Distributed Tracking, Decoding, and Demodulation Using Wireless Sensor Networks

---

A THESIS

SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL

OF THE UNIVERSITY OF MINNESOTA

BY

Hao Zhu

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

Professor Georgios B. Giannakis, Advisor

December 2009

Acknowledgments

First and foremost, my deepest gratitude goes to my advisor Prof. Georgios B. Giannakis. I would like to thank him for giving me the opportunity to start this challenging yet exciting endeavor. His advice has been extraordinary by all means, dedicating extensive amounts of time to help me along my first footsteps in research, improve my technical writing and oral presentation skills, and even learn things together to deepen my understanding of the discipline. It is his earnest passion for research that touches and inspires me most, encouraging me to join this exploration of a fascinating realm.

Thanks to Prof. Nihar Jindal and Prof. Zhi-Quan Luo for enlightening me on the subjects they taught me. I hope that part of this knowledge is reflected in this thesis. I am also thankful to Prof. Ofer Zeitouni for agreeing to serve on my M. Sc. thesis committee, representing my minor emphasis in Mathematics. I hope this work may be interesting as an application of mathematics.

The work in this thesis is a result of a very fruitful side-by-side collaboration with Yannis Schizas and Dr. Alfonso Cano. Not only they deserve my gratitude for sharing with me their invaluable research experience, but significant credit is due to their contributions to the work reported here. The results of this thesis further benefited from discussions with current and former members of the SPiNCOM group: Dr. Daniele Angelosante, Juan-Andrés Bazerque, Yannis Delis, Shahrokh Farahmand, Pedro Forero, Nikos Gatsis, Dr. Vassilis Kekatos, Dr. Seung-Jun Kim, Prof. Antonio Marques, Gonzalo Mateos, Eric Msechu, Ketan Rajawat, Prof. Alejandro Ribeiro, Dr. Tairan Wang, Yuchen Wu, and Dr. Yingqun Yu. At this point, I would also like to thank my colleagues of the ITiS group in Tsinghua University; and specially Prof. Xiqin Wang, whose encouragement was instrumental for me to pursue my graduate studies in the US.

My sincere gratitude are also due to my family and friends, for their continuous generosity in supporting me, for their precious time and sincerity to care about me, for their pleasant company and sharing of those exciting moments. Especially to my parents, our love is always the essential strength for us to face up everything coming to us and rejoice upon everything rejoicing us.

Hao Zhu, Minneapolis, November 18, 2009.

Distributed Tracking, Decoding, and Demodulation Using Wireless Sensor Networks

Abstract

Recent advances in hardware technology have led to the emergence of small, low-power, and possibly mobile sensors with limited onboard processing and wireless communication capabilities. When deployed in large numbers over space, these individually primitive sensors can cooperate to form an intelligent network, the wireless sensor network (WSN), capable of measuring aspects or identities of the operational environment with unprecedented accuracy. This is a promising technology ideal for applications as diverse as environmental and healthcare monitoring, smart-house climate control, tactical surveillance, space exploration, and intelligent transportation, to name a few. The advent of WSNs enables re-thinking the field of distributed processing, whereby distributed sensors collaborate to perform power-efficient tracking of nonstationary processes, and reduced-complexity detection of multiple hypotheses. In this thesis, WSN-based distributed tracking and detection algorithms are developed, and analyzed in terms of their optimality, robustness, as well as performance.

The underlying mobility and spatial diversity offered by WSNs gives rise to the interest in distributed tracking of nonstationary signals, and motivates well the distributed counterpart of Kalman filtering developed in this thesis, that is based on judicious sharing of sensor observations. Different from the traditional (centralized) Kalman filter, the low-energy budget per sensor necessitates transmission of reduced-dimensionality data and awareness to imperfect sensor links as integral parts of the distributed design. Adhering to these operational conditions, optimal transmission schemes are developed to minimize the corresponding tracking error by judicious allocation of each sensor's limited power in order to facilitate the fusion of most informative observations.

Through wireless broadcast communications, WSNs offer a suitable platform to realize cooperative information exchange. To comply with their low-complexity radio frequency circuit, individual sensors collaborate to eliminate the ambiguity and detect the broadcasting message, either coded or modulated. As the number of candidate messages grows exponentially, traditional distributed detection algorithms cannot operate with the sensors' limited computation and communication capabilities. This motivates the reduced-complexity distributed decoding and demodulation algorithms of this thesis that rely on in-network one-hop communications to achieve consensus on the sufficient statistics required to decipher the broadcasted message. For both algorithms, the robustness to imperfect inter-sensor links affected by additive noise or random link failures is established, and error rate analysis is provided to evaluate their performance.

Contents

List of Figures

Chapter 1

Wireless Sensor Networks

Wireless sensor networks (WSNs), generally defined as a collection of wireless sensors, have elicited interest in various communities in recent years. A wireless sensor is a device capable of performing the following tasks: sensing the physical environment, performing signal processing tasks, and communicating information using a wireless transceiver. This generic outline itself only seems to be union of the centuries-old fields of sensing and control, and the decades-old field of wireless communications. However, it is the *combination* of these fields that has led to a whole new era of research and development.

Indeed, the great potential that WSNs hold is due to wireless communication abilities of the sensors. The fundamental premise that many simple devices working in synergy offer added value is the key behind the popularity of WSNs. The wireless communication abilities of the sensors enables their collaborative operation, and leads to a whole new dimension of possibilities with regards to environmental, commercial, and military applications. What's more, the impact of WSNs becomes evident in respect of the possibility for distributed inference tasks such as estimation and detection. These tasks play a critical role in e.g., monitoring ambient temperature, humidity, illumination density, as well as localization. Other applications include the detection of either critical events, e.g., the start of a fire to trigger a remote alarm, or the detection of a common message intended for the network. Even though it is the distributed structure of a WSN which provides the means of tackling the aforementioned applications, this is simply not sufficient. Specifically,

it has been recognized that sensors should be empowered with appropriate signal processing tools, that explicitly take into account the distributed nature of the sensed data and operational constraints of the WSNs. In this thesis, novel distributed estimation and detection algorithms are developed and analyzed in terms of their stability and performance. They enable power-efficient tracking of nonstationary processes, and reduced-complexity detection of multiple-hypothesis messages.

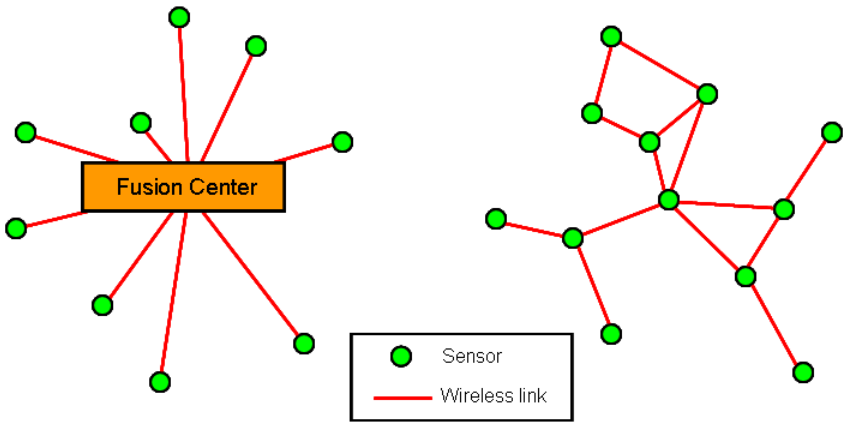
1.1 Challenges and Topologies for WSNs

Formidable challenges, reckoned as customary in the WSN-based applications, will be considered integral to our WSN setup in the rest of the thesis. In order to render large scale deployments economically feasible, sensors are supposed to be inexpensive devices with constrained resources, limited computational, storage, and communication capabilities. Thus, the quality of sensor observations is considered low, and the sensor processing capabilities limited. Moreover, the stringent power and bandwidth available greatly challenges the operation of inter-sensor communications. As a result, the wireless sensor links are expected to be unreliable, intermittent, and transient. The network, on the other hand, is considered to consist of a large number of sensors randomly distributed in a geographical area of interest. Accordingly, the design of a WSN and any specific application should follow the rules of easy deployment, awareness, and robustness to the imperfect wireless links, as well as survivability under limited energy for long periods of time.

In respect to these operational constraints, two kinds of WSN topologies have prevailed as the most suitable for distributed inference tasks. They are characterized by the presence or absence of a central processing unit, also known as (a.k.a.) the fusion center (FC).

If an FC is present, the WSN is hierarchical in the sense that sensors act as information gathering devices for the FC that is in charge of processing this information; see Fig 1.1[1](a). The individual sensors are also capable of simple processing – such as compression and/or quantization – of the acquired raw data before transmission. Sensors collect data from the environment, and transmit pertinent to the information over the unreliable wireless

[1]This figure is taken from [28, pg. 3].



(a) (b)

Figure 1.1: (a) An FC-based WSN where sensors communicate with a fusion center via wireless links; (b) An ad hoc topology devoid of hierarchies.

interface. The FC collects information from different sensors that it later processes to reach a global decision. In contrast to the resource-constrained sensors, it is common to assume that an FC is a device with sufficient processing capabilities, and memory capacity. Hence, this topology may also include a feedback channel from the FC to the sensor to facilitate the local processing per sensor.

When the FC is absent, *ad hoc* WSNs are devoid of hierarchies; see Fig 1.1(b). Here, the network itself is responsible for processing the collected information, and to this end sensors communicate with each other through the shared wireless medium. This so-termed *in-network* processing is based on sensor-to-sensor message exchanges, and allows the local information to percolate across a possibly large geographical area. For such ad hoc WSN topologies, the distributed in-network processing usually consists of two steps: (i) a communication step, where sensors exchange information by broadcasting to either the whole network or nearby neighbors through the shared wireless medium; and (ii) an update step, where each sensor utilizes received information to refine its local estimate. In most cases,

this two-step cooperation leads to consensus on the local estimate across the WSN.

Usually, FC-based topologies benchmark the centralized performance as the network-wide information becomes available for processing at the central unit. The feedback channel from the FC can also compensate for the low-quality estimates obtained at low-cost sensors by assisting the sensor-level processing and sometimes even lessen the sensors' energy shortage.

Nevertheless, compared with FC-based operation, distributed in-network processing performed by ad hoc WSNs is based on cooperative communications, and offers two attractive features: (i) it scales favorably with regards to power requirements, thus prolonging the WSN lifetime; and (ii) it avoids isolating the point(s) of failure, thus enhancing fault tolerance and WSN survivability. Indeed, remote sensors added to expand the geographical area covered by a large-scale WSN have to consume increased power to reach the FC(s); and if the FC fails, the WSN falls in outage. By contrast, if sensors implement in-network processing, information is exchanged only among one-hop neighbors, which keeps the communication overhead per sensor at an affordable level within its neighborhood, as the WSN size grows. Furthermore, error rate performance degrades gracefully even when inter-sensor communications fail, so long as the network remains connected. As a result, it is more likely that an ad hoc WSN can survive for a longer time.

1.2 Motivation and Context

1.2.1 Distributed Estimation and Tracking using WSNs

As its term "sensor" suggests, among the most popular applications of WSNs stem from their sensing ability. Estimation is not only useful as an indication of the field state in itself, but also serves as the first step for distributed control. In addition, maneuverable sensors can monitor a constantly changing environment, or the perhaps dynamic behavior of underlying sources which induce a time-varying process of interest. In such dynamic setups, the parameter estimation task boils down to tracking a nonstationary process. Albeit challenged by the aforementioned WSN operational constraints, distributed estimation and

tracking using WSNs can afford more flexibility and tractability when compared to the traditional centralized algorithms.

Consider as a typical example in the monitoring of temperature over a certain environment. The difficulty for a centralized sensing system is that there is no single indicator, but a space varying field. This spatial variability of the field can be more easily tackled by a distributed network of sensors. Another canonical example is that of target tracking. While a centralized tracker will do just fine, a distributed network can collect more accurate observations given the greater likelihood a sensor has to be close to the target. More details about these applications can be found in the numerical examples of Section 2.4. In a nutshell, with the advent of multi-functional WSNs, distributed estimation and tracking using either FC-based or ad hoc WSN topologies have drawn a lot of interest recently [27, 53].

Without channel-aware dimensionality reduction, power savings brought by FC feedback have been studied in [26] for tracking an auto-regressive process. In order to save power and bandwidth, other existing approaches rely on reduced-dimensionality analog-amplitude or quantized data to track the dynamical process of interest [11, 40]; see also [52] where compression is performed via wavelets. In addition to saving communication resources, dimensionality reduction is well motivated because the sampling rate at a sensor may be different from the data rate supported by the sensor-to-FC or the inter-sensor communication links. In this case, dimensionality-reducing matrices are useful for matching the sampling rate, dictated by a desirable accuracy in estimation, with the data rate that can be supported by the channel. However, transmit-power as well as the fading and receiver noise have not been fully accounted by existing approaches. Furthermore, except for [11], estimation based on reduced-dimensionality sensor data has dealt primarily with stationary signals [44, 53, 56]. A recent alternative for data compression using WSNs relies on compressive sampling of *sparse* signals. The latter is effected via random projections [4]; see also [37] where gossiping algorithms are developed for computing random projections in a distributed fashion. Especially for ad hoc topologies, among the first distributed KF-based trackers using sensors communicating any-to-any over ideal links was the one in [39]. Other existing distributed KF approaches reduce the communication cost either by allowing only

single-hop communications among neighboring sensors and utilizing consensus-based techniques [32, 46], or, by having sensors quantize their data before transmission to all other sensors [40].

In Chapter 2, power-efficient and channel-aware Kalman filtering (KF)-based tracking is pursued for signals that are not necessarily sparse based on analog-amplitude reduced-dimensionality multi-sensor data in both FC-based and ad hoc WSNs.

1.2.2 Distributed Detection using WSNs

As mentioned earlier, the wireless communication feature enables the cooperation among sensors. As a result, collaboration among wireless sensors capitalizes on the premise that limited individual processing and communication capabilities can be greatly enhanced when sharing information with neighboring sensors. In the cooperative broadcast scenario, all users are interested in the message sent, but may not have sufficient signal quality to individually determine the message - a case motivating well the need for cooperation. Inspired by the distributed estimation approach of Chapter 2, and the increasing interest in wireless cooperative communications [13,15], we formulated approaches for global recovery of a class of information-bearing messages through local exchanges among neighboring sensors.

Fig. 1.2 depicts an ad hoc WSN deployed to detect a common message sent from an access point (AP). For efficiency reasons, the AP transmits only limited redundant information, and the sensors cannot request retransmissions when errors are detected. For instance, the helicopter in Fig. 1.2 flies over a WSN deployed on the ground, and broadcasting lasts for only a short period of time. In addition, due to the conceivably low signal-to-noise-ratio (SNR) conditions and the low-cost transceiver hardware, each sensor may be unable to reliably detect the broadcasted message. For example, if the receiver at the sensor can only afford one antenna, the AP's multi-antenna transmission can not be uniquely decoded by individual sensors. In either case, sensors have to collaborate in order to decipher the locally received message. Furthermore, due to communication and energy constraints, inter-sensor communications may be restricted to one-hop transmissions. In this context, the objective here is to develop distributed algorithms for each sensor to decode the common message

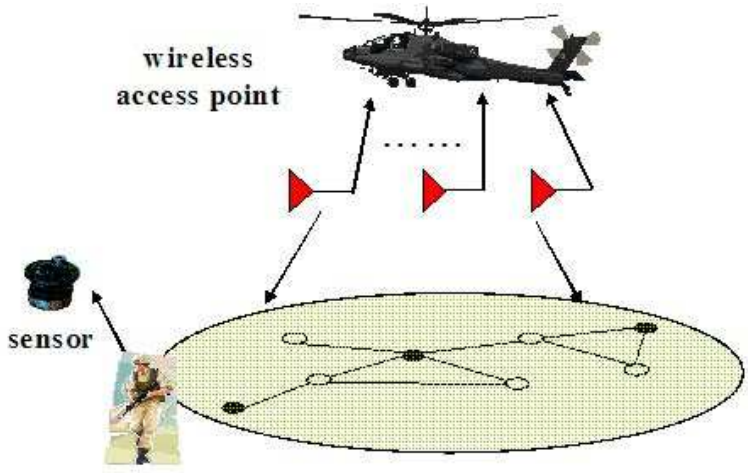


Figure 1.2: A group of sensors collaboratively detects a common message

transmitted from the AP, with accuracy as high as if it had received all other sensors' data. This holds promise to bring local probability of error performance as close as possible to global (i.e., centralized) performance.

From an information-theoretic perspective, the problem of interactive decoding of a common message over a broadcast channel was studied in [15] and [13]. Limited to a pair of users, these works seek the optimal number of conversation rounds [15], or, the achievable capacity region [13]. From a signal processing viewpoint, various distributed algorithms have been developed to exploit collaboration among neighboring sensors for detection-estimation problems, mostly through iterative exchanges of information-bearing messages. The specific problem of distributed consensus averaging (CA) of data collected across sensors has been considered in many works; see e.g., [21,36,54] and references thereof. A more general algorithm for distributed parameter estimation [45] is also available using the method of multipliers (MoM) [7, Sec. 3.4.4]. Distributed hypotheses testing (DHT) approaches, whereby sensors agree on the optimal hypothesis using either CA or belief propagation can be found in [30], [43], and [23].

In this context, Chapter 3 considers the case where the AP broadcasts coded messages,

and the detection problem becomes to efficiently decide one out of multiple candidate codewords; while in Chapter 4 a common space-time matrix is transmitted from a multi-antenna AP, and the distributed demodulation task amounts to selecting one out of the possible signal constellation points transmitted. In this sense, the distributed message detectors pursued here are related to the distributed multiple hypothesis testing techniques developed using WSNs in e.g., [30, 43] and [23]. However, these detectors are not suitable for the message recovery problem because their communication as well as computation complexity grows with the number of hypotheses, which is usually exponential in the message length. To tackle this, reduced-complexity and task-specific algorithms are necessary in lieu of the general one which can be afforded only when the number of hypotheses is small.

1.3 Thesis Outline and Contributions

Chapter 2 deals with the distributed Kalman tracking problem, for both FC-based and ad hoc WSNs. In FC-based topologies, optimal linear dimensionality-reducing operators (matrices) are derived to compress sensor data and minimize the MSE of state estimates at the FC, while adhering to power constraints prescribed at each sensor. An attractive feature of this FC-based approach is the utilization of feedback from the FC to the sensors, which allows them to remove redundancy from their observations, lower the dimensionality of their innovation processes, and thus gain in power efficiency. Compared to [26], the dimensionality reduction approach of the present thesis is attractive because it enhances power savings. The novel approach also includes as a special case an earlier work in [44] for distributed estimation of stationary signals, while it differs from [11] because it accounts for fading channels and additive reception noise.

In ad hoc WSNs, tracking schemes are developed to effect scalability and gain resilience to FC failures. Compared to the approach in [40] which relies on coarsely quantized message exchanges, the approach here pursues *channel-aware* dimensionality reduction of analog-amplitude observations. Specifically, a single sensor broadcasts per time slot its reduced-dimensionality data to all sensors in its range, which rely on the received information to perform MSE-optimal tracking. Such an approach enables even sensors with less informative

observations to acquire high-quality data from active sensors, and boost their tracking performance. The dimensionality-reducing operator employed by the broadcasting sensor to compress its data is chosen to minimize the state estimation MSE at the sensor with the worst reception signal-to-noise ratio (SNR). The material in Chapter 2 has been reported in [62] and [58].

With *in-network* processing, Chapter 3 considers two specific distributed channel decoding problems of interest: (a) distributed in-network maximum-likelihood (ML) block decoding; and (b) distributed in-network *a posteriori* probability (APP) evaluation. The ultimate goal is to obtain tools such as distributed versions of the Viterbi algorithm for block ML decoding and the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm for APP evaluation to enable (possibly iterative) decoding [35, Sec. 8.8]. The solution of (a) and (b) becomes possible by having sensors consent on minimal sufficient statistics, namely average log-likelihood ratios (LLRs) of the distributed data. Compared to the distributed detection approaches in [30, 43] and [23], consenting on the average LLR incurs communication complexity that grows linearly in the message length.

Relative to the existing consensus averaging (CA) approaches mentioned in Section 1.2.2, this thesis leverages them for distributed channel decoding in the presence of imperfect inter-sensor links. Specifically, two iterative algorithms are derived: (i) CA with a single iteration (CA-SI); and (ii) CA using the MoM (CA-MoM). The CA-SI algorithm offers resilience to inter-sensor link failures; yet under noisy links the variance of the resultant iterates grows unbounded. Introducing a vanishing stepsize, the CA-SI algorithm can ensure bounded variance at the price of slower convergence rate. On the other hand, the CA-MoM algorithm converges in the presence of inter-sensor noise without a vanishing stepsize. Compared to [45], the CA-MoM algorithm here offers two distinct novelties: it does not require a bridge sensor set with which sensors need to communicate, thus offering a *fully* distributed CA approach; and it is provably convergent in the presence of inter-sensor link failures. To evaluate the algorithm's performance, error bounds on the average bit-error rate (BER) as a function of the number of iterations are also derived. These bounds enable sensors to quantify the number of iterations needed to reach a prescribed error

performance. Numerical tests demonstrate that only a few consensus iterations suffice for the novel distributed decoders to approach the performance of their centralized counterpart. The results in Chapter 3 have been reported in [63] and [60].

Starting from modulated messages, Chapter 4 develops two algorithms for solving *distributed* demodulation and equalization problems: (a) distributed consensus on demodulated symbols (DC-DS); and (b) distributed consensus on sufficient statistics (DC-SS). In the DC-DS approach, finite-alphabet constraints on the transmitted symbols are relaxed, and the distributed demodulation task is formulated as a convex optimization problem that is solved iteratively in a distributed fashion using the MoM. Distributed versions of the centralized zero-forcing (ZF) and minimum mean-square error (MMSE) demodulators are special cases of this algorithm. An analytical approximation to the symbol error rate (SER) per iteration of the algorithm is also derived. In the DC-SS algorithm, on the other hand, sensors iteratively reach network-wide consensus on the average of (cross-) covariances of locally available per-sensor data vectors with the channel matrix, which constitute sufficient statistics for maximum likelihood (ML) demodulation. Upon obtaining sufficient statistics per sensor it becomes possible to perform general (possibly non-linear) demodulation, including distributed sphere decoding (SD) and ZF/MMSE demodulation. Per-iteration pairwise error probability (PEP) bounds of the ML demodulator are included, establishing that full diversity is achieved in a finite number of iterations. An analytical approximation to the SER per iteration for linear demodulators is also provided. Compared to related distributed detection schemes, both DC-DS and DC-SS can afford reduced overhead in inter-sensor communications, irrespective of the number of hypotheses, which is exponential in the dimensionality of the space-time matrix and the constellation size. Simulated tests demonstrate faster convergence for the DC-SS under ideal inter-sensor links, while the DC-DS offers robustness to non-ideal links. The results of Chapter 4 are included in [59,61] and [64].

