\mathcal{D}\right]\end{aligned} \tag{4.63}$$

where we used the fact that the matrices $\mathcal{M}$ and $\mathcal{D}$ are symmetric and block diagonal. Since, however, the update noise vector is characterized in terms of an upper bound on its variance according to (4.40), it is more convenient to work with a set of inequality recursions based on (4.58)–(4.59) to bound the mean-square performance of each node—see Theorem 1 further ahead.

To this end, first note that in view of (4.59) being a convex combination of $\left\{\widetilde{\boldsymbol{\psi}}_{\ell,i}\right\}$, by Jensen's inequality [BV04] it can be established for all k that

$$\|\widetilde{\boldsymbol{w}}_{k,i}\|^2 \leq \sum_{\ell\in\mathcal{N}_k} a_{\ell k} \left\|\widetilde{\boldsymbol{\psi}}_{\ell,i}\right\|^2. \tag{4.64}$$

Next, a variance relation for (4.58) is established by equating the squared Euclidean norms of both sides, conditioning on $\boldsymbol{\mathcal{F}}_{i-1}$, taking the expectation, and using (4.53) and (4.39):

$$\begin{aligned}\mathbb{E}\left\{\left\|\widetilde{\boldsymbol{\psi}}_{k,i}\right\|^2 \big| \boldsymbol{\mathcal{F}}_{i-1}\right\} &= \|\widetilde{\boldsymbol{w}}_{k,i-1}\|^2_{\Sigma_k} \\ &\quad + \mu_k^2\, \mathbb{E}\left\{\left\|\boldsymbol{v}^g_{k,i}(\boldsymbol{w}_{k,i-1})\right\|^2 \big| \boldsymbol{\mathcal{F}}_{i-1}\right\}\end{aligned} \tag{4.65}$$

where

$$\Sigma_k = (I - \mu_k D_k)^2. \tag{4.66}$$

Taking the expectation again, with respect to $\boldsymbol{\mathcal{F}}_{i-1}$, leads to

$$\mathbb{E}\left\|\widetilde{\boldsymbol{\psi}}_{k,i}\right\|^2 = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{k,i-1}\|^2_{\Sigma_k} + \mu_k^2 \mathbb{E}\left\|\boldsymbol{v}^g_{k,i}(\boldsymbol{w}_{k,i-1})\right\|^2. \tag{4.67}$$

In order to bound (4.67), we need to bound the matrix Σ_k, as well as the term $\mathbb{E}\left\|\boldsymbol{v}^g_{k,i}(\boldsymbol{w}_{k,i-1})\right\|^2$. While the latter is bounded by the property (4.40), the following lemma bounds the matrix Σ_k.

Lemma 2 (Bounds for Σ_k). The matrix Σ_k for all k is symmetric, nonnegative definite, and satisfies

$$0 \leq \Sigma_k \leq \kappa_k^2 I_M \tag{4.68}$$

where

$$\kappa_k \triangleq \max\left\{\left|1-\mu_k\bar{\lambda}_{U,k}\right|, \left|1-\mu_k\bar{\lambda}_{L,k}\right|\right\}. \tag{4.69}$$

Proof. See Appendix A.6. ■

Using (4.68) and (4.40), (4.67) can be replaced with

$$\mathbb{E}\left\|\widetilde{\boldsymbol{\psi}}_{k,i}\right\|^2 \leq \left(\kappa_k^2 + \mu_k^2\beta_k\right)\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{k,i-1}\|^2 + \mu_k^2\sigma^2_{g,k}. \tag{4.70}$$

Introducing the following network mean-square-error vectors:

$$\mathcal{Y}_i \triangleq \mathrm{col}\left\{\mathbb{E}\left\|\widetilde{\boldsymbol{\psi}}_{1,i}\right\|^2, \ldots, \mathbb{E}\left\|\widetilde{\boldsymbol{\psi}}_{N,i}\right\|^2\right\} \tag{4.71}$$

$$\mathcal{W}_i \triangleq \mathrm{col}\left\{\mathbb{E}\left\|\widetilde{\boldsymbol{w}}_{1,i}\right\|^2, \ldots, \mathbb{E}\left\|\widetilde{\boldsymbol{w}}_{N,i}\right\|^2\right\} \tag{4.72}$$

and the matrices

$$\Gamma \triangleq \mathrm{diag}\left\{\kappa_1^2 + \mu_1^2\beta_1, \ldots, \kappa_N^2 + \mu_N^2\beta_N\right\} \tag{4.73}$$

$$\Omega \triangleq \mathrm{diag}\left\{\mu_1, \ldots, \mu_N\right\} \tag{4.74}$$

$$\Sigma_g \triangleq \mathrm{diag}\left\{\sigma_{g,1}^2, \ldots, \sigma_{g,N}^2\right\} \tag{4.75}$$

then (4.64) and (4.70) imply that

$$\mathcal{Y}_i \preceq \Gamma\mathcal{W}_{i-1} + \Omega^2\Sigma_g\mathbb{1} \tag{4.76}$$

$$\mathcal{W}_i \preceq A^T\mathcal{Y}_i \tag{4.77}$$

where $\preceq$ denotes entry-wise comparison. Since the entries of the matrix A are non-negative, we can combine the inequalities (4.76)–(4.77), which leads to the following relation for sufficiently large i:

$$\mathcal{W}_i \preceq A^T\Gamma\mathcal{W}_{i-1} + A^T\Omega^2\Sigma_g\mathbb{1}. \tag{4.78}$$

In the following theorem, we use (4.78) to prove that under a certain condition on the step-sizes $\{\mu_k\}$, the mean-square error vector $\mathcal{W}_i$ is bounded as $i \to \infty$. This result will be used subsequently to evaluate the steady-state error expressions for sufficiently small step-sizes.

Theorem 1 (Mean-Square Stability). If the step-sizes $\{\mu_k\}$ satisfy the following condition:

$$0 < \mu_k < \min\left\{\frac{2\bar{\lambda}_{U,k}}{\bar{\lambda}_{U,k}^2 + \beta_k}, \frac{2\bar{\lambda}_{L,k}}{\bar{\lambda}_{L,k}^2 + \beta_k}\right\} \tag{4.79}$$

for all k, then it holds that

$$\limsup_{i\to\infty} \|\mathcal{W}_i\|_\infty \leq \frac{\max\limits_{1\leq k\leq N} \mu_k^2\sigma_{g,k}^2}{1 - \max\limits_{1\leq k\leq N}(\kappa_k^2 + \mu_k^2\beta_k)}. \tag{4.80}$$

Proof. The proof follows that of Theorem 1 in [CS12] closely. ■

Expression (4.80) bounds the mean-square error of the worst-performing node in the network. It can further be established that for sufficiently small step-sizes, each $\boldsymbol{w}_{k,i}$ for all k will get close to w^o at steady-state. This can be shown by assuming that the

step-sizes are small enough such that the nonnegative factor κ_k defined in (4.69) can be expressed as

$$\kappa_k = 1 - \mu_k \bar{\lambda}_{L,k}. \tag{4.81}$$

Substituting (4.81) into the upper bound in (4.80) and ignoring the $\mathcal{O}(\mu_k^2)$ terms in the denominator, it can be readily established that

$$\limsup_{i\to\infty} \|\mathcal{W}_i\|_\infty = \mathcal{O}(\mu_{\max}) \tag{4.82}$$

where $\mu_{\max} \triangleq \max_{1\leq k\leq N} \mu_k$.

Bearing this result in mind, the update noise vector in (4.62) can be approximated at steady-state in terms of its N blocks as

$$\begin{aligned} \boldsymbol{v}_{k,i}^g(\boldsymbol{w}_{k,i-1}) &\approx \boldsymbol{v}_{k,i}^g(w^o) \\ &= \hat{\boldsymbol{g}}_{k,i}(w^o) - g_k(w^o) \\ &= \hat{\boldsymbol{g}}_{k,i}(w^o) \\ &= -\boldsymbol{u}_{k,i}^T \boldsymbol{\alpha}_{k,i}^T \boldsymbol{\varphi}_{v,k,i} \end{aligned} \tag{4.83}$$

for each k, and its covariance matrix is given by

$$\begin{aligned} \mathcal{R}_g &\approx \left\{ \mathbb{E}\, \boldsymbol{v}_i^g \boldsymbol{v}_i^{g^T} \right\} \Big|_{\boldsymbol{w}_{i-1}=\mathbb{1}_N \otimes w^o} \\ &= \operatorname{diag}\left\{ s_1(\infty) R_{u,1}, \ldots, s_N(\infty) R_{u,N} \right\} \end{aligned} \tag{4.84}$$

under the model assumptions, (*D-A1*), and (*D-A4*), where

$$s_k(i) = \operatorname{Tr}(R_{\alpha_{k,i}} R_{\varphi_{v,k}}) \tag{4.85}$$

$$s_k(\infty) \triangleq \lim_{i\to\infty} s_k(i) = \operatorname{Tr}(R_{\alpha_{k,\infty}} R_{\varphi_{v,k}}) \tag{4.86}$$

with $R_{\varphi_{v,k}} \triangleq \mathbb{E}\, \boldsymbol{\varphi}_{v,k,i} \boldsymbol{\varphi}_{v,k,i}^T$, where the time subscript i has been dropped from the latter time-invariant moment.

Therefore, an approximate weighted variance relation for (4.63) that holds at steady-state is

$$\mathbb{E}\, \|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2 \approx \mathbb{E}\, \|\widetilde{\boldsymbol{w}}_{i-1}\|_{\Sigma'}^2 + \operatorname{Tr}(\Sigma \mathcal{A}^T \mathcal{M} \mathcal{R}_g \mathcal{M} \mathcal{A}) \tag{4.87}$$

$$\Sigma' = [I - \mathcal{M}\mathcal{D}]\, \mathcal{A}\Sigma\mathcal{A}^T\, [I - \mathcal{M}\mathcal{D}] \tag{4.88}$$

In the following, we denote by $\operatorname{bvec}(X)$ for an arbitrary square matrix X with block entries of size $M \times M$ each the vector obtained by vectorizing each block entry of the matrix and then stacking the resulting columns on top of each other. Let $\sigma \triangleq \operatorname{bvec}(\Sigma)$. We also state the following properties in terms of the block Kronecker product [KNW91]:

$$\begin{aligned} \operatorname{bvec}(X\Sigma Y) &= \left(Y^T \otimes_b X\right) \sigma \\ \operatorname{Tr}(\Sigma X) &= \left[\operatorname{bvec}(X^T)\right]^T \sigma \end{aligned} \tag{4.89}$$

Hence, vectorizing (4.87)–(4.88) leads to

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\sigma^2 \approx \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|_{\mathcal{F}\sigma}^2 + \left[\mathrm{bvec}(\mathcal{A}^T\mathcal{M}\mathcal{R}_g\mathcal{M}\mathcal{A})\right]^T\sigma \tag{4.90}$$

where the matrix $\mathcal{F}$ of size $M^2\times M^2$ is given by

$$\mathcal{F} = \left([I-\mathcal{M}\mathcal{D}]\,\mathcal{A}\right)\otimes_b\left([I-\mathcal{M}\mathcal{D}]\,\mathcal{A}\right), \tag{4.91}$$

and where we are using the notation $\|X\|_\Sigma^2$ and $\|X\|_\sigma^2$ interchangeably. The recursion (4.90) converges to a steady-state value if the matrix $\mathcal{F}$ is stable, which is ensured if the step-sizes are chosen according to

$$0<\mu_k<\frac{2}{\bar{\lambda}_{U,k}} \tag{4.92}$$

for each k—see [ASZSon,CS12,Say14b], which is guaranteed for sufficiently small step-sizes and also by condition (4.79).

Let

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|_\sigma^2 \triangleq \lim_{i\to\infty}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|_\sigma^2. \tag{4.93}$$

Taking the limit of (4.90) as $i\to\infty$,

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|_{(I-\mathcal{F})\sigma}^2 \approx \left[\mathrm{bvec}(\mathcal{A}^T\mathcal{M}\mathcal{R}_g\mathcal{M}\mathcal{A})\right]^T\sigma. \tag{4.94}$$

The node and network mean-square deviations (MSDs) and excess MSEs (EMSEs) can be computed by appropriate selection of the free parameter σ in (4.94). For example, the MSD of node k can be computed by weighting $\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|^2$ with a block matrix that has an identity matrix at the (k,k)th block and zeroes elsewhere; and the EMSE of node k can be computed by weighting $\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|^2$ with a block matrix that has $R_{u,k}$ at the (k,k)th block and zeroes elsewhere. The MSD and EMSE of node k are hence given by

$$\begin{aligned}\mathrm{MSD}_k &\triangleq \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{k,\infty}\|^2 &(4.95)\\ &\approx \left[\mathrm{bvec}(\mathcal{A}^T\mathcal{M}\mathcal{R}_g\mathcal{M}\mathcal{A})\right]^T(I-\mathcal{F})^{-1}q_k\\ \mathrm{EMSE}_k &\triangleq \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{k,\infty}\|_{R_{u,k}}^2 &(4.96)\\ &\approx \left[\mathrm{bvec}(\mathcal{A}^T\mathcal{M}\mathcal{R}_g\mathcal{M}\mathcal{A})\right]^T(I-\mathcal{F})^{-1}r_k\end{aligned}$$

with

$$q_k \triangleq \mathrm{bvec}(\mathrm{diag}\,\{e_k\}\otimes I_M) \tag{4.97}$$

$$r_k \triangleq \mathrm{bvec}(\mathrm{diag}\,\{e_k\}\otimes R_{u,k}) \tag{4.98}$$

being the vectorized versions of the aforementioned weighting block matrices, where e_k is the all-zero vector of length N and kth entry equal to 1. The network MSD and EMSE are defined as the average MSD and EMSE across all nodes, respectively:

$$\text{MSD} \triangleq \frac{1}{N}\sum_{k=1}^{N}\text{MSD}_k = \frac{1}{N}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|^2 \tag{4.99}$$

$$\begin{aligned}\text{EMSE} &\triangleq \frac{1}{N}\sum_{k=1}^{N}\text{EMSE}_k \\ &= \frac{1}{N}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_\infty\|^2_{\text{diag}\{R_{u,1},\ldots,R_{u,N}\}}\end{aligned} \tag{4.100}$$

Note that the invertibility of the matrix $(I-\mathcal{F})$ in (4.95) and (4.96) is guaranteed by the stability of $\mathcal{F}$.

Remark 10. As previously mentioned in Remark 9, the diffusion LMS algorithm is recovered when $B_k = 1$ and $\phi_{k,1}(x) = x$ for all k. It can be verified that in this case, $p_k(i) = 1$ and $s_k(i) = \sigma_{v,k}^2$ for all k and i. By substituting these values into the expression for $\mathcal{D}$ in (4.45) and (4.61) and that for $\mathcal{R}_g$ in (4.84), and then substituting into (4.95)–(4.96) and (4.99)–(4.100), the well-known steady-state mean-square performance expressions for the diffusion LMS algorithm are recovered [CS10, STC+13, Say14b, Say14a].

4.2.4 Comparison With the Diffusion LMS Algorithm

It is instructive to compare the optimal steady-state mean-square performance, minimized with respect to the combination policy $\{a_{\ell k}\}$, of the diffusion LMS and robust diffusion algorithms for *connected* networks, i.e., there is a path connecting any pair of nodes in the network, where the step-sizes $\{\mu_k\}$, regressor covariance matrices $\{R_{u,k}\}$, and noise pdfs $\{f_{\boldsymbol{v}_k}(v)\}$ are the same across the nodes, i.e., $\mu_k \equiv \mu$, $R_{u,k} \equiv R_u$, and $f_{\boldsymbol{v}_k}(v) \equiv f_{\boldsymbol{v}}(v)$ for all k. It follows that the moments $\{s_k(\infty)\}$ and $\{p_k(\infty)\}$ are the same across the nodes: $s_k(\infty) \equiv s(\infty)$ and $p_k(\infty) \equiv p(\infty)$ for all k. In this case, following [ZS12, CS15a, CS15b, Say14a], for sufficiently small step-sizes, the steady-state mean-square performance is equivalent across the nodes up to $\mathcal{O}(\mu)$, for both the diffusion LMS and robust diffusion algorithms, with the optimal MSD and EMSE for the diffusion LMS algorithm given by

$$\text{MSD}^{\text{opt,dLMS}} = \frac{\mu M\sigma_v^2}{2N} \tag{4.101}$$

$$\text{EMSE}^{\text{opt,dLMS}} = \frac{\mu\,\text{Tr}(R_u)\sigma_v^2}{2N} \tag{4.102}$$

and for the robust diffusion algorithm by

$$\text{MSD}^{\text{opt,d-rob}} = \frac{\mu M}{2N}\cdot\frac{s(\infty)}{p(\infty)} \tag{4.103}$$

$$\text{EMSE}^{\text{opt,d-rob}} = \frac{\mu\,\text{Tr}(R_u)}{2N}\cdot\frac{s(\infty)}{p(\infty)} \tag{4.104}$$

which shows the same N-fold performance improvement with respect to the stand-alone counterpart as the diffusion LMS algorithm [ASZSon]. These optimal MSDs and EMSEs are achieved, for example, by the Metropolis rule:

$$a_{\ell k}^{\text{opt}} = \begin{cases} \frac{1}{\max\{n_k, n_\ell\}}, & \ell \in \mathcal{N}_k \backslash \{k\}, \\ 1 - \sum\limits_{m \in \mathcal{N}_k \backslash \{k\}} a_{mk}^{\text{opt}}, & \ell = k. \end{cases} \tag{4.105}$$

The reader is referred to [ZS12, CS15a, CS15b, Say14a] for further details on the procedure that led to the optimal combination policy (4.105).