The thesis is summarized and interesting open problems are included in Chapter 5.

1.4 Notational conventions

The following notational conventions will be adopted throughout the subsequent chapters. Upper (lower) bold face letters will be used for matrices (column vectors); $[\cdot]_{ij}$ ($[\cdot]_i$) for the $(i,j)-$th (ith) entry of a matrix (vector); diag($\mathbf{x}$) is a diagonal matrix with $\mathbf{x}$ on its diagonal; bdiag($\cdot$) a block diagonal matrix; $(\cdot)^T$ denotes transposition; $(\cdot)^{\mathcal{H}}$ Hermitian transpose; tr($\cdot$) the matrix trace; rank($\cdot$) the matrix rank; $\mathbf{I}_N$ the $N \times N$ identity matrix; $\mathbf{1}_N$ the $N \times 1$ vector of all ones; $\mathbf{0}_N$ the $N \times 1$ vector of all zeros; $\| \cdot \|$ the Frobenius norm; $| \cdot |$ the cardinality of a set; $E[.]$ the expectation; vec $[.]$ the matrix vectorization; $(\mathcal{C})\mathcal{N}(\mu, \sigma^2)$ a (complex) Gaussian distribution with mean μ and variance σ^2; $p[x = z|y]$ the probability density function (pdf) of the continuous random variable (r.v.) x, given the r.v. y, evaluated at z; $p[x = z|y]$ the probability mass function (pmf) of the discrete r.v. x, given the r.v. y, evaluated at z.

Chapter 2

Distributed Kalman Tracking with Dimensionality Reduction

Estimation of nonstationary dynamical processes is of paramount importance in various applications including target tracking and navigation. The goal of this chapter is to perform such tasks in a distributed fashion, using data collected at power-limited sensors which either communicate with a fusion center (FC) over noisy links, or, communicate with each other over non-ideal channels in an ad hoc setting. In FC-based WSNs with a prescribed power budget, linear dimensionality reducing operators which account for the sensor-to-FC channel are derived per sensor to minimize the mean-square error (MSE) of Kalman filtered state estimates formed at the FC. Using these operators and state predictions fed back from the FC online, sensors reduce the dimensionality of their local innovation sequences and communicate them to the FC where tracking estimates are corrected. Analytical and numerical results advocate that the novel channel-aware distributed tracker outperforms competing alternatives. In ad hoc WSNs deployed to perform distributed tracking, one sensor broadcasts reduced-dimensionality data per time slot, according to a pre-specified transmission order. The dimensionality reducing operators employed by the broadcasting sensor are selected to meet its transmit-power budget, while minimizing the state estimation MSE of the sensor with the lowest receiving signal-to-noise ratio (SNR). Based on the received reduced-dimensionality data from the broadcasting sensor, every sensor in range

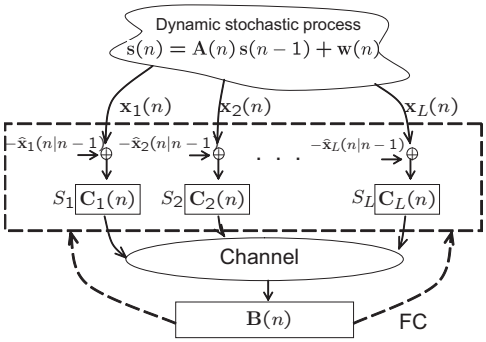


Figure 2.1: Distributed setup for estimating a dynamic process using FC-based WSNs.

performs the MSE optimal tracking. Corroborating distributed target tracking simulations based on distance-only observations illustrate that the novel scheme provides sensors with accurate estimates at affordable communication cost.

Next, we will start with the data models associated with both kinds of WSN topologies.

2.1 WSN Topologies and Problem Statement

Consider a WSN with L sensors $\{S_\ell\}_{\ell=1}^{L}$ deployed to estimate a dynamical nonstationary process $\mathbf{s}(n) \in \mathbb{R}^p$. The goal is to estimate $\mathbf{s}(n)$ based on reduced-dimensionality distributed sensor data for two WSN configurations. The first configuration depicted by the topology in Fig. 2.1 comprises sensors that are linked with an FC, where information gathering and tracking of $\mathbf{s}(n)$ takes place. The second configuration consists of sensors without an FC (ad hoc topology), where tracking of $\mathbf{s}(n)$ is accomplished via information exchanges among neighboring sensors; see Fig. 2.2.

The state of interest $\mathbf{s}(n)$, assumed to be obtained by sampling uniformly the corresponding continuous-time waveform, obeys the discrete-time difference equation

$$\mathbf{s}(n) = \mathbf{A}(n)\mathbf{s}(n-1) + \mathbf{w}(n) \tag{2.1}$$

where $\mathbf{A}(n) \in \mathbb{R}^{p\times p}$ denotes the state transition matrix and $\mathbf{w}(n) \in \mathbb{R}^p$ denotes zero-mean additive white Gaussian noise (AWGN) with covariance matrix $\mathbf{\Sigma}_{ww}(n)$. Sensor S_ℓ observes

$$\mathbf{x}_\ell(n) = \mathbf{H}_\ell(n)\mathbf{s}(n) + \mathbf{v}_\ell(n), \quad \ell = 1, \ldots, L \tag{2.2}$$

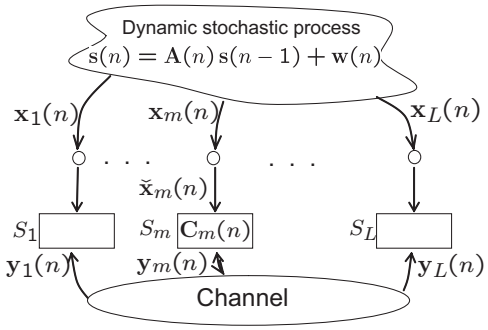


Figure 2.2: Distributed setup for estimating a dynamic process using ad hoc WSNs.

where $\mathbf{H}_\ell(n) \in \mathbb{R}^{N_\ell \times p}$ and the AWGN $\mathbf{v}_\ell(n) \in \mathbb{R}^{N_\ell}$ is zero-mean with (cross-) covariance matrix $\mathbf{\Sigma}_{v_\ell v_m}(n) := E[\mathbf{v}_\ell(n)\mathbf{v}_m{}^T(n)]$, and uncorrelated with $\mathbf{w}(n)$. Different from [40], the observation noise is allowed to be correlated across sensors. Each $N_\ell \times 1$ vector $\mathbf{x}_\ell(n)$ at sensor S_ℓ can be formed by stacking N_ℓ scalar observations collected by sampling faster than the rate $\mathbf{s}(n)$ varies.

2.1.1 Fusion Center-based WSNs

In the envisioned FC-based WSN, sensor S_ℓ relies on a $k_\ell \times N_\ell$ matrix $\mathbf{C}_\ell(n)$ with $k_\ell < N_\ell$ to form the reduced-dimensionality vector $\mathbf{C}_\ell(n)[\mathbf{x}_\ell(n) - \hat{\mathbf{x}}_\ell(n|n-1)]$, where $\hat{\mathbf{x}}_\ell(n|n-1)$ is a vector subtracted from $\mathbf{x}_\ell(n)$ to save transmission power. (It will turn out that $\hat{\mathbf{x}}_\ell(n|n-1)$ is in fact a predictor of $\mathbf{x}_\ell(n)$ based on data to be specified soon.) Furthermore, it is assumed that:

(a2.1) *The S_ℓ-to-FC link comprises a $k_\ell \times k_\ell$ full rank channel matrix $\mathbf{D}_\ell$ along with zero-mean AWGN $\mathbf{z}_\ell(n)$ at the FC with covariance $\mathbf{\Sigma}_{z_\ell z_\ell}$, which is uncorrelated with $\mathbf{s}(n)$, $\{\mathbf{x}_\ell(n)\}_{\ell=1}^L$ and across channels. Sensors transmit over orthogonal channels so that the FC can receive separately the vectors*

$$\mathbf{y}_\ell(n) = \mathbf{D}_\ell \mathbf{C}_\ell(n)[\mathbf{x}_\ell(n) - \hat{\mathbf{x}}_\ell(n|n-1)] + \mathbf{z}_\ell(n),\ \forall \ell = 1, \ldots, L; \tag{2.3}$$

(a2.2) *The covariance matrices $\mathbf{\Sigma}_{ww}(n)$, $\{\mathbf{\Sigma}_{v_\ell v_m}(n)\}_{\ell,m=1}^L$ as well as matrices $\mathbf{A}(n)$ and $\{\mathbf{H}_\ell(n)\}_{\ell=1}^L$, and channel matrices $\{\mathbf{D}_\ell\}_{\ell=1}^L$ and $\{\mathbf{\Sigma}_{z_\ell z_\ell}\}_{\ell=1}^L$ are known at the FC;*

(a2.3) *The feedback channel from the FC to the sensors is ideal.*

Assumption (a2.1) is satisfied by analog or digital frequency-division multiplexing, e.g., multicarrier transmissions, where each entry of the reduced dimensionality vector rides on a subcarrier with nonzero channel gain. Time-invariance of the channels $\{\mathbf{D}_\ell\}_{\ell=1}^L$ is justified when the sensors are static. Matrices $\{\mathbf{D}_\ell\}_{\ell=1}^L$ in (a2.2) can be estimated at the FC using training symbols as detailed in e.g., [6, pg. 383]. The matrices $\mathbf{A}(n)$, $\{\mathbf{H}_\ell(n)\}_{\ell=1}^L$, $\mathbf{\Sigma}_{ww}(n)$ and $\mathbf{\Sigma}_{v_\ell v_m}(n)$ can be obtained from the physics of the problem, i.e., using standard kinematic models with constant velocity or acceleration that are well documented in the tracking literature; see e.g., [5, Chap. 6]. Finally (a2.3), that is also assumed by [11], is reasonable since the FC has sufficient resources to mitigate errors in the FC-to-sensor channel through powerful error control codes.

The FC concatenates $\mathbf{y}_\ell(n)$ for $\ell = 1, \ldots, L$, to form [cf. (2.3)]

$$\mathbf{y}(n) = \text{bdiag}(\mathbf{D}_1\mathbf{C}_1(n), \ldots, \mathbf{D}_L\mathbf{C}_L(n))[\mathbf{x}(n) - \hat{\mathbf{x}}(n|n-1)] + \mathbf{z}(n) \tag{2.4}$$

where $\mathbf{x}(n) := [\mathbf{x}_1^T(n) \ldots \mathbf{x}_L^T(n)]^T$, and likewise for the aggregate predictor $\hat{\mathbf{x}}(n|n-1)$ and the FC noise vector $\mathbf{z}(n)$.

The goal of the FC is to track $\mathbf{s}(n)$ using the received data $\{\mathbf{y}(\nu)\}_{\nu=0}^n$ in (2.4). Specifically, the FC seeks the minimum (M-) MSE optimal state estimate $\hat{\mathbf{s}}(n|n) = E[\mathbf{s}(n)|\mathbf{y}(0), \ldots, \mathbf{y}(n)]$, recursively (via a KF) based on the observations given by (2.4). To this end, it is well known that the filtered estimate $\hat{\mathbf{s}}(n|n)$ can be obtained from the predictor $\hat{\mathbf{s}}(n|n-1) = \mathbf{A}(n)\hat{\mathbf{s}}(n-1|n-1)$ after correcting it using the innovations

$$\tilde{\mathbf{y}}(n|n-1) := \mathbf{y}(n) - \hat{\mathbf{y}}(n|n-1), \text{ where } \hat{\mathbf{y}}(n|n-1) := E[\mathbf{y}(n)|\mathbf{y}(0), \ldots, \mathbf{y}(n-1)]; \tag{2.5}$$

i.e., see e.g., [25, eq. (13.61)]

$$\hat{\mathbf{s}}(n|n) = \hat{\mathbf{s}}(n|n-1) + E[\mathbf{s}(n)|\tilde{\mathbf{y}}(n|n-1)]. \tag{2.6}$$

The correction term in (2.6) is a function of the innovation $\tilde{\mathbf{y}}(n|n-1)$ which clearly depends on the selection of the local predictor $\hat{\mathbf{x}}_\ell(n|n-1)$, as summarized by the following lemma (see Appendix 2.5.1 for the proof).

Lemma 2.1 *If* $\hat{\mathbf{x}}_\ell(n|n-1) = E[\mathbf{x}_\ell(n)|\mathbf{y}(0),\ldots,\mathbf{y}(n-1)]$*, then* $\hat{\mathbf{y}}_\ell(n|n-1) = \mathbf{0}$*; thus* $\tilde{\mathbf{y}}_\ell(n|n-1) = \mathbf{y}_\ell(n)$.

The choice of $\hat{\mathbf{x}}_\ell(n|n-1)$ in Lemma 2.1 enables each sensor S_ℓ to save considerable power because S_ℓ needs to transmit only the non-redundant local innovation $\tilde{\mathbf{x}}_\ell(n|n-1) := \mathbf{x}_\ell(n) - \hat{\mathbf{x}}_\ell(n|n-1)$, which has much smaller variance than its raw data $\mathbf{x}_\ell(n)$. In addition, this innovation that is received at the FC, as $\mathbf{y}_\ell(n)$ coincides with $\tilde{\mathbf{y}}_\ell(n|n-1)$, is all that the FC needs to implement the correction step (2.6) of the KF. Since from (2.2) it holds that $\hat{\mathbf{x}}_\ell(n|n-1) = \mathbf{H}_\ell(n)\hat{\mathbf{s}}(n|n-1)$, the FC must feed back the predictor $\hat{\mathbf{s}}(n|n-1)$ for sensor S_ℓ to form $\hat{\mathbf{x}}_\ell(n|n-1)$ [cf. (a2.3) and Fig. 2.1].

To proceed with the dimensionality reduction task, note that Gaussianity and uncorrelatedness of $\mathbf{z}_\ell(n)$ and $\mathbf{v}_\ell(n)$ imply that $E[\mathbf{s}(n)|\tilde{\mathbf{y}}(n|n-1)]$ in (2.6) is a linear function of $\tilde{\mathbf{y}}(n|n-1) = \mathbf{y}(n)$, i.e., $E[\mathbf{s}(n)|\tilde{\mathbf{y}}(n|n-1)] = \mathbf{B}(n)\tilde{\mathbf{y}}(n|n-1)$, where $\mathbf{B}(n)$ denotes the linear operator applied to the received data $\mathbf{y}(n)$ by the FC. Further, it follows from (2.4) and (2.6) that $\hat{\mathbf{s}}(n|n)$ depends on both $\{\mathbf{C}_\ell(n)\}_{\ell=1}^L$ and $\mathbf{B}(n)$. Thus, for a prescribed average transmit-power P_ℓ per sensor S_ℓ the constrained optimization problem of interest is:

$$\begin{aligned}(\mathbf{B}^o(n), \{\mathbf{C}_\ell^o(n)\}_{\ell=1}^L) &= \arg\min E[\|\mathbf{s}(n) - \hat{\mathbf{s}}(n|n)\|^2] \\ \text{s.t.} \quad &\text{tr}(\mathbf{C}_\ell(n)\mathbf{\Sigma}_{\tilde{x}_\ell\tilde{x}_\ell}(n)\mathbf{C}_\ell^T(n)) \leq P_\ell, \quad \ell = 1,\ldots,L \end{aligned} \tag{2.7}$$

where $\mathbf{\Sigma}_{\tilde{x}_\ell\tilde{x}_\ell}(n)$ denotes the covariance matrix of $\tilde{\mathbf{x}}_\ell(n|n-1)$. Besides transmission, sensors consume power also in the receive mode. The latter is not minimized here but if P_ℓ^{Rx} denotes average receive power and P^{tot} the total power per sensor, the prescribed transmit power in (2.7) is given by $P_\ell := P^{tot} - P_\ell^{Rx}$.

Postponing the solution of (2.7) until Section 2.2, it is useful to outline the envisioned communication protocol that is assumed to rely on orthogonal access, e.g., TDMA. Supposing that clock/timing synchronization has been established, each TDMA slot (time index) n is divided into $2L+1$ sub-slots. During the first sub-slot, the FC broadcasts to all receiving sensors $\hat{\mathbf{s}}(n|n-1)$; and in the next L sub-slots the FC feeds back one after the other $\mathbf{C}_\ell(n)$ for $\ell = 1,\ldots,L$, each to the corresponding sensor S_ℓ. Because each sensor's sensing unit is decoupled from its transceiver unit, over these first $L+1$ sub-plots the L sensors

acquire observations and each, say the ℓth, forms $\tilde{\mathbf{x}}_\ell(n|n-1)$. Over the last L sub-slots, each sensor transmits to the FC in a round-robin fashion its reduced dimensionality data, e.g., S_ℓ transmits $\mathbf{C}_\ell(n)\tilde{\mathbf{x}}(n|n-1)$.

2.1.2 Ad Hoc WSNs

Relative to FC-based WSNs, the distinct feature of ad hoc WSNs is that without a central processing unit the network itself is in charge of tracking the state $\mathbf{s}(n)$. As a result, distributed tracking algorithms can be obtained that are robust to FC failures. Similar to [40], communication among sensors takes place over a shared wireless channel that can afford transmission of a single message per time slot. Specifically, at time slot n there is a single transmitting (broadcasting) sensor S_m which, via a $k \times N_m$ matrix $\mathbf{C}_m(n)$ with $k < N_m$, forms and broadcasts to all other sensors in its range the reduced-dimensionality vector $\mathbf{C}_m(n)\check{\mathbf{x}}_m(n)$, where $\check{\mathbf{x}}_m(n)$ is a function of S_m's observations that will be specified later in Section 2.3; see also Fig. 2.2. The mapping between time slot n and sensor index m is considered given, and can be determined using existing sensor selection (a.k.a. scheduling) schemes [19]. Further, it is assumed that:

(a2.4) *The broadcasting channel from S_m to S_ℓ, with $\ell = 1, \ldots, L$ and $\ell \neq m$, is characterized by a $k \times k$ full rank channel matrix $\mathbf{D}_\ell^m$ and a zero-mean AWGN $\mathbf{z}_\ell(n)$ with covariance matrix $\mathbf{\Sigma}_{z_\ell z_\ell}$, which is uncorrelated with $\mathbf{s}(n)$, $\{\mathbf{x}_\ell(n)\}_{\ell=1}^L$ and across channels. At time slot n, sensor S_ℓ receives the vector*

$$\mathbf{y}_\ell^m(n) = \begin{cases} \mathbf{D}_\ell^m \mathbf{C}_m(n)\check{\mathbf{x}}_m(n) + \mathbf{z}_\ell(n), & \ell \neq m \\ \mathbf{C}_m(n)\check{\mathbf{x}}_m(n), & \ell = m \end{cases} ; \tag{2.8}$$

(a2.5) *Covariance matrices $\mathbf{\Sigma}_{ww}(n)$, $\{\mathbf{\Sigma}_{v_\ell v_m}(n)\}_{\ell,m=1}^L$, and matrices $\mathbf{A}(n)$, $\{\mathbf{H}_\ell(n)\}_{\ell=1}^L$, as well as channel matrices $\{\mathbf{D}_\ell^m\}_{\ell,m=1}^L$ and $\{\mathbf{\Sigma}_{z_\ell z_\ell}\}_{\ell=1}^L$ are assumed available to every sensor.*

As explained in Section 2.1.1, the matrices in (a2.5) can be obtained from the physics of the problem, or during a training phase. These matrices are acquired off-line to enable on-line tracking of $\mathbf{s}(n)$. For simplicity in exposition and since there is a prescribed mapping

between time slot index n and broadcasting sensor index m, in the subsequent analysis we omit the superscript index m in $\mathbf{y}_\ell^m(n)$.

Each sensor S_ℓ wishes to track $\mathbf{s}(n)$ by forming the MMSE estimate of $\mathbf{s}(n)$ given $\{\mathbf{y}_\ell(\nu)\}_{\nu=0}^n$. This will be accomplished through Kalman filtering at each sensor. The state estimate of $\mathbf{s}(n)$ will be obtained at sensor S_ℓ using the prediction $\hat{\mathbf{s}}_\ell(n|n-1) = \mathbf{A}(n)\hat{\mathbf{s}}_\ell(n-1|n-1)$ and the received data $\mathbf{y}_\ell(n)$ as

$$\hat{\mathbf{s}}_\ell(n|n) = \hat{\mathbf{s}}_\ell(n|n-1) + E[\mathbf{s}(n)|\tilde{\mathbf{y}}_\ell(n|n-1)] \tag{2.9}$$

where $\tilde{\mathbf{y}}_\ell(n|n-1) := \mathbf{y}_\ell(n) - \hat{\mathbf{y}}_\ell(n|n-1)$. Unlike (2.6), the predictor-corrector equation (2.9) for ad hoc WSNs is sensor specific.

Note that $\mathbf{y}_\ell(n)$, and thus $\hat{\mathbf{s}}_\ell(n|n)$, is a function of $\mathbf{C}_m(n)$ while $E[\mathbf{s}(n)|\tilde{\mathbf{y}}_\ell(n|n-1)]$ is again, due to Gaussianity, a linear function of the form $\mathbf{B}(n)\tilde{\mathbf{y}}_\ell(n|n-1)$. One way to determine $\mathbf{C}_m(n)$ at the broadcasting sensor S_m could be to solve per slot n the minimization problem

$$\begin{aligned}(\mathbf{B}^o(n),\mathbf{C}_m^o(n)) &= \arg\min E[\|\mathbf{s}(n) - \hat{\mathbf{s}}_\ell(n|n)\|^2] \\ \text{s.t.} \quad & \operatorname{tr}(\mathbf{C}_m(n)\mathbf{\Sigma}_{\check{x}_m\check{x}_m}(n)\mathbf{C}_m^T(n)) \leq P_m.\end{aligned} \tag{2.10}$$

However, (2.10) would then yield a compression matrix $\mathbf{C}_m^o(n)$ that depends on the receiving sensor S_ℓ with $\ell \neq m$, though the broadcasting sensor S_m can only afford broadcasting with a common dimensionality-reducing matrix over all channel links with the receiving sensors $\{S_\ell\}_{\ell=1,\ell\neq m}^L$. Note that determining $\mathbf{C}_m^o(n)$ in a closed form even for a fixed $\ell \neq m$, appears to be impossible.

In order to bypass these difficulties, $\mathbf{C}_m^o(n)$ will be determined in order to minimize the MSE between $\mathbf{s}(n)$ and its reconstruction $\hat{\mathbf{s}}_{\bar{\ell}}(n) = \mathbf{B}(n)\mathbf{y}_{\bar{\ell}}(n)$ obtained at sensor $S_{\bar{\ell}}$, where $S_{\bar{\ell}}$ is the sensor whose reception link from S_m is the worst in the sense that it has the weakest receiving SNR; i.e.,

$$\bar{\ell}_m := \arg \min_{\ell=1,\ldots,L,\ \ell\neq m} \operatorname{tr}\left(\mathbf{D}_\ell^m \mathbf{\Sigma}_{z_\ell z_\ell}^{-1} (\mathbf{D}_\ell^m)^T\right). \tag{2.11}$$

Thus, for a prescribed power P_m at broadcasting sensor S_m, matrix $\mathbf{C}_m(n)$ will be determined as the optimal solution of

$$(\mathbf{B}^o(n),\mathbf{C}_m^o(n)) = \arg\min E[\|\mathbf{s}(n) - \mathbf{B}(n)\mathbf{y}_{\bar{\ell}_m}(n)\|^2]$$
$$\text{s.t.} \quad \text{tr}(\mathbf{C}_m(n)\mathbf{\Sigma}_{\check{x}_m\check{x}_m}(n)\mathbf{C}_m^T(n)) \leq P_m \tag{2.12}$$

where $\mathbf{\Sigma}_{\check{x}_m\check{x}_m}(n)$ denotes the covariance matrix of $\check{\mathbf{x}}_m(n)$ at the broadcasting sensor S_m.

Remark 2.1 In both WSN settings KF-based tracking algorithms will be developed based on reduced-dimensionality sensor data. In FC-based WSNs, the FC must feed back to the sensors $\{\mathbf{C}_\ell^o(n)\}_{\ell=1}^L$ and $\hat{\mathbf{s}}(n|n-1)$ per time slot, that is $\left(\sum_{\ell=1}^L k_\ell N_\ell + p\right)$ scalars. The extra p parameters required relative to [11] is a small increase in the feedback overhead especially in target tracking applications where typically $p = 4$. Thus, [11] and the approaches considered here essentially incur comparable feedback overhead. On the other hand, both approaches in the present chapter remain applicable even if the observation and channel noise vectors do not adhere to a Gaussian distribution–case in which the resultant KF yields the linear (L-) MMSE optimal estimates of $\mathbf{s}(n)$.

Remark 2.2 Although exposition here deals with dimensionality reduction of analog-amplitude data, the model considered does not exclude digital communications. In fact, dimensionality reduction of analog-amplitude samples is the stage preceding rate-optimal bit loading for a quadratic distortion metric of a source coder, e.g., [12, pg. 345]. From another perspective, digital transmissions can be viewed as analog ones with quantization errors lumped in the additive (generally non-) Gaussian noise terms. To ensure that the noise terms $\mathbf{z}_\ell(n)$ are temporarily uncorrelated, the number of quantization bits per sensor and time slot must be sufficiently large.

Remark 2.3 The minimization problems in (2.7) and (2.12), are related to those in [44]. However, the system and data settings here are different from those in [44]. Specifically, [44] deals with distributed parameter estimation of *stationary* signals whereas the setup here entails distributed state tracking for *nonstationary* signals. From this point of view, the

approach here subsumes [44]. Finally, the present setup addresses *both* FC-based and ad hoc topologies, whereas only FC-based WSNs are considered in [44].

2.2 Reduced-Dimensionality KF in FC-based WSNs

State estimation is dealt with in this section using reduced-dimensionality data transmitted from distributed sensors to an FC. The operation of the reduced-dimensionality tracker will be described first, followed by the development of optimal matrices $\{\mathbf{C}_\ell(n)\}_{\ell=1}^L$ and $\mathbf{B}(n)$.

Supposing that all quantities at step $n-1$ are available, the FC relies on $\hat{\mathbf{s}}(n-1|n-1)$ to obtain the state predictor

$$\hat{\mathbf{s}}(n|n-1) = \mathbf{A}(n)\hat{\mathbf{s}}(n-1|n-1) \tag{2.13}$$

and on the estimation error covariance matrix $\mathbf{M}(n-1|n-1) := E[(\mathbf{s}(n-1) - \hat{\mathbf{s}}(n-1|n-1))(\mathbf{s}(n-1) - \hat{\mathbf{s}}(n-1|n-1))^T]$ to find the covariance matrix of the state innovation (prediction error) $\tilde{\mathbf{s}}(n|n-1) := \mathbf{s}(n) - \hat{\mathbf{s}}(n|n-1)$ as

$$\mathbf{M}(n|n-1) = \mathbf{A}(n)\mathbf{M}(n-1|n-1)\mathbf{A}^T(n) + \mathbf{\Sigma}_{ww}(n). \tag{2.14}$$

With MSE optimal $\{\mathbf{C}_\ell(n)\}_{\ell=1}^L$ and $\mathbf{B}(n)$ assumed determined as described in the ensuing section, recall that the FC feeds back to sensor S_ℓ the matrix $\mathbf{C}_\ell(n)$ and the vector $\hat{\mathbf{s}}(n|n-1)$.

Using the feedback and its local observation, sensor S_ℓ forms the innovation $\tilde{\mathbf{x}}_\ell(n|n-1) := \mathbf{x}_\ell(n) - \hat{\mathbf{x}}_\ell(n|n-1) = \mathbf{x}_\ell(n) - \mathbf{H}_\ell(n)\hat{\mathbf{s}}(n|n-1)$ and transmits to the FC $\mathbf{C}_\ell(n)\tilde{\mathbf{x}}_\ell(n|n-1)$. The FC receives each $\mathbf{y}_\ell(n)$ separately as in (2.3) and relies on $\mathbf{y}(n)$ to obtain the filtered estimate (corrector)

$$\hat{\mathbf{s}}(n|n) = \hat{\mathbf{s}}(n|n-1) + \textstyle\sum_{\ell=1}^L \mathbf{B}_\ell(n)\tilde{\mathbf{y}}_\ell(n|n-1) \tag{2.15}$$

where $\mathbf{B}_\ell(n)$ represents the columns of matrix $\mathbf{B}(n)$ with indices $\left(\sum_{m=1}^{\ell-1} k_m + 1\right)$ through $\left(\sum_{m=1}^{\ell} k_m\right)$ and $\mathbf{B}(n) := [\mathbf{B}_1(n) \ldots \mathbf{B}_L(n)]$. (Recall Lemma 2.1 that $\tilde{\mathbf{y}}_\ell(n|n-1) = \mathbf{y}_\ell(n)$.) Matrix $\mathbf{B}(n)$ represents the KF gain. Further, as $\mathbf{B}(n)$ is chosen so that $\hat{\mathbf{s}}(n|n)$ is the (L)MMSE estimator of $\mathbf{s}(n)$ based on $\mathbf{y}(n)$, the orthogonality principle and the linearity of

the expected value operator imply that the filtered error covariance matrix (ECM) can be updated as

$$\begin{aligned}\mathbf{M}(n|n) &= E\left[\left(\tilde{\mathbf{s}}(n|n-1)-\sum_{\ell=1}^{L}\mathbf{B}_\ell(n)\tilde{\mathbf{y}}_\ell(n|n-1)\right)\left(\tilde{\mathbf{s}}(n|n-1)-\sum_{\ell=1}^{L}\mathbf{B}_\ell(n)\tilde{\mathbf{y}}_\ell(n|n-1)\right)^T\right] \\ &= \left[\mathbf{I}-\sum_{\ell=1}^{L}\mathbf{B}_\ell(n)\mathbf{D}_\ell\mathbf{C}_\ell(n)\mathbf{H}_\ell(n)\right]\mathbf{M}(n|n-1) \qquad (2.16)\end{aligned}$$

where the second equality comes from the fact that the estimation error $(\mathbf{s}(n)-\hat{\mathbf{s}}(n|n))$ is uncorrelated with the received data $\{\mathbf{y}(\nu)\}_{\nu=0}^{n}$.

2.2.1 Dimensionality-Reducing Matrices for a Single-Sensor Setup

Before tackling the multi-sensor setup, consider first the design of MSE optimal dimensionality reducing matrix $\mathbf{C}_1(n)$ for a WSN with a single sensor, which turns out to be a basic step for the multi-sensor problem. Using (2.4) with $L=1$ the optimization problem in (2.7) can be rewritten as follows [cf. (2.15)]

$$\begin{aligned}(\mathbf{B}^o(n),\mathbf{C}_1^o(n)) = \arg\min E\left[\|\tilde{\mathbf{s}}(n|n-1)-\mathbf{B}(n)[\mathbf{D}_1\mathbf{C}_1(n)\tilde{\mathbf{x}}_1(n|n-1)+\mathbf{z}_1(n)]\|^2\right] \\ \text{s.t.}\quad \operatorname{tr}(\mathbf{C}_1(n)\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}(n)\mathbf{C}_1^T(n)) \le P_1. \qquad (2.17)\end{aligned}$$

After straightforward algebraic manipulations, the Lagrangian function associated with (2.17) can be expressed as

$$\begin{aligned}&J(\mathbf{B}(n),\mathbf{C}_1(n),\mu) \\ =\ & J_o(n)+\operatorname{tr}(\mathbf{B}(n)\mathbf{\Sigma}_{z_1z_1}\mathbf{B}^T(n))+\operatorname{tr}\left[\left(\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)-\mathbf{B}(n)\mathbf{D}_1\mathbf{C}_1(n)\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}(n)\right)\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}^{-1}(n)\right. \\ &\left.\left(\mathbf{\Sigma}_{\tilde{x}_1\tilde{s}}(n)-\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}(n)\mathbf{C}_1^T(n)\mathbf{D}_1^T\mathbf{B}^T(n)\right)\right]+\mu\left[\operatorname{tr}(\mathbf{C}_1(n)\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}(n)\mathbf{C}_1^T(n))-P_1\right] \qquad (2.18)\end{aligned}$$

where μ is the corresponding Lagrange multiplier and $J_o(n) := \operatorname{tr}[\mathbf{M}(n|n-1)-\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}^{-1}(n)\mathbf{\Sigma}_{\tilde{x}_1\tilde{s}}(n)]$ is the minimum MSE achieved when estimating $\tilde{\mathbf{s}}(n|n-1)$ based on $\tilde{\mathbf{x}}_1(n|n-1)$ without channel distortion and additive noise at the FC; and for any $\ell, m \in [1,L]$

$$\mathbf{\Sigma}_{\tilde{s}\tilde{x}_\ell}(n) := E\left[\tilde{\mathbf{s}}(n|n-1)\tilde{\mathbf{x}}_\ell^T(n|n-1)\right] = \mathbf{M}(n|n-1)\mathbf{H}_\ell^T(n) \qquad (2.19)$$

$$\mathbf{\Sigma}_{\tilde{x}_\ell\tilde{x}_m}(n) := E\left[\tilde{\mathbf{x}}_\ell(n|n-1)\tilde{\mathbf{x}}_m^T(n|n-1)\right] = \mathbf{H}_\ell(n)\mathbf{M}(n|n-1)\mathbf{H}_m^T(n)+\mathbf{\Sigma}_{v_\ell v_m}(n). \qquad (2.20)$$

The Lagrangian function J in (2.18) resembles that in [44, eq. (24)], which deals with distributed estimation of *stationary* signals via reduced-dimensionality observations. However, it is important to stress that here distributed state estimation pertains to nonstationary processes.

In order to obtain $\mathbf{B}^o(n)$ and $\mathbf{C}_1^o(n)$, we will first simplify the Lagrangian in (2.18) using appropriate matrix decompositions. To this end, consider the SVD (see e.g., [18]) $\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n) = \mathbf{U}_{\tilde{s}\tilde{x}}(n)\ \mathbf{S}_{\tilde{s}\tilde{x}}(n)\mathbf{V}_{\tilde{s}\tilde{x}}^T(n)$, as well as the eigen-decompositions $\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}(n) = \mathbf{Q}_{\tilde{x}}(n)\mathbf{\Lambda}_{\tilde{x}}(n)\mathbf{Q}_{\tilde{x}}^T(n)$ and $\mathbf{D}_1^T\mathbf{\Sigma}_{z_1z_1}^{-1}\mathbf{D}_1 = \mathbf{Q}_{zd}\mathbf{\Lambda}_{zd}\mathbf{Q}_{zd}^T$, where $\mathbf{\Lambda}_{\tilde{x}}(n) := \text{diag}(\lambda_{\tilde{x},1}(n), \cdots, \lambda_{\tilde{x},N_1}(n))$ with $\lambda_{\tilde{x},1}(n) \geq \cdots \geq \lambda_{\tilde{x},N_1}(n) > 0$ and $\mathbf{\Lambda}_{zd} := \text{diag}(\lambda_{zd,1}, \cdots, \lambda_{zd,k_1})$ with $\lambda_{zd,1} \geq \cdots \geq \lambda_{zd,k_1} > 0$. Notice that $\lambda_{zd,i}$ captures the SNR of the ith entry in the received vector $\mathbf{y}_1(n)$ at the FC. Further, define $\mathbf{G}(n) := \mathbf{Q}_{\tilde{x}}^T(n)\ \mathbf{V}_{\tilde{s}\tilde{x}}(n)\mathbf{S}_{\tilde{s}\tilde{x}}^T(n)\mathbf{S}_{\tilde{s}\tilde{x}}(n)\mathbf{V}_{\tilde{s}\tilde{x}}^T(n)\mathbf{Q}_{\tilde{x}}(n)$ and let $\rho_g := \text{rank}(\mathbf{G}(n)) = \text{rank}(\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n))$, while matrix $\mathbf{G}_{\tilde{x}}(n) := \mathbf{\Lambda}_{\tilde{x}}^{-1/2}(n)\mathbf{G}(n)\mathbf{\Lambda}_{\tilde{x}}^{-1/2}(n)$ can be factorized as $\mathbf{G}_{\tilde{x}}(n) = \mathbf{Q}_{g\tilde{x}}(n)\mathbf{\Lambda}_{g\tilde{x}}(n)\mathbf{Q}_{g\tilde{x}}^T(n)$, with $\mathbf{\Lambda}_{g\tilde{x}}(n) := \text{diag}(\lambda_{g\tilde{x},1}(n), \cdots, \lambda_{g\tilde{x},\rho_g}(n), 0, \cdots, 0)$ and $\lambda_{g\tilde{x},1}(n) \geq \ldots \geq \lambda_{g\tilde{x},\rho_g}(n) > 0$. Moreover, let $\mathbf{V}_g(n) := \mathbf{\Lambda}_{\tilde{x}}^{-1/2}(n)\mathbf{Q}_{g\tilde{x}}(n)$ denote the invertible matrix which simultaneously diagonalizes the matrices $\mathbf{G}(n)$ and $\mathbf{\Lambda}_{\tilde{x}}(n)$.

Relying on these matrices and their corresponding decompositions it is possible to establish the following proposition (the proof follows using the arguments in [44, Appendices G and H]).

Proposition 2.1 *Under (a2.1), (a2.2) with $L = 1$, and for $k_1 \leq \rho_g := rank(\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)))$, the optimal matrices $\mathbf{C}_1^o(n)$ and $\mathbf{B}^o(n)$ that minimize (2.7) are:*

$$\begin{aligned}\mathbf{C}_1^o(n) =& \mathbf{Q}_{zd}\mathbf{\Phi}_C^o\mathbf{V}_g^T(n)\mathbf{Q}_{\tilde{x}}^T(n) \\ \mathbf{B}^o(n) =& \mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{Q}_{\tilde{x}}(n)\mathbf{V}_g(n)\mathbf{\Phi}_C^{oT}\left(\mathbf{\Phi}_C^o\mathbf{\Phi}_C^{oT} + \mathbf{\Lambda}_{zd}^{-1}\right)^{-1}\mathbf{\Lambda}_{zd}^{-1}\mathbf{Q}_{zd}^T\mathbf{D}_1^T\mathbf{\Sigma}_{z_1z_1}^{-1}\end{aligned} \tag{2.21}$$

where the $k_1 \times N_1$ matrix $\mathbf{\Phi}_C^o := [\phi_{c,ij}^o]$ is diagonal with diagonal entries given by

$$\phi_{c,ii}^o = \begin{cases} \pm\sqrt{\left(\frac{\lambda_{g\tilde{x},i}(n)}{\mu^o\lambda_{zd,i}}\right)^{1/2} - \frac{1}{\lambda_{zd,i}}}, & i = 1, \ldots, \kappa \\ 0, & i = \kappa+1, \ldots, k_1 \end{cases}. \tag{2.22}$$

Index κ is the maximum integer in $[1, k_1]$ for which $\{\phi_{c,ii}^o\}_{i=1}^{\kappa}$ are strictly positive; and μ^o

is the Lagrange multiplier, associated with $\mathbf{C}_1^o(n)$ *and* $\mathbf{B}^o(n)$*, given by*

$$\mu^o = \frac{(\sum_{i=1}^{\kappa}(\lambda_{g\tilde{x},i}(n)\lambda_{zd,i}^{-1})^{1/2})^2}{(P_1 + \sum_{i=1}^{\kappa}\lambda_{zd,i}^{-1})^2}. \tag{2.23}$$

Although $\mathbf{\Phi}_C^o$, μ^o, κ and ρ_g are all functions of n, the time index is dropped for notational brevity.

The optimal matrices $\mathbf{C}_1^o(n)$ and $\mathbf{B}^o(n)$ obtained in Proposition 2.1 are given in closed form as a function of the state and observation model parameters $\mathbf{A}(n)$, $\mathbf{\Sigma}_{ww}(n)$, $\mathbf{H}_1(n)$ and $\mathbf{\Sigma}_{v_1v_1}(n)$, as well as the channel matrix $\mathbf{D}_1$, the received noise covariance $\mathbf{\Sigma}_{z_1z_1}$ and the transmission power P_1.

Remark 2.4 Intuitively, the optimal $\mathbf{C}_1^o(n)$ in Proposition 2.1 selects the entries of $\tilde{\mathbf{x}}_1(n|n-1)$ in which $\mathbf{s}(n)$ is strongest and the channel imperfections weakest, and allocates power among them in a water-filling like manner. It is also worth mentioning that (2.22) dictates a minimum power per sensor. Indeed, in order to ensure that $\text{rank}(\mathbf{\Phi}_C^o) = \kappa$, it must hold that $\mu^o < \lambda_{g\tilde{x},\kappa}(n)\lambda_{zd,\kappa}$, which implies that the power must satisfy

$$P_1 > \frac{\sum_{i=1}^{\kappa}(\lambda_{g\tilde{x},i}(n)\lambda_{zd,i}^{-1})^{1/2}}{\sqrt{\lambda_{g\tilde{x},\kappa}(n)\lambda_{zd,\kappa}}} - \sum_{i=1}^{\kappa}\lambda_{zd,i}^{-1}, \quad \forall n. \tag{2.24}$$

Notice that when the sensor power P_1 is relatively low, κ will be equal to one no matter how many orthogonal channels are available. This way S_1 transmits the most informative entry of $\tilde{\mathbf{x}}_1(n|n-1)$ (associated with $\lambda_{g\tilde{x},1}(n)$), using the most reliable channel (associated with $\lambda_{zd,1}$). On the contrary, if the sensor has much more available power, P_1 is allocated judiciously to all the available channels so that the error between $\mathbf{s}(n)$ and $\hat{\mathbf{s}}(n|n)$ is as small as possible. Besides, when $k_1 > \rho_g$ there is no improvement in the MSE obtained in (2.17), i.e., $\phi_{c,ii}^o = 0$ for any $i > \rho_g$. This is exactly the reason why dimensionality reduction is considered within the interval $[1, \rho_g]$ in Proposition 2.1.

The computations involved in obtaining $\mathbf{C}_1^o(n)$ and $\mathbf{B}^o(n)$ per time slot n, include those needed for: (i) evaluating the SVD of $\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)$, $\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}(n)$ and $\mathbf{G}_{\tilde{x}}(n)$ [cf. (2.21)], and (ii) performing the necessary matrix multiplications. Thus, the complexity for evaluating $\mathbf{C}_1^o(n)$ is $\mathcal{O}(N_1^3)$ per time step n [18, Sec. 5.4].

2.2.2 Dimensionality-Reducing Matrices for the Multi-Sensor Setup

In the multi-sensor scenario it turns out that the minimization problem in (2.7) does not lead to a closed-form solution, and in general incurs complexity that grows exponentially with L [27, 44]. For this reason, we resort to a block coordinate descent algorithm which converges at least to a stationary point of the cost in (2.7). For the L-sensor setup, the cost function in (2.7) becomes [cf. (2.15)]

$$J\left(\{\mathbf{B}_\ell(n)\}_{\ell=1}^L, \{\mathbf{C}_\ell(n)\}_{\ell=1}^L\right) = E\left[\left\|\tilde{\mathbf{s}}(n|n-1) - \sum_{\ell=1}^L \mathbf{B}_\ell(n)[\mathbf{D}_\ell \mathbf{C}_\ell(n)\tilde{\mathbf{x}}_\ell(n|n-1) + \mathbf{z}_\ell(n)]\right\|^2\right]. \tag{2.25}$$

Specifically, suppose temporarily that matrices $\{\mathbf{B}_l(n)\}_{l=1,l\neq\ell}^L$ and $\{\mathbf{C}_l(n)\}_{l=1,l\neq\ell}^L$ are fixed and satisfy the power constraints $\mathrm{tr}(\mathbf{C}_l(n)\mathbf{\Sigma}_{\tilde{x}_l\tilde{x}_l}(n)\mathbf{C}_l^T(n)) = P_l$, for $l = 1, \ldots, L$ and $l \neq \ell$. Upon defining

$$\tilde{\mathbf{s}}_{-\ell}(n) := \mathbf{s}(n) - \hat{\mathbf{s}}(n|n-1) - \textstyle\sum_{l=1,l\neq\ell}^L \mathbf{B}_l(n)\tilde{\mathbf{y}}_l(n|n-1) \tag{2.26}$$

the cost in (2.25) can be written as

$$J(\mathbf{B}_\ell(n), \mathbf{C}_\ell(n)) = E[\|\tilde{\mathbf{s}}_{-\ell}(n) - \mathbf{B}_\ell(n)\tilde{\mathbf{y}}_\ell(n|n-1)\|^2] \tag{2.27}$$

which is a function of $\mathbf{C}_\ell(n)$ and $\mathbf{B}_\ell(n)$. Interestingly, (2.27) falls under the realm of Proposition 2.1. This means that when $\{\mathbf{B}_l(n)\}_{l=1,l\neq\ell}^L$ and $\{\mathbf{C}_l(n)\}_{l=1,l\neq\ell}^L$ are given, the matrices $\mathbf{B}_\ell(n)$ and $\mathbf{C}_\ell(n)$ minimizing (2.27) under the power constraint $\mathrm{tr}(\mathbf{C}_\ell(n)\mathbf{\Sigma}_{\tilde{x}_\ell\tilde{x}_\ell}(n)\mathbf{C}_\ell^T(n)) \leq P_\ell$ can be directly obtained from Proposition 2.1, after setting $\tilde{\mathbf{s}}(n|n-1) \equiv \tilde{\mathbf{s}}_{-\ell}(n)$ and $\tilde{\mathbf{y}}_1(n|n-1) = \tilde{\mathbf{y}}_\ell(n|n-1)$ in (2.17).