4.3 Simulation Results

We consider a network of $N = 20$ nodes, seeking to estimate a unit-norm signal vector w^o of size $M = 5$. The performance of the ATC diffusion LMS algorithm [CS10, STC+13, Say14b, Say14a], the ATC diffusion LMS algorithm with an adaptive combination policy [YS13, Say14b], and the ATC robust diffusion estimation algorithm developed in this work will be compared. The regressors $\{u_{k,i}\}$ and noise samples $\{v_k(i)\}$ are drawn independently of one another, independently across time and space, and identically distributed across time: the regressors from a multivariate zero-mean Gaussian distribution with covariances $\{R_{u,k}\}$, and the noise samples according to an ε-contaminated Gaussian mixture model with pdf

$$f_{\boldsymbol{v}_k}(v) = (1-\varepsilon)\,\mathcal{N}(0, \bar{\sigma}_{v,k}^2) + \varepsilon \mathcal{N}(0, \kappa \bar{\sigma}_{v,k}^2)$$

where $\left\{\bar{\sigma}_{v,k}^2\right\}$ are the nominal noise variances. Herein, κ is set to 100. The network topology, regressor covariance traces, and nominal noise variances are shown in Fig. 4.1. The weighting coefficients $a_{\ell k}$ are chosen according to the relative-degree rule, i.e., $a_{\ell k} = n_\ell / \Sigma_{m \in \mathcal{N}_k} n_m$. The step-sizes $\{\mu_k\}$ are set to be the same across the nodes. While the adaptation rate of the diffusion LMS algorithm with the adaptive combination policy is the same as that of the diffusion LMS algorithm with the static combination policy, the adaptation rate of the robust algorithm was adjusted to achieve the same steady-state network performance as the diffusion LMS algorithm with the static combination policy for the case of no contamination ($\varepsilon = 0$) for fair comparison: $\mu^{\text{dLMS}} = 0.02$ and $\mu^{\text{d-rob}} = 0.02$. For the robust algorithm, we consider two basis functions for every node k, i.e., $B_k \equiv B = 2$ for all k, where $\phi_{k,1}(x) = x$ and $\phi_{k,2}(x) = \tanh(x)$. The initial estimates of the basis weights, $\alpha_{k,-1}$, are set to $\frac{1}{B}\mathbb{1}$ for every node k. For the smoothing recursions, zero initial conditions are assumed, and ν_k is set to 0.9 for every node k. Finally, ϵ is set to 10^{-6}. All simulation results are obtained by averaging over 200 experiments.

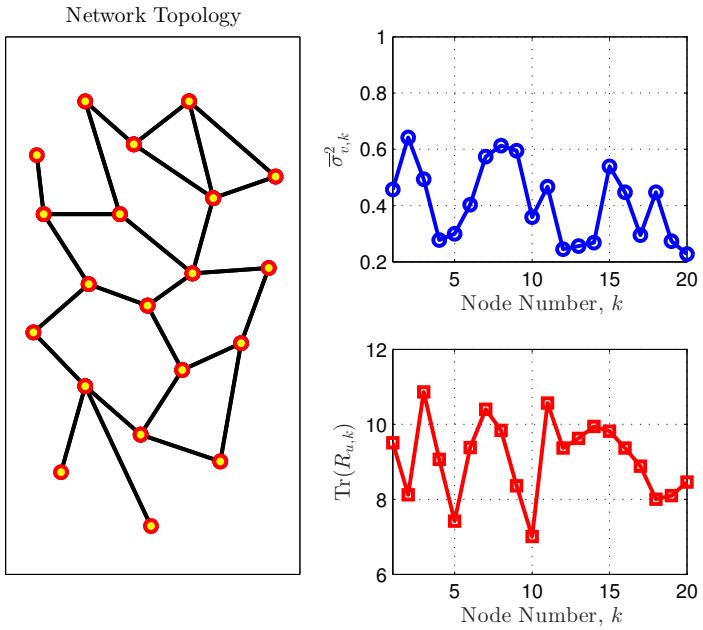


Figure 4.1: Network topology, node nominal noise variances $\bar{\sigma}^2_{v,k}$, and regressor covariance traces $\mathrm{Tr}(R_{u,k})$, for $N = 20$ nodes.

The transient network MSD and EMSE for the aforementioned algorithms in the presence of uncontaminated Gaussian noise ($\varepsilon = 0$) are plotted in Figs. 4.2a and 4.2b, respectively. The corresponding steady-state MSDs and EMSEs of each node, obtained by averaging the last 100 samples of the respective curves, are plotted in Figs. 4.2c and 4.2d, respectively. Also plotted throughout are the theoretical steady-state MSD and EMSE of the robust algorithm according to (4.95)–(4.96) and (4.99)–(4.100) for verification; the limiting values of the moments $\mathbb{E}\,\boldsymbol{\alpha}_{k,i}$ and $R_{\alpha_{k,i}}$ for every k are approximated using the respective sample average over the last 100 samples and across the experiments. The same curves in the presence of contaminated Gaussian noise, with $\varepsilon = 0.1$ and $\kappa = 100$, are plotted in Figs. 4.3a–4.3d. In the absence of contamination, the diffusion LMS and the robust diffusion algorithms perform just as well. The diffusion LMS algorithm with the adaptive combination policy outperforms both since network statistical knowledge is being estimated and incorporated into the algorithm on-the-fly. The robust diffusion algorithm outperforms both in the presence of contamination, however.

Depicted in Figs. 4.4a and 4.4b are the steady-state network MSD and EMSE of the algorithms, obtained by averaging the last 100 samples of the respective curves, when the measurements are corrupted with contaminated Gaussian noise with increasing contamination ratio ε and $\kappa = 100$. Also plotted are the theoretical steady-state network MSD and EMSE of the robust algorithm according to (4.99)–(4.100) for verification. It

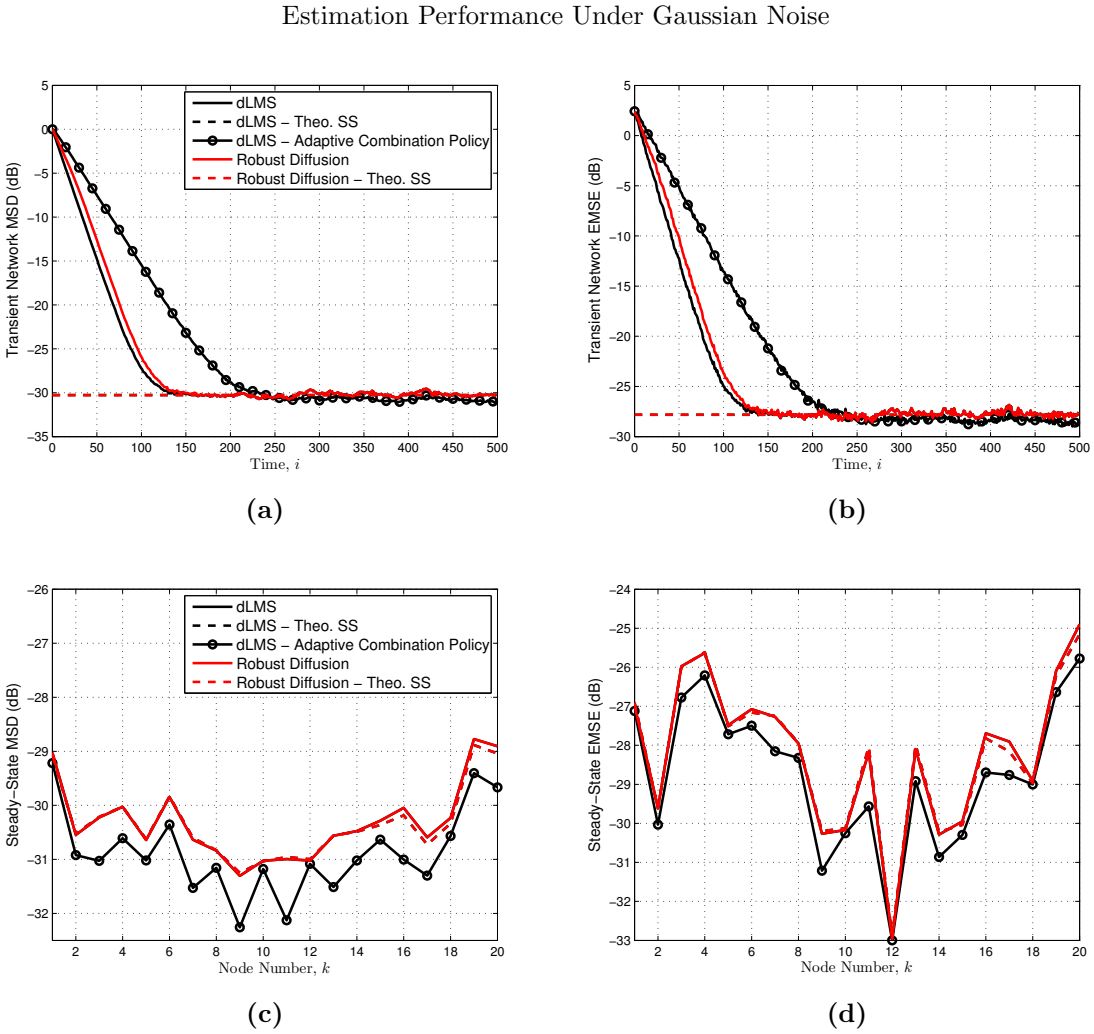


Figure 4.2: Figs. 4.2a and 4.2b: Simulated network MSD and EMSE learning curves (solid lines) and theoretical steady-state MSD and EMSE (dashed lines) for diffusion LMS (black), diffusion LMS with an adaptive combination policy (black, circled), and the robust algorithm (red) with $B = 2$ basis functions across the nodes under uncontaminated Gaussian noise ($\varepsilon = 0$). While the adaptation rate (equal across the nodes) of diffusion LMS with the adaptive combination policy is the same as that of diffusion LMS with the static combination policy, the adaptation rate of the robust algorithm was adjusted to achieve the same steady-state network performance as diffusion LMS with the static combination policy. Figs. 4.2c and 4.2d: Simulated and theoretical (solid and dashed lines, respectively) steady-state MSD and EMSE across the individual nodes for diffusion LMS (black), diffusion LMS with an adaptive combination policy (black, circled), and the robust algorithm (red).

is obvious that only the robust algorithm remains relatively insensitive to the increase in contamination.

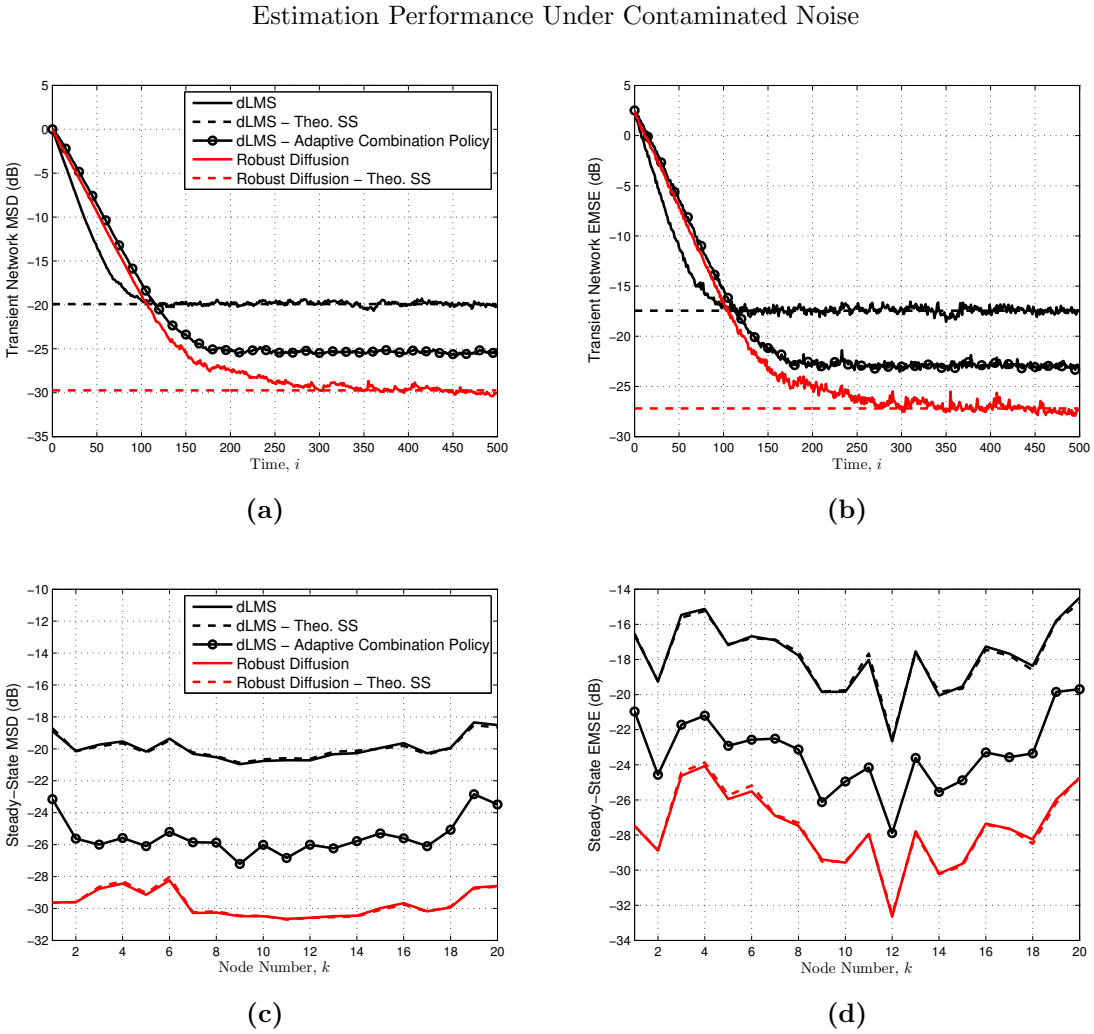
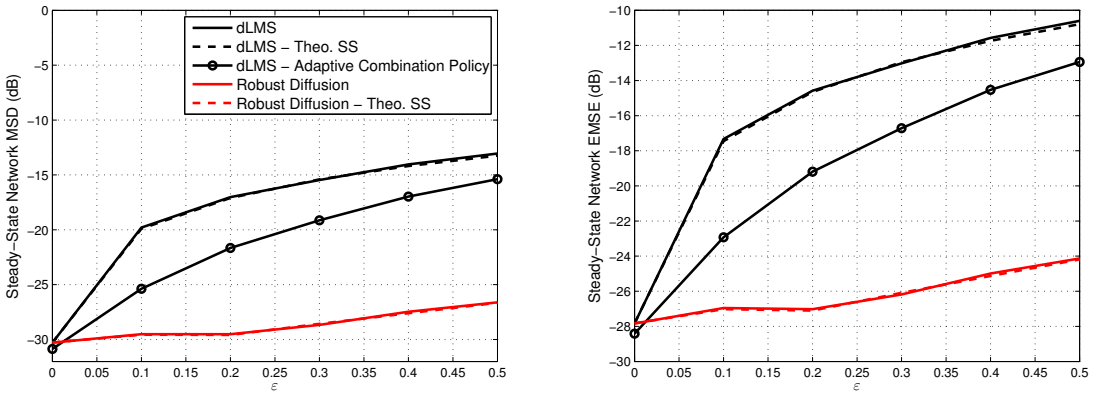


Figure 4.3: Figs. 4.3a and 4.3b: Simulated network MSD and EMSE learning curves (solid lines) and theoretical steady-state MSD and EMSE (dashed lines) for diffusion LMS (black), diffusion LMS with an adaptive combination policy (black, circled), and the robust algorithm (red) with $B = 2$ basis functions across the nodes under contaminated Gaussian noise, with $\varepsilon = 0.1$ and $\kappa = 100$. The adaptation rates are the same as in Figs. 4.3a–4.3b. Figs. 4.3c and 4.3d: Simulated and theoretical (solid and dashed lines, respectively) steady-state MSD and EMSE across the individual nodes for diffusion LMS (black), diffusion LMS with an adaptive combination policy (black, circled), and the robust algorithm (red).

4.4 Conclusion

A robust diffusion adaptation algorithm of the ATC variety was developed for distributed estimation over networks where the measurements may be corrupted by impulsive noise. Each node in the network runs a combination of its neighbors' estimates through a robust adaptive filter to ameliorate the effects of contamination, leading to better overall network performance matching that of a centralized strategy at steady-



(a) **(b)**

Figure 4.4: Simulated and theoretical steady-state network MSD (4.4a) and EMSE (4.4a) (solid and dashed lines, respectively) for diffusion LMS (black), diffusion LMS with an adaptive combination policy (black, circled), and the robust algorithm (red) with $B = 2$ basis functions across the nodes. The measurements are corrupted with contaminated Gaussian noise with increasing contamination ratio ε and $\kappa = 100$. The adaptation rate (equal across the nodes) for each algorithm is kept constant for all ε and is chosen as follows: While the adaptation rate of diffusion LMS with the adaptive combination policy is the same as that of diffusion LMS with the static combination policy, the adaptation rate of the robust algorithm was adjusted to achieve the same steady-state network performance as diffusion LMS with the static combination policy at $\varepsilon = 0$ (no contamination).

state. The robust adaptive update rule employs an adaptive error nonlinearity. The error nonlinearity was chosen to be a convex combination of preselected basis functions where the combination coefficients are adapted jointly with the weight vector such that the MSE relative to the local optimal error nonlinearity is minimized in each iteration. While knowledge of the nature of the noise, impulsive or otherwise, serves to guide the choice of basis functions, exact distributional knowledge is not required, which endows the algorithm with robustness and flexibility. The transient and steady-state behavior of the algorithm were analyzed in the mean and mean-square sense using the energy conservation framework subject to a set of reasonable assumptions given the nonlinear and stochastic nature of the algorithm, rendered even more complicated by the coupling of the estimation problems across the network. The performance of the algorithm was illustrated in simulation in an impulsive noise scenario, revealing the robustness of the proposed diffusion strategy, which outmatched even the LMS-based diffusion strategy employing a noise-adaptive combination policy.

Chapter 5
Robust Adaptive Detection Over Networks

In this chapter, the problem of distributed detection over adaptive networks in the presence of impulsive noise is considered. To this end, each node in a graph topology cooperates with its neighbors, diffusing information through the network, in order to detect events using local neighborhood measurements that are corrupted by impulsive noise.

In Ch. 4, a robust diffusion adaptation algorithm was developed for parameter estimation purposes in the presence of impulsive contamination across the network. In this chapter, this algorithm is extended to a detection context, where the algorithm serves as basis for the construction of robust adaptive test-statistics and detection thresholds. The performance of the resulting algorithm is examined and supporting simulations are provided.