In order to apply the corresponding matrix decomposition, it is also necessary to update the (cross-) covariance matrices as follows

$$\begin{aligned}
&\mathbf{\Sigma}_{\tilde{s}_{-\ell}\tilde{s}_{-\ell}}(n) := E[\tilde{\mathbf{s}}_{-\ell}(n)\tilde{\mathbf{s}}_{-\ell}^T(n)] \\
=&\mathbf{M}(n|n-1) - \sum_{l=1,l\neq\ell}^L \mathbf{B}_l(n)\mathbf{D}_l\mathbf{C}_l(n)\mathbf{\Sigma}_{\tilde{x}_l\tilde{s}}(n) - \sum_{l=1,l\neq\ell}^L \mathbf{\Sigma}_{\tilde{s}\tilde{x}_l}(n)\mathbf{C}_l^T(n)\mathbf{D}_l^T\mathbf{B}_l^T(n) \\
&+ \sum_{l=1,l\neq\ell}^L \mathbf{B}_l(n)\mathbf{\Sigma}_{z_l z_l}\mathbf{B}_l^T(n) + \sum_{l=1,l\neq\ell}^L \sum_{m=1,m\neq\ell}^L \mathbf{B}_l(n)\mathbf{D}_l\mathbf{C}_l(n)\mathbf{\Sigma}_{\tilde{x}_l\tilde{x}_m}(n)\mathbf{C}_m^T(n)\mathbf{D}_m^T\mathbf{B}_m^T(n)\ ,
\end{aligned} \tag{2.28}$$

$$\mathbf{\Sigma}_{\tilde{s}_{-\ell}\tilde{x}_\ell}(n) := E[\tilde{\mathbf{s}}_{-\ell}(n)\tilde{\mathbf{x}}_\ell^T(n|n-1)] = \mathbf{\Sigma}_{\tilde{s}\tilde{x}_\ell}(n) - \sum_{l=1,l\neq\ell}^{L} \mathbf{B}_l(n)\mathbf{D}_l\mathbf{C}_l(n)\mathbf{\Sigma}_{\tilde{x}_l\tilde{x}_\ell}(n)\ . \quad (2.29)$$

In summary, the following proposition has been established:

Proposition 2.2 *Under (a2.1), (a2.2) and for given matrices* $\{\mathbf{B}_l(n)\}_{l=1,l\neq\ell}^L$ *and* $\{\mathbf{C}_l(n)\}_{l=1,l\neq\ell}^L$ *satisfying* $tr(\mathbf{C}_l(n)\mathbf{\Sigma}_{\tilde{x}_l\tilde{x}_l}(n)\mathbf{C}_l^T(n)) = P_l$*, the optimal* $\mathbf{B}_\ell^o(n)$ *and* $\mathbf{C}_\ell^o(n)$ *matrices minimizing (2.27) subject to* $tr(\mathbf{C}_\ell(n)\ \mathbf{\Sigma}_{\tilde{x}_\ell\tilde{x}_\ell}(n)\mathbf{C}_\ell^T(n)) \leq P_\ell$ *are provided by Proposition 2.1, after setting* $\mathbf{s}(n) - \hat{\mathbf{s}}(n|n-1) \equiv \tilde{\mathbf{s}}_{-\ell}(n)$ *and* $\tilde{\mathbf{y}}_1(n|n-1) = \tilde{\mathbf{y}}_\ell(n|n-1)$*, and using the (cross-) covariances in (2.28) and (2.29).*

Based on Proposition 2.2, a block coordinate descent algorithm follows whereby the FC determines in an alternating fashion (successively for $\ell = 1, \ldots, L$) the matrices $\mathbf{C}_\ell(n)$ and $\mathbf{B}_\ell(n)$, which are guaranteed to converge at least to a stationary point of (2.7). This round-robin scheme is tabulated as Algorithm 1.

Algorithm 1 Fusion Center: Solving for Optimal Matrices

Initialize randomly $\{\mathbf{C}_\ell^{(0)}(n)\}_{\ell=1}^L$ and $\{\mathbf{B}_\ell^{(0)}(n)\}_{\ell=1}^L$ so that $\text{tr}\left(\mathbf{C}_\ell^{(0)}(n)\mathbf{\Sigma}_{\tilde{x}_\ell\tilde{x}_\ell}(n)(\mathbf{C}_\ell^{(0)}(n))^T\right) = P_\ell$.

for $k = 1, \ldots$ **do**

  for $\ell = 1, \ldots, L$ **do**

    Given $\mathbf{C}_1^{(k)}(n), \mathbf{B}_1^{(k)}(n), \ldots, \mathbf{C}_{\ell-1}^{(k)}(n), \mathbf{B}_{\ell-1}^{(k)}(n), \mathbf{C}_{\ell+1}^{(k-1)}(n), \mathbf{B}_{\ell+1}^{(k-1)}(n), \ldots, \mathbf{C}_L^{(k-1)}(n)$, $\mathbf{B}_L^{(k-1)}(n)$, determine $\mathbf{C}_\ell^{(k)}(n), \mathbf{B}_\ell^{(k)}(n)$ using Proposition 2.2.

  end for

  if $|J(\{\mathbf{B}_\ell^{(k)}(n)\}_{\ell=1}^L, \{\mathbf{C}_\ell^{(k)}(n)\}_{\ell=1}^L) - J(\{\mathbf{B}_\ell^{(k-1)}(n)\}_{\ell=1}^L, \{\mathbf{C}_\ell^{(k-1)}(n)\}_{\ell=1}^L)| < \epsilon$ for a prescribed ϵ **then**

    break

  end if

end for

Remark 2.5 The KF in this subsection relies on reduced-dimensionality sensor data and is both channel- and power-aware. The MMSE optimal dimensionality-reducing matrices in the single-sensor setup as well as those in the multi-sensor case select the most 'informative' entries of $\tilde{\mathbf{x}}_\ell(n|n-1)$ and communicate them to the FC through the most reliable

subchannels of $\mathbf{D}_\ell$. On the other hand, the channel-unaware approach in [11] is challenged by error propagation when AWGN is present in the sensor-to-FC channels. Further, the feedback link from the FC to sensors enables forming and reducing the dimensionality of the innovation $\tilde{\mathbf{x}}_\ell(n|n-1)$ that has much smaller dynamic range than $\mathbf{x}_\ell(n)$, and thus effects transmit-power savings. Such savings are not available in [11] or [44], where *raw* sensor observations are compressed directly without preprocessing. It is also worth mentioning that the novel reduced-dimensionality KF offers a neat generalization of the stationary results in [44] to nonstationary Markov processes.

Algorithm 2–A Fusion Center: Reception and Estimation

Initialize prior estimate $\hat{\mathbf{s}}(0|0)$ and ECM $\mathbf{M}(0|0)$

for $n = 1, 2, \ldots$ **do**

Compute $\hat{\mathbf{s}}(n|n-1)$ and $\mathbf{M}(n|n-1)$ using (2.13) and (2.14)

Compute optimal matrices $\{\mathbf{B}_\ell^o(n)\}_{\ell=1}^L$ and $\{\mathbf{C}_\ell^o(n)\}_{\ell=1}^L$ using Algorithm 1

Transmit $\mathbf{C}_\ell^o(n)$ and $\hat{\mathbf{s}}(n|n-1)$ to sensor S_ℓ for $\ell = 1, \ldots, L$

Receive $\mathbf{y}(n)$ from sensor transmissions

Compute $\hat{\mathbf{s}}(n|n)$ and $\mathbf{M}(n|n)$ using (2.15) and (2.16)

end for

Algorithm 2–B Sensor S_ℓ: Observation and Transmission

Observes $\mathbf{x}_\ell(n)$ and receives the feedback $\hat{\mathbf{s}}(n|n-1)$ and $\mathbf{C}_\ell^o(n)$ from FC

Forms $\hat{\mathbf{x}}_l(n|n-1) := \mathbf{H}_l(n)\hat{\mathbf{s}}(n|n-1)$ used to construct the innovations $\tilde{\mathbf{x}}_\ell(n|n-1)$ thereafter.

Forms $\mathbf{C}_\ell^o(n)\tilde{\mathbf{x}}_\ell(n|n-1)$ and transmits this to the FC

The reduced-dimensionality KF scheme is run at the FC to track the dynamical process $\mathbf{s}(n)$, and is summarized as Algorithm 2–A. The associated computational complexity for implementing Algorithm 1 is $\mathcal{O}(\sum_{\ell=1}^L N_\ell^3)$ per iteration. The data acquisition and dimensionality reduction carried out at the sensors is summarized under Algorithm 2–B, where each sensor S_ℓ only needs to perform a matrix-vector multiplication operation that incurs computational complexity $\mathcal{O}(k_\ell N_\ell)$. Note that at each time-step sensors have to receive from the FC $\mathbf{C}_\ell^o(n)$, which certainly consumes power. However, matrices $\mathbf{C}_\ell^o(n)$ depend only on the covariance of the sensor data $\mathbf{x}_\ell(n)$ and that of $\mathbf{s}(n)$. With matrices

$\mathbf{A}(n), \mathbf{H}_\ell(n), \mathbf{\Sigma}_{ww}(n), \mathbf{\Sigma}_{z_\ell z_\ell}$, $\mathbf{D}_\ell$ and $\mathbf{\Sigma}_{v_\ell v_m}(n)$ available, $\mathbf{C}_\ell(n)$ can thus be determined and stored at the sensors offline pretty much as KF covariance iterations can be run offline. In this case, transmission of the dimensionality-reducing matrices from the FC to the sensors is not an issue.

2.2.3 Reduced-Dimensionality KF with Ideal Links

If sensors utilize powerful error control codes, then the adverse effects of non-ideal links can be mitigated and the sensor-to-FC channel links can be considered essentially ideal. In this case, it is interesting to examine how the reduced-dimensionality KF schemes developed in Sections 2.2.1 and 2.2.2 specialize to ideal channels. To this end, let us replace assumption (a2.1) with:

(a2.1') *The link from sensor S_ℓ to the FC is ideal, so that the FC receives separately the vectors*

$$\mathbf{y}_\ell(n) = \mathbf{C}_\ell(n)[\mathbf{x}_\ell(n) - \hat{\mathbf{x}}_\ell(n|n-1)], \;\; \forall \ell = 1, \ldots, L.$$

Notice that (a2.1') boils down to (a2.1) if the multiplicative fading matrix $\mathbf{D}_\ell$ is set equal to the $k_\ell \times k_\ell$ identity matrix $\mathbf{I}_{k_\ell}$, and the noise covariance matrices $\mathbf{\Sigma}_{z_\ell z_\ell}$ are set equal to $\mathbf{0}$. The optimization problem in (2.7) then becomes

$$(\{\mathbf{B}_\ell^o(n)\}_{\ell=1}^L, \{\mathbf{C}_\ell^o(n)\}_{\ell=1}^L) = \arg\min E\left[\left\|\tilde{\mathbf{s}}(n|n-1) - \sum_{\ell=1}^L \mathbf{B}_\ell(n)\mathbf{C}_\ell(n)\tilde{\mathbf{x}}_\ell(n|n-1)\right\|^2\right]. \quad (2.30)$$

The power constraint per sensor in (2.7) is not necessary for ideal sensor-to-FC links. This holds because the required transmission power can be made negligible by scaling the optimal $\mathbf{C}_\ell^o(n)$ in (2.30) with an arbitrarily small factor α and correspondingly multiplying $\mathbf{B}_\ell^o(n)$ by $1/\alpha$ without inducing loss in performance.

The optimal solution for the minimization problem in (2.30), obtained from (2.7) under the assumption of ideal sensor-to-FC links and after setting $L = 1$, can be easily found if (2.30) is viewed as a canonical correlation analysis (CCA) problem. From this perspective, it follows readily from [9, p. 368] that:

Proposition 2.3 *For $L = 1$ and $k_1 \leq \min(N_1, p)$, the optimal matrices $\mathbf{C}_1^o(n)$ at the sensor and $\mathbf{B}^o(n)$ at the FC that minimize (2.30) under (a2.1') and (a2.2), are:*

$$\mathbf{C}_1^o(n) = \mathbf{F}\mathbf{Q}_{\tilde{s}\tilde{x}_1,k_1}^T(n)\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}^{-1}(n) \ , \quad \mathbf{B}^o(n) = \mathbf{Q}_{\tilde{s}\tilde{x}_1,k_1}(n)\mathbf{F}^{-1} \tag{2.31}$$

where $\mathbf{F}$ is any $k_1 \times k_1$ invertible matrix and $\mathbf{Q}_{\tilde{s}\tilde{x}_1,k_1}(n)$ is the orthonormal matrix formed by the eigenvectors corresponding to the k_1 largest eigenvalues $\{\lambda_{sx_1,i}(n)\}_{i=1}^{k_1}$ of $\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{\Sigma}_{\tilde{x}_1\tilde{x}_1}^{-1}(n)\mathbf{\Sigma}_{\tilde{x}_1\tilde{s}}(n)$.

Interestingly, the results of Proposition 2.1 generalize the noise-free CCA analysis in Proposition 2.3 to non-ideal sensor-to-FC links. Recall that in the classical CCA setup, estimation of $\mathbf{s}(n)$ is performed using noiseless reduced-dimensionality data $\mathbf{C}_1^o(n)\mathbf{y}_1(n)$ which is not the case in our setup. Specifically, it is possible to establish that (see Appendix 2.5.2 for the proof):

Corollary 2.1 *Proposition 2.1 generalizes the CCA analysis summarized in Proposition 2.3 to non-ideal channel links; i.e., the optimal matrices in (2.21) coincide with (2.31) if $\mathbf{D}_1 = \mathbf{I}_{k_1}$ and $\mathbf{\Sigma}_{z_1z_1} = \mathbf{0}$.*

Clearly, Proposition 2.1 allows determination of the MSE optimal dimensionality-reducing operator $\mathbf{C}_1^o(n)$ and the Kalman gain $\mathbf{B}^o(n)$, even when fading and noise are present. These cannot be handled by the classical CCA approach, especially for the non-stationary time-varying setup considered here. Further, notice that if fading and reception noise are not taken into account, estimation errors can accumulate. This is the case with the channel-unaware approach in [11], as will be confirmed with simulated tests presented next. Also, if $p < N_1$ and the dimensionality of $\tilde{\mathbf{x}}_1$ is reduced down to p, the reduced-dimensionality KF coincides with the classical one and there is no loss in MSE performance.

2.3 Reduced-Dimensionality KF in Ad Hoc WSNs

In this section, distributed state tracking algorithms are developed based on reduced-dimensionality data collected by sensors configured in an ad hoc topology. The motivation

behind ad hoc configurations is that information processing is performed across sensors and not at a central location. As a consequence, the resultant tracking schemes exhibit resilience to FC failures. On the other hand, the reduced-dimensionality tracking algorithms developed for FC-based topologies offer better MSE performance since all the available data per time slot are collected and processed at the FC. However, their ad hoc counterparts are flexible to trade-off estimation accuracy for robustness. In addition, in the ad hoc setup only one sensor transmits per time slot, which further improves power efficiency.

A possible approach to reduce the communication cost could be to have each sensor perform tracking based on its own (local) observations. However, some sensors may not attain adequate tracking performance simply because they acquire poor observations. This motivates a communication scheme in which sensors with high-quality observations broadcast their data to all other sensors in range. In target tracking for example, sensors that are closer to the target acquire more accurate observations than sensors located further away. Thus, remote sensors can perform better if they receive and utilize information from sensors in the vicinity of the target. Next, we first outline the steps of the reduced-dimensionality KF running locally at each sensor, and then elaborate on the selection of the dimensionality reducing operator $\mathbf{C}_m(n)$ as well as the data vector $\tilde{\mathbf{x}}_m(n)$ whose dimensionality is reduced at sensor S_m.

Suppose that at time slot $n-1$, each sensor S_ℓ has available a local state estimate $\hat{\mathbf{s}}_\ell(n-1|n-1)$ and the corresponding covariance matrix $\mathbf{M}_\ell(n-1|n-1)$. Then, each sensor carries out the prediction phase, which relies on $\hat{\mathbf{s}}_\ell(n-1|n-1)$ to obtain the state predictor as

$$\hat{\mathbf{s}}_\ell(n|n-1) = \mathbf{A}(n)\hat{\mathbf{s}}_\ell(n-1|n-1) \tag{2.32}$$

and also on $\mathbf{M}_\ell(n-1|n-1)$ to update the prediction ECM via

$$\mathbf{M}_\ell(n|n-1) = \mathbf{A}(n)\mathbf{M}_\ell(n-1|n-1)\mathbf{A}^T(n) + \mathbf{\Sigma}_{ww}(n). \tag{2.33}$$

Consider temporarily that $\check{\mathbf{x}}_m(n)$ and $\mathbf{C}_m(n)$ are given (means of selecting them optimally will follow in the next subsections). At time slot n, sensor S_m broadcasts the reduced-dimensionality vector $\mathbf{C}_m(n)\check{\mathbf{x}}_m(n)$ to all other sensors in range. Each of these sensors, say

the ℓth, receives $\mathbf{y}_\ell(n)$ in (2.8) and corrects its corresponding state estimate as

$$\hat{\mathbf{s}}_\ell(n|n) = \hat{\mathbf{s}}_\ell(n|n-1) + \mathbf{K}_\ell(n)\tilde{\mathbf{y}}_\ell(n|n-1) \tag{2.34}$$

where $\mathbf{K}_\ell(n)$ is the Kalman gain that depends on the statistics of $\tilde{\mathbf{y}}_\ell(n|n-1)$. The gain $\mathbf{K}_\ell(n)$ will be chosen to obtain $\hat{\mathbf{s}}_\ell(n|n)$ as the MMSE estimate of state $\mathbf{s}(n)$ based on the data sequence $\{\mathbf{y}_\ell(\nu)\}_{\nu=0}^n$. The associated ECM can be written as

$$\begin{aligned}\mathbf{M}_\ell(n|n) &= E\left[(\mathbf{s}(n) - \hat{\mathbf{s}}_\ell(n|n))(\mathbf{s}(n) - \hat{\mathbf{s}}_\ell(n|n))^T\right] \\ &= E\left[(\tilde{\mathbf{s}}_\ell(n|n-1) - \mathbf{K}_\ell(n)\tilde{\mathbf{y}}_\ell(n|n-1))\tilde{\mathbf{s}}_\ell^T(n|n-1)\right] \\ &= \mathbf{M}_\ell(n|n-1) - \mathbf{K}_\ell(n)\mathbf{\Sigma}_{\tilde{y}_\ell\tilde{s}_\ell}(n)\end{aligned} \tag{2.35}$$

where $\mathbf{\Sigma}_{\tilde{y}_\ell\tilde{s}_\ell}(n)$ is the cross-covariance of $\tilde{\mathbf{y}}_\ell(n|n-1)$ with $\tilde{\mathbf{s}}_\ell(n|n-1)$, and the Kalman gain is

$$\mathbf{K}_\ell(n) = \mathbf{\Sigma}_{\tilde{s}_\ell\tilde{y}_\ell}(n)\mathbf{\Sigma}_{\tilde{y}_\ell\tilde{y}_\ell}^{-1}(n). \tag{2.36}$$

The issue of which sensor broadcasts at slot n depends on the underlying scheduling algorithm and is assumed given. One possibility is to have only neighboring sensors with high quality observations broadcast their reduced-dimensionality data allowing the rest, even those sensors with less informative data, to form accurate state estimates.

Further, recall that in FC-based WSNs, state tracking when links are ideal can be viewed as a special case of the one for non-ideal links. However, this is not the case in ad hoc WSNs, where reduced-dimensionality KF for ideal and non-ideal inter-sensor links must be treated separately.

2.3.1 Ideal Sensor Links

Under operational conditions elaborated in Section 2.2.3 here too we assume ideal links between sensors. Thus, it follows from (a2.4) that at time slot n sensor S_ℓ receives the vector

$$\mathbf{y}_\ell(n) = \mathbf{C}_m(n)\check{\mathbf{x}}_m(n), \;\; \forall \ell = 1, \ldots, L. \tag{2.37}$$

Since the received sequence $\{\mathbf{y}_\ell(\nu)\}_{\nu=0}^n$ is the same for all $\ell = 1, \ldots, L$, every sensor at time slot n computes the same state estimate given by $\hat{\mathbf{s}}_\ell(n|n) = E[\mathbf{s}(n)|\mathbf{y}_\ell(0), \ldots, \mathbf{y}_\ell(n)]$. Therefore, the sensor index ℓ in (2.32)-(2.37) can be dropped for simplicity in exposition. Similar to Section 2.2.1, let $\check{\mathbf{x}}_m(n)$ denote the observation innovation $\tilde{\mathbf{x}}_m(n|n-1)$, broadcasted by sensor S_m to gain in power efficiency. Since $\check{\mathbf{x}}_m(n) = \tilde{\mathbf{x}}_m(n|n-1) := \mathbf{x}_m(n) - \hat{\mathbf{x}}_m(n|n-1)$, it follows that $\tilde{\mathbf{y}}(n|n-1) = \mathbf{y}(n)$, which implies that the received data across sensors contain the minimal information required to perform the correction step in (2.34).

Because all sensor-to-sensor links are ideal, the power constraint in (2.12) can be dropped as discussed in Section 2.2.3. Further, as the received sequence $\{\mathbf{y}_\ell(\nu)\}_{\nu=0}^n$ is the same for all sensors it follows that the optimal $\mathbf{C}^o(n)$ in (2.12) is the same for all $\ell = 1, \ldots, L$. Keeping these properties in mind, it is possible to reformulate the optimization problem in (2.12) as (details in Appendix 2.5.3)

$$(\mathbf{B}^o(n), \mathbf{C}_m^o(n)) = \arg\min E\left[\|\tilde{\mathbf{s}}(n|n-1) - \mathbf{B}(n)\mathbf{C}_m(n)\tilde{\mathbf{x}}_m(n|n-1)\|^2\right]. \tag{2.38}$$

Similar to the optimization problem in (2.30) with $L = 1$, (2.38) can also be solved using the CCA approach. Thus, the optimal matrices $\mathbf{C}_m^o(n)$ and $\mathbf{B}^o(n)$ are provided by Proposition 2.3 as

$$\mathbf{C}_m^o(n) = \mathbf{F}\mathbf{Q}_{\tilde{s}\tilde{x}_m,k}^T(n)\mathbf{\Sigma}_{\tilde{s}\tilde{x}_m}(n)\mathbf{\Sigma}_{\tilde{x}_m\tilde{x}_m}^{-1}(n), \quad \mathbf{B}^o(n) = \mathbf{Q}_{\tilde{s}\tilde{x}_m,k}(n)\mathbf{F}^{-1}. \tag{2.39}$$

From (2.38) it follows that $\mathbf{B}^o(n)\tilde{\mathbf{y}}(n|n-1)$ is the MMSE optimal estimate of the innovation $\tilde{\mathbf{s}}(n|n-1)$ based on $\tilde{\mathbf{y}}(n|n-1)$. Thus, $\mathbf{B}^o(n)$ plays the role of the Kalman gain $\mathbf{K}_\ell(n)$ associated with the KF steps in (2.32)-(2.35). Further, the ECM in (2.35) can be updated readily if one takes into account that

$$\mathbf{\Sigma}_{\tilde{y}_\ell\tilde{s}_\ell}(n) = \mathbf{C}_m^o(n)\mathbf{H}_m(n)\mathbf{M}(n|n-1), \ \ \forall \ell = 1, \ldots, L. \tag{2.40}$$

2.3.2 Non-ideal Sensor Links

When sensor links are non-ideal, the received data $\{\mathbf{y}_\ell(\nu)\}_{\nu=0}^n$ differ from sensor to sensor, due to the presence of additive reception noise as well as multiplicative fading. The fact that data received by different sensors are different, renders the corresponding filtered and

predicted state estimates also different. Unfortunately, this unavoidable fact does not allow KF via reduced-dimensionality innovations, as explained next.