In Sec. 5.1, the robust diffusion adaptation algorithm developed in Ch. 4 is extended to solve the problem of robust distributed detection over adaptive networks, and subsequently analyzed in Sec. 5.2 using the energy conservation analysis framework [Say03,Say14b,Say14a]. In Sec. 5.3, simulation results are presented. Conclusions are drawn in Sec. 5.4.[1]

5.1 Distributed Detection

5.1.1 Data Model and Problem Formulation

The same network model as in Ch. 4 is considered here, but the data model is slightly different. At each time index $i \geq 0$, each node k has access to a real-valued scalar measurement $d_k(i)$ arising from realizations of the random process $\boldsymbol{d}_k(i)$. These measurements relate to an unknown real-valued vector parameter w^o of size M according to

$$\boldsymbol{d}_k(i) = u_{k,i} w^o + \boldsymbol{v}_k(i) \tag{5.1}$$

where $u_{k,i}$ is now a known *deterministic* real-valued row regression vector of size M; and $\boldsymbol{v}_k(i)$ is a real-valued scalar wide-sense stationary zero-mean impulsive noise process

[1] An early short version of the work in this chapter was presented at ICASSP 2014 [ASZS14].

with variance $\sigma_{v,k}^2$. The random variables $\boldsymbol{v}_k(i)$ and $\boldsymbol{v}_\ell(j)$ are spatially and temporally independent for $k \neq \ell$ or $i \neq j$. It is still assumed that the noise probability density functions (pdfs), $f_{\boldsymbol{v}_k}(v)$, are symmetric for all nodes k, i.e., $\mathbb{E}\,\boldsymbol{v}_k^{2p-1}(i) = 0$, $p = 1, 2, \ldots$ Model (5.1) was used in [CS11]; however, in [CS11], the noise was restricted to being Gaussian distributed.

The objective is for every node in the network to establish the presence or absence of a known signal given noisy observations, which constitutes a simple binary hypothesis test:

$$w^o = \begin{cases} 0 & \text{under } \mathcal{H}_0 \\ w_s & \text{under } \mathcal{H}_1 \end{cases} \tag{5.2}$$

where w_s is known. The approach from [CS11] is followed.

The data from all nodes $1, \ldots, N$ at time index i are arranged into vectors and matrices as follows:

$$\boldsymbol{d}_i = \operatorname{col}\{\boldsymbol{d}_1(i), \ldots, \boldsymbol{d}_N(i)\} \tag{5.3}$$

$$U_i = \operatorname{col}\{u_{1,i}, \ldots, u_{N,i}\} \tag{5.4}$$

$$\boldsymbol{v}_i = \operatorname{col}\{\boldsymbol{v}_1(i), \ldots, \boldsymbol{v}_N(i)\} \tag{5.5}$$

$$R_v = \operatorname{diag}\left\{\sigma_{v,1}^2, \ldots, \sigma_{v,N}^2\right\} \tag{5.6}$$

Then, the data $\boldsymbol{d}_i$, U_i, and $\boldsymbol{v}_i$ is collected from all time indices $i, i-1 \ldots, 0$ in the same manner to obtain

$$\boldsymbol{d}_{0:i} = \operatorname{col}\{\boldsymbol{d}_i, \boldsymbol{d}_{i-1}, \ldots, \boldsymbol{d}_0\} \tag{5.7}$$

$$U_{0:i} = \operatorname{col}\{U_i, U_{i-1}, \ldots, U_0\} \tag{5.8}$$

$$\boldsymbol{v}_{0:i} = \operatorname{col}\{\boldsymbol{v}_i, \boldsymbol{v}_{i-1}, \ldots, \boldsymbol{v}_0\} \tag{5.9}$$

$$R_{v,0:i} = \operatorname{diag}\{R_v, \ldots, R_v\} \tag{5.10}$$

The data model (5.1) may therefore be expressed compactly as

$$\boldsymbol{d}_{0:i} = U_{0:i} w^o + \boldsymbol{v}_{0:i}. \tag{5.11}$$

5.1.2 Neyman–Pearson-Based Detection

Based on the Neyman–Pearson (NP) criterion [Kay98a], the detector that maximizes the detection probability $P_{d,i}$ given a target false-alarm probability P_f is the likelihood-ratio test leading to a comparison test of the form

$$T_i(\boldsymbol{d}_{0:i}) \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}} \gamma_i. \tag{5.12}$$

If the noise random vector $\boldsymbol{v}_{0:i}$ is Gaussian distributed, i.e., $\boldsymbol{v}_{0:i} \sim \mathcal{N}(0, R_{v,0:i})$, then the test-statistic $T_i(\boldsymbol{d}_{0:i})$ is given by [CS11, Kay98a]

$$T_i(\boldsymbol{d}_{0:i}) = w_s^T U_{0:i}^T R_{v,0:i}^{-1} \boldsymbol{d}_{0:i}. \tag{5.13}$$

The threshold γ_i is computed from the target false-alarm probability as $\gamma_i = \sigma_i \, \mathrm{Q}^{-1}(P_f)$, where $\mathrm{Q}(\cdot)$ is the right-tail Gaussian probability function

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} \, dt \tag{5.14}$$

and $\sigma_i^2 = w_s^T U_{0:i}^T R_{v,0:i}^{-1} U_{0:i} w_s$.

Assuming the matrix $U_{0:i}$, of size $(i+1)N \times M$, is full-rank with $M \leq N$, the minimum-variance unbiased (MVU) estimator of w^o given $\boldsymbol{d}_{0:i}$ in (5.11) is given by the Gauss–Markov theorem [Say03]:

$$\boldsymbol{w}_i^{\mathrm{mvu}} = \left(U_{0:i}^T R_{v,0:i}^{-1} U_{0:i}\right)^{-1} U_{0:i}^T R_{v,0:i}^{-1} \boldsymbol{d}_{0:i}. \tag{5.15}$$

Thus, the NP-optimal test-statistic in (5.13) under Gaussian noise can be rewritten in terms of $\boldsymbol{w}_i^{\mathrm{mvu}}$ in (5.15) as

$$T_i(\boldsymbol{w}_i^{\mathrm{mvu}}) = w_s^T U_{0:i}^T R_{v,0:i}^{-1} U_{0:i} \boldsymbol{w}_i^{\mathrm{mvu}}. \tag{5.16}$$

Note that $\boldsymbol{w}_i^{\mathrm{mvu}}$ in (5.15) is the solution to a weighted least-squares problem:

$$\boldsymbol{w}_i^{\mathrm{mvu}} = \arg\min_w \|\boldsymbol{d}_{0:i} - U_{0:i} w\|_{R_{v,0:i}^{-1}}^2 . \tag{5.17}$$

5.1.3 Robust Diffusion Detection Algorithm

The computation, at each node in the network, at time index i of the NP-optimal test-statistic $T_i(\boldsymbol{d}_{0:i})$ or $T_i(\boldsymbol{w}_i^{\mathrm{mvu}})$, using (5.13) or (5.16), respectively, and the MVU estimator $\boldsymbol{w}_i^{\mathrm{mvu}}$, using (5.15), requires that each node have access to the data $\{\boldsymbol{d}_k(j), u_{k,j}, \sigma_{v,k}^2\}$ from all nodes $1, \ldots, N$ and all time indices $j = 0, \ldots, i$. Since a node can only communicate with its neighbors, adaptive diffusion algorithms present themselves as a viable technique for the approximation of $\boldsymbol{w}_i^{\mathrm{mvu}}$ at each node in the network in a distributed fashion by means of local interactions and in-network processing, as explained in [CS10, CS11, STC+13, Say14b, Say14a]. However, the algorithms developed in [CS11] work well for distributed detection under the Gaussian assumption on the measurement noise. Here, a more robust adaptive diffusion algorithm is considered, based on the stand-alone counterpart in Ch. 3, and it is shown how to extend the distributed formulation of [CS11] to accommodate impulsive noise scenarios.

Table 5.1: Robust Diffusion Detection Algorithm

Initializations: w_s, A, ϵ, B_k, $\{\phi_{k,b}(x)\}$, Π_k, $\alpha_{k,-1} \in \Omega_{++,k}$, $\widehat{R}_{\varphi_{k,-1}}$, $\widehat{\varphi}'_{k,-1}$, ν_k, $\widehat{\lambda}_k(-1)$, μ_k, $\varphi'_{0,k}$, $\widehat{R}_{\widetilde{w}_{k,-1}} = 0$, $P_{f,k}$ for every node k. Start with $w_{k,-1} = 0$ for every node k. For every time index $i \geq 0$, repeat

1. Run one iteration of the **Robust Diffusion Estimation Algorithm** in Table 4.1.
2. **Decision:** for every node k, repeat
   (a) Test-statistic:
   If $i < M-1$, set $Q_{k,i}$ to I_M. Otherwise, if $i \geq M-1$,

$$\begin{cases} \mu_k(i) = \mu_k\left(\alpha_{k,i}^T \varphi'_{0,k}\right) \\ Q_{k,i} = \left(\sum_{j=0}^{i} \mu_k(j) u_{k,j}^T u_{k,j}\right)\left(\sum_{j=0}^{i} \mu_k^2(j) u_{k,j}^T u_{k,j}\right)^{-1} \end{cases} \tag{5.19}$$

$$T_{k,i} = w_s^T Q_{k,i} w_{k,i} \tag{5.20}$$

   (b) Threshold:

$$\widehat{p}_k(i) = \alpha_{k,i}^T \widehat{\varphi}'_{k,i} \tag{5.21a}$$

$$\widehat{s}_k(i) = \alpha_{k,i}^T \widehat{R}_{\varphi_{k,i}} \alpha_{k,i} \tag{5.21b}$$

$$\widehat{R}_{\widetilde{w}_{k,i}}^{A=I} = \left[I - \mu_k \widehat{p}_k(i) u_{k,i}^T u_{k,i}\right] \widehat{R}_{\widetilde{w}_{k,i-1}}^{A=I} \left[I - \mu_k \widehat{p}_k(i) u_{k,i}^T u_{k,i}\right] + \mu_k^2 \widehat{s}_k(i) u_{k,i}^T u_{k,i} \tag{5.21c}$$

$$\left(\widehat{\sigma}_{k,i}^{A=I}\right)^2 = w_s^T Q_{k,i} \widehat{R}_{\widetilde{w}_{k,i}}^{A=I} Q_{k,i} w_s \tag{5.21d}$$

$$\gamma_{k,i} = \frac{1}{\sqrt{g}} \widehat{\sigma}_{k,i}^{A=I}\, \mathrm{Q}^{-1}(P_{f,k,i}) \tag{5.21e}$$

   (c) Test:

$$T_{k,i} \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}} \gamma_{k,i} \tag{5.22}$$

Effectively, at each time index i, each node has an estimate $w_{k,i}$ for w^o, which is not necessarily the MVU estimate. Nodes can then compute local test-statistics $T_{k,i}(\boldsymbol{w}_{k,i})$ that will be defined further ahead. The resulting algorithm is listed in Table 5.1. All parameters are defined analogously to those in Table 4.1 while

$$\varphi'_{0,k} \triangleq \mathrm{col}\left\{\phi'_{k,1}(x=0), \ldots, \phi'_{k,B}(x=0)\right\}. \tag{5.18}$$

Note that the factors $\left\{\frac{1}{\sigma_{v,k}^2}\right\}$ that turn up in the local gradients that are computed based on (5.17) have been absorbed into the step-sizes $\{\mu_k\}$. In order to derive the appropriate test, we focus our attention on the adaptation step for the kth node:

$$\psi_{k,i} = w_{k,i-1} + \mu_k u_{k,i}^T \sum_{b=1}^{B_k} \alpha_{k,i}(b) \phi_{k,b}\big(e_k(i)\big). \tag{5.23}$$

Linearizing the sign-preserving, monotonically increasing error nonlinearities $\phi_{k,b}\big(\boldsymbol{e}_k(i)\big)$, $b = 1, \ldots, B_k$, by a Taylor series around $e_k(i) = 0$ gives $\phi_{k,b}\big(\boldsymbol{e}_k(i)\big) \approx$

$\phi'_{k,b}(x=0)\boldsymbol{e}_k(i)$, $b=1,\ldots,B_k$. The adaptation and combination steps in the algorithm can be combined as

$$\begin{aligned} w_{k,i} \approx & \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\left[I-\mu_\ell\left(\alpha_{\ell,i}^T\varphi'_{0,\ell}\right)u_{\ell,i}^T u_{\ell,i}\right]w_{k,i-1} \\ & + \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\mu_\ell\left(\alpha_{\ell,i}^T\varphi'_{0,\ell}\right)u_{\ell,i}^T d_\ell(i). \end{aligned} \tag{5.24}$$

Let $\mu_k(i)=\mu_k\left(\alpha_{k,i}^T\varphi'_{0,k}\right)$, $C_{k,i}=I-\mu_k(i)u_{k,i}^T u_{k,i}$, and $E_k=\operatorname{diag}\{e_k\}$, where e_k is the all-zero vector of length N and kth entry equal to 1. By induction based on (5.24), it can be verified that $w_{k,i}\approx K_{k,i}d_{0:i}$, where

$$K_{k,i}=\left[\sum_{\ell\in\mathcal{N}_k} a_{\ell k}\mu_\ell(i)U_i^T E_\ell \quad \sum_{\ell\in\mathcal{N}_k} a_{\ell k}C_{\ell,i}K_{\ell,i-1}\right]. \tag{5.25}$$

At this point, the assumption is made that the independent, non-identically distributed random entries of $\boldsymbol{d_{0:i}}$ with finite means and variances satisfy the Lindeberg condition such that the Lindeberg–Feller Central Limit Theorem holds asymptotically, as $i\to\infty$ [Lin22]. In other words, let

$$\Delta_i^2 \triangleq \sum_{j=0}^{i}\sum_{k=1}^{N}\mathbb{E}\big(\boldsymbol{d}_k(j)-\mathbb{E}\,\boldsymbol{d}_k(j)\big)^2 \tag{5.26}$$

and suppose that for every $\epsilon'>0$, the following condition holds:

$$\lim_{i\to\infty}\frac{1}{\Delta_i^2}\sum_{j=0}^{i}\sum_{k=1}^{N}\mathbb{E}\big[\big(\boldsymbol{d}_k(j)-\mathbb{E}\,\boldsymbol{d}_k(j)\big)^2\cdot\mathbf{1}_{\{|\boldsymbol{d}_k(j)-\mathbb{E}\,\boldsymbol{d}_k(j)|>\epsilon'\Delta_i\}}\big]=0 \tag{5.27}$$

where $\mathbf{1}_{\{\ldots\}}$ is the indicator function. Then the distribution of the random variables $\frac{1}{\Delta_i}\sum_{j=0}^{i}\sum_{k=1}^{N}\big(\boldsymbol{d}_k(j)-\mathbb{E}\,\boldsymbol{d}_k(j)\big)$ converges towards the standard normal distribution $\mathcal{N}(0,1)$. From this assumption, it follows that the estimators $\{\boldsymbol{w}_{k,i}\}$ are asymptotically approximately Gaussian distributed. In this case, if $K_{k,i}$ is full-rank with $M\le N$, and motivated by (5.11) and (5.13), a near-optimal NP detector at the kth node is given by

$$T_{k,i}(\boldsymbol{w}_{k,i})\underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}}\gamma_{k,i} \tag{5.28}$$

with the local test-statistic given by

$$T_{k,i}(\boldsymbol{w}_{k,i})=w_s^T Q_{k,i}^{\mathrm{opt}}\boldsymbol{w}_{k,i} \tag{5.29}$$

where

$$Q_{k,i}^{\mathrm{opt}}=(K_{k,i}U_{0:i})^T\left(K_{k,i}R_{v,0:i}K_{k,i}^T\right)^{-1}. \tag{5.30}$$

The threshold at the kth node, $\gamma_{k,i}$, is to be computed in a distributed manner as well in terms of the target false-alarm probability. This is addressed in Sec. 5.2.

In order to reduce the communication and computational burden at each node, we may approximate $Q_{k,i}^{\text{opt}}$ in (5.30). If the diffusion operation is overlooked by setting A to the identity matrix in (5.25), a reasonable substitute for $Q_{k,i}^{\text{opt}}$ under small step-sizes $\{\mu_k\}$ is

$$Q_{k,i} = \left(\sum_{j=0}^{i} \mu_k(j) u_{k,j}^T u_{k,j}\right) \left(\sum_{j=0}^{i} \mu_k^2(j) u_{k,j}^T u_{k,j}\right)^{-1} \tag{5.31}$$

for $i \geq M-1$, assuming invertibility, which is guaranteed if the basis functions are monotonically increasing and if the following matrix is full-rank:

$$U_{k,0:i} \triangleq \text{col}\{u_{k,0}, \ldots, u_{k,i}\}. \tag{5.32}$$

For $i < M-1$, $Q_{k,i}$ is set to I_M. The derivation can be found in Appendix A.7.

The two running sums in (5.31) can be computed recursively. Since the inverted expression in (5.31) constitutes a running sum of rank-one matrices, we may appeal to the Sherman–Morrison formula for matrix inversion to simplify the computation [HJ90].

5.2 Performance of Robust Diffusion Detection Algorithm

In this section, the detection performance of the robust diffusion detection algorithm is analyzed subject to the data model described in Sec. 5.1.1. Let $\widetilde{\boldsymbol{w}}_{k,i} \triangleq w^o - \boldsymbol{w}_{k,i}$ denote node k's weight-error vector at time index i. The following assumptions are made:

- (*D-A1*`) $\boldsymbol{\alpha}_{k,i}$ is independent of $\boldsymbol{v}_\ell(i)$ and $\widetilde{\boldsymbol{w}}_{\ell,i-1}$ for all k, ℓ, and i.
- (*D-A2*`) The step-sizes $\{\mu_k\}$ are sufficiently small.
- (*D-A3*`) The basis functions $\{\phi_{k,b}(x)\}$ are sign-preserving, odd-symmetric, monotonically increasing, and differentiable.

Note that the assumptions (*D-A1*`)–(*D-A3*`) are similar to (*D-A1*)–(*D-A3*) that were used in Sec. 4.2 in the performance analysis of the robust diffusion estimation algorithm, with the exception that the deterministic regressors do not appear here.