Suppose that the broadcasting sensor S_m at time slot n forms the reduced-dimensionality vector $\mathbf{C}_m(n)\ \tilde{\mathbf{x}}_m(n|n-1)$; i.e., $\check{\mathbf{x}}_m(n) = \tilde{\mathbf{x}}_m(n|n-1)$. Then, sensor S_ℓ receives the vector

$$\mathbf{y}_\ell(n) = \mathbf{D}_\ell^m \mathbf{C}_m(n)\tilde{\mathbf{x}}_m(n|n-1) + \mathbf{z}_\ell(n).$$

Clearly, sensor S_ℓ can perform the correction step (2.34) so long as it has available the innovation $\tilde{\mathbf{y}}_\ell(n|n-1) = \mathbf{y}_\ell(n) - \hat{\mathbf{y}}_\ell(n|n-1)$, where

$$\hat{\mathbf{y}}_\ell(n|n-1) = \mathbf{D}_\ell^m \mathbf{C}_m(n)\mathbf{H}_m(n)\big\{\hat{\mathbf{s}}_\ell(n|n-1) - E\left[\hat{\mathbf{s}}_m(n|n-1)|\mathbf{y}_\ell(0),\ldots,\mathbf{y}_\ell(n-1)\right]\big\}.$$

Since links are non-ideal, $\mathbf{y}_\ell(n) \neq \mathbf{y}_m(n)$ for $\ell \neq m$, from which we infer that $E[\hat{\mathbf{s}}_m(n|n-1)|\mathbf{y}_\ell(0),\ldots,\mathbf{y}_\ell(n-1)] \neq \hat{\mathbf{s}}_\ell(n|n-1)$. In fact, computing $E[\hat{\mathbf{s}}_m(n|n-1)|\mathbf{y}_\ell(0),\ldots,\mathbf{y}_\ell(n-1)]$ at sensor S_ℓ requires that S_m transmits to S_ℓ the cross-covariance between $\hat{\mathbf{s}}_m(n|n-1)$ and $\hat{\mathbf{s}}_{m'(\nu)}(\nu|\nu-1)$ for $\nu = 1, 2, \ldots, n$, with $m'(\nu)$ denoting the broadcasting sensor at time slot ν. However, this requirement renders inter-sensor communications quite demanding in terms of power and complexity.

This challenge can be bypassed by having S_m perform dimensionality reduction on its raw observation vector $\mathbf{x}_m(n)$; i.e., $\check{\mathbf{x}}_m(n) = \mathbf{x}_m(n)$. In this case, the innovation $\tilde{\mathbf{y}}_\ell(n|n-1)$ in (2.34) takes the form

$$\tilde{\mathbf{y}}_\ell(n|n-1) = \mathbf{y}_\ell(n) - \hat{\mathbf{y}}_\ell(n|n-1) = \mathbf{y}_\ell(n) - \mathbf{D}_\ell^m \mathbf{C}_m(n)\mathbf{H}_m(n)\hat{\mathbf{s}}_\ell(n|n-1)$$

which can be computed easily at sensor S_ℓ for $\ell = 1, \ldots, L$ and $\ell \neq m$. This task requires only the compression matrix $\mathbf{C}_m(n)$ that can also be found locally at S_ℓ [cf. (a2.5)], and matrix $\mathbf{H}_m(n)$ which is available at sensor S_l.

To proceed, consider writing explicitly the (cross-) covariance matrices

$$\mathbf{\Sigma}_{ss}(n) := E[\mathbf{s}(n)\mathbf{s}^T(n)] = \mathbf{A}(n)\mathbf{\Sigma}_{ss}(n-1)\mathbf{A}^T(n) + \mathbf{\Sigma}_{ww}(n) \tag{2.41}$$

$$\mathbf{\Sigma}_{sx_m}(n) := E[\mathbf{s}(n)\mathbf{x}_m^T(n)] = \mathbf{\Sigma}_{ss}(n)\mathbf{H}_m^T(n) \tag{2.42}$$

$$\mathbf{\Sigma}_{x_m x_m}(n) := E[\mathbf{x}_m(n)\mathbf{x}_m^T(n)] = \mathbf{H}_m(n)\mathbf{\Sigma}_{ss}(n)\mathbf{H}_m^T(n) + \mathbf{\Sigma}_{v_m v_m}(n) \tag{2.43}$$

which are required to obtain $\mathbf{C}_m(n)$. Based on these, the optimization in (2.12) can be rewritten as

$$(\mathbf{B}^o(n), \mathbf{C}_m^o(n)) = \arg\min E[\|\mathbf{s}(n) - \mathbf{B}(n)(\mathbf{D}_{\bar{\ell}}^m \mathbf{C}_m(n)\mathbf{x}_m(n) + \mathbf{z}_{\bar{\ell}}(n))\|^2]$$
$$\text{s.t.} \quad \text{tr}(\mathbf{C}_m(n)\mathbf{\Sigma}_{x_m x_m}(n)\mathbf{C}_m{}^T(n)) \leq P_m. \tag{2.44}$$

The optimal solution of (2.44) is provided by Proposition 2.1, since the minimization problem in (2.44) resembles the one in (2.17). Thus, the optimal $\mathbf{C}_m^o(n)$ can be obtained using the (cross-) covariance matrices of $\mathbf{s}(n)$ and $\mathbf{x}_m(n)$ in (2.41)-(2.43), the corresponding channel matrices $\mathbf{D}_{\bar{l}}^m$ and the reception noise covariance $\mathbf{\Sigma}_{z_{\bar{l}} z_{\bar{l}}}$. Due to (a2.5), the quantities that are necessary for solving (2.44) are all available per sensor, which implies that each sensor S_ℓ can locally obtain the optimal matrix $\mathbf{C}_m^o(n)$. Upon receiving $\mathbf{y}_l(n)$, sensor S_l is then able to perform the steps of the reduced-dimensionality KF as outlined next.

At time slot n, the broadcasting sensor S_m computes the innovation of its reduced-dimensionality observation, namely

$$\tilde{\mathbf{y}}_m(n|n-1) = \mathbf{C}_m^o(n)[\mathbf{x}_m(n) - \mathbf{H}_m(n)\hat{\mathbf{s}}_m(n|n-1)] \tag{2.45}$$

and the Kalman gain $\mathbf{K}_m(n) = \mathbf{\Sigma}_{\tilde{s}_m \tilde{y}_m}(n)\mathbf{\Sigma}^{-1}_{\tilde{y}_m \tilde{y}_m}(n)$ with

$$\mathbf{\Sigma}_{\tilde{s}_m \tilde{y}_m}(n) = \mathbf{M}_m(n|n-1)\mathbf{H}_m^T(n)\mathbf{C}_m^{o\,T}(n) \tag{2.46}$$

$$\mathbf{\Sigma}_{\tilde{y}_m \tilde{y}_m}(n) = \mathbf{C}_m^o(n)(\mathbf{H}_m(n)\mathbf{M}_m(n|n-1)\mathbf{H}_m^T(n) + \mathbf{\Sigma}_{v_m v_m}(n))\mathbf{C}_m^{o\,T}(n). \tag{2.47}$$

Then, sensor S_m performs the correction step in (2.34) and updates its filtered state ECM as specified in (2.35). Notice that dimensionality reduction is not necessary at the broadcasting sensor, since it can readily utilize $\mathbf{x}_m(n)$ and perform tracking through the ordinary KF recursions. Though, for symmetry in exposition dimensionality reduction is performed across all sensors.

Similarly, after receiving $\mathbf{y}_\ell(n)$ sensor S_ℓ, with $\ell \neq m$, computes the innovation $\tilde{\mathbf{y}}_\ell(n|n-1) = \mathbf{y}_\ell(n) - \mathbf{D}_\ell^m \mathbf{C}_m(n)\mathbf{H}_m(n)\hat{\mathbf{s}}_\ell(n|n-1)$, as well as the Kalman gain $\mathbf{K}_\ell(n) =$

$\mathbf{\Sigma}_{\tilde{s}_\ell\tilde{y}_\ell}(n)\mathbf{\Sigma}^{-1}_{\tilde{y}_\ell\tilde{y}_\ell}(n)$ where

$$\mathbf{\Sigma}_{\tilde{s}_\ell\tilde{y}_\ell}(n) = \mathbf{M}_\ell(n|n-1)\mathbf{H}_m^T(n)\mathbf{C}_m^{o\,T}(n)\mathbf{D}_\ell^{mT} \tag{2.48}$$

$$\mathbf{\Sigma}_{\tilde{y}_\ell\tilde{y}_\ell}(n) = \mathbf{D}_\ell^m\mathbf{C}_m^o(n)\big[\mathbf{H}_m(n)\mathbf{M}_\ell(n|n-1)\mathbf{H}_m^T(n) + \mathbf{\Sigma}_{v_mv_m}(n)\big]\mathbf{C}_m^{o\,T}(n)\mathbf{D}_\ell^{mT} + \mathbf{\Sigma}_{z_\ell z_\ell} \tag{2.49}$$

and forms $\hat{\mathbf{s}}_\ell(n|n)$ through (2.34). Using (2.48), (2.49) and (2.34) the correction step can be performed locally once sensor S_ℓ receives the reduced-dimensionality data from the broadcasting sensor S_m, without any extra communications.

The reduced-dimensionality KF schemes in this section (for either ideal or non-ideal links) entail two phases. In the first phase the broadcasting sensor forms $\check{\mathbf{x}}_m(n)$ from $\mathbf{x}_m(n)$, performs dimensionality reduction of $\check{\mathbf{x}}_m(n)$ and broadcasts $\mathbf{C}_m^o(n)\check{\mathbf{x}}_m(n)$ to all other sensors. This phase is tabulated as Algorithm 3–A. The second phase involves reception of $\mathbf{y}_\ell(n)$ at all sensors in range, through (non-)ideal links and implementation of properly designed KF recursions. This phase is tabulated as Algorithm 3–B.

Algorithm 3–A Broadcasting Sensor S_m: Observation and transmission

Compute matrix $\mathbf{C}_m^o(n)$ after updating the corresponding (cross-) covariance matrices

Form $\check{\mathbf{x}}_m(n) = \begin{cases} \tilde{\mathbf{x}}_m(n|n-1), & \text{for ideal links} \\ \mathbf{x}_m(n), & \text{for non-ideal links} \end{cases}$

Construct and transmit $\mathbf{C}_m^o(n)\check{\mathbf{x}}_m(n)$

Algorithm 3–B Sensor S_ℓ: Reception and estimation

Initialize prior estimate $\hat{\mathbf{s}}_\ell(0|0)$ and ECM $\mathbf{M}_\ell(0|0)$

for $n = 1, 2, \ldots$ **do**

Compute $\hat{\mathbf{s}}_\ell(n|n-1)$ and $\mathbf{M}_\ell(n|n-1)$ using (2.32) and (2.33)

Compute $\mathbf{C}_m^o(n)$ via (2.44).

Receive $\mathbf{y}_\ell(n)$

Compute $\hat{\mathbf{s}}_\ell(n|n)$ and $\mathbf{M}_\ell(n|n)$ using (2.34) and (2.35).

end for

Algorithm 3–A is run at the broadcasting sensor and provides reduced-dimensionality data to the rest of the neighboring (listening) sensors; Algorithm 3–B is run by all sensors in range to keep track of the state via the filtered estimate $\hat{\mathbf{s}}(n|n)$. The computational

complexity per sensor is determined by the steps involved in determining the optimal $\mathbf{C}_m^o(n)$ in Proposition 2.1, which is $\mathcal{O}(N_m^3)$ per time slot n.

Remark 2.6 The tracking schemes summarized as Algorithm 3 offer resilience to FC failures as well as savings in transmit-power. The price paid is increase in the estimation MSE as will become more apparent in the next subsection. However, at any time slot only one sensor is broadcasting while the rest are just listening. This way the lifetime of the ad hoc network can be extended relative to the FC-based approach. Additionally, the distributed tracking schemes developed for ad hoc WSNs scale better than Algorithm 2 as the number of sensors increases, and offer operational robustness (to FC failures). On the other hand, the distributed tracking schemes for ad hoc and FC-based WSNs have complementary strengths. Indeed, if estimation accuracy is at a premium, the FC-based tracker can be applied, whereas when robustness, scalability and extended WSN lifetime are more important the ad hoc distributed schemes should be preferred.

2.4 Numerical Examples

2.4.1 Algorithm 2 for FC-based WSNs

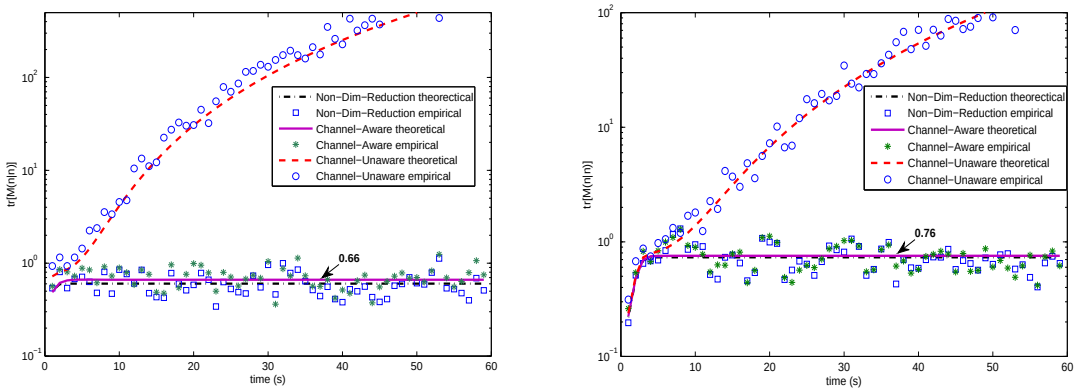
First we test the performance of our channel-aware reduced-dimensionality KF for FC-based WSNs and compare it with [11]. Consider a WSN deployed for measuring e.g., room temperature. A common state zero-acceleration propagation model is adopted from [40], where the temperature contained in the first entry of $\mathbf{s}(n)$ evolves according to

$$\mathbf{s}(n) = \begin{pmatrix} T(n) \\ \dot{T}(n) \end{pmatrix} = \begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix} \mathbf{s}(n-1) + \mathbf{w}(n) \tag{2.50}$$

with $\alpha = 0.05$ and covariance matrix

$$\mathbf{\Sigma}_{ww}(n) = \begin{pmatrix} \alpha^3/3 & \alpha^2/2 \\ \alpha^2/2 & \alpha \end{pmatrix}. \tag{2.51}$$

The WSN comprises $L = 50$ sensors, where each sensor S_ℓ acquires N_ℓ temperature measurements. The observation matrix $\mathbf{H}_l(n)$ has dimensionality $N_\ell \times 2$ with its first column

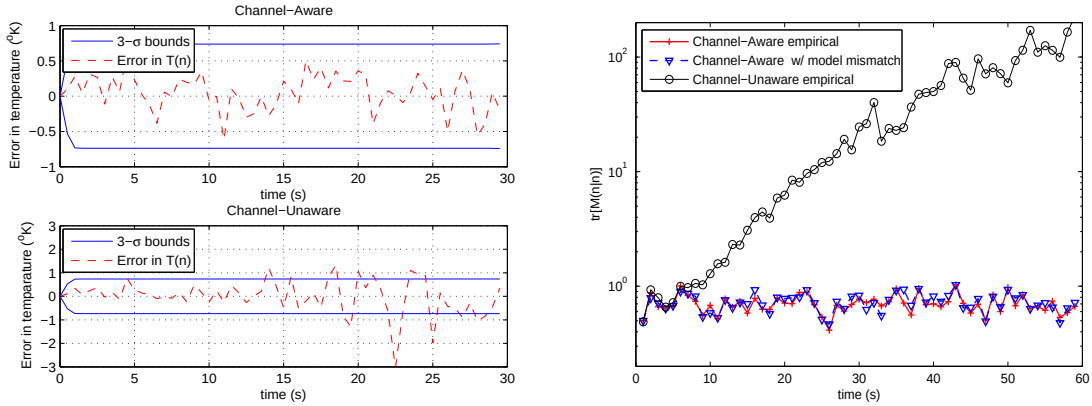


(a) (b)

Figure 2.3: Trace of the correction ECM versus time for (a) $N_\ell = 10$, and (b) $N_\ell = 5$.

equal to the all-one vector $\mathbf{1}$ and the second one $-\beta_\ell \mathbf{1}$, where β_ℓ is chosen randomly within the range $[0.1, 0.3]$; and $\mathbf{\Sigma}_{v_\ell v_\ell} = 2\mathbf{I}_\ell$ for $\ell = 1, \ldots, 50$. Each sensor S_l reduces the dimensionality of $\tilde{\mathbf{x}}_l(n|n-1)$ to $k_\ell = 1$. Furthermore, sensors here transmit only scalar data over scalar channels with $\mathbf{D}_\ell = 1$, while the reception noise variance $\mathbf{\Sigma}_{z_\ell z_\ell} = \sigma^2_{z_\ell}$ is set so that the $\text{SNR}_\ell := 10 \log_{10}(P_\ell / \sigma^2_{z_\ell})$ is equal to 20 dB for $\ell = 1, \ldots, L$. When the reduced dimension is $k_\ell = 1$, it turns out that the matrices obtained by either Proposition 2.1 for $L = 1$, or, Algorithm 1 have the form $\mathbf{C}^o_\ell(n) = c_{P,\ell}\mathbf{1}$, where $c_{P,\ell}$ is chosen to satisfy the power constraint in (2.7). This happens because the dominant eigenvector of $\mathbf{\Sigma}_{\tilde{x}_\ell \tilde{x}_\ell}(n)$ and $\mathbf{G}_{\tilde{x}}(n)$ in this case, introduced before Proposition 2.1, is the scaled all-one vector, i.e., $\frac{1}{\sqrt{N}}\mathbf{1}$. Thus, for this special case $\mathbf{C}^o_\ell(n)$ is a scaled all-one vector with the scaling constant determined by the prescribed transmit-power. Intuitively, this is expected since the sample average of the entries in $\tilde{\mathbf{x}}_\ell(n|n-1)$ is a sufficient statistic for estimating $\mathbf{s}(n)$ in the presence of Gaussian noise.

Figs. 2.3 (a) and (b) depict the MSE of the novel tracker along with that of [11], and the one corresponding to the clairvoyant KF which does not involve dimensionality reduction. The MSE is obtained from the trace of $\mathbf{M}(n|n)$ (theoretical), as well as through Monte Carlo simulations (empirical) for both $N_\ell = 10$ and $N_\ell = 5$ and $\{k_\ell = 1\}_{\ell=1}^L$. The comparison is done so that sensors utilize the same sampling rate both for the clairvoyant



(a) (b)

Figure 2.4: (a) Error in temperature and the 3–σ bounds versus time; and (b) MSE performance of in the presence of model mismatch.

KF and the reduced-dimensionality KF. Under this equal footing, each sensor acquires and transmits N_ℓ scalars in the clairvoyant setup; but when dimensionality-reduction is applied, each sensor acquires N_ℓ observations and compresses them down to a scalar. Thus, the required bandwidth in the clairvoyant KF is N_ℓ times larger that the one needed when dimensionality reduction is applied. Figs. 2.3 (a) and (b) demonstrate that increasing N_ℓ from 5 to 10 reduces the trace of ECM by approximately 13% (from 0.76 to 0.66), which quantifies the improvement in state estimation performance. However, the computational burden required per sensor increases since complexity is in the order of $\mathcal{O}(k_\ell N_\ell)$.

Moreover, the MSE in the novel channel-aware approach reaches steady-state, while in the channel-unaware approach [11] the MSE eventually diverges. This is expected since the channel-unaware approach does not account for noise when designing the dimensionality-reducing operators in (2.7). Interestingly, the tracking performance achieved by the proposed reduced-dimensionality KF is close to that enjoyed by the clairvoyant KF even when the compression ratio is 10. Despite the fact that the clairvoyant KF needs N_ℓ times more bandwidth than the reduced-dimensionality KF, the MSE amplification factor penalizing for latter scheme due to compression is much smaller than the compression ratio N_ℓ. This advocates that sensor observations are 'judiciously squeezed' without much information lost.

Similar conclusions can be drawn from Fig. 2.4(a) that depicts the error in temperature, namely $T(n) - \hat{T}(n|n)$, compared with the 3–σ bounds (see e.g., [5, Sec. 5.4]) for both channel-aware and channel-unaware approaches with $N_\ell = 10$. Notice that the estimation error in the channel-aware approach falls within the 3–σ bounds obtained by the reduced-dimensionality KF, thus corroborating its MSE optimality. However, the error associated with [11] does not comply with the 3–σ bounds obtained by the corresponding KF since it does not account for the presence of noise in the sensor-to-FC channels. Fig. 2.4(b) displays the empirically computed MSE achieved by our approach when the channel matrices $\mathbf{D}_\ell$ are not perfectly known at the FC. Here the reduced-dimensionality KF uses matrices $\hat{\mathbf{D}}_\ell$, where the perturbation is upper bounded as $\|\hat{\mathbf{D}}_\ell - \mathbf{D}_\ell\| \leq 0.5$ for $\ell = 1, \ldots, L$. Clearly, the proposed tracking scheme exhibits robustness to model mismatch.

2.4.2 Algorithm 3 for Ad Hoc WSNs

Target tracking based on distance-only measurements is a common problem in power-limited distributed estimation using WSNs (see e.g., [14]) for which estimation with low communication cost is particularly attractive. Consider $L = 100$ sensors randomly deployed uniformly in a square region of $2D \times 2D$ square meters, where $D = 2$ km, and suppose that using any of the existing localization algorithms, e.g., [33], every sensor has available position vectors $\mathbf{q}_\ell$ of all sensors in range.

The sensors are deployed to track the position $\mathbf{q}(n) := [q_{o1}(n), q_{o2}(n)]^T$ of a target. The initial position $\mathbf{q}(0)$ is randomly placed within 10 meters from the reference point $[0, 0]^T$. The associated state model used in the tracking process accounts for both the target position $\mathbf{q}(n)$ and its velocity $\mathbf{v}(n) := [v_1(n), v_2(n)]^T$ which is initialized as $\mathbf{v}(0) = [5,\ 3]^T$, but not for the acceleration which is lumped into the random noise term. The maneuvering target's position and velocity evolve according to

$$\mathbf{s}(n) := \begin{pmatrix} \mathbf{q}(n) \\ \mathbf{v}(n) \end{pmatrix} = \begin{pmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{q}(n-1) \\ \mathbf{v}(n-1) \end{pmatrix} + \mathbf{w}(n) \tag{2.52}$$

where T_s is the sampling period and the covariance matrix of $\mathbf{w}(n)$ is given by

$$\mathbf{\Sigma}_{ww}(n) = \begin{pmatrix} T_s^3/3 & 0 & T_s^2/2 & 0 \\ 0 & T_s^3/3 & 0 & T_s^2/2 \\ T_s^2/2 & 0 & T_s & 0 \\ 0 & T_s^2/2 & 0 & T_s \end{pmatrix} \sigma_w^2. \tag{2.53}$$

Consider a radar located far from the WSN, tracking a target moving in the direction of the field where sensors are deployed. Sensors observe the power of the signal return off the target that is modeled as

$$x_m(n) = \mathcal{P}/\|\mathbf{q}(n) - \mathbf{q}_m\|^2 + v(n) \tag{2.54}$$

where $\mathcal{P} = \mathcal{P}_R/\left(d_R^{\text{WSN}}\right)^2$ and depends on the transmission power of the radar $\mathcal{P}_R$, the distance between the radar and the WSN d_R^{WSN}, as well as the path-loss propagation law. Also, $\|\mathbf{q}(n) - \mathbf{q}_m\|$ denotes the distance between the target and the broadcasting sensor S_m, while $v(n)$ in (2.54) stands for zero-mean observation noise with variance σ_v^2.

Mimicking the steps applied to an extended (E)KF, see e.g., [25, Sec. 13.7], sensor S_ℓ linearizes (2.54) in the neighborhood of $\hat{\mathbf{q}}_\ell(n|n-1)$ to obtain

$$x_m(n) - x_m^o(n) \simeq \mathbf{h}_m^T(n)\mathbf{q}(n) + v(n) \tag{2.55}$$

where $\mathbf{h}_m(n) := -2\mathcal{P}\hat{\mathbf{q}}_\ell(n|n-1)/\|\hat{\mathbf{q}}_\ell(n|n-1) - \mathbf{q}_m\|^4$ and $x_m^o(n) = \mathcal{P}/\|\hat{\mathbf{q}}_\ell(n|n-1) - \mathbf{q}_m\|$. Note that (2.55) is a linear observation model with all the parameters locally available at S_ℓ.

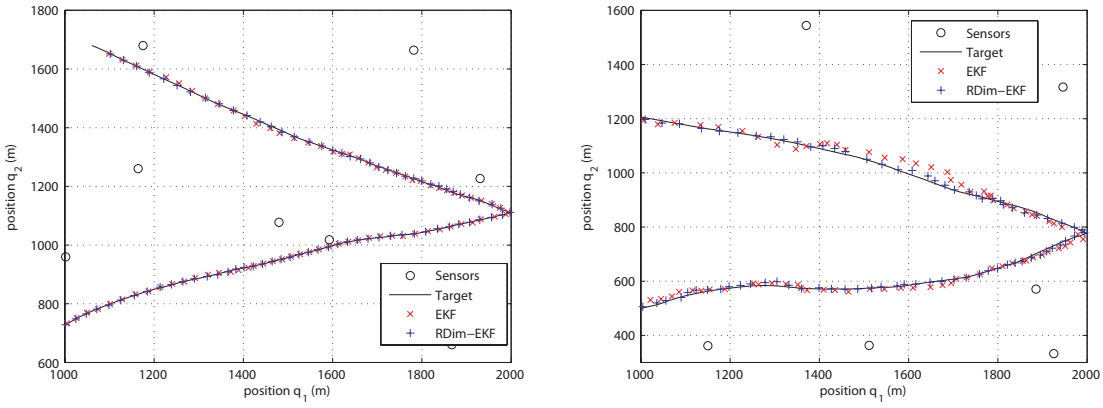
With the *linearized* model in (2.52) and (2.55) obeying (2.1)-(2.2), it is possible apply the KF scheme outlined in Algorithms 3–A and 3–B to track the target's position $\mathbf{q}(n)$. This amounts to an extended KF based on reduced-dimensionality data (abbreviated as RDim-EKF in the plots).

Similar to Section 2.4.1, the broadcasting sensor collects $N_m = 10$ samples successively in time and reduces their dimensionality to $k_m = 1$. With a sufficiently small sampling period, we can assume that the target position $\mathbf{q}(n)$ remains invariant during these N_ℓ successive sampling periods. Further, the order in which sensors broadcast is specified by how close

they are to the target. Specifically, the broadcasting sensor evaluates the Euclidean distance of each sensor from the target's estimated location, namely $\|\hat{\mathbf{q}}_m(n|n) - \mathbf{q}_\ell\|$, and selects the next one which is closer to the estimated sensor location. An overview of information-driven sensor selection schemes tailored for target tracking applications can be found in [57]. However, different from the signal model established in Section 2.1, the observation model (2.54) is a non-linear one and therefore the broadcasting sensor must rely on numerical methods to find $\mathbf{\Sigma}_{sx_m}(n)$ and $\mathbf{\Sigma}_{x_m x_m}(n)$ that are necessary for obtaining $\mathbf{C}_m(n)$ when links are non-ideal. The reception quality at sensor S_ℓ from sensor S_m is characterized by the SNR $SNR_\ell^m = (1/\|\mathbf{q}_m - \mathbf{q}_\ell\|^2)\overline{SNR}$, where: (i) $1/\|\mathbf{q}_m - \mathbf{q}_\ell\|^2$ accounts for the path-loss since S_m and S_ℓ have inter-sensor distance $\|\mathbf{q}_m - \mathbf{q}_\ell\|$; and (ii) $\overline{SNR}$ is the reception SNR without path-loss that is set to 20dB.

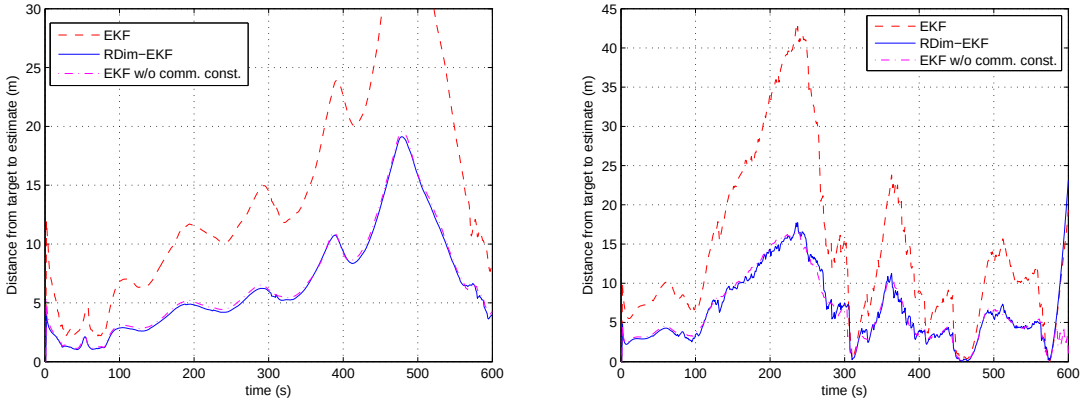
Next, the tracking performance of RDim-EKF is compared against the clairvoyant EKF. The parameters are set to be $\sigma_w = 0.15m/s^2$, $\sigma_v = 1$, and $\overline{SNR} = 20dB$. Note that the clairvoyant EKF entails no dimensionality reduction but for fairness the communication cost as well as the communication rate are kept equal as in Algorithm 3. To ensure identical communication cost each broadcasting sensor in EKF acquires one sample and broadcasts it to the other sensors, while during the same interval in RDim-EKF each sensor acquires N_m samples and implements Algorithm 3 to only broadcast a scalar sample to the rest of the sensors.

Figs. 2.5(a) and 2.5(b) depict the true and estimated target trajectories obtained by Algorithm 3 and the EKF at a specific sensor. Figs. 2.6(a) and 2.6(b) show that the standard deviation of the estimation error associated with Algorithm 3 is in the order of $2m - 15m$ for ideal links, and $5m - 20m$ for non-ideal ones, both of which are much smaller than those of the clairvoyant EKF, and almost identical to the EKF without the communication rate constraint. This confirms that the former outperforms the latter for the same communication cost. This should be expected because the information broadcasted in Algorithm 3 is more informative about the target position than in EKF, since it is constructed by judicious dimensionality reduction of multiple, namely N_ℓ, observations. Similar conclusions can be drawn from Figs. 2.7(a) and 2.7(b), where clearly Algorithm



(a) (b)

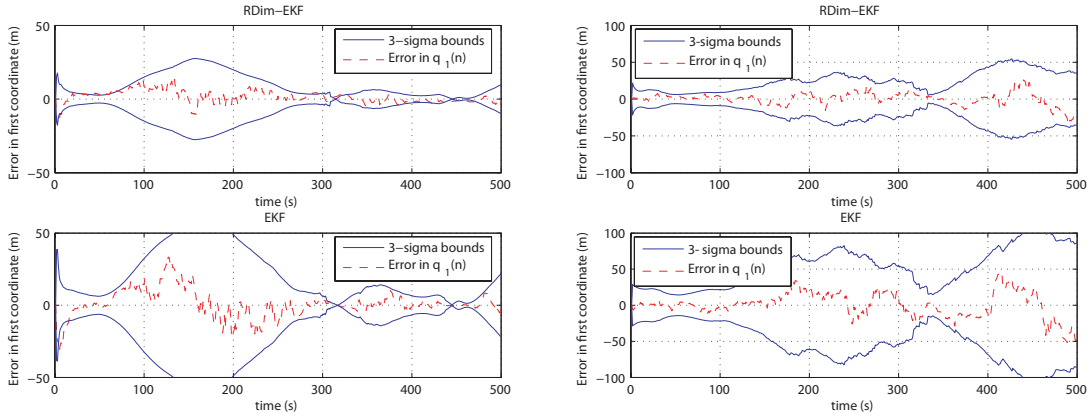
Figure 2.5: Target tracking with EKF and reduced-dimensionality EKF under: (a) ideal sensor links, and (b) non-ideal links.



(a) (b)

Figure 2.6: Standard deviation of the RDim-EKF estimate versus time for (a) ideal links, and (b) non-ideal links.

3 outperforms the EKF, while the corresponding estimation errors of both fall within the corresponding 3–σ bounds. Notice that in Figs. 2.6(a) and 2.6(b) the estimation error's standard deviation for the EKF with no communication rate constraint is also plotted as a baseline. In this case, each sensor transmits 10 scalars per time slot n, instead of 1 as in



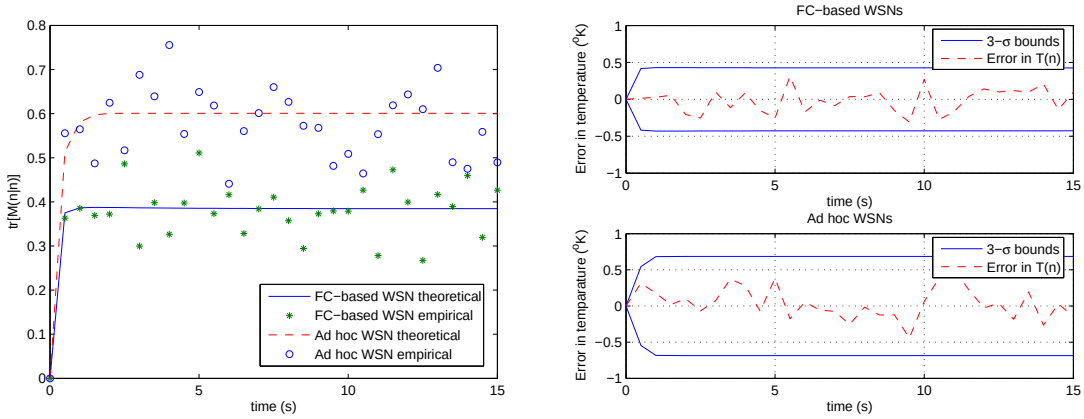
(a) (b)

Figure 2.7: Position error $q_1(n) - \hat{q}_1(n|n)$ and its 3–σ bounds versus time under (a) ideal, and (b) non-ideal links.

RD-EKF. It can be seen that the RDim-EKF has almost the same performance with the EKF when no communication rate constraint is imposed. Intuitively, this is expected since for the Gaussian observation noise setup here, a sufficient statistic for estimating $\mathbf{s}(n)$ will be the sample average of each sensor's N_ℓ observations. Thus, dimensionality reduction is expected to introduce a negligible performance loss in MSE.

2.4.3 Comparisons of Algorithms 2 and 3

Next, the tracking performance of Algorithms 2 and 3 is compared in order to gauge the loss in estimation MSE when giving more emphasis to robustness. The system model is similar to the one in Section 2.4.1 and both the FC-based and ad hoc WSNs have $L = 5$ sensors. Each sensor acquires data vectors of size $N_\ell = 10$ and reduces them to $k_\ell = 1$. Note that within each iteration of Algorithm 2 the FC receives data from all five sensors. For fairness in comparison (same communication rate per sensor) all sensors in the ad hoc WSN broadcast in a round-robin fashion during one iteration of Algorithm 3. The available transmission power per sensor is P for the FC-based and $P/5$ for the ad hoc WSN. As a result, Algorithm 2 consumes 5 times more power than Algorithm 3 per iteration. Fading is not incorporated and reception noise variance is set so that SNR= 20 dB for the sensor-



(a) (b)

Figure 2.8: (a) Trace of the correction ECM for versus time; (b) Error in temperature and the corresponding 3–σ bounds versus time.

to-FC links. Fig. 2.8(a) depicts the estimation MSE (both theoretical and empirical) corresponding to Algorithms 2 and 3. The tracking performance of Algorithm 2 in FC-based WSNs as expected outperforms Algorithm 3 in the ad hoc setup. The performance loss experienced by Algorithm 3 is roughly 50%. However, recall that in every step of Algorithm 2 the power consumption is $5P$, while in Algorithm 3 it is P over the same bandwidth. This implies that Algorithm 3 consumes 80% less total power than Algorithm 2, with the first leading to improved estimation performance and the second exhibiting improved robustness and lower power consumption. Similar conclusions can be drawn from Fig. 2.8(b).

2.5 Appendices

2.5.1 Proof of Lemma 2.1

The predictor of $\mathbf{y}_\ell(n)$ based on past observations can be written as [cf. (a2.1)]

$$\begin{aligned}\hat{\mathbf{y}}_\ell(n|n-1) =& E\big[\mathbf{D}_\ell\mathbf{C}_\ell(n)[\mathbf{x}_\ell(n)-\hat{\mathbf{x}}_\ell(n|n-1)]\Big|\mathbf{y}(0),\ldots,\mathbf{y}(n-1)\big]\\ =&\mathbf{D}_\ell\mathbf{C}_\ell(n)[\hat{\mathbf{x}}_\ell(n|n-1)-\hat{\mathbf{x}}_\ell(n|n-1)]=\mathbf{0}\end{aligned} \tag{2.56}$$

where for the last equality we used that the predictor of $\mathbf{x}_\ell(n)$ is given by $\hat{\mathbf{x}}_\ell(n|n-1) = E[\hat{\mathbf{x}}_\ell(n|n-1)|\mathbf{y}(0),\ldots,\mathbf{y}(n-1)]$. □

2.5.2 Proof of Corollary 2.1

Upon setting the channel matrices $\mathbf{D}_1 = \mathbf{I}_{k_1}$ and $\mathbf{\Sigma}_{z_1z_1} = \mathbf{0}$, it follows readily that $\mathbf{Q}_{zd} = \mathbf{I}_{k_1}$ and $\mathbf{\Lambda}_{zd}^{-1} = \mathbf{0}$; thus, $\lambda_{zd,i}^{-1} = 0$ for $i = 1,\ldots,k_1$. The fact that the power inequality in (2.24) holds for any $\kappa \in [1, k_1]$, implies that $\text{rank}(\mathbf{\Phi}_C^o) = \kappa = k_1$ (full rank). An inspection of the Lagrange function reveals that the diagonal entries of the MSE optimal operator $\mathbf{\Phi}_C^o$ are

$$\phi_{c,ii}^o = \pm\frac{P_1(\lambda_{g\tilde{x},i}(n))^{1/2}}{\sum_{j=1}^{k_1}(\lambda_{g\tilde{x},j}(n))^{1/2}},\quad i=1,\ldots,k_1. \tag{2.57}$$

Next, it is shown that the matrices obtained in Proposition 2.1 coincide with these in Proposition 2.3. To this end, let $\mathbf{\Phi}^o \in \mathbf{R}^{k_1\times k_1}$ be the diagonal matrix whose ith diagonal entry is equal to $\phi_{c,ii}^o$ in (2.57), while $\mathbf{I}_{k_1\times N_1}$ denotes the $k_1 \times N_1$ matrix obtained from the $N_1 \times N_1$ identity matrix after removing its last $N_1 - k_1$ rows. Thus, matrix $\mathbf{\Phi}_C^o$ can be rewritten as

$$\mathbf{\Phi}_C^o = \mathbf{\Phi}^o\mathbf{I}_{k_1\times N_1}. \tag{2.58}$$

Substituting the matrices $\mathbf{\Lambda}_{zd}^{-1}$ and $\mathbf{Q}_{zd}$ in (2.21) as specified earlier we obtain that

$$\mathbf{B}^o(n) = \mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{Q}_{\tilde{x}}(n)\mathbf{\Lambda}_{\tilde{x}}^{-1/2}(n)\mathbf{Q}_{g\tilde{x}}(n)\mathbf{I}_{k_1\times N_1}^T(\mathbf{\Phi}^o)^{-1}. \tag{2.59}$$

Notice that the right and left eigenvector matrices of $\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{Q}_{\tilde{x}}(n)\mathbf{\Lambda}_{\tilde{x}}^{-1/2}(n)$ coincide with the unitary matrices $\mathbf{Q}_{g\tilde{x}}(n)$ and $\mathbf{Q}_{\tilde{s}\tilde{x}_1}(n)$, respectively. Further, let $\mathbf{S}_{p\times N_1}$ denote the diagonal matrix containing the singular values of $\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{Q}_{\tilde{x}}(n)\mathbf{\Lambda}_{\tilde{x}}^{-1/2}(n)$. Using these properties

it follows that

$$\mathbf{B}^o(n)=\mathbf{Q}_{\tilde{s}\tilde{x}_1}(n)\mathbf{S}_{p\times k_1}\mathbf{I}^T_{k_1\times N_1}(\mathbf{\Phi}^o)^{-1}=\mathbf{Q}_{\tilde{s}\tilde{x}_1}(n)\mathbf{I}^T_{k_1\times p}\mathbf{S}_{k_1\times k_1}(\mathbf{\Phi}^o)^{-1}=\mathbf{Q}_{\tilde{s}\tilde{x}_1,k_1}(n)\mathbf{S}_{k_1\times k_1}(\mathbf{\Phi}^o)^{-1} \tag{2.60}$$

where $\mathbf{S}_{k_1\times k_1}$ denotes the $k_1\times k_1$ upper left part of $\mathbf{S}_{p\times N_1}$. From (2.60) it follows readily that $\mathbf{B}^o(n)$ obtained from Proposition 2.1 coincides with the one in Proposition 2.3 after setting $\mathbf{F}=(\mathbf{\Phi}^o)\mathbf{S}^{-1}_{k_1\times k_1}$. Next, it is shown that the optimal compression matrix $\mathbf{C}^o(n)$ provided by CCA coincides with the one in Proposition 2.3 when sensor links are ideal. To this end, $\mathbf{C}^o(n)$ in (2.21) can be re-expressed as

$$\begin{aligned}\mathbf{C}^o(n)&=\mathbf{\Phi}^o\mathbf{I}_{k_1\times N_1}\mathbf{Q}^T_{g\tilde{x}}(n)\mathbf{\Lambda}^{1/2}_{\tilde{x}}(n)\mathbf{Q}^T_{\tilde{x}}(n)\mathbf{\Sigma}^{-1}_{\tilde{x}_1\tilde{x}_1}(n)\\&=\mathbf{F}(\mathbf{S}_{k_1\times k_1}\mathbf{I}_{k_1\times N_1}\mathbf{Q}^T_{g\tilde{x}}(n)\mathbf{\Lambda}^{1/2}_{\tilde{x}}(n)\mathbf{Q}^T_{\tilde{x}}(n)\mathbf{\Sigma}^{-1}_{\tilde{x}_1\tilde{x}_1}(n))\end{aligned} \tag{2.61}$$

where $\mathbf{F}=(\mathbf{\Phi}^o)\mathbf{S}^{-1}_{k_1\times k_1}$. Now, recall: (i) that $\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{Q}_{\tilde{x}}(n)\mathbf{\Lambda}^{-1/2}_{\tilde{x}}(n)=\mathbf{Q}_{\tilde{s}\tilde{x}_1}(n)\mathbf{S}_{p\times N_1}\mathbf{Q}^T_{g\tilde{x}}(n)$ and (ii) that $\mathbf{S}_{k_1\times k_1}\mathbf{I}_{k_1\times N_1}=\mathbf{I}_{k_1\times p}\mathbf{S}_{p\times N_1}$. Using the latter, (2.61) can be re-written as

$$\begin{aligned}\mathbf{C}^o(n)&=\mathbf{F}(\mathbf{I}_{k_1\times p}\mathbf{Q}^T_{\tilde{s}\tilde{x}_1}(n)\mathbf{Q}_{\tilde{s}\tilde{x}_1}(n)\mathbf{S}_{p\times N_1}\mathbf{Q}^T_{g\tilde{x}}(n)\mathbf{\Lambda}^{1/2}_{\tilde{x}}(n)\mathbf{Q}^T_{\tilde{x}}(n)\mathbf{\Sigma}^{-1}_{\tilde{x}_1\tilde{x}_1}) \qquad (2.62)\\&=\mathbf{F}\mathbf{Q}^T_{\tilde{s}\tilde{x}_1,k_1}(n)\mathbf{\Sigma}_{\tilde{s}\tilde{x}_1}(n)\mathbf{\Sigma}^{-1}_{\tilde{x}_1\tilde{x}_1}(n).\end{aligned}$$ □

2.5.3 Proof of Equation (2.38)

Recall from (2.37) that $\mathbf{y}_{\bar{\ell}}(n)=\mathbf{C}_m(n)\tilde{\mathbf{x}}_m(n|n-1)$, $\forall\ell\in[1,L]$. Adding and subtracting $\tilde{\mathbf{s}}(n|n-1)$ inside the Euclidean norm of the cost in (2.12) yields

$$\begin{aligned}&E[\|\mathbf{s}(n)-\mathbf{B}(n)\mathbf{y}_{\bar{\ell}}(n)\|^2]\\=&E[\|\mathbf{s}(n)-\tilde{\mathbf{s}}(n|n-1)+\tilde{\mathbf{s}}(n|n-1)-\mathbf{B}(n)\mathbf{C}_m(n)\tilde{\mathbf{x}}_m(n|n-1)\|^2]\\=&E[\|\hat{\mathbf{s}}(n|n-1)\|^2]+E[\|\tilde{\mathbf{s}}(n|n-1)-\mathbf{B}(n)\mathbf{C}_m(n)\tilde{\mathbf{x}}_m(n|n-1)\|^2]\end{aligned} \tag{2.63}$$

where the second equality stems from the fact that the predictor $\hat{\mathbf{s}}(n|n-1)$ is uncorrelated with the innovation signals $\tilde{\mathbf{s}}(n|n-1)$ and $\tilde{\mathbf{x}}_m(n|n-1)$. Since $E[\|\hat{\mathbf{s}}(n|n-1)\|^2]$ does not depend on $\mathbf{B}(n)$ and $\mathbf{C}_m(n)$, the first term in (2.63) can be dropped leading to the MSE cost given in (2.38). □

Chapter 3

Distributed In-Network Channel Decoding

Average log-likelihood ratios (LLRs) constitute sufficient statistics for centralized maximum-likelihood block decoding as well as for *a posteriori* probability evaluation which enables bit-wise (possibly iterative) decoding. By acquiring such average LLRs per sensor it becomes possible to perform these decoding tasks in a low-complexity *distributed* fashion using wireless sensor networks. At affordable communication overhead, the resultant distributed decoders rely on local message exchanges among single-hop neighboring sensors to achieve iteratively consensus on the average LLRs per sensor. Furthermore, the decoders exhibit robustness to non-ideal inter-sensor links affected by additive noise and random link failures. Pairwise error probability bounds benchmark the decoding performance as a function of the number of consensus iterations. Interestingly, simulated tests corroborating the analytical findings demonstrate that only a few consensus iterations suffice for the novel distributed decoders to approach the performance of their centralized counterparts.