5.2.1 Error Recursions

We recall the diffusion recursions for each node k:

$$\begin{aligned}\boldsymbol{\psi}_{k,i} &= \boldsymbol{w}_{k,i-1} + \mu_k u_{k,i}^T \boldsymbol{h}_k(i) \\ \boldsymbol{w}_{k,i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\psi}_{\ell,i}\end{aligned} \tag{5.33}$$

which lead to the following network weight-error recursion:

$$\widetilde{\boldsymbol{w}}_i = \mathcal{A}^T \widetilde{\boldsymbol{w}}_{i-1} - \mathcal{A}^T \mathcal{M} \mathcal{U}_i^T \boldsymbol{h}_i \tag{5.34}$$

where

$$\mathcal{U}_i = \operatorname{diag}\{u_{1,i}, \ldots, u_{N,i}\} \tag{5.35}$$

and the rest of the quantities are defined analogously to Sec. 4.2 through (4.18)–(4.20), (4.23), and (4.24).

5.2.2 Mean Performance

Taking the expectation of both sides of (5.34),

$$\mathbb{E}\, \widetilde{\boldsymbol{w}}_i = \mathcal{A}^T \mathbb{E}\, \widetilde{\boldsymbol{w}}_{i-1} - \mathcal{A}^T \mathcal{M} \mathcal{U}_i^T \mathbb{E}\, \boldsymbol{h}_i. \tag{5.36}$$

From model (5.1), it holds that $\boldsymbol{e}_k(i) = u_{k,i} \widetilde{\boldsymbol{w}}_{k,i-1} + \boldsymbol{v}_k(i)$. Moreover, each $\boldsymbol{h}_k(i)$ can be approximated using a first-order Taylor series approximation of the basis functions $\{\phi_{k,b}(x)\}$ around $u_{k,i} \widetilde{\boldsymbol{w}}_{k,i-1} = 0$ for all $i \geq 0$ as follows:

$$\begin{aligned}\boldsymbol{h}_k(i) &= \sum_{b=1}^{B} \boldsymbol{\alpha}_{k,i}(b) \phi_{k,b}\big(\boldsymbol{e}_k(i)\big) \\ &\approx \sum_{b=1}^{B} \boldsymbol{\alpha}_{k,i}(b) \boldsymbol{\phi}_{v,k,b}(i) + u_{k,i} \widetilde{\boldsymbol{w}}_{k,i-1}(i) \sum_{b=1}^{B} \boldsymbol{\alpha}_{k,i}(b) \boldsymbol{\phi}'_{v,k,b}(i)\end{aligned} \tag{5.37}$$

where $\boldsymbol{\phi}_{v,k,b}(i)$ is given by (4.27). Taking the expectation,

$$\mathbb{E}\, \boldsymbol{h}_k(i) = p_k(i) u_{k,i} \mathbb{E}\, \widetilde{\boldsymbol{w}}_{k,i-1} \tag{5.38}$$

where, in addition to (*D-A1*`)–(*D-A3*`), we invoked the model assumptions on the noise samples $\boldsymbol{v}_k(i)$ being spatially and temporally independent with symmetric pdfs for all k; and $p_k(i)$ is given by (4.31). Introducing the following matrices:

$$P_i \triangleq \operatorname{diag}\{p_1(i), \ldots, p_N(i)\} \tag{5.39}$$

$$\mathcal{P}_i \triangleq P_i \otimes I_M \tag{5.40}$$

it follows that

$$\mathbb{E}\,\boldsymbol{h}_i = \mathcal{U}_i\mathcal{P}_i\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}. \tag{5.41}$$

The mean weight-error recursion (5.36) then becomes

$$\mathbb{E}\,\widetilde{\boldsymbol{w}}_i = \mathcal{A}^T\left[I - \mathcal{M}\mathcal{U}_i^T\mathcal{U}_i\mathcal{P}_i\right]\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}. \tag{5.42}$$

We again refer to [CS11, Lemma 2] for sufficient conditions for the asymptotic unbiasedness of the weight estimates $\{\boldsymbol{w}_{k,i}\}$. In particular, the weight estimates $\{\boldsymbol{w}_{k,i}\}$ are asymptotically unbiased for all nodes $k = 1,\ldots,N$, if there exists a time index i^*, a number $0 < \theta < 1$, and a submultiplicative norm $\|\cdot\|$ such that $\left\|\mathcal{A}^T\left[I - \mathcal{M}\mathcal{U}_i^T\mathcal{U}_i\mathcal{P}_i\right]\right\| \leq \theta < 1$ for all $i > i^*$. Similarly as in the prequel, special cases can be outlined where this condition is satisfied in terms of well-known norms.

5.2.3 Mean-Square Performance

The weight-error covariance matrix at time index i is defined as

$$R_{\widetilde{w}_i} \triangleq \mathbb{E}\left(\widetilde{\boldsymbol{w}}_i - \mathbb{E}\,\widetilde{\boldsymbol{w}}_i\right)\left(\widetilde{\boldsymbol{w}}_i - \mathbb{E}\,\widetilde{\boldsymbol{w}}_i\right)^T. \tag{5.43}$$

From (5.34), (5.42), and (5.41), it follows that

$$\begin{aligned} R_{\widetilde{w}_i} = {} & \mathcal{A}^T R_{\widetilde{w}_{i-1}}\mathcal{A} - \mathcal{A}^T\mathbb{E}\left(\widetilde{\boldsymbol{w}}_{i-1}\boldsymbol{h}_i^T\right)\mathcal{U}_i\mathcal{M}\mathcal{A} \\ & - \mathcal{A}^T\mathcal{M}\mathcal{U}_i^T\mathbb{E}\left(\boldsymbol{h}_i\widetilde{\boldsymbol{w}}_{i-1}^T\right)\mathcal{A} \\ & + \mathcal{A}^T\mathcal{M}\mathcal{U}_i^T\mathbb{E}\left(\boldsymbol{h}_i\boldsymbol{h}_i^T\right)\mathcal{U}_i\mathcal{M}\mathcal{A} \\ & + \mathcal{A}^T\left(\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}^T\right)\mathcal{P}_i\mathcal{U}_i^T\mathcal{U}_i\mathcal{M}\mathcal{A} \\ & + \mathcal{A}^T\mathcal{M}\mathcal{U}_i^T\mathcal{U}_i\mathcal{P}_i\left(\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}^T\right)\mathcal{A} \\ & - \mathcal{A}^T\mathcal{M}\mathcal{U}_i^T\mathcal{U}_i\mathcal{P}_i\left(\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}^T\right)\mathcal{P}_i\mathcal{U}_i^T\mathcal{U}_i\mathcal{M}\mathcal{A} \end{aligned} \tag{5.44}$$

with $R_{\widetilde{w}_{-1}} = 0$, if $w_{k,-1} = 0$ for all k. A couple of terms need to be evaluated:

1. Evaluating $\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}\boldsymbol{h}_i^T$:
   Invoking (*D-A1*$\backprime$)–(*D-A3*$\backprime$), in addition to the model assumptions, it holds for any pair k and $\ell \in \{1,\ldots,N\}$ that

$$\mathbb{E}\,\widetilde{\boldsymbol{w}}_{k,i-1}\boldsymbol{h}_\ell(i) = p_\ell(i)\mathbb{E}\left(\widetilde{\boldsymbol{w}}_{k,i-1}\widetilde{\boldsymbol{w}}_{\ell,i-1}^T\right)u_{\ell,i}^T \tag{5.45}$$

   so that

$$\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}\boldsymbol{h}_i^T = \left(R_{\widetilde{w}_{i-1}} + \mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}^T\right)\mathcal{U}_i^T P_i. \tag{5.46}$$

2. Evaluating $\mathbb{E}\,\boldsymbol{h}_i\boldsymbol{h}_i^T$:
   a) Diagonal entries – $\mathbb{E}\,\boldsymbol{h}_k^2(i)$:
   By squaring (5.37) and invoking (*D-A1*`)–(*D-A3*`), in addition to the model assumptions, it follows that

$$\mathbb{E}\,\boldsymbol{h}_k^2(i) = s_k(i) + t_{k,k}(i)u_{k,i}\mathbb{E}\left(\widetilde{\boldsymbol{w}}_{k,i-1}\widetilde{\boldsymbol{w}}_{k,i-1}^T\right)u_{k,i}^T \tag{5.47}$$

   where

$$s_k(i) = \mathrm{Tr}(R_{\alpha_{k,i}}R_{\varphi_{v,k}}) \tag{5.48}$$
$$t_{k,k}(i) = \mathrm{Tr}(R_{\alpha_{k,i}}R_{\varphi'_{v,k}}) \tag{5.49}$$

   with

$$R_{\alpha_{k,i}} \triangleq \mathbb{E}\,\boldsymbol{\alpha}_{k,i}\boldsymbol{\alpha}_{k,i}^T \tag{5.50}$$
$$R_{\varphi_{v,k}} \triangleq \mathbb{E}\,\boldsymbol{\varphi}_{v,k,i}\boldsymbol{\varphi}_{v,k,i}^T \tag{5.51}$$
$$R_{\varphi'_{v,k}} \triangleq \mathbb{E}\,\boldsymbol{\varphi}'_{v,k,i}\boldsymbol{\varphi}'^T_{v,k,i} \tag{5.52}$$

   where $\boldsymbol{\varphi}_{v,k,i}$ is given by (4.28); a primed vector denotes entry-wise differentiation; and the time subscript i has been dropped from the latter two time-invariant moments.
   b) Off-diagonal entries – $\mathbb{E}\,\boldsymbol{h}_k(i)\boldsymbol{h}_\ell(i)$, $k \neq \ell$:
   Invoking (*D-A1*`)–(*D-A3*`), in addition to the model assumptions, it follows that

$$\mathbb{E}\,\boldsymbol{h}_k(i)\boldsymbol{h}_\ell(i) = t_{k,\ell}(i)u_{k,i}\mathbb{E}\left(\widetilde{\boldsymbol{w}}_{k,i-1}\widetilde{\boldsymbol{w}}_{\ell,i-1}^T\right)u_{\ell,i}^T \tag{5.53}$$

   where

$$t_{k,\ell}(i) = \mathbb{E}\left(\boldsymbol{\alpha}_{k,i}^T\boldsymbol{\varphi}'_{v,k,i}\right)\mathbb{E}\left(\boldsymbol{\alpha}_{\ell,i}^T\boldsymbol{\varphi}'_{v,\ell,i}\right) \tag{5.54}$$

   since, under (*D-A1*`), $\boldsymbol{\alpha}_{k,i}$ and $\boldsymbol{\alpha}_{\ell,i}$ are independent for $k \neq \ell$. Define

$$S_i \triangleq \mathrm{diag}\,\{s_1(i), \ldots, s_N(i)\} \tag{5.55}$$

   and the matrix T_i whose (k,ℓ)th entry is $t_{k,\ell}(i)$. It follows that

$$\mathbb{E}\,\boldsymbol{h}_i\boldsymbol{h}_i^T = S_i + T_i \odot \left[\left(\mathcal{U}_i R_{\widetilde{w}_{i-1}}\mathcal{U}_i^T\right) + \left(\mathcal{U}_i\left(\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}\mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1}^T\right)\mathcal{U}_i^T\right)\right] \tag{5.56}$$

   where $\odot$ is the Hadamard (entry-wise) matrix product.

Substituting (5.46) and (5.56) into (5.44), the covariance recursion becomes

$$\begin{aligned} R_{\widetilde{w}_i} = {} & \mathcal{A}^T R_{\widetilde{w}_{i-1}}\mathcal{A} - \mathcal{A}^T R_{\widetilde{w}_{i-1}}\mathcal{U}_i^T P_i\mathcal{U}_i\mathcal{M}\mathcal{A} \\ & - \mathcal{A}^T\mathcal{M}\mathcal{U}_i^T P_i\mathcal{U}_i R_{\widetilde{w}_{i-1}}\mathcal{A} \end{aligned} \tag{5.57}$$

$$
\begin{aligned}
&+ \mathcal{A}^T \mathcal{M} \mathcal{U}_i^T S_i \mathcal{U}_i \mathcal{M} \mathcal{A} \\
&+ \mathcal{A}^T \mathcal{M} \mathcal{U}_i^T \left[T_i \odot \left(\mathcal{U}_i R_{\widetilde{w}_{i-1}} \mathcal{U}_i^T \right) \right] \mathcal{U}_i \mathcal{M} \mathcal{A} \\
&+ \mathcal{A}^T \mathcal{M} \mathcal{U}_i^T \left[T_i \odot \left(\mathcal{U}_i \left(\mathbb{E}\, \widetilde{\boldsymbol{w}}_{i-1} \mathbb{E}\, \widetilde{\boldsymbol{w}}_{i-1}^T \right) \mathcal{U}_i^T \right) \right] \mathcal{U}_i \mathcal{M} \mathcal{A} \\
&- \mathcal{A}^T \mathcal{M} \mathcal{U}_i^T P_i \mathcal{U}_i \left(\mathbb{E}\, \widetilde{\boldsymbol{w}}_{i-1} \mathbb{E}\, \widetilde{\boldsymbol{w}}_{i-1}^T \right) \mathcal{U}_i^T P_i \mathcal{U}_i \mathcal{M} \mathcal{A},
\end{aligned}
$$

with $R_{\widetilde{w}_{-1}} = 0$.

5.2.4 Detection Performance

In order to evaluate the asymptotic detection performance of the algorithm, as $i \to \infty$, the following additional assumption, similar to (*D-A4*) in Sec. 4.2.3, is made:

- (*D-A4'*) For sufficiently large i, the process $\{\boldsymbol{\alpha}_{k,i}\}$ is i.i.d. for all k, where the first- and second-order moments $\mathbb{E}\, \boldsymbol{\alpha}_{k,i}$ and $R_{\alpha_{k,i}}$ have reached finite constant values, as defined in (4.36).

The following lemma establishes the limiting value of $\boldsymbol{Q}_{k,i}$ for all k when the regressors are regarded as having been drawn from a distribution. That is, the following additional assumption is made:

- (*D-A5'*) The regressor sequence $\{u_{k,i}\}$ for each node k is a realization of an i.i.d. random process $\{\boldsymbol{u}_{k,i}\}$ with second-order moment denoted as $R_{u,k} = \mathbb{E}\, \boldsymbol{u}_{k,i}^T \boldsymbol{u}_{k,i}$ and assumed to be positive definite. Moreover, the vectors $\boldsymbol{u}_{k,i}$ and $\boldsymbol{\alpha}_{k,i}$ are assumed to be asymptotically independent for all k.

Lemma 3 (Limiting Value of $\boldsymbol{Q}_{k,i}$). Asymptotically, as $i \to \infty$, under (*D-A4'*) and (*D-A5'*), the random matrix $\boldsymbol{Q}_{k,i}$ can be approximated by a deterministic constant matrix:

$$
\lim_{i \to \infty} \boldsymbol{Q}_{k,i} = \eta_k I_M \tag{5.58}
$$

where

$$
\eta_k \triangleq \lim_{i \to \infty} \frac{\mathbb{E}\, \boldsymbol{\mu}_k(i)}{\mathbb{E}\, \boldsymbol{\mu}_k^2(i)} \tag{5.59}
$$

with $\boldsymbol{\mu}_k(i) = \mu_k \left(\boldsymbol{\alpha}_{k,i}^T \varphi'_{0,k} \right)$ and $\varphi'_{0,k}$ given by (5.18).

Proof. See Appendix A.8. ■

It is worth noting that the constants η_k, for all k, have no bearing on the resulting detection performance. For sufficiently large i, the test-statistics $T_{k,i}(\boldsymbol{w}_{k,i}) \approx \eta_k w_s^T \boldsymbol{w}_{k,i}$ are distributed as

$$T_{k,i}(\boldsymbol{w}_{k,i}) \sim \mathcal{N}(\eta_k w_s^T \mathbb{E}\, \boldsymbol{w}_{k,i}, \sigma_{k,i}^2) \tag{5.60}$$

where $\sigma_{k,i}^2 = \eta_k^2 w_s^T R_{\widetilde{w}_{k,i}} w_s$, with $R_{\widetilde{w}_{k,i}}$ denoting node k's weight-error covariance matrix at time index i:

$$R_{\widetilde{w}_{k,i}} \triangleq \mathbb{E}\,(\widetilde{\boldsymbol{w}}_{k,i} - \mathbb{E}\,\widetilde{\boldsymbol{w}}_{k,i})\,(\widetilde{\boldsymbol{w}}_{k,i} - \mathbb{E}\,\widetilde{\boldsymbol{w}}_{k,i})^T. \tag{5.61}$$

Hence, the detection, false-alarm, and miss probabilities at each node k and time index i are asymptotically given by

$$P_{d,k,i} = \mathrm{Q}\Big(\frac{\gamma_{k,i} - \eta_k w_s^T w_s + \eta_k w_s^T \mathbb{E}\,\widetilde{\boldsymbol{w}}_{k,i}}{\sigma_{k,i}}\Big) \tag{5.62}$$

$$P_{f,k,i} = \mathrm{Q}\Big(\frac{\gamma_{k,i} + \eta_k w_s^T \mathbb{E}\,\widetilde{\boldsymbol{w}}_{k,i}}{\sigma_{k,i}}\Big) \tag{5.63}$$

and $P_{m,k,i} = 1 - P_{d,k,i}$. Given target false-alarm probabilities at each node k and time index i, under the assumption of asymptotic unbiasedness of the weight estimates $\{\boldsymbol{w}_{k,i}\}$, the corresponding detection thresholds may subsequently be approximated, in a distributed manner, as

$$\gamma_{k,i} = \frac{1}{\sqrt{g}} \widehat{\sigma}_{k,i}^{A=I}\, \mathrm{Q}^{-1}(P_{f,k,i}) \tag{5.64}$$

where $\big(\widehat{\sigma}_{k,i}^{A=I}\big)^2 = w_s^T Q_{k,i} \widehat{R}_{\widetilde{w}_{k,i}}^{A=I} Q_{k,i} w_s$, with $\widehat{R}_{\widetilde{w}_{k,i}}^{A=I}$ given by the following recursion:

$$\begin{aligned} \widehat{R}_{\widetilde{w}_{k,i}}^{A=I} &= \big[I - \mu_k \widehat{p}_k(i) u_{k,i}^T u_{k,i}\big]\, \widehat{R}_{\widetilde{w}_{k,i-1}}^{A=I}\, \big[I - \mu_k \widehat{p}_k(i) u_{k,i}^T u_{k,i}\big] \\ &\quad + \mu_k^2 \widehat{s}_k(i) u_{k,i}^T u_{k,i}, \qquad \widehat{R}_{\widetilde{w}_{k,-1}}^{A=I} = 0. \end{aligned} \tag{5.65}$$

The estimated moments $\widehat{p}_k(i)$ and $\widehat{s}_k(i)$ are stochastic approximations of their true counterparts, reusing smoothed estimates from the algorithm:

$$\widehat{p}_k(i) = \alpha_{k,i}^T \widehat{\varphi}'_{k,i}, \qquad \widehat{s}_k(i) = \alpha_{k,i}^T \widehat{R}_{\varphi_{k,i}} \alpha_{k,i} \tag{5.66}$$

A sketch of the derivation can be found in Appendix A.9.