3.1 Modeling and Problem Statement

Consider the access point (AP) depicted in Fig. 3.1 broadcasting an $N \times 1$ coded block $\mathbf{x}$ to a set of J wireless low-cost, power-limited sensors. The WSN is modeled as a graph

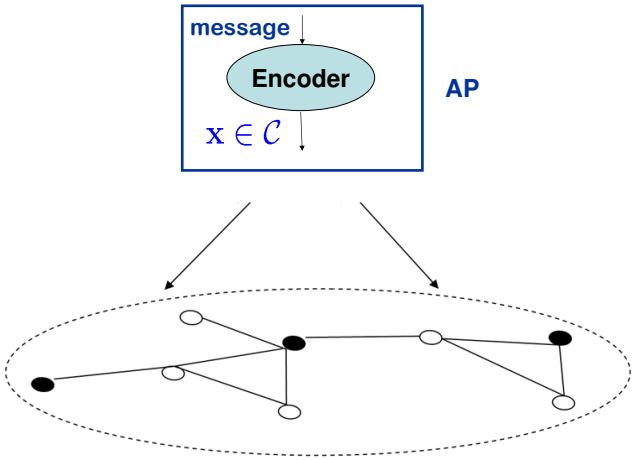


Figure 3.1: Cooperative decoding system model.

$\mathcal{G} := \{\mathcal{E}, \mathcal{J}\}$, where $\mathcal{J} := \{1, \ldots, J\}$ denotes the set of sensors, and $\mathcal{E} \subset \mathcal{J} \times \mathcal{J}$ the set of available communication links (graph edges). The set of neighbors of a given sensor j is denoted by $\mathcal{N}_j \subseteq \mathcal{J}$. Graph connectivity is captured by the $J \times J$ adjacency matrix $\mathbf{A}$, where $[\mathbf{A}]_{ji} = 1$ if $i \in \mathcal{N}_j$, otherwise $[\mathbf{A}]_{ji} = 0$. The graph Laplacian matrix is defined as $\mathbf{L} := \mathbf{D} - \mathbf{A}$, where $\mathbf{D} := \text{diag}(|\mathcal{N}_1|, \ldots, |\mathcal{N}_J|)$ is the degree matrix.

Let $\mathbf{y}_j$ denote the $N \times 1$ block of symbols received at sensor j. The system operates under the following assumptions:

(a3.1) *The WSN is connected, i.e., there is a (possibly multi-hop) path connecting any two nodes;*

(a3.2) *Sensors communicate with single-hop neighbors over symmetric links that can be:*

(a3.2a) *ideal and time-invariant per-block; or*

(a3.2b) *time-invariant corrupted by additive zero-mean noise with variance σ_i^2 at sensor $i \in \mathcal{N}_j$, and uncorrelated across j and i; or*

(a3.2c) *time-varying in the sense that the link between neighboring sensors j and i is allowed to fail with probability $1 - p_{ji}$, where $0 < p_{ji} < 1$.*

(a3.3) *Entry x_n of $\mathbf{x}$ is binary $\{0,1\}$ and all sensors know the codebook $\mathcal{C}$ that $\mathbf{x}$ belongs to;*

(a3.4) *AP-sensor channels are discrete, memoryless, symmetric, and conditionally independent across sensors; and sensor j knows the conditional pdf $p[\mathbf{y}_j|\mathbf{x}]$ characterizing its channel with the AP.*

Note that the connectivity in (a3.1) need not be strong, meaning that a sensor must have just enough power to reach single-hop neighbors but not all other sensors in the network. The ideal (virtually error-free) inter-sensor communications under (a3.2a) are possible with sufficiently powerful error control codes; and are considered to benchmark the imperfect link performance allowed under (a3.2b) and (a3.2c). Similar to [45], the noisy inter-sensor link model (a3.2b) includes quantization errors and receiver noise. The random failures considered in (a3.2c) encompass: (i) communication among sensors using error-detection codes such as cyclic redundancy check (CRC) codes; and (ii) fading channels. Link failures under (a3.2c) are different from sensor failures. The former allow for link recovery since $p_{ij} > 0$, whereas the latter are associated with permanent link failures. Binary codewords are assumed in (a3.3) for simplicity in exposition, but generalizations to any finite alphabet are possible as shown in Section 3.2. Channels with finite memory are also covered by (a3.4) provided that e.g., multi-carrier modulation is employed [35, Ch. 11.2.3]. Channel knowledge in (a3.4) is assumed acquired via training. The AP-sensor link models include the binary symmetric channel, case in which the probability in (a3.4) depends only on the cross-over probability when $x_n \in \{0,1\}$; and the binary phase shift keying (BPSK) transmission over the additive white Gaussian noise (AWGN) channel, case in which $\mathbf{y}_j = \sqrt{E_s}(2\mathbf{x} - \mathbf{1}_N) + \boldsymbol{\epsilon}_j$, $j \in \mathcal{J}$, where E_s denotes average symbol energy and $\boldsymbol{\epsilon}_j$ the AWGN.

Given $\mathbf{y}_j$ per sensor j, the objective is to find under (a3.1)-(a3.4) and without a fusion center *distributed* solutions of:

(P1) *the centralized maximum-likelihood (ML) block decoder*

$$\hat{\mathbf{x}}_{ML} = \arg\max_{\mathbf{x}\in\mathcal{C}} p[\mathbf{y}_1, \ldots, \mathbf{y}_J|\mathbf{x}]; \tag{3.1}$$

(P2) *the centralized a posteriori probability (APP) evaluator*

$$\Pr[x_n|\mathbf{y}_1,\ldots,\mathbf{y}_J] \quad \forall n=1,\ldots,N. \tag{3.2}$$

The ultimate goal is to obtain tools such as *distributed* versions of the Viterbi algorithm (DiVA) for ML decoding and the BCJR algorithm (DiBCJR) for APP evaluation to enable (possibly iterative) decoding via in-network message exchanges among sensors. It is also desirable to have such algorithms incur low computational cost and affordable overhead in inter-sensor communications. Furthermore, the design must be geared toward efficient operation in the power-limited regime that is typical of WSN applications; i.e., maximizing data rates is not of prime interest here.

Remark 3.1 *(Decoding as hypothesis testing)* Due to the conditional independence in (a3.4), ML decoding of equiprobable codewords amounts to a multiple hypotheses test aiming to maximize the sum of per-sensor log-likelihoods, i.e., $\hat{\mathbf{x}}_{ML} = \arg\max_{\mathbf{x}\in\mathcal{C}} \sum_{j=1}^{J} \log p(\mathbf{y}_j|\mathbf{x})$. Taking linear (N,K) binary block codes as an example, the number of hypotheses equals the cardinality of the codebook $|\mathcal{C}| = 2^K$; or, generally $|\mathcal{C}| = Q^K$ for a finite alphabet of size Q. Viewed as a test of multiple hypotheses, distributed ML decoding with centralized performance is possible so long as each sensor can acquire the average log-likelihood over all sensors *for each candidate* $\mathbf{x}\in\mathcal{C}$. To obtain this average, belief consensus and belief propagation approaches were developed in [30] and [43] in the context of distributed detection. Belief consensus in [30] relies on consensus averaging (CA) – a scheme orginally developed in [31, 54] for distributed estimation of sample averages. Relative to the aforementioned contributions, the present chapter exploits the special structure of the channel decoding problem and aims at low-communication-complexity solutions with quantifiable error performance and pragmatic inter-sensor communication links.

3.2 Consensus-based Distributed Decoding

In this section, (P1) and (P2) are cast in equivalent forms amenable to distributed implementation. Since AP-sensor channels are memoryless and independent across sensors [cf.

(a3.4)], the log-likelihood of $\{\mathbf{y}_j\}_{j=1}^J$ given $\mathbf{x}$ is

$$\log p[\mathbf{y}_1, \ldots, \mathbf{y}_J|\mathbf{x}] = \sum_{j=1}^{J} \log p[\mathbf{y}_j|\mathbf{x}] = \sum_{j=1}^{J}\sum_{n=1}^{N} \log p[y_{jn}|x_n]. \tag{3.3}$$

Furthermore, with $x_n \in \{0, 1\}$, each summand in (3.3) can be written as

$$\log p[y_{jn}|x_n] = -x_n \log\left(\frac{p[y_{jn}|x_n = 0]}{p[y_{jn}|x_n = 1]}\right) + \log p[y_{jn}|x_n = 0] \tag{3.4}$$

where clearly the term $\log p[y_{jn}|x_n = 0]$ is constant with respect to (w.r.t.) the variable x_n. Upon defining the local log-likelihood ratio (LLR) corresponding to the n-th bit of the codeword at sensor j as

$$\gamma_{jn} := \log\left(\frac{p[y_{jn}|x_n = 0]}{p[y_{jn}|x_n = 1]}\right) \tag{3.5}$$

the centralized ML decoding problem in (3.1) is equivalent to

$$\hat{\mathbf{x}}_{ML} = \arg\min_{\mathbf{x}\in\mathcal{C}} \sum_{j=1}^{J}\sum_{n=1}^{N} \gamma_{jn}x_n = \arg\min_{\mathbf{x}\in\mathcal{C}} \bar{\boldsymbol{\gamma}}^T\mathbf{x} \tag{3.6}$$

where

$$\bar{\boldsymbol{\gamma}} := \frac{1}{J}\sum_{j=1}^{J} \boldsymbol{\gamma}_j \tag{3.7}$$

and the vector $\boldsymbol{\gamma}_j := [\gamma_{j1}, \ldots, \gamma_{jN}]^T$ collects the N scalar local LLRs per sensor j.

Equation (3.6) reveals that ML optimal decoding amounts to a linear program [16]. But since $\mathcal{C}$ is available to all sensors [cf. (a3.3)], to solve (3.6) in a distributed fashion it suffices for each sensor to acquire the $N \times 1$ average LLR vector $\bar{\boldsymbol{\gamma}}$. Correspondingly, Appendix 3.6.1 shows that for a general alphabet of size Q each sensor must consent to $(Q-1)$ such average LLR vectors (one per pair of alphabet values). It is worth stressing though that, unlike [30, 43], the communication complexity here does not depend on the codebook size $|\mathcal{C}| = Q^K$. The next example provides further insight on this point.

Example 3.1 *(Sufficient consensus with AWGN AP-sensor channels.)* Consider the AWGN model $\mathbf{y}_j = \sqrt{E_s}(2\mathbf{x} - \mathbf{1}_N) + \boldsymbol{\epsilon}_j$, where $\boldsymbol{\epsilon}_j \sim \mathcal{N}(0, N_{0,j}\mathbf{I}_N/2)$. The ML decoding problem in this special case amounts to

$$\hat{\mathbf{x}}_{ML} = \arg\max_{\mathbf{x}\in\mathcal{C}} \left(\sum_{j=1}^{J} \frac{2\sqrt{E_s}\mathbf{y}_j}{JN_{0,j}}\right)^T (2\mathbf{x} - \mathbf{1}_N) - \left(\sum_{j=1}^{J} \frac{E_s}{JN_{0,j}}\right) \|2\mathbf{x} - \mathbf{1}_N\|^2. \tag{3.8}$$

Since $(2x_n - 1)$ takes values $\{-1, 1\}$, the term $\|2\mathbf{x} - \mathbf{1}_N\|^2$ in (3.8) is constant and thus irrelevant to the maximization. Upon comparing (3.8) with (3.6), the average LLR here is $\bar{\boldsymbol{\gamma}} = -\frac{1}{J}\sum_{j=1}^{J} \frac{4\sqrt{E_s}\mathbf{y}_j}{N_{0,j}}$. This is reasonable since the weighted average data in the AWGN model constitute sufficient statistics for ML decoding [35, Ch. 5]. Notice that the local receive SNR is $\frac{2E_s}{N_{0,j}}$, whereas the global receive SNR after maximum-ratio combining is $\sum_{j=1}^{J} \frac{2E_s}{N_{0,j}}$, which is precisely the SNR of $\bar{\gamma}_n$, $\forall n$. Intuitively, if sensors acquire in a distributed fashion their weighted average data, they can subsequently carry out ML decoding with their local SNR boosted to its global value – thanks to averaging – that is otherwise reachable only when all data $\{\mathbf{y}_j\}_{j=1}^{J}$ are centrally available.

Let us now turn our attention from ML block decoding to the APP evaluation task in (P2). In this problem, sensor $j \in \mathcal{J}$ knows $p[\mathbf{y}_j|\mathbf{x}]$ and seeks $\Pr[x_n|\mathbf{y}_1, \ldots, \mathbf{y}_J]$ for $n = 1, \ldots, N$. The latter is completely determined by the following ratio (since numerator and denominator must sum up to one)

$$l_n := \frac{\Pr[x_n = 0|\mathbf{y}_1, \ldots, \mathbf{y}_J]}{\Pr[x_n = 1|\mathbf{y}_1, \ldots, \mathbf{y}_J]} = \frac{\sum_{\mathbf{x}\in\mathcal{C}_n^0} p[\mathbf{y}_1, \ldots, \mathbf{y}_J|\mathbf{x}]}{\sum_{\mathbf{x}\in\mathcal{C}_n^1} p[\mathbf{y}_1, \ldots, \mathbf{y}_J|\mathbf{x}]} \tag{3.9}$$

$$= \frac{\sum_{\mathbf{x}\in\mathcal{C}_n^0} \exp\left[-\left(\sum_{j=1}^{J} \boldsymbol{\gamma}_j\right)^T \mathbf{x}\right]}{\sum_{\mathbf{x}\in\mathcal{C}_n^1} \exp\left[-\left(\sum_{j=1}^{J} \boldsymbol{\gamma}_j\right)^T \mathbf{x}\right]} = \frac{\sum_{\mathbf{x}\in\mathcal{C}_n^0} \exp\left[-J\bar{\boldsymbol{\gamma}}^T\mathbf{x}\right]}{\sum_{\mathbf{x}\in\mathcal{C}_n^1} \exp\left[-J\bar{\boldsymbol{\gamma}}^T\mathbf{x}\right]} \tag{3.10}$$

where $\mathcal{C}_n^i := \{\mathbf{x} \in \mathcal{C}|x_n = i\}$ for $i \in \{0, 1\}$ is the set of codewords in $\mathcal{C}$ whose n-th entry is equal to i; and equation (3.9) is due to Bayes' theorem while (3.10) follows from (3.3)-(3.5). Again, (3.10) reveals that if each sensor can acquire the average LLR vector $\bar{\boldsymbol{\gamma}}$ in a distributed fashion, distributed APP evaluation is also possible regardless of the codebook size $|\mathcal{C}| = Q^K$. Summarizing, it has been established that:

Proposition 3.1 *For distributed ML decoding and APP evaluation tasks to attain performance of their centralized counterparts under (a3.1)-(a3.4), it suffices for each sensor to obtain, i.e., consent on the average LLR vector $\bar{\boldsymbol{\gamma}}$ – a task incurring linear communication complexity in the codeword size N.*

The average LLR is the sufficient statistic for a number of centralized decoding algorithms [42, Ch. 4]. The major implication of Proposition 3.1 is that acquiring the average

LLR in a distributed fashion gives rise to distributed versions of these algorithms: DiVA, DiBCJR, and distributed iterative (LDPC or turbo) decoding algorithms. The key enabler of these schemes is distributed evaluation of the average LLR, the topic dealt with in the ensuing section.

3.3 Consensus on average LLR

The major issue in Proposition 3.1 is whether the average LLR vector in (3.7) can indeed be found distributedly even when sensors can only communicate with one-hop neighbors. This will be shown possible through iterative consensus schemes tailored for the setup described by (a3.1)-(a3.4). The resulting algorithms will generate (local) iterates $\bar{\gamma}_{jn}(k)$, which can be viewed as *local estimates* of the n-th entry $\bar{\gamma}_n$ of $\bar{\boldsymbol{\gamma}}$ in (3.7) per sensor j and iteration k. These local estimates will be required to converge to $\bar{\gamma}_n$. But more importantly, if the iterations stop before finding the exact $\bar{\gamma}_n$, each sensor will be allowed to decode using the estimate available up to that instant. Therefore, an approximate sufficient statistic for (P1) and (P2) will become available per sensor j and iteration k.

If the network operates under (a3.2a) or (a3.2c), the average $\bar{\boldsymbol{\gamma}}$ in (3.7) can be found in a distributed fashion by applying the CA algorithm of [31, 36, 54], which is based on a single iterative (SI) equation involving $\bar{\gamma}_{jn}(k)$. However, when inter-sensor links suffer from noise as in (a3.2b), these algorithms fail to converge. The solution advocated in [20] enables convergence at the price of considerably reducing convergence speed. Motivated by these facts, this chapter will introduce novel CA alternatives based on the method of multipliers (CA-MoM), in which the average value is obtained as the iterative solution of a convex optimization problem. Modified iterations of the CA-MoM will be further developed to cope with random link failures in (a3.2c).

3.3.1 CA-SI Algorithms

With its local information $\{\gamma_{jn}\}_{n=1}^{N}$ available, sensor j seeks the global $\bar{\boldsymbol{\gamma}}$ in (3.7); i.e., $\bar{\gamma}_n = \frac{1}{J}\sum_{j=1}^{J}\gamma_{jn}$ for $n = 1, \ldots, N$. The CA-SI algorithm yields iterates $\bar{\gamma}_{jn}(k)$ per sensor j

converging to the desired $\bar{\gamma}_n \ \forall n$, as $k \to \infty$. In the present context, these iterates per sensor j exploit LLR information from single-hop neighboring sensors $\mathcal{N}_j$ through the following linear recursion

$$\bar{\gamma}_{jn}(k) = \bar{\gamma}_{jn}(k-1) + \sum_{i \in \mathcal{N}_j} W_{ji}[\bar{\gamma}_{in}(k-1) - \bar{\gamma}_{jn}(k-1)] \ , \quad j \in \mathcal{J}, \quad n = 1, \ldots, N. \quad (3.11)$$

With the initialization $\bar{\gamma}_{jn}(0) = \gamma_{jn}$ and properly selected weights W_{ji}, convergence to $\bar{\gamma}_n$ $\forall j \in \mathcal{J}$ as $k \to \infty$ is ensured [54]. With $\bar{\boldsymbol{\gamma}}_n(k) := [\bar{\gamma}_{1n}(k), \ldots, \bar{\gamma}_{Jn}(k)]^T$ and $W_{ji} = 0$ for $i \notin \mathcal{N}_j$, concatenating (3.11) for $j = 1, \ldots, J$, yields the vector recursion $\bar{\boldsymbol{\gamma}}_n(k) = \mathbf{W}\bar{\boldsymbol{\gamma}}_n(k-1)$, where $[\mathbf{W}]_{ji \in \mathcal{J}} = W_{ji}$. The weight matrix $\mathbf{W}$ guarantees convergence if [54]: (i) $\mathbf{1}^T\mathbf{W} = \mathbf{1}^T$; (ii) $\mathbf{W}\mathbf{1} = \mathbf{1}$; and (iii) $\rho(\mathbf{W} - (1/J)\mathbf{1}\mathbf{1}^T) < 1$, where $\rho(\cdot)$ denotes the spectral radius.

The iterations in (3.11) are known to exhibit resilience to random link failures [36]. However, if additive noise is present when neighboring sensors exchange $\bar{\gamma}_{jn}(k)$, the variance of $\bar{\gamma}_{jn}(k)$ in (3.11) grows unbounded with k [20]. To avoid the latter, [20] advocated scaling the correction term in (3.11) by a vanishing stepsize $\alpha(k)$; see also [3, 34, 38]. In this case, (3.11) is replaced by

$$\bar{\gamma}_{jn}(k) = \bar{\gamma}_{jn}(k-1) + \alpha(k) \sum_{i \in \mathcal{N}_j} W_{ji}[\bar{\gamma}_{in}(k-1) - (\bar{\gamma}_{jn}(k-1) + \eta_{jin}(k))] \quad (3.12)$$

where $\eta_{jin}(k)$ denotes additive noise at sensor j, and $\alpha(k)$ is chosen to satisfy

$$\alpha(k) > 0, \quad \sum_{k=1}^{\infty} \alpha(k) = \infty, \quad \text{and} \ \sum_{k=1}^{\infty} \alpha^2(k) < \infty. \quad (3.13)$$

The third condition in (3.13) dictates $\alpha(k)$ to vanish as $k \to \infty$, while the second condition demands this convergence to be sufficiently slow. Typical choice of a stepsize satisfying (3.13) is $\alpha(k) = 1/k$. All the iterates $\{\bar{\gamma}_{jn}(k)\}_{j=1}^{J}$ in (3.12) reach consensus to an estimator of $\bar{\gamma}_n$ that has bounded variance [20]; see also [24]. However, similar to stochastic approximation iterations encountered in adaptive signal processing, the price paid is considerably slower convergence and inability to track changes; see e.g., [49, Ch. 9] and also the simulations in Section 3.5. This renders the CA-SI less desirable for distributed decoding in the presence of noise. These considerations motivate our distributed, iterative CA decoder based on the method of multipliers.