For comparison, the least-mean-squares (LMS)-based algorithm uses the following recursion [CS11]:

$$\begin{aligned} R_{\widetilde{w}_{k,i}}^{A=I} &= \big[I - \mu_k u_{k,i}^T u_{k,i}\big]\, R_{\widetilde{w}_{k,i-1}}^{A=I}\, \big[I - \mu_k u_{k,i}^T u_{k,i}\big] \\ &\quad + \mu_k^2 \sigma_{v,k}^2 u_{k,i}^T u_{k,i}, \qquad R_{\widetilde{w}_{k,-1}}^{A=I} = 0. \end{aligned} \tag{5.67}$$

The correction factor $g^{-\frac{1}{2}}$ accounts for the gain incurred by the diffusion process and can be estimated offline (cf. [CS11]).

5.3 Simulation Results

The network and simulation setups are the same as those in Ch. 4, except that the same set of regressors is maintained throughout the experiments, with the nodes seeking to detect a unit-norm signal vector w_s of size $M = 5$. The target false-alarm probabilities $P_{f,k,i}$ are set to 10^{-2} for every node k and time index i. The diffusion LMS-based detection algorithm and the robust diffusion detection algorithm are compared. All parameters are the same as in Sec. 4.3. The factor g was estimated offline and found to be 20. All simulation results are obtained by averaging over $10,000$ experiments.

In Fig. 5.1, the resulting best-case, average, and worst-case performance of both algorithms across the network is displayed for various degrees of contamination and $\kappa = 100$. While achieving better detection performance overall, the robustness of the algorithm developed in this work figures prominently with respect to the false-alarm performance.

5.4 Conclusion

The robust diffusion adaptation algorithm developed in Ch. 4 was extended in this chapter to solve the problem of distributed detection over adaptive networks where the measurements may be corrupted by impulsive noise. The weight estimates generated by the robust algorithm are used as basis for the design of robust local detectors, where the form of the test-statistics and the rule for the computation of the detection thresholds were motivated by the analysis of the algorithm dynamics. Each node in the network cooperates with its neighbors, utilizing their estimates, to update its local detector. Effectively, information pertaining to the event of interest percolates across the network, leading to enhanced detection performance. Exact knowledge of the noise distribution is not required since the robust diffusion detection algorithm is capable of learning it on-the-fly and adapting its parameters accordingly. The transient behavior of the algorithm was analyzed using the energy conservation framework subject to a set of reasonable assumptions given the nonlinear and stochastic nature of the algorithm. The detection performance was also established. The performance of the algorithm was illustrated in simulation in an impulsive noise scenario, revealing the robustness of the proposed diffusion detection algorithm, particularly in terms of false-alarm rate.

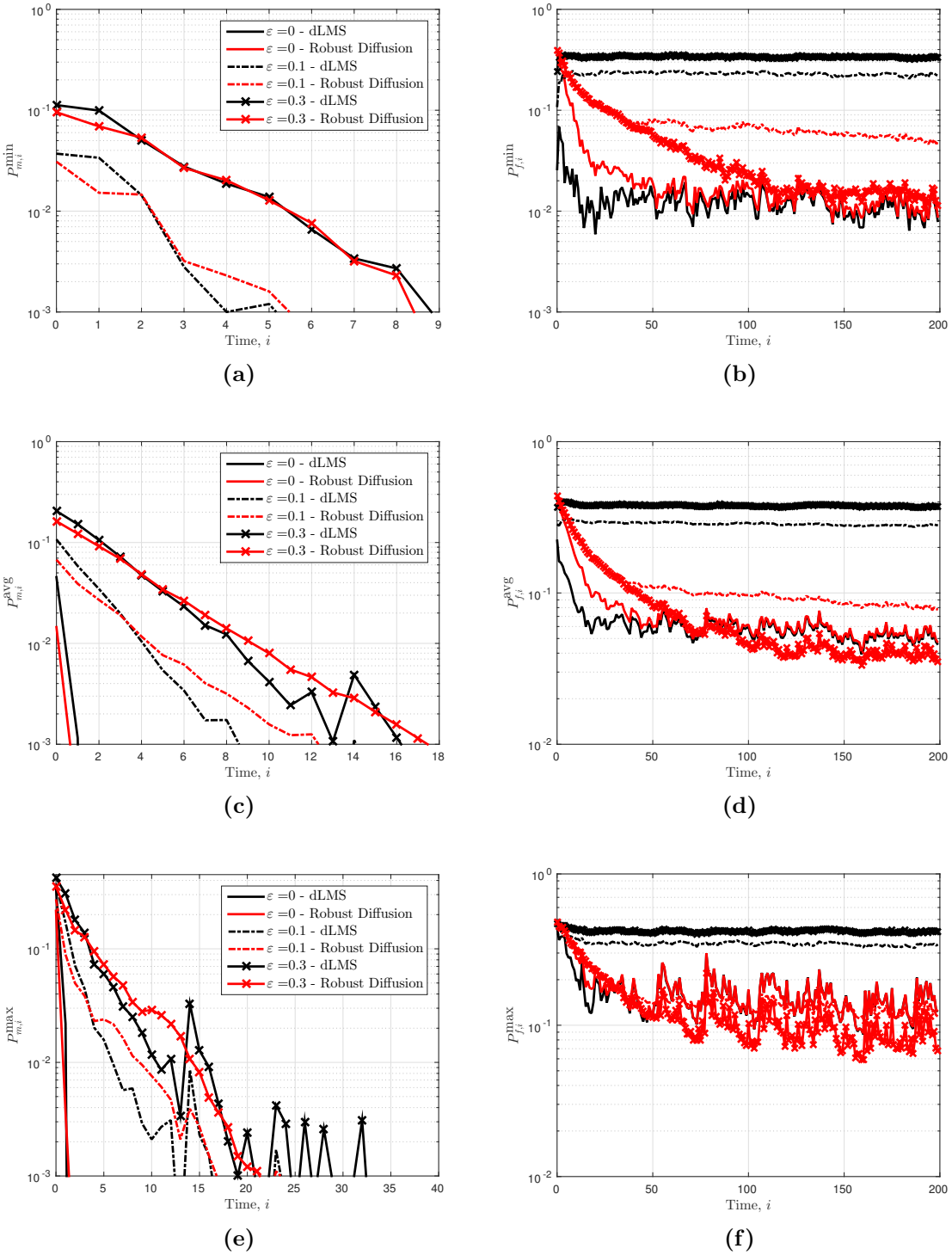


Figure 5.1: Best-case (Figs. 5.1a and 5.1b), average (Figs. 5.1c and 5.1d), and worst-case (Figs. 5.1e and 5.1f) detection and false-alarm performance across the network of diffusion LMS-based detection (black) and robust diffusion detection (red) with $B = 2$ basis functions across the nodes. The measurements are corrupted with contaminated Gaussian noise with increasing contamination ratio ε and $\kappa = 100$. The adaptation rate (equal across the nodes) for each algorithm is kept constant for all ε and is chosen such that the robust algorithm achieves the same steady-state network estimation performance as diffusion LMS at $\varepsilon = 0$ (no contamination).

Chapter 6

Summary, Conclusions, and Future Research Directions

6.1 Summary and Conclusions

First, a robust adaptive filtering algorithm of the least-mean-squares (LMS) type was developed that employs an adaptive error nonlinearity. The error nonlinearity was chosen to be a convex combination of preselected basis functions where the combination coefficients are adapted jointly with the weight vector such that the mean-square-error (MSE) relative to the optimal error nonlinearity is minimized in each iteration. While knowledge of the nature of the noise, impulsive or otherwise, serves to guide the choice of basis functions, exact distributional knowledge is not required since the robust algorithm is capable of learning it on-the-fly and adapting its parameters accordingly.

Then, a robust diffusion adaptation algorithm of the adapt-then-combine (ATC) variety was developed as a natural extension of its stand-alone counterpart for distributed estimation over networks where the measurements may be corrupted by impulsive noise. Each node in the network runs a combination of its neighbors' estimates through one iteration of a local robust adaptive filter update to ameliorate the effects of contamination, leading to better overall network performance matching that of a centralized strategy at steady-state. The robust adaptive update rule again employs an adaptive error nonlinearity that is a convex combination of preselected basis functions where the combination coefficients are adapted jointly with the weight vector such that the MSE relative to the local optimal error nonlinearity is minimized in each iteration.

Finally, the robust diffusion adaptation algorithm developed was extended further to solve the problem of distributed detection over adaptive networks where the measurements may be corrupted by impulsive noise. The weight estimates generated by the robust algorithm are used as basis for the design of robust local detectors, where the form of the test-statistics and the rule for the computation of the detection thresholds were motivated by the analysis of the algorithm dynamics. Each node in the network cooperates with its neighbors, utilizing their estimates, to update its local detector. Effectively, information pertaining to the event of interest percolates across the network, leading to enhanced detection performance.

The computational complexity of the robust algorithm was summarized, revealing that it introduces $\mathcal{O}(2M + B^3)$ additional complexity per iteration compared to the LMS algorithm. The transient and steady-state behavior of the robust algorithm in both its stand-alone and distributed varieties were analyzed in the mean and mean-square sense using the energy conservation framework subject to a set of reasonable assumptions given the nonlinear and stochastic nature of the algorithm, rendered even more complicated by the coupling of the estimation problems across the network in multi-agent scenarios. The performance of the algorithm was also examined in the context of distributed detection. Comprehensive simulations in an impulsive noise scenario served to illustrate the performance of the algorithm for single- and multi-agent adaptation, revealing the robustness of the proposed strategies. Good agreement between theory and practice was obtained.

6.2 Future Research Directions

Transient analysis without appeal to linearization. In the analysis of the robust algorithm in its stand-alone and distributed varieties in Chs. 3–5, we appealed to a truncated Taylor series approximation for the basis functions constituting the error nonlinearity. This so-called linearization approach leads to results that are in agreement with practice towards steady-state and for sufficiently small step-sizes. Ideally, one would like to be able to analyze the transient behavior of the nested algorithm more accurately so as to draw valuable insight into its dynamics. Such insight would guide the selection of the algorithm parameters for performance enhancement, or even reveal ways in which the implementation of the algorithm can be simplified. One work where accurate modeling of coupling in an algorithm was possible is [TYS10], using the energy conservation framework. In the stochastic approximation literature, algorithms with state-dependent noise are typically analyzed using averaging analysis or the ordinary-differential-equation (ODE) method [KY03], where their stability and convergence are established in an almost-sure or weak sense, which does not necessarily imply mean-square stability. Perhaps these methods can be combined with energy-based arguments towards more accurate analysis of nested algorithms.

Relaxation of error nonlinearity restrictions. In the analysis of the robust algorithm in its stand-alone and distributed varieties in Chs. 3–5, the basis functions were assumed to be sign-preserving, odd-symmetric—in view of the symmetry assumption on the noise probability density function (pdf), monotonically increasing, and twice

differentiable. These restrictions are naturally inherited by the error nonlinearity constructed from the basis functions. In this respect, the analysis provided in this dissertation does not encompass some error nonlinearities that have proved their merit in mitigating impulsive disturbances, such as those of redescending nature. Additionally, some of the minimax error nonlinearities yielded by distribution-free approaches to robustness over broader noise distribution classes (see [SV02, PT79]) admittedly do not lend themselves to analysis under the restrictions imposed here.

Relaxation of local observability condition to global observability condition. In the course of the analysis of the robust diffusion adaptation algorithm, a local observability condition was imposed on every node in the network. As a matter of fact, this condition can be relaxed to one of global observability, so that nodes with partial information can still recover the parameter of interest, permitting a measure of flexibility and diversity in the network. The analysis of the algorithm after a sufficiently large number of iterations can henceforth be conducted using the relaxed assumption by appealing to a Jordan canonical decomposition of the combination policy matrix and subsequent transformation of the network weight-error recursion, in a manner similar to [Say14a, CS15a, CS15b]. Such a transformation would flesh out the intricate relationship between network topology and steady-state performance.

Locally optimum detection. Since the form of the optimal adaptive filtering error nonlinearity relates quite closely to that of the test-statistic for locally optimum detection [Kay98a], it would be interesting to appropriate the robust algorithm developed in this dissertation for the context of locally optimum detection in non-Gaussian noise and examine its performance [Kas88].

Network-MSE–optimal agent-specific error nonlinearities. The robust diffusion adaptation algorithm developed in this dissertation has each agent employ its local-MSE–optimal error nonlinearity in the hope that the network performance would be enhanced as a result. A more powerful design should stem from optimizing the network MSE with respect to each of the agents' error nonlinearities. It is as yet unclear if such an approach could spur a distributed, cooperative, adaptive strategy for nonlinearity synthesis on-the-fly.

Appendix

A.1 Price's Theorem

Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be scalar real-valued zero-mean jointly Gaussian random variables with their correlation denoted by $\rho = \mathbb{E}\,\boldsymbol{xy}$. Then, according to Price's Theorem [Pri58], for any function $f(\boldsymbol{x}, \boldsymbol{y})$, the following equality holds:

$$\frac{\partial^n \mathbb{E}\, f(\boldsymbol{x}, \boldsymbol{y})}{\partial \rho^n} = \mathbb{E}\left(\frac{\partial^{2n} f(\boldsymbol{x}, \boldsymbol{y})}{\partial \boldsymbol{x}^n \partial\, \boldsymbol{y}^n}\right) \tag{1}$$

in terms of the n-th and $2n$-th order partial derivatives, assuming the derivatives and integrals in question exist. Two results, stated in [Say03, P. 333] and summarized here, are of interest:

Result 1: Assume that $f(\boldsymbol{x}, \boldsymbol{y})$ has the form $f(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x} g(\boldsymbol{y})$ and choose $n = 1$. Then, it holds that $\frac{\partial \mathbb{E}\, \boldsymbol{x} g(\boldsymbol{y})}{\partial \rho} = \mathbb{E}\, \frac{dg}{d\boldsymbol{y}}$. Integrating both sides over ρ leads to

$$\mathbb{E}\, \boldsymbol{x} g(\boldsymbol{y}) = \mathbb{E}\, \boldsymbol{xy} \cdot \mathbb{E}\, \frac{dg}{d\boldsymbol{y}}. \tag{2}$$

□

Result 2: Since from Result 1 it holds that $\mathbb{E}\, \boldsymbol{y} g(\boldsymbol{y}) = \sigma_y^2 \cdot \mathbb{E}\, \frac{dg}{d\boldsymbol{y}}$, where $\sigma_y^2 = \mathbb{E}\, \boldsymbol{y}^2$, it can further be established that

$$\mathbb{E}\, \boldsymbol{x} g(\boldsymbol{y}) = \frac{\mathbb{E}\, \boldsymbol{xy}}{\sigma_y^2} \cdot \mathbb{E}\, \boldsymbol{y} g(\boldsymbol{y}). \tag{3}$$

□

Result 3: Assume further that $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent of a third scalar real-valued zero-mean random variable $\boldsymbol{z}$. Then, as a consequence of Result 2, it holds that

$$\mathbb{E}\, \boldsymbol{x} f(\boldsymbol{y} + \boldsymbol{z}) = \frac{\mathbb{E}\, \boldsymbol{xy}}{\mathbb{E}\, \boldsymbol{y}^2} \cdot \mathbb{E}\, \boldsymbol{y} f(\boldsymbol{y} + \boldsymbol{z}). \tag{4}$$

□

A.2 Lower Bound on MSD in (2.62)

First, note that under the assumptions outlined in Chapter 2, it cannot be established that the adaptive filtering algorithm (2.37) leads to an asymptotically unbiased estimate of the optimal weight vector w^o, i.e., $\lim_{i\to\infty} \mathbb{E}\,\boldsymbol{w}_i = w^o$, or equivalently, $\lim_{i\to\infty} \mathbb{E}\,\widetilde{\boldsymbol{w}}_i = 0$. More assumptions are called for to establish asymptotic unbiasedness, as will be seen in Chapter 3. There, conditions for mean stability and asymptotic unbiasedness are derived. For now, suffice it to say that in [ANS01], a condition for mean-square stability of the adaptive filtering algorithm (2.37) was derived. As is well-known, mean-square stability implies mean stability [PP02].

Hence, assume that for all $i \geq 0$, $\boldsymbol{T}_i(w^o) \triangleq \boldsymbol{w}_i - w^o + w^o$ is a biased estimator for the parameter w^o. Let the bias be given by $b_i(w^o) = \mathbb{E}\,\boldsymbol{w}_i - w^o$. The estimator $\boldsymbol{T}_i(w^o)$ can then be regarded as an unbiased estimator of a function of the parameter w^o, say $g_i(w^o)$, such that

$$g_i(w^o) = \mathbb{E}\,\boldsymbol{T}_i(w^o) = b_i(w^o) + w^o. \tag{5}$$

Then, recalling that $\widetilde{\boldsymbol{w}}_i = w^o - \boldsymbol{w}_i$, and noting that its covariance is given by

$$\begin{aligned} R_{\widetilde{w}_i} &\triangleq \mathbb{E}\,(\widetilde{\boldsymbol{w}}_i - \mathbb{E}\,\widetilde{\boldsymbol{w}}_i)\,(\widetilde{\boldsymbol{w}}_i - \mathbb{E}\,\widetilde{\boldsymbol{w}}_i)^T \\ &= \mathbb{E}\,(\boldsymbol{w}_i - \mathbb{E}\,\boldsymbol{w}_i)\,(\boldsymbol{w}_i - \mathbb{E}\,\boldsymbol{w}_i)^T \\ &\triangleq R_{w_i} \\ &= \mathbb{E}\big(\boldsymbol{T}_i(w^o) - \mathbb{E}\,\boldsymbol{T}_i(w^o)\big)\big(\boldsymbol{T}_i(w^o) - \mathbb{E}\,\boldsymbol{T}_i(w^o)\big)^T \end{aligned} \tag{6}$$

which is the covariance of the estimator $\boldsymbol{T}_i(w^o)$, it follows that a lower bound for $R_{\widetilde{w}_i}$ is the Cramér–Rao lower bound [Kay98b]:

$$\begin{aligned} R_{\widetilde{w}_i} &\geq [\nabla_{w^o} g_i(w^o)]\, I_{F,i}^{-1}(w^o)\, [\nabla_{w^o} g_i(w^o)]^T \\ &= [I + \nabla_{w^o} b_i(w^o)]\, I_{F,i}^{-1}(w^o)\, [I + \nabla_{w^o} b_i(w^o)]^T \end{aligned} \tag{7}$$

where $I_{F,i}(w^o)$ is the Fisher information matrix, defined in what follows. Let $f\big(d(0),\ldots,d(i);w^o\big)$ denote the likelihood function of the observations $d(0),\ldots,d(i)$, which satisfy the data model (2.1). The likelihood function is parametrized by w^o. Assume that the following regularity condition holds for all w^o:

$$\mathbb{E}\,\nabla_{w^o} \ln f\big(\boldsymbol{d}(0),\ldots,\boldsymbol{d}(i);w^o\big) = 0. \tag{8}$$

Then, the Fisher information matrix is given by [Kay98b]

$$I_{F,i}(w^o) = -\mathbb{E}\,\nabla_{w^o}^2 \ln f\big(\boldsymbol{d}(0),\ldots,\boldsymbol{d}(i);w^o\big). \tag{9}$$

Since the matrices on either side of the inequality in (7) are nonnegative definite, their respective diagonal entries are nonnegative, from which follows that

$$\begin{aligned}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i - \mathbb{E}\,\widetilde{\boldsymbol{w}}_i\|^2 &= \mathrm{Tr}(R_{\widetilde{w}_i}) \\ &\geq \mathrm{Tr}\left([I + \nabla_{w^o} b_i(w^o)]\, I_{F,i}^{-1}(w^o)\, [I + \nabla_{w^o} b_i(w^o)]^T\right). \end{aligned} \tag{10}$$

Adding the squared bias term

$$\|b_i(w^o)\|^2 = \|\mathbb{E}\,\widetilde{\boldsymbol{w}}_i\|^2 \tag{11}$$

to either side and using bias–variance decomposition [Kay98b], it follows that, for all $i \geq 0$,

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2 \geq \mathrm{Tr}\left([I + \nabla_{w^o} b_i(w^o)]\, I_{F,i}^{-1}(w^o)\, [I + \nabla_{w^o} b_i(w^o)]^T\right) + \|\mathbb{E}\,\widetilde{\boldsymbol{w}}_i\|^2. \tag{12}$$

Since at steady-state, as $i \to \infty$,

$$\mathbb{E}\,\widetilde{\boldsymbol{w}}_i = \mathbb{E}\,\widetilde{\boldsymbol{w}}_{i-1} = r \tag{13}$$

$$\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2 = \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}\|^2 = \mathrm{MSD} \tag{14}$$

where r is some constant vector, then this implies that there exists a time index i^* sufficiently large such that for all $i \geq i^*$, the following inequality holds:

$$\begin{aligned}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i\|^2 &\geq \mathrm{Tr}\left([I + \nabla_{w^o} b_{i^*}(w^o)]\, I_{F,i^*}^{-1}(w^o)\, [I + \nabla_{w^o} b_{i^*}(w^o)]^T\right) + \|\mathbb{E}\,\widetilde{\boldsymbol{w}}_i\|^2 \\ &\geq \mathrm{Tr}\left([I + \nabla_{w^o} b_{i^*}(w^o)]\, I_{F,i^*}^{-1}(w^o)\, [I + \nabla_{w^o} b_{i^*}(w^o)]^T\right) \end{aligned} \tag{15}$$

In other words, the steady-state MSD is lower-bounded as

$$\mathrm{MSD} \geq \lambda \tag{16}$$

where

$$\lambda \triangleq \mathrm{Tr}\left([I + \nabla_{w^o} b_{i^*}(w^o)]\, I_{F,i^*}^{-1}(w^o)\, [I + \nabla_{w^o} b_{i^*}(w^o)]^T\right). \tag{17}$$

A.3 Derivation of Optimal Error Nonlinearity

It was established in Sec. 2.2.4 that the following relation holds:

$$\frac{\mathbb{E}\,h^2\big(\boldsymbol{e}^\star\big)}{\mathbb{E}\,h'\big(\boldsymbol{e}^\star\big)} \geq \alpha \tag{18}$$

where $\alpha > 0$.

It was claimed in [ANS01] that if the error nonlinearity $h(\cdot)$ is chosen as

$$\hat{h}(\cdot) \triangleq -\alpha \frac{f'_{\boldsymbol{e}^\star}(\cdot)}{f_{\boldsymbol{e}^\star}(\cdot)} \tag{19}$$

then the resulting ratio in (18) will achieve the lower bound α. The proof from [ANS01] is reproduced here.

Proof. The ratio $\mathbb{E}\, h^2(\boldsymbol{e}^\star)/\mathbb{E}\, h'(\boldsymbol{e}^\star)$ will be evaluated for the choice of error nonlinearity (19). Using integration by parts, the moment in the denominator can be expressed as

$$\begin{aligned}\mathbb{E}\, h'(\boldsymbol{e}^\star) &= \int_{-\infty}^{\infty} h'(e) f_{\boldsymbol{e}^\star}(e)\, de \\ &= h(e) f_{\boldsymbol{e}^\star}(e)\big|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} h(e) f'_{\boldsymbol{e}^\star}(e)\, de \end{aligned} \tag{20}$$

For the choice (19), this gives

$$\mathbb{E}\, \hat{h}'(\boldsymbol{e}^\star) = -\alpha f'_{\boldsymbol{e}^\star}(e)\big|_{-\infty}^{\infty} + \alpha \int_{-\infty}^{\infty} \frac{\big(f'_{\boldsymbol{e}^\star}(e)\big)^2}{f_{\boldsymbol{e}^\star}(e)}\, de \tag{21}$$

which under the assumption

$$\lim_{e^\star \to \pm\infty} f'_{\boldsymbol{e}^\star}(e^\star) = 0 \tag{22}$$

evaluates to

$$\mathbb{E}\, \hat{h}'(\boldsymbol{e}^\star) = \alpha \int_{-\infty}^{\infty} \frac{\big(f'_{\boldsymbol{e}^\star}(e)\big)^2}{f_{\boldsymbol{e}^\star}(e)}\, de. \tag{23}$$

On the other hand, for the same choice (19), the moment in the numerator evaluates to

$$\begin{aligned}\mathbb{E}\, \hat{h}^2(\boldsymbol{e}^\star) &= \alpha^2 \int_{-\infty}^{\infty} \left(\frac{f'_{\boldsymbol{e}^\star}(e)}{f_{\boldsymbol{e}^\star}(e)}\right)^2 f_{\boldsymbol{e}^\star}(e)\, de \\ &= \alpha^2 \int_{-\infty}^{\infty} \frac{\big(f'_{\boldsymbol{e}^\star}(e)\big)^2}{f_{\boldsymbol{e}^\star}(e)}\, de \end{aligned} \tag{24}$$

Hence,

$$\frac{\mathbb{E}\, \hat{h}^2(\boldsymbol{e}^\star)}{\mathbb{E}\, \hat{h}'(\boldsymbol{e}^\star)} = \alpha. \tag{25}$$

That is, the lower bound is attained. ■

A.4 Condition (3.13)

Using (3.8) we have

$$\begin{aligned}\mathbb{E}\, \phi_b(x) h_{2,i}^{\text{opt}}(x) &= \int_{-\infty}^{\infty} \phi_b(x) h_{2,i}^{\text{opt}} f_{\boldsymbol{e}(i)}(x)\, dx \\ &= -\int_{-\infty}^{\infty} \phi_b(x) \frac{f'_{\boldsymbol{e}(i)}(x)}{f_{\boldsymbol{e}(i)}(x)} f_{\boldsymbol{e}(i)}(x)\, dx \end{aligned}$$

$$
\begin{aligned}
&= -\int_{-\infty}^{\infty} \phi_b(x) f'_{\boldsymbol{e}(i)}(x)\, dx \\
&= -\phi_b(x) f_{\boldsymbol{e}(i)}(x)\big|_{-\infty}^{\infty} \\
&\quad + \int_{-\infty}^{\infty} \phi'_b(x) f_{\boldsymbol{e}(i)}(x)\, dx \\
&= \mathbb{E}\, \phi'_b(x)
\end{aligned}
$$

where in the last line we exploited the assumption

$$\lim_{x \to \pm\infty} \phi_b(x) f_{\boldsymbol{e}(i)}(x) = 0. \tag{27}$$

A.5 Proof of Lemma 1

Randomness: Writing out the approximation of the update vector $\hat{\boldsymbol{g}}_{k,i}$ in (4.38) in terms of $\boldsymbol{w} \in \boldsymbol{\mathcal{F}}_{i-1}$,

$$\hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}) \approx -\boldsymbol{u}_{k,i}^T \left[\boldsymbol{\alpha}_{k,i}^T \boldsymbol{\varphi}_{v,k,i} + \boldsymbol{u}_{k,i} \left(w^o - \boldsymbol{w}\right) \boldsymbol{\alpha}_{k,i}^T \boldsymbol{\varphi}'_{v,k,i} \right]. \tag{28}$$

Hence, for sufficiently large i, under the model assumptions and *(D-A1)*–*(D-A4)*, it holds that

$$
\begin{aligned}
\mathbb{E}\left\{ \hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}) | \boldsymbol{\mathcal{F}}_{i-1} \right\} &= -p_k(\infty) R_{u,k} \left(w^o - \boldsymbol{w}\right) \\
&\triangleq g_k(\boldsymbol{w})
\end{aligned} \tag{29}
$$

proving (4.39), where

$$p_k(\infty) \triangleq \lim_{i \to \infty} p_k(i) = \left(\mathbb{E}\, \boldsymbol{\alpha}_{k,\infty}\right)^T \overline{\boldsymbol{\varphi}'_{v,k}} \tag{30}$$

and $p_k(i)$ was defined in (4.31). Now let

$$\boldsymbol{v}_{k,i}^g(\boldsymbol{w}) \triangleq \hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}) - g_k(\boldsymbol{w}). \tag{31}$$

Taking the conditional expectation of the squared Euclidean norm of both sides of (31),

$$
\begin{aligned}
\mathbb{E}\left\{ \left\| \boldsymbol{v}_{k,i}^g(\boldsymbol{w}) \right\|^2 \middle| \boldsymbol{\mathcal{F}}_{i-1} \right\} &= \mathbb{E}\left\{ \left\| g_k(\boldsymbol{w}) \right\|^2 \middle| \boldsymbol{\mathcal{F}}_{i-1} \right\} + \mathbb{E}\left\{ \left\| \hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w}) \right\|^2 \middle| \boldsymbol{\mathcal{F}}_{i-1} \right\} \\
&\quad - 2\mathbb{E}\left\{ \hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w})^T g_k(\boldsymbol{w}) \middle| \boldsymbol{\mathcal{F}}_{i-1} \right\}.
\end{aligned} \tag{32}
$$

As for the first term in (32), referring to (29),

$$\mathbb{E}\left\{ \left\| g_k(\boldsymbol{w}) \right\|^2 \middle| \boldsymbol{\mathcal{F}}_{i-1} \right\} = p_k^2(\infty) \left\| w^o - \boldsymbol{w} \right\|_{R_{u,k}^2}^2. \tag{33}$$

Similarly for the third term in (32),

$$
\begin{aligned}
\mathbb{E}\left\{ \hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w})^T g_k(\boldsymbol{w}) \middle| \boldsymbol{\mathcal{F}}_{i-1} \right\} &= \mathbb{E}\left\{ \left\| g_k(\boldsymbol{w}) \right\|^2 \middle| \boldsymbol{\mathcal{F}}_{i-1} \right\} \\
&= p_k^2(\infty) \left\| w^o - \boldsymbol{w} \right\|_{R_{u,k}^2}^2
\end{aligned} \tag{34}
$$

which is nonnegative. As for the second term in (32), referring to (28),

$$\begin{aligned}
&\mathbb{E}\left\{\|\hat{\boldsymbol{g}}_{k,i}(\boldsymbol{w})\|^2\big|\boldsymbol{\mathcal{F}}_{i-1}\right\}\\
&\quad=\mathbb{E}\left\{\|\boldsymbol{u}_{k,i}\|^2\left[\boldsymbol{\alpha}_{k,i}^T\boldsymbol{\varphi}_{v,k,i}+\boldsymbol{u}_{k,i}\,(w^o-\boldsymbol{w})\,\boldsymbol{\alpha}_{k,i}^T\boldsymbol{\varphi}'_{v,k,i}\right]^2\big|\boldsymbol{\mathcal{F}}_{i-1}\right\}\\
&\quad=\mathbb{E}\,\|\boldsymbol{u}_{k,i}\|^2\,\mathbb{E}\left(\boldsymbol{\alpha}_{k,i}^T\boldsymbol{\varphi}_{v,k,i}\right)^2+\|w^o-\boldsymbol{w}\|^2_{\mathbb{E}\|\boldsymbol{u}_{k,i}\|^2\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}}\,\mathbb{E}\left(\boldsymbol{\alpha}_{k,i}^T\boldsymbol{\varphi}'_{v,k,i}\right)^2 \qquad (35a)\\
&\quad=s_k(\infty)\,\mathrm{Tr}(R_{u,k})+t_k(\infty)\,\|w^o-\boldsymbol{w}\|^2_{\mathbb{E}\|\boldsymbol{u}_{k,i}\|^2\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}} \qquad (35b)
\end{aligned}$$

under the model assumptions and (*D-A1*)–(*D-A4*), where

$$s_k(i)=\mathbb{E}\left(\boldsymbol{\alpha}_{k,i}^T\boldsymbol{\varphi}_{v,k,i}\right)^2=\mathrm{Tr}(R_{\alpha_{k,i}}R_{\varphi_{v,k}}) \qquad (36)$$

$$s_k(\infty)\triangleq\lim_{i\to\infty}s_k(i) \qquad (37)$$

and

$$t_k(i)=\mathbb{E}\left(\boldsymbol{\alpha}_{k,i}^T\boldsymbol{\varphi}'_{v,k,i}\right)^2=\mathrm{Tr}(R_{\alpha_{k,i}}R_{\varphi'_{v,k}}) \qquad (38)$$

$$t_k(\infty)\triangleq\lim_{i\to\infty}t_k(i) \qquad (39)$$

with

$$R_{\varphi_{v,k}}\triangleq\mathbb{E}\,\boldsymbol{\varphi}_{v,k,i}\boldsymbol{\varphi}_{v,k,i}^T \qquad (40)$$

$$R_{\varphi'_{v,k}}\triangleq\mathbb{E}\,\boldsymbol{\varphi}'_{v,k,i}\boldsymbol{\varphi}'^T_{v,k,i} \qquad (41)$$

where the time subscript i has been dropped from the time-invariant moments. Note that the moments $p_k(i)$, $s_k(i)$, and $t_k(i)$ for all k and i are nonnegative. In particular, the moment $p_k(i)$ is positive under the model assumption on the noise pdfs being even-symmetric, under assumption (*D-A3*) on the basis functions $\{\phi_{k,b}(x)\}$ being odd-symmetric and monotonically increasing, and under the convexity of the entries of $\alpha_{k,i}$. Substituting (33), (35b), and (34) into (32),

$$\begin{aligned}
&\mathbb{E}\left\{\left\|\boldsymbol{v}_{k,i}^g(\boldsymbol{w})\right\|^2\big|\boldsymbol{\mathcal{F}}_{i-1}\right\}\\
&\quad=p_k^2(\infty)\,\|w^o-\boldsymbol{w}\|^2_{R_{u,k}^2}-2p_k^2(\infty)\,\|w^o-\boldsymbol{w}\|^2_{R_{u,k}^2}+t_k(\infty)\,\|w^o-\boldsymbol{w}\|^2_{\mathbb{E}\|\boldsymbol{u}_{k,i}\|^2\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}}\\
&\qquad+s_k(\infty)\,\mathrm{Tr}(R_{u,k})\\
&\quad\le t_k(\infty)\,\|w^o-\boldsymbol{w}\|^2_{\mathbb{E}\|\boldsymbol{u}_{k,i}\|^2\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}}+s_k(\infty)\,\mathrm{Tr}(R_{u,k})\\
&\quad\le t_k(\infty)\left\|\mathbb{E}\,\|\boldsymbol{u}_{k,i}\|^2\,\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|\cdot\|w^o-\boldsymbol{w}\|^2+s_k(\infty)\,\mathrm{Tr}(R_{u,k}) \qquad (42)
\end{aligned}$$

Making the identifications

$$\begin{aligned}
\beta_k&\triangleq t_k(\infty)\left\|\mathbb{E}\,\|\boldsymbol{u}_{k,i}\|^2\,\boldsymbol{u}_{k,i}^T\boldsymbol{u}_{k,i}\right\|\\
\sigma_{g,k}^2&\triangleq s_k(\infty)\,\mathrm{Tr}(R_{u,k}) \qquad (43)
\end{aligned}$$

for each k, the property (4.40) is proved.

Lipschitz: Evaluating the left-hand side of (4.41) and using (29),

$$\begin{aligned}\|g_k(x) - g_k(y)\| &\leq p_k(\infty)\,\|R_{u,k}\|\,\|x-y\| \\ &= p_k(\infty)\lambda_{\max}(R_{u,k})\,\|x-y\| \end{aligned} \tag{44}$$

Noting that the moment $p_k(i)$ is positive for all k and i and that $\lambda_{\max}(R_{u,k})$ is also positive for all k since $R_{u,k} > 0$, then property (4.41) is proved to hold by making the identification

$$\lambda_{U,k} \triangleq p_k(\infty)\lambda_{\max}(R_{u,k}). \tag{45}$$

Strong monotonicity: Evaluating the left-hand side of (4.42) and using (29),

$$\begin{aligned}(x-y)^T\,[g_k(x) - g_k(y)] &= p_k(\infty)\,\|x-y\|^2_{R_{u,k}} \\ &\geq p_k(\infty)\lambda_{\min}(R_{u,k})\,\|x-y\|^2 \end{aligned} \tag{46}$$

Noting that the moment $p_k(i)$ is positive for all k and i and that $\lambda_{\min}(R_{u,k})$ is also positive for all k since $R_{u,k} > 0$, then property (4.42) is proved to hold by making the identification

$$\lambda_{L,k} \triangleq p_k(\infty)\lambda_{\min}(R_{u,k}). \tag{47}$$

■

A.6 Proof of Lemma 2

Since the matrix $D_k = p_k(\infty)R_{u,k}$ is symmetric, the matrix $\Sigma_k = (I_M - \mu_k D_k)^2$ is symmetric; it is also nonnegative definite. From (4.46) and (4.52), it holds that $\bar{\lambda}_{L,k} I_M \leq D_k \leq \bar{\lambda}_{U,k} I_M$, such that

$$I_M - \mu_k D_k \geq \left(1 - \mu_k \bar{\lambda}_{U,k}\right) I_M \tag{48a}$$

$$I_M - \mu_k D_k \leq \left(1 - \mu_k \bar{\lambda}_{L,k}\right) I_M \tag{48b}$$

and, hence,

$$\lambda(I_M - \mu_k D_k) \geq 1 - \mu_k \bar{\lambda}_{U,k} \tag{49a}$$

$$\lambda(I_M - \mu_k D_k) \leq 1 - \mu_k \bar{\lambda}_{L,k} \tag{49b}$$

Since

$$\lambda(\Sigma_k) = [\lambda(I_M - \mu_k D_k)]^2 \geq 0, \tag{50}$$

then, by (49a)–(49b),

$$\lambda(\Sigma_k) \leq \max\left\{\left|1 - \mu_k \bar{\lambda}_{U,k}\right|^2, \left|1 - \mu_k \bar{\lambda}_{L,k}\right|^2\right\} = \kappa_k^2, \tag{51}$$

which is equivalent to (4.68), where κ_k was defined in (4.69). ■

A.7 Derivation of $Q_{k,i}$ in (5.31)

Setting A to the identity matrix in (5.25) leads to

$$K_{k,i} = \begin{bmatrix} \mu_k(i)U_i^T E_k & \mu_k(i-1)C_{k,i}U_{i-1}^T E_k & \dots & \mu_k(0)\left(C_{k,i}\dots C_{k,1}\right)U_0^T E_k \end{bmatrix} \tag{52}$$

so that

$$K_{k,i}U_{0:i} = \sum_{j=0}^{i} \mu_k(j)\left(C_{k,i}\dots C_{k,j+1}\right)u_{k,j}^T u_{k,j} \tag{53}$$

and

$$K_{k,i}R_{v,0:i}K_{k,i}^T = \sigma_{v,k}^2 \sum_{j=0}^{i} \mu_k^2(j)\left(C_{k,i}\dots C_{k,j+1}\right)u_{k,j}^T u_{k,j}\left(C_{k,i}\dots C_{k,j+1}\right)^T. \tag{54}$$

For small step-sizes $\{\mu_k\}$, we employ the approximation:

$$C_{k,i}\dots C_{k,j+1} \approx I - \sum_{m=j+1}^{i} \mu_k(m)u_{k,m}^T u_{k,m}. \tag{55}$$

Therefore, for small step-sizes $\{\mu_k\}$, expressions (53) and (54) can be approximated as

$$K_{k,i}U_{0:i} \approx \sum_{j=0}^{i} \mu_k(j)u_{k,j}^T u_{k,j} \tag{56}$$

$$K_{k,i}R_{v,0:i}K_{k,i}^T \approx \sigma_{v,k}^2 \sum_{j=0}^{i} \mu_k^2(j)u_{k,j}^T u_{k,j} \tag{57}$$

Since constants multiplying each test-statistic $T_{k,i}$, such as $\sigma_{v,k}^2$, do not affect the resulting detection performance, the choice of $Q_{k,i}$ in (5.31) is justified.

A.8 Proof of Lemma 3

Recall that

$$\boldsymbol{Q}_{k,i} = \left(\sum_{j=0}^{i} \boldsymbol{\mu}_k(j)\boldsymbol{u}_{k,j}^T\boldsymbol{u}_{k,j}\right)\left(\sum_{j=0}^{i} \boldsymbol{\mu}_k^2(j)\boldsymbol{u}_{k,j}^T\boldsymbol{u}_{k,j}\right)^{-1} \tag{58}$$

for $i \geq M-1$, where the matrix $\boldsymbol{U}_{k,0:i} = \operatorname{col}\{\boldsymbol{u}_{k,0},\dots,\boldsymbol{u}_{k,i}\}$ is assumed to be full-rank and the basis functions are assumed to be monotonically increasing to guarantee invertibility above. Under assumption (*D-A4'*), there exists a time index i^* such that the moments $\mathbb{E}\,\boldsymbol{\mu}_k(i)$ and $\mathbb{E}\,\boldsymbol{\mu}_k^2(i)$ would have reached finite constant values for all $i \geq i^*$ denoted by

$$\mathbb{E}\,\boldsymbol{\mu}_k(\infty) = \mu_k\left(\mathbb{E}\,\boldsymbol{\alpha}_{k,\infty}\right)^T\varphi_{0,k}' \tag{59}$$

$$\mathbb{E}\,\boldsymbol{\mu}_k^2(\infty) = \mu_k^2\left(\varphi_{0,k}'^T R_{\alpha_{k,\infty}}\varphi_{0,k}'\right) \tag{60}$$

where $\mathbb{E}\,\boldsymbol{\alpha}_{k,\infty}$ and $R_{\alpha_{k,\infty}}$ were defined in (4.36). Then, under (*D-A4'*)–(*D-A5'*), the process $\left\{\boldsymbol{\mu}_k(i)\boldsymbol{u}_{k,j}^T\boldsymbol{u}_{k,j}\right\}$ is i.i.d. and, hence, $\boldsymbol{Q}_{k,i}$ can be expressed as

$$\begin{aligned}\boldsymbol{Q}_{k,i} &= \left(\sum_{j=0}^{i^*-1}\boldsymbol{\mu}_k(j)\boldsymbol{u}_{k,j}^T\boldsymbol{u}_{k,j} + \sum_{j'=i^*}^{i}\boldsymbol{\mu}_k(j')\boldsymbol{u}_{k,j'}^T\boldsymbol{u}_{k,j'}\right) \\ &\quad\cdot\left(\sum_{j=0}^{i^*-1}\boldsymbol{\mu}_k^2(j)\boldsymbol{u}_{k,j}^T\boldsymbol{u}_{k,j} + \sum_{j'=i^*}^{i}\boldsymbol{\mu}_k^2(j')\boldsymbol{u}_{k,j'}^T\boldsymbol{u}_{k,j'}\right)^{-1} \\ &= \left(\boldsymbol{\Delta}_{k,i^*-1}^{(1)} + (i-i^*+1)\,\mathbb{E}\,\boldsymbol{\mu}_k(\infty)R_{u,k}\right)\left(\boldsymbol{\Delta}_{k,i^*-1}^{(2)} + (i-i^*+1)\,\mathbb{E}\,\boldsymbol{\mu}_k^2(\infty)R_{u,k}\right)^{-1}\end{aligned} \tag{61}$$

where $\boldsymbol{\Delta}_{k,i^*-1}^{(1)}$ and $\boldsymbol{\Delta}_{k,i^*-1}^{(2)}$ represent summations over transient terms and are finite. It follows that

$$\lim_{i\to\infty}\boldsymbol{Q}_{k,i} = \frac{\mathbb{E}\,\boldsymbol{\mu}_k(\infty)}{\mathbb{E}\,\boldsymbol{\mu}_k^2(\infty)}I_M = \eta_k I_M. \tag{62}$$

■

A.9 Derivation of (5.65)—Recursion for $\widehat{R}^{A=I}_{\widetilde{w}_{k,i}}$

Note that the covariance recursion in (5.57) cannot be computed in a distributed fashion across the nodes. In addition to that, it involves the computation of the moments $p_k(i)$, $s_k(i)$, and $t_{k,\ell}(i)$, for all k and ℓ. The approximate recursions in (5.65), for each node k, can be arrived at by setting $\mathcal{A}$ in (5.57) to the identity matrix and replacing the matrices P_i, S_i, and T_i with stochastic approximations $\widehat{P}_i$, $\widehat{S}_i$, and $\widehat{T}_i$. The matrices $\widehat{P}_i$ and $\widehat{S}_i$ are diagonal with entries $\widehat{p}_k(i)$ and $\widehat{s}_k(i)$, respectively, given by (5.66), for $k = 1,\ldots,N$. On the other hand, the (k,ℓ)th entry of the matrix $\widehat{T}_i$ is set to

$$\widehat{t}_{k,\ell}(i) = \widehat{p}_k(i)\widehat{p}_\ell(i). \tag{63}$$

Noting that

$$\widehat{T}_i \odot X = \widehat{P}_i X \widehat{P}_i \tag{64}$$

for any $N\times N$ matrix X, the covariance recursion in (5.57), with $\mathcal{A}$ set to I, is therefore approximated by

$$\widehat{R}^{A=I}_{\widetilde{w}_i} = \left[I - \mathcal{M}\mathcal{U}_i^T\widehat{P}_i\mathcal{U}_i\right]\widehat{R}^{A=I}_{\widetilde{w}_{i-1}}\left[I - \mathcal{U}_i^T\widehat{P}_i\mathcal{U}_i\mathcal{M}\right] + \mathcal{M}\mathcal{U}_i^T\widehat{S}_i\mathcal{U}_i\mathcal{M}. \tag{65}$$

The matrix equation in (65) is block diagonal with the kth equation given by (5.65).

List of Abbreviations

i.i.d.	independent and identically distributed
pdf	probability density function
AR	autoregressive
ATC	adapt-then-combine
CTA	combine-then-adapt
EMSE	excess mean-square error
FIR	finite-impulse response
LMF	least-mean-fourth
LMM	least-mean M-estimate
LMMN	least-mean mixed-norm
LMS	least-mean-squares
MA	moving-average
MC	Monte Carlo
MVU	minimum-variance unbiased
ML	maximum-likelihood
MMSE	minimum mean-square-error
MSD	mean-square deviation
MSE	mean-square error
NP	Neyman–Pearson
NLMS	normalized least-mean-squares
RMN	robust mixed-norm
SNR	signal-to-noise ratio

List of Notation and Symbols

$\mathbb{R}$	set of real numbers
$\mathbb{R}_+$	set of positive real numbers
$\mathbb{R}_{++}$	set of nonnegative real numbers
$\mathbb{R}^N$	set of vectors of size N on $\mathbb{R}$
$\mathbb{R}_+^N$	set of vectors of size N on $\mathbb{R}_+$
$\mathbb{R}_{++}^N$	set of vectors of size N on $\mathbb{R}_{++}$
$\boldsymbol{x}$	boldface lowercase letter denotes a random scalar or vector
$\boldsymbol{X}$	boldface uppercase letter denotes a random matrix
x	normal-font lowercase letter denotes a scalar or vector
X	normal-font uppercase letter denotes a matrix
$x(i)$	scalar quantities are indexed using parenthesis
x_i	vector quantities are indexed using subscripts
$X > 0$	X is positive definite
$X \geq 0$	X is nonnegative definite
$(\cdot)^T$	matrix transposition
$(\cdot)^{-1}$	matrix inversion
$(\cdot)^\dagger$	matrix pseudoinversion
$\mathrm{Tr}(\cdot)$	matrix trace
$\|x\|$	Euclidean norm of x
$\|x\|^2$	squared Euclidean norm of x
$\|x\|_\Sigma^2$	weighted squared Euclidean norm of x, $x^T\Sigma x$
$\|X\|_2$	2-induced norm of A (maximum singular value of A)
$\|X\|_1$	1-induced norm of A (maximum absolute column sum of A)
$\|X\|_\infty$	∞-induced norm of A (maximum absolute row sum of A)
$\lambda_{\max}(A)$	maximum eigenvalue of A
$\lambda_{\min}(A)$	minimum eigenvalue of A
$\lambda_n(A)$	eigenvalues of $N \times N$ matrix A, $n = 1, \ldots, N$

$\mathrm{col}\{\cdot\}$	stacks its arguments vertically
$\mathrm{diag}\{\cdot\}$	- forms a diagonal matrix from its arguments - recovers the vector comprising the diagonal of its matrix argument
$\mathrm{vec}(\cdot)$	vectorizes its matrix argument, stacking columns on top of one another
$\mathrm{vec}^{-1}(\cdot)$	inverse operation of $\mathrm{vec}(\cdot)$
$\mathrm{bvec}(\cdot)$	block-vectorizes its matrix argument
$\mathrm{bvec}^{-1}(\cdot)$	inverse operation of $\mathrm{bvec}(\cdot)$
$X \otimes Y$	Kronecker product of X and Y
$X \otimes_b Y$	block Kronecker product of X and Y
$X \odot Y$	Hadamard (entry-wise) product of X and Y
$\mathbb{1}_N$	all-one vector of size N (suppressed if obvious)
I_N	identity matrix of size N (suppressed if obvious)
$f'(x)$	first derivative of scalar or vector function $f(x)$, $x \in \mathbb{R}$
$f''(x)$	second derivative of scalar or vector function $f(x)$, $x \in \mathbb{R}$
$\nabla_x f(x)$	gradient of scalar or vector function $f(x)$ with respect to vector x (operating on a column vector producing a row vector)
$\lvert\cdot\rvert$	absolute value function
$\exp(\cdot)$	natural exponential function
$\ln(\cdot)$	natural logarithm function
$\tanh(\cdot)$	hyperbolic tangent function
$\mathrm{sech}(\cdot)$	hyperbolic secant function
$\mathrm{sgm}(\cdot)$	sigmoid function, $\mathrm{sgm}(x) \triangleq \frac{1}{1+e^{-x}}$
$\mathbb{E}$	expectation operator
$f_{\boldsymbol{x}}(x)$	probability density function (pdf) of the random variable x
$\mathcal{N}(0, \sigma^2)$	Gaussian distribution with mean 0 and variance σ^2
$Q(\cdot)$	right-tail Gaussian probability function, $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}}\, dt$
$x \triangleq y$	x is defined as y
$x \equiv y$	x is identically equal to y

$\mathcal{O}(N)$	of the order of N
w^o	optimal weight vector
$d(i)$	reference signal at time index i
u_i	regressor at time index i (row vector)
$v(i)$	measurement noise at time index i
w_i	weight estimate at time index i
$\widetilde{w}_i$	weight-error vector at time index i
$e(i)$	output estimation error at time index i
$e_a(i)$	*a priori* estimation error at time index i
R_u	covariance matrix of the regression data
σ_v^2	noise variance
$\mathcal{N}_k$	neighborhood of node k
n_k	degree of node k
$a_{\ell k}$	weight used by node k to scale the data it receives from node ℓ
A	combination policy matrix
$d_k(i)$	reference signal of node k at time index i
$u_{k,i}$	regressor of node k at time index i (row vector)
$v_k(i)$	measurement noise of node k at time index i
$w_{k,i}$	weight estimate of node k at time index i
$\widetilde{w}_{k,i}$	weight-error vector of node k at time index i
$e_k(i)$	output error of node k at time index i
$e_{a,k}(i)$	*a priori* error of node k at time index i
$R_{u,k}$	covariance matrix of the regression data of node k
$\sigma_{v,k}^2$	noise variance of node k

References

[AM97] T. Aboulnasr and K. Mayyas, "A robust variable step-size LMS-type algorithm: Analysis and simulations," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 631–639, Mar. 1997.

[ANS01] T. Y. Al-Naffouri and A. H. Sayed, "Adaptive filters with error nonlinearities: Mean-square analysis and optimum design," *EURASIP Journal on Advances in Signal Processing*, vol. 4, no. 1, pp. 192–205, Dec. 2001.

[ANS03] ——, "Transient analysis of adaptive filters with error nonlinearities," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 653–663, Mar. 2003.

[ANSK00] T. Y. Al-Naffouri, A. H. Sayed, and T. Kailath, "On the selection of optimal nonlinearities for stochastic gradient adaptive algorithms," in *Proc. IEEE ICASSP*, vol. 1, Istanbul, Turkey, Jun. 2000, pp. 464–467 vol.1.

[ASZS13] S. Al-Sayed, A. M. Zoubir, and A. H. Sayed, "An optimal error nonlinearity for robust adaptation against impulsive noise," in *Proc. IEEE SPAWC*, Darmstadt, Germany, Jun. 2013, pp. 415–419.

[ASZS14] ——, "Robust distributed detection over adaptive diffusion networks," in *Proc. IEEE ICASSP*, Florence, Italy, May 2014, pp. 7233–7237.

[ASZSon] ——, "Robust adaptation in impulsive noise," *IEEE Trans. Signal Process.*, submitted for publication.

[BB90] N. J. Bershad and M. Bonnet, "Saturation effects in LMS adaptive echo cancellation for binary data," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 10, pp. 1687–1696, Oct. 1990.

[Ber88] N. J. Bershad, "On error-saturation nonlinearities in LMS adaptation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 4, pp. 440–452, Apr. 1988.

[Ber08] ——, "On error saturation nonlinearities for LMS adaptation in impulsive noise," *IEEE Trans. Signal Process.*, vol. 56, no. 9, pp. 4526–4530, Sep. 2008.

[BGM+01] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*. Berlin: Springer-Verlag, 2001.

[BKR97] T. K. Blankenship, D. M. Kriztman, and T. S. Rappaport, "Measurements and simulation of radio frequency impulsive noise in hospitals and clinics," in *Proc. IEEE Vehicular Technology Conference*, vol. 3, Phoenix, AZ, USA, May 1997, pp. 1942–1946.

[BMC07] C. Boukis, D. P. Mandic, and A. G. Constantinides, "A generalised mixed norm stochastic gradient algorithm," in *Proc. International Conference on Digital Signal Processing*, Cardiff, Wales, UK, Jul. 2007, pp. 27–30.

[BMP87] A. Benveniste, M. Métivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*. NY: Springer-Verlag, 1987.

[BPT+03] A. K. Barros, J. Principe, Y. Takeuchi, C. H. Sales, and N. Ohnishi, "An algorithm based on the even moments of the error," in *Proc. IEEE Workshop on Neural Networks for Signal Processing (NNSP)*, Toulouse, France, Sep. 2003, pp. 879–885.

[Bro62] J. Brown, Jr., "Asymmetric non-mean-square error criteria," *IRE Trans. Automat. Contr.*, vol. 7, no. 1, pp. 64–66, Jan. 1962.

[BS07] S. Barbarossa and G. Scutari, "Bio-inspired sensor network design," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 26–35, May 2007.

[BSD13] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Distributed detection and estimation in wireless sensor networks," *CoRR*, 2013. [Online]. Available: http://arxiv.org/abs/1307.1448

[BV04] S. Boyd and L. Vanderberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[BZ02] R. F. Brcich and A. M. Zoubir, "Robust estimation with parametric score function estimation," in *Proc. IEEE ICASSP*, vol. 2, Orlando, FL, USA, May 2002, pp. 1149–1152.

[CA97] J. Chambers and A. Avlonitis, "A robust mixed-norm adaptive filter algorithm," *IEEE Signal Process. Lett.*, vol. 4, no. 2, pp. 46–48, Feb. 1997.

[CDF+03] S. Camazine, J. L. Deneubourg, N. R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau, *Self-Organization in Biological Systems*. Princeton: Princeton University Press, 2003.

[CES04] D. Culler, D. Estrin, and M. Srivastava, "Overview of sensor networks," *Computer*, vol. 37, no. 8, pp. 41–49, Aug. 2004.

[Cla90] F. Clarke, *Optimization and Nonsmooth Analysis*. PA: SIAM, 1990.

[CM81] T. Claasen and W. Mecklenbraüker, "Comparison of the convergence of two algorithms for adaptive FIR digital filters," *IEEE Trans. Circuits Syst.*, vol. 28, no. 6, pp. 510–518, Jun. 1981.

[CM90] S. H. Cho and V. J. Mathews, "Tracking analysis of the sign algorithm in nonstationary environments," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 12, pp. 2046–2057, Dec. 1990.

[Coh05] L. Cohen, "The history of noise [on the 100th anniversary of its birth]," *IEEE Signal Process. Mag.*, vol. 22, no. 6, pp. 20–45, Nov. 2005.

[CS10] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, Mar. 2010.

[CS11] ——, "Distributed detection over adaptive networks using diffusion adaptation," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 1917–1932, May 2011.

[CS12] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4289–4305, Aug. 2012.

[CS15a] ——, "On the learning behavior of adaptive networks—Part I: Transient analysis," *IEEE Trans. Information Theory*, vol. 61, no. 6, pp. 3487–3517, Jun. 2015.

[CS15b] ——, "On the learning behavior of adaptive networks—Part II: Performance analysis," *IEEE Trans. Information Theory*, vol. 61, no. 6, pp. 3518–3548, Jun. 2015.

[CST11] S. Chouvardas, K. Slavakis, and S. Theodoridis, "Adaptive robust distributed learning in diffusion sensor networks," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4692–4707, Oct. 2011.

[CTC94] J. A. Chambers, O. Tanrikulu, and A. G. Constantinides, "Least mean mixed-norm adaptive filtering," *Electron. Lett.*, vol. 30, no. 19, pp. 1574–1575, Sep. 1994.

[DM94] S. C. Douglas and T. H.-Y. Meng, "Stochastic gradient adaptation under general error criteria," *IEEE Trans. Signal Process.*, vol. 42, no. 6, pp. 1335–1351, Jun. 1994.

[DN03] H. A. David and H. N. Nagaraja, *Order Statistics*. NJ: Wiley, 2003.

[Dut82] D. Duttweiler, "Adaptive filter performance with nonlinearities in the correlation multiplier," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 30, no. 4, pp. 578–586, Aug. 1982.

[EP06] D. Erdogmus and J. C. Principe, "From linear adaptive filtering to nonlinear information processing - The design and analysis of information processing systems," *IEEE Signal Process. Mag.*, vol. 23, no. 6, pp. 14–33, Nov. 2006.

[Ewe92] E. Eweda, "Convergence analysis and design of an adaptive filter with finite-bit power-of-two quantized error," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 39, no. 2, pp. 113–115, Feb. 1992.

[GB14] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[Ger69] A. Gersho, "Some aspects of linear estimation with non-mean-square error criteria," in *Proc. Asilomar Conference on Circuits, Systems and Computers*, Pacific Grove, CA, 1969.

[Ger84] ——, "Adaptive filtering with binary reinforcement," *IEEE Trans. Inform. Theory*, vol. 30, no. 2, pp. 191–199, Mar. 1984.

[GGSB00] T. Gänsler, S. L. Gay, M. M. Sondhi, and J. Benesty, "Double-talk robust fast converging algorithms for network echo cancellation," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 656–663, Nov. 2000.

[Ham68] F. R. Hampel, "Contributions to the Theory of Robust Estimation," Ph.D. dissertation, University of California, Berkeley, 1968.

[Hay01] S. Haykin, *Communication Systems*. NJ: Wiley, 2001.

[Hay12] ——, *Cognitive Dynamic Systems*. Cambridge: Cambridge University Press, 2012.

[Hay13] ——, *Adaptive Filter Theory*. NJ: Prentice Hall, 2013.

[HC92] T. I. Haweel and P. M. Clarkson, "A class of order statistic LMS algorithms," *IEEE Trans. Signal Process.*, vol. 40, no. 1, pp. 44–53, Jan. 1992.

[HJ90] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge: Cambridge University Press, 1990.

[HR09] P. J. Huber and E. M. Ronchetti, *Robust Statistics*. NJ: Wiley, 2009.

[HRRS86] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics: The Approach Based on Influence Functions*. NJ: Wiley, 1986.

[Hub64] P. J. Huber, "Robust estimation of a location parameter," *Ann. Math. Statist.*, vol. 35, no. 1, pp. 73–101, 1964.

[Kas88] S. A. Kassam, *Signal Detection in Non-Gaussian Noise*. NY: Springer-Verlag, 1988.

[Kay98a] S. M. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*. NJ: Prentice Hall, 1998.

[Kay98b] ——, *Fundamentals of Statistical Signal Processing: Estimation Theory*. NJ: Prentice Hall, 1998.

[Kel70] J. L. Kelly, Jr., "Self-adaptive echo canceller," Patent US 3,500,000, 1970.

[KJ92] R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *IEEE Trans. Signal Process.*, vol. 40, no. 7, pp. 1633–1642, Jul. 1992.

[KNW91] R. H. Koning, H. Neudecker, and T. Wansbeek, "Block Kronecker products and the vecb operator," *Linear Algebra and its Applications*, vol. 149, pp. 165–184, 1991.

[KP83] S. Kassam and H. V. Poor, "Robust signal processing for communication systems," *IEEE Commun. Mag.*, vol. 21, no. 1, pp. 20–28, Jan. 1983.

[KP85] S. A. Kassam and H. V. Poor, "Robust techniques for signal processing: A survey," *Proc. IEEE*, vol. 73, no. 3, pp. 433–481, Mar. 1985.

[KY03] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. NY: Springer-Verlag, 2003.

[Lin22] J. W. Lindeberg, "Eine neue Herleitung des Exponentialgesetzes in der Wahrscheinlichkeitsrechnung," *Mathematische Zeitschrift*, vol. 15, no. 1, pp. 211–225, 1922.

[Mat91] V. J. Mathews, "Performance analysis of adaptive filters equipped with the dual sign algorithm," *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 85–91, Jan. 1991.

[MC87] V. J. Mathews and S. H. Cho, "Improved convergence analysis of stochastic gradient adaptive filters using the sign algorithm," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 4, pp. 450–454, Apr. 1987.

[Mid77] D. Middleton, "Statistical-physical models of electromagnetic interference," *IEEE Trans. Electromagn. Compat.*, vol. EMC-19, no. 3, pp. 106–127, Aug. 1977.

[Mid99] ——, "Non-Gaussian noise models in signal processing for telecommunications: New methods and results for class A and class B noise models," *IEEE Trans. Inf. Theory*, vol. 45, no. 4, pp. 1129–1149, May 1999.

[Nut85] A. H. Nuttall, "Evaluation of densities and distributions via Hermite and generalized Laguerre series employing high-order expansion coefficients determined recursively via moments or cumulants," Naval Underwater Systems Center, Tech. Rep. 7377, Feb. 1985.

[PC95] D. I. Pazaitis and A. G. Constantinides, "LMS+F algorithm," vol. 31, no. 17, pp. 1423–1424, Aug. 1995.

[PC96] ——, "An intelligent LMS+F algorithm," in *Proc. IEEE SSAP*, Corfu, Greece, Jun. 1996, pp. 486–489.

[PC99] ——, "A novel kurtosis driven variable step-size adaptive algorithm," *IEEE Trans. Signal Process.*, vol. 47, no. 3, pp. 864–872, Mar. 1999.

[Pol87] B. T. Polyak, *Introduction to Optimization*. NY: Optimization Software, 1987.

[PP02] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*. NY: McGraw-Hill, 2002.

[Pri58] R. Price, "A useful theorem for nonlinear devices having Gaussian inputs," *IRE Trans. Inform. Theory*, vol. 4, no. 2, pp. 69–72, Jun. 1958.

[PT79] B. T. Polyak and Y. Z. Tsypkin, "Adaptive estimation algorithms: convergence, optimality, stability," *Avtomat. i Telemekh.*, no. 3, pp. 71–84, Mar. 1979.

[RC93] P. J. Rousseeuw and C. Croux, "Alternatives to the median absolute deviation," *Journal of the American Statistical Association*, vol. 88, no. 424, pp. 1273–1283, 1993.

[Say03] A. H. Sayed, *Fundamentals of Adaptive Filtering*. NJ: Wiley, 2003.

[Say08] ——, *Adaptive Filters*. NJ: Wiley, 2008.

[Say14a] ——, "Adaptation, learning, and optimization over networks," in *Foundations and Trends in Machine Learning*, vol. 7, no. 4-5, 2014, pp. 311–801.

[Say14b] ——, "Diffusion adaptation over networks," in *Academic Press Library in Signal Processing*, R. Chellapa and S. Theodoridis, Eds., vol. 3. Academic Press, Elsevier, 2014, pp. 323–454.

[Sch91] L. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Boston: Addison-Wesley Publishing Company, 1991.

[Set92] W. A. Sethares, "Adaptive algorithms with nonlinear data and error functions," *IEEE Trans. Signal Process.*, vol. 40, no. 9, pp. 2199–2206, Sep. 1992.

[She58] S. Sherman, "Non-mean-square error criteria," *IRE Trans. Inform. Theory*, vol. 4, no. 3, pp. 125–126, Sep. 1958.

[SJ89] W. A. Sethares and C. R. Johnson, "A comparison of two quantized state adaptive algorithms," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 1, pp. 138–143, Jan. 1989.

[SN93] M. Shao and C. L. Nikias, "Signal processing with fractional lower order moments: stable processes and their applications," *Proc. IEEE*, vol. 81, no. 7, pp. 986–1010, Jul. 1993.

[SSS04] H.-C. Shin, A. H. Sayed, and W.-J. Song, "Variable step-size NLMS and affine projection algorithms," *IEEE Signal Process. Lett.*, vol. 11, no. 2, pp. 132–135, Feb. 2004.

[STC+13] 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.*, vol. 30, no. 3, pp. 155–171, May 2013.

[SV02] G. L. Shevlyakov and N. O. Vilchevski, *Robustness in Data Analysis: Criteria and Methods*. Zeist, The Netherlands: VSP, 2002.

[TBG00] A. Taleb, R. Brcich, and M. Green, "Suboptimal robust estimation for signal plus noise models," in *Proc. Asilomar Conference on Signals, Systems and Computers*, vol. 2, Nov. 2000, pp. 837–841.

[TC96] O. Tanrikulu and J. A. Chambers, "Convergence and steady-state properties of the least-mean mixed-norm (LMMN) adaptive algorithm," *IEE Proc.-Vis. Image Signal Process.*, vol. 143, no. 3, pp. 137–142, Jun. 1996.

[TS12] S.-Y. Tu and A. H. Sayed, "Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6217–6234, Dec. 2012.

[TSY11] S. Theodoridis, K. Slavakis, and I. Yamada, "Adaptive learning in a world of projections," *IEEE Signal Process. Mag.*, vol. 28, no. 1, pp. 97–123, Jan. 2011.

[Tuk60] J. W. Tukey, "A survey of sampling from contaminated distributions," in *Contributions to Probability and Statistics*, I. Olkin, Ed., vol. 3. Stanford: Stanford University Press, 1960, pp. 448–485.

[TYS10] N. Takahashi, I. Yamada, and A. H. Sayed, "Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4795–4810, Sep. 2010.

[VRBT08] L. R. Vega, H. Rey, J. Benesty, and S. Tressens, "A new robust variable step-size NLMS algorithm," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1878–1893, May 2008.

[VV11] V. V. Veeravalli and P. K. Varshney, "Distributed inference in wireless sensor networks," *Phil. Trans. R. Soc. A*, vol. 370, no. 1958, pp. 100–117, 2011.

[WGM+75] B. Widrow, J. R. Glover, Jr., J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. Dong, Jr., and R. C. Goodlin, "Adaptive noise cancelling: Principles and applications," *Proc. IEEE*, vol. 63, no. 12, pp. 1692–1716, Dec. 1975.

[WH60] B. Widrow and M. E. Hoff, "Adaptive switching circuits," in *IRE WESCON Convention Record, Pt. 4*, 1960, pp. 96–104.

[WKL91] V. Weerackody, S. A. Kassam, and K. R. Laker, "Convergence analysis of an algorithm for blind equalization," *IEEE Trans. Commun.*, vol. 39, no. 6, pp. 856–865, Jun. 1991.

[WS85] B. Widrow and S. D. Stearns, *Adaptive Signal Processing.* NJ: Prentice Hall, 1985.

[WW84] E. Walach and B. Widrow, "The least-mean fourth (LMF) adaptive algorithm and its family," vol. 30, no. 2, pp. 275–283, Mar. 1984.

[XL86] P. Xue and B. Liu, "Adaptive equalizer using finite-bit power-of-two quantizer," *IEEE Trans. Signal Process.*, vol. 34, no. 6, pp. 1603–1611, Dec. 1986.

[YO05] I. Yamada and N. Ogura, "Adaptive projected subgradient method for asymptotic minimization of sequence of nonnegative convex functions," *Numerical Functional Analysis and Optimization*, vol. 25, no. 7-8, pp. 593–617, 2005.

[YS13] C.-K. Yu and A. H. Sayed, "A strategy for adjusting combination weights over adaptive networks," in *Proc. IEEE ICASSP*, Vancouver, Canada, May 2013, pp. 4579–4583.

[Zak64] M. Zakai, "General error criteria," *IEEE Trans. Inf. Theory*, vol. 10, no. 1, pp. 94–95, Jan. 1964.

[ZB02] A. M. Zoubir and R. F. Brcich, "Multiuser detection in heavy tailed noise," *Digital Signal Processing*, vol. 12, no. 2-3, pp. 262–273, 2002.

[ZCN00] Y. Zou, S.-C. Chan, and T.-S. Ng, "Least mean M-estimate algorithms for robust adaptive filtering in impulse noise," *IEEE Trans. Circuits Syst. IIII, Analog Digit. Signal Process.*, vol. 47, no. 12, pp. 1564–1569, Dec. 2000.

[ZKCM12] A. M. Zoubir, V. Koivunen, Y. Chakhchoukh, and M. Muma, "Robust estimation in signal processing: A tutorial-style treatment of fundamental concepts," *IEEE Signal Process. Mag.*, vol. 29, no. 4, pp. 61–80, Jul. 2012.

[ZLCH08] Y. Zhang, N. Li, J. Chambers, and Y. Hao, "New gradient-based variable step size LMS algorithms," *EURASIP Journal on Advances in Signal Processing*, pp. 1–9, Jan. 2008.

[ZS12] X. Zhao and A. H. Sayed, "Performance limits for distributed estimation over LMS adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5107–5124, Oct. 2012.

Lebenslauf

Name:	Sara Al-Sayed
Anschrift:	Havelstr. 24
Geburtsdatum:	16.06.1986
Geburtsort:	Kairo, Ägypten
Familienstand:	ledig

Schulausbildung

1991–2004	Ramses College for Girls in Kairo, Ägypten

Studium

2009–2010	Studium der Elektrotechnik an der Universität Ulm, Studienabschluß: Master of Science
2004–2009	Studium der Elektrotechnik an der German University in Cairo, Studienabschluß: Bachelor of Science

Berufstätigkeit

seit 2016	wissenschaftlicher Mitarbeiterin am Fachgebiet Bioinspirierte Kommunikationssysteme, Institut für Nachrichtentechnik, Technische Universität Darmstadt
2011–2015	wissenschaftlicher Mitarbeiterin am Fachgebiet Signalverarbeitung, Institut für Nachrichtentechnik, Technische Universität Darmstadt

Erklärung laut §9 der Promotionsordnung

Ich versichere hiermit, dass ich die vorliegende Dissertation allein und nur unter Verwendung der angegebenen Literatur verfasst habe. Die Arbeit hat bisher noch nicht zu Prüfungszwecken gedient.

Darmstadt, 8. Februar 2016