3.3.2 CA-MoM Algorithm with Ideal Links

The CA-MoM approach originates from the well-known fact that the sample average can be viewed as the solution of a least-squares cost, namely $\bar{\gamma}_n = \arg\min_\theta \frac{1}{2}\sum_{j=1}^{J}(\gamma_{jn}-\theta)^2$. The crux of distributing this centralized cost is to use *local auxiliary* variables $\bar{\gamma}_{jn}$ for the wanted average per sensor j along with constraints to ensure agreement of these variables among neighboring sensors; hence, the term consensus. The variables $\bar{\gamma}_{jn}$ can be written as the minimizer of the following consensus-constrained quadratic cost function

$$\begin{aligned}&\min_{\bar{\boldsymbol{\gamma}}_n} \frac{1}{2}\sum_{j=1}^{J}(\gamma_{jn}-\bar{\gamma}_{jn})^2\\ &\text{s.t. } \bar{\gamma}_{jn}-\bar{\gamma}_{in}=0, \quad j\in\mathcal{J},\ i\in\mathcal{N}_j\end{aligned} \tag{3.14}$$

where $\bar{\boldsymbol{\gamma}}_n := [\bar{\gamma}_{1n},\ldots,\bar{\gamma}_{Jn}]^T$. Since the network is connected under (a3.1), the neighborhood consensus constraint in (3.14) ensures that consensus is achieved over the entire network; i.e., $\bar{\gamma}_{jn}=\bar{\gamma}_{in}\ \forall j,i\in\mathcal{J}$. This provides a sufficient condition to guarantee that the optimum of (3.14) is achieved at the sample average.

The goal is to solve (3.14) distributedly using the MoM [7, Sec. 3.4.4]. The latter entails an additional set of variables $\boldsymbol{z}_n := \{\{z_{jin}, z'_{jin}\}_{i\in\mathcal{N}_j}\}_{j\in\mathcal{J}}$, through which (3.14) can be equivalently written as

$$\begin{aligned}\min_{\bar{\boldsymbol{\gamma}}_j,\boldsymbol{z}_n} \quad & \sum_{j=1}^{J}\frac{1}{2}(\gamma_{jn}-\bar{\gamma}_{jn})^2\\ \text{s.t.} \quad & \bar{\gamma}_{jn}-z_{jin}=0, \quad j\in\mathcal{J},\ i\in\mathcal{N}_j\\ & -\bar{\gamma}_{in}-z'_{jin}=0, \quad j\in\mathcal{J},\ i\in\mathcal{N}_j\\ & z_{jin}+z'_{jin}=0, \quad j\in\mathcal{J}, i\in\mathcal{N}_j.\end{aligned} \tag{3.15}$$

Let v_{jin} and v'_{jin} denote the Lagrange multipliers associated with the constraints $\bar{\gamma}_{jn}-z_{jin}=0$ and $-\bar{\gamma}_{in}-z'_{jin}=0$, respectively. Likewise, define the set $\mathcal{C}_z := \{\boldsymbol{z}_n : z_{jin}+z'_{jin}=0,\ \forall j\in\mathcal{J},\ i\in\mathcal{N}_j\}$ that represents the constraints on the entries of $\boldsymbol{z}_n$. With $c>0$ denoting a

penalty coefficient, consider the augmented Lagrangian function of (3.15)

$$\begin{aligned}\mathcal{L}_a(\bar{\boldsymbol{\gamma}}_n, \boldsymbol{z}_n, \mathbf{v}_n, \mathbf{v}'_n) &= \frac{1}{2}\sum_{j=1}^{J}(\gamma_{jn}-\bar{\gamma}_{jn})^2 + \sum_{j=1}^{J}\sum_{i\in\mathcal{N}_j}\left\{v_{jin}(\bar{\gamma}_{jn}-z_{jin}) + v'_{jin}(-\bar{\gamma}_{in}-z'_{jin})\right\} \\ &\quad + \frac{c}{2}\sum_{j=1}^{J}\sum_{i\in\mathcal{N}_j}\left\{(\bar{\gamma}_{jn}-z_{jin})^2 + (-\bar{\gamma}_{in}-z'_{jin})^2\right\} \end{aligned} \tag{3.16}$$

where the $2|\mathcal{E}|\times 1$ vector $\mathbf{v}_n := [\{v_{1in}\}_{i\in\mathcal{N}_1}, \ldots, \{v_{Jin}\}_{i\in\mathcal{N}_J}]^T$, and likewise for $\mathbf{v}'_n$.

The alternating-direction MoM operates by minimizing $\mathcal{L}_a$ in (3.16) cyclically w.r.t. one set of variables given the other variables. The $(k+1)$st iteration of the MoM solver of (3.16) becomes (see also [7, pg. 255])

$$\bar{\boldsymbol{\gamma}}_n(k+1) = \arg\min_{\bar{\boldsymbol{\gamma}}_n}\ \mathcal{L}_a\left(\bar{\boldsymbol{\gamma}}_n, \boldsymbol{z}_n(k), \mathbf{v}_n(k), \mathbf{v}'_n(k)\right) \tag{3.17a}$$

$$\boldsymbol{z}_n(k+1) = \arg\min_{\boldsymbol{z}_n\in\mathcal{C}_z}\ \mathcal{L}_a\left(\bar{\boldsymbol{\gamma}}_n(k+1), \boldsymbol{z}_n, \mathbf{v}_n(k), \mathbf{v}'_n(k)\right) \tag{3.17b}$$

$$v_{jin}(k+1) = v_{jin}(k) + c(\bar{\gamma}_{jn}(k+1) - z_{jin}(k+1)),\quad j\in\mathcal{J},\ i\in\mathcal{N}_j \tag{3.17c}$$

$$v'_{jin}(k+1) = v'_{jin}(k) + c(-\bar{\gamma}_{in}(k+1) - z'_{jin}(k+1)),\quad j\in\mathcal{J},\ i\in\mathcal{N}_j \tag{3.17d}$$

where (3.17c) and (3.17d) are due to the fact that the multipliers in (3.16) appear linearly in $\mathcal{L}_a$.

It is proved in Appendix 3.6.2 that if $v_{jin}(k)$ and $v'_{jin}(k)$ are initialized identically $\forall(j,i)\in\mathcal{E}$; i.e., $v_{jin}(0) = v'_{jin}(0)$, the variables $\mathbf{v}'_n(k)$ and $\boldsymbol{z}_n(k+1)$ can be eliminated and (3.17a)-(3.17d) can be simplified to

$$v_{jin}(k) = v_{jin}(k-1) + \frac{c}{2}(\bar{\gamma}_{jn}(k) - \bar{\gamma}_{in}(k)),\quad j\in\mathcal{J},\ i\in\mathcal{N}_j \tag{3.18a}$$

$$\bar{\gamma}_{jn}(k+1) = \frac{1}{1+2c|\mathcal{N}_j|}\left\{\gamma_{jn} - \sum_{i\in\mathcal{N}_j}\left[v_{jin}(k) - v_{ijn}(k) - c(\bar{\gamma}_{jn}(k)+\bar{\gamma}_{in}(k))\right]\right\},\quad j\in\mathcal{J}. \tag{3.18b}$$

The iterations (3.18a)-(3.18b) constitute the CA-MoM algorithm. Sensor $j\in\mathcal{J}$ maintains the local estimate of the average LLR $\bar{\gamma}_{jn}(k)$ and all the multipliers $\{v_{jin}(k)\}_{i\in\mathcal{N}_j}$. During the k-th iteration, sensor j receives the broadcasted estimates $\bar{\gamma}_{in}(k)$ from all its neighboring sensors $i\in\mathcal{N}_j$, and updates the corresponding multipliers via (3.18a). It then transmits

back the updated multiplier $v_{jin}(k)$ to each of its neighboring sensors $i \in \mathcal{N}_j$, based on which each sensor j is able to determine $\bar{\gamma}_{jn}(k+1)$ via (3.18b). Subsequently, all sensors $j \in \mathcal{J}$ broadcast their updated estimates $\bar{\gamma}_{jn}(k+1)$ to their neighbors, thus completing the k-th iteration and initializing the next one.

The iterates in (3.18a)-(3.18b) are provably convergent, as asserted in the following proposition.

Proposition 3.2 *(CA-MoM with ideal inter-sensor links) Under (a3.2a), the CA-MoM iterations (3.18a)-(3.18b) with arbitrary initialization of $\bar{\gamma}_{jn}(1)$ and $v_{jin}(0)$, $\forall (j,i) \in \mathcal{E}$ and $c > 0$, reach consensus to the average LLR in (3.7) as $k \to \infty$; i.e.,*

$$\lim_{k\to\infty} \bar{\gamma}_{jn}(k) = \bar{\gamma}_n, \quad \forall j \in \mathcal{J},\ n = 1, \ldots, N. \tag{3.19}$$

Proof: Iterations (3.17a)-(3.17d) follow directly from the MoM approach in [7, pg. 255]; and as shown in Appendix 3.6.2, they are equivalent to (3.18a)-(3.18b). Since the cost function in (3.15) is convex and its constraints comply with [7, Assumption 4.1, pg. 255], the iterates in (3.17a)-(3.17d) converge as established by [7, Proposition 4.2, pg. 256]. The aforementioned equivalence readily establishes the convergence of (3.18a)-(3.18b) too. □

Remark 3.2 *(Comparison with [45].)* A MoM-based consensus approach was also developed in [45] for distributed estimation. However, to form the MoM cost in [45], a subset of sensors called bridge sensors is required. In turn, an algorithm to find the bridge sensor set is necessary, and has to be re-run whenever a bridge sensor fails. Setting all sensors in the network to act as bridge sensors is possible, but the communication overhead is unnecessarily increased. Compared to [45], the consensus-constrained formulation in (3.15) does not require such a bridge sensor set, and in this sense, it offers a *fully* distributed approach. In addition to bypassing the need for bridge sensors, the approach here relative to [45], has provable convergence when links fail randomly – an issue dealt with in Section 3.3.4.

Remark 3.3 *(Comparison with CA-SI for ideal links.)* It has been shown that the CA-SI scheme in (3.11) can be derived using a primal-dual solver of (3.14) [36]. However, weight

selections in [36] and [54] ensuring convergence of (3.11) without requiring knowledge of the global WSN topology per sensor, are rather limited. One possible choice is to select $\mathbf{W} = \mathbf{I}_J - (1/J)\mathbf{L}$, where $\mathbf{L}$ denotes the Laplacian of the graph. Since MoM-based iterations on the other hand are convergent $\forall\ c > 0$, they are more flexible to accelerate convergence speed, even when inter-sensor links are ideal; see also [55, pg. 69]. In addition, MoM-based solvers can cope with noisy links as elaborated next.

3.3.3 CA-MoM Algorithm with Noisy Links

When inter-sensor links are corrupted by additive noise as in (a3.2b), different from the CA-SI iterates in (3.11), the CA-MoM iterates in (3.18a)-(3.18b) will be shown to exhibit finite variance without resorting to a vanishing stepsize. To prove this claim, let $\eta_{jin}(k)$ and $\bar{\eta}_{jin}(k)$ denote the additive noise present in sensor $j \in \mathcal{J}$ when receiving $v_{ijn}(k)$ and $\bar{\gamma}_{in}(k)$ from sensor $i \in \mathcal{N}_j$, respectively. The iterations in (3.18a)-(3.18b) now take the form

$$v_{jin}(k) = v_{jin}(k-1) + \frac{c}{2}[\bar{\gamma}_{jn}(k) - (\bar{\gamma}_{in}(k) + \bar{\eta}_{jin}(k))], \quad j \in \mathcal{J},\ i \in \mathcal{N}_j \tag{3.20a}$$

$$\bar{\gamma}_{jn}(k+1) = \frac{1}{1+2c|\mathcal{N}_j|}\Big\{\gamma_{jn} - \sum_{i\in\mathcal{N}_j}[v_{jin}(k) - (v_{ijn}(k) + \eta_{jin}(k)) - c(\bar{\gamma}_{jn}(k) + \bar{\gamma}_{in}(k) + \bar{\eta}_{jin}(k))]\Big\}. \tag{3.20b}$$

Iterations (3.20a)-(3.20b) can be interpreted as stochastic gradient updates; see e.g., [7, Sec. 7.8]. Viewed from this vantage point, the noise causes $\bar{\gamma}_{jn}(k)$ to fluctuate around the noise-free optimal solution of (3.14). The magnitude of fluctuations is proportional to the noise variance. Our convergence claim in this case can be summarized as follows.

Proposition 3.3 *(CA-MoM with noisy inter-sensor links) Under (a3.2b), the CA-MoM iterations (3.20a)-(3.20b) reach consensus to the average LLR in (3.7) in the mean sense as $k \to \infty$; i.e.,*

$$\lim_{k\to\infty} E\{\bar{\gamma}_{jn}(k)\} = \bar{\gamma}_n \quad \forall j \in \mathcal{J},\ n = 1, \ldots, N. \tag{3.21}$$

In addition, the variance of $\bar{\gamma}_{jn}(k)$ converges to a bounded value, which depends on the noise variance and the WSN topology.

Proof: See Appendix 3.6.3. □

Remark 3.4 *(Comparison with CA-SI with noisy links.)* Different from the CA-SI algorithm in (3.11), Proposition 3.3 ensures that as iterations evolve, the variance of $\bar{\gamma}_{jn}(k)$ remains bounded without a vanishing stepsize required by (3.12). As the simulations of Section 3.5 will confirm, bypassing the need for a vanishing stepsize further enables the CA-MoM to converge faster than the noise-resilient CA-SI in (3.12).

3.3.4 CA-MoM Algorithm with Link Failures

In the presence of random link failures, the adjacency matrix $\mathbf{A}(k)$ at the k-th iteration adheres to an independent and identically distributed (i.i.d.) Bernoulli process with $\Pr[[\mathbf{A}(k)]_{ji} = 1] = p_{ij} > 0$ as per (a3.2c). Whenever a link fails between sensors, the corresponding multipliers become unavailable. The idea to cope with this challenge is to introduce an extra variable $v_{ijn}^{(j)}(k)$ per sensor j to keep track of neighbor i's multiplier information. The CA-MoM iterations (3.18a)-(3.18b) take now the form

$$v_{jin}(k) = \begin{cases} v_{jin}(k-1) + \frac{c}{2}[\bar{\gamma}_{jn}(k) - \bar{\gamma}_{in}(k)] & \text{if } [\mathbf{A}(k)]_{ji} = 1 \\ v_{jin}(k-1) & \text{if } [\mathbf{A}(k)]_{ji} = 0 \end{cases} \tag{3.22a}$$

$$v_{ijn}^{(j)}(k) = \begin{cases} v_{ijn}(k) & \text{if } [\mathbf{A}(k)]_{ji} = 1 \\ v_{ijn}^{(j)}(k-1) & \text{if } [\mathbf{A}(k)]_{ji} = 0 \end{cases} \tag{3.22b}$$

$$\bar{\gamma}_{jn}(k+1) = \frac{1}{1+2c|\mathcal{N}_j|}\Bigg\{\gamma_{jn} - \sum_{i\in\mathcal{N}_j} p_{ji}[v_{jin}(k) - v_{ijn}^{(j)}(k)] + 2c|\mathcal{N}_j|\bar{\gamma}_{jn}(k) + c\sum_{i\in\mathcal{N}_j(k)}(\bar{\gamma}_{in}(k) - \bar{\gamma}_{jn}(k))\Bigg\}. \tag{3.22c}$$

Sensor j locally stores the multipliers $\{v_{ijn}^{(j)}(k)\}_{i\in\mathcal{N}_j}$ of its neighboring sensors and freezes the multiplier pair $\{v_{jin}(k), v_{ijn}^{(j)}(k)\}$ whenever the link (j,i) fails at the k-th iteration. Notice that p_{ji} is assumed known when updating $\bar{\gamma}_{jn}(k)$ in (3.22c). This way, sensor j keeps track of the unavailable multipliers by updating the local ones $v_{ijn}^{(j)}(k)$ through (3.22b), since the freezing strategy maintains the multiplier pair unchanged during the link failures.

Since each sensor j does not have to wait until information $\{\bar{\gamma}_{in}(k)\}_{i\in\mathcal{N}_j}$ from all its neighbors becomes available in order to update $\bar{\gamma}_{jn}(k+1)$ in (3.22c), iterations (3.22a)-(3.22c) can be seen as an *asynchronous* implementation of the CA-MoM. Convergence of the algorithm can be asserted in the mean-square sense (m.s.s.) as summarized next.

Proposition 3.4 *(CA-MoM with random failures of inter-sensor links) Under (a3.2c), the iterations (3.22a)-(3.22c) consent on the average LLR of (3.7) in the m.s.s. as $k \to \infty$; i.e.,*

$$\lim_{k\to\infty} E\left\{|\bar{\gamma}_{jn}(k) - \bar{\gamma}_n|^2\right\} = 0, \quad \forall j \in \mathcal{J}, \quad n = 1, \ldots, N. \tag{3.23}$$

Proof: See Appendix 3.6.4. □

Compared to the CA-MoM with ideal links, the asynchronous implementation here only requires extra information on the link probability p_{ji} to update the $\bar{\gamma}_{jn}(k+1)$ in (3.22c). It is possible to relax (a3.2c) and allow for a real-time acquisition of p_{ji}. Specifically, at iteration k sensor j can substitute the running estimate $p_{ji}(k) = \frac{1}{k}\sum_{k'=1}^{k}[\mathbf{A}(k')]_{ji}$ for p_{ji} into (3.22c). With the i.i.d. Bernoulli process in (a3.2c), the asymptotic convergence of $p_{ji}(k)$ to the true p_{ji} is guaranteed, and (3.23) holds even when p_{ji} is acquired on-the-fly.

3.4 Performance Analysis

In Section 3.3, suitable iterative algorithms to solve (P1) and (P2) were developed under conditions (a3.2a)-(a3.2c) regarding inter-sensor communications. This section derives bounds for the BER performance of the CA-SI and CA-MoM algorithms as a function of the number of iterations k, even if consensus on the *exact* LLR vector has not been achieved. Recall also that ML block decoding minimizes the block error rate, whereas bitwise decoding using the APP minimizes the BER. For this reason, the analysis here will be focused on finding bounds for the BER of distributed ML decoders, with the understanding that those also bound the bitwise distributed APP decoder performance too.

3.4.1 Pairwise Probability Bound

The probability of error P_e of the ML decoder in (3.1) can be bounded in terms of the pairwise error probability (PEP), which is the probability of erroneously decoding $\mathbf{x}$ as $\mathbf{x}' \in \mathcal{C}$ with $\mathbf{x}' \neq \mathbf{x}$. Letting $P_{\mathbf{x}\to\mathbf{x}'}$ denote the PEP, it holds that $P_e \leq \sum_{\mathbf{x}\in\mathcal{C}} P_{\mathbf{x}} \sum_{\mathbf{x}'\neq\mathbf{x}} P_{\mathbf{x}\to\mathbf{x}'}$, where $P_{\mathbf{x}}$ is the probability of transmitting $\mathbf{x}$. For a given LLR vector $\boldsymbol{\gamma}$, the PEP can be expressed as [cf. (3.6)]

$$\begin{aligned} P_{\mathbf{x}\to\mathbf{x}'} &= \Pr\left[\boldsymbol{\gamma}^T(\mathbf{x}-\mathbf{x}') \geq 0|\mathbf{x}\right] \\ &\leq E\left\{\exp\left[\lambda\boldsymbol{\gamma}^T(\mathbf{x}-\mathbf{x}')\right]|\mathbf{x}\right\} \end{aligned} \tag{3.24}$$

where the inequality follows from the Chernoff bound, see e.g., [35, Sec. 2.4], and is satisfied for all $\lambda > 0$. Upon setting $\boldsymbol{\gamma} = \bar{\boldsymbol{\gamma}}$, the PEP in (3.24) corresponds to that of the centralized decoder, whereas if $\boldsymbol{\gamma} = \boldsymbol{\gamma}_j$, the PEP pertains to the local performance at sensor j.

Because the AP-sensor channels are memoryless, the Chernoff bound in (3.24) reduces to

$$P_{\mathbf{x}\to\mathbf{x}'} \leq \prod_{n=1}^{N}\left[\int_{\gamma_n} \exp\left[\lambda\gamma_n(x_n - x_n')\right] p[\gamma_n|x_n]d\gamma_n\right]. \tag{3.25}$$

Notice that $(x_n - x_n')$ can only take values 1, 0 or -1. Moreover, under (a3.4), the LLR is symmetrically distributed; hence, $p[\gamma_n|x_n = 0] = p[-\gamma_n|x_n = 1]$. The upper bound in (3.25) can thus be rewritten as

$$P_{\mathbf{x}\to\mathbf{x}'} \leq \prod_{n=1}^{N}\left(\Delta^{(\lambda)}\right)^{|x_n - x_n'|} \tag{3.26}$$

where, due to symmetry, it suffices to consider only the case $x_n = 0$ in

$$\Delta^{(\lambda)} := \int_{\gamma_n} \exp(-\lambda\gamma_n) p[\gamma_n|x_n = 0]d\gamma_n. \tag{3.27}$$

Parameter $\Delta^{(\lambda)}$ encapsulates the channel effects in decoding performance. It is independent of the codebook and the bit index n since the channel statistics are assumed invariant w.r.t. n.

To make the upper bound in (3.26) as tight as possible, consider the minimum achievable $\Delta := \min_{\lambda>0} \Delta^{(\lambda)}$. Not surprisingly, Δ plays the same role as the Chernoff or Bhatacharyya

parameters in the error performance analysis in [35, Sec. 6.8]. Here, Δ is derived as a function of the sufficient statistic $\boldsymbol{\gamma}$, whereas in [35, Sec. 6.8] it is expressed in terms of the received block $\mathbf{y}$. Clearly, for AWGN channels $\mathbf{y}$ and $\boldsymbol{\gamma}$ are closely related, as illustrated in the following example.

Example 3.2 *(Performance analysis for AWGN AP-sensor channels.)* Consider again the AWGN AP-sensor channel using BPSK modulation as in Example 3.1. The input-output relationship per bit can be written as

$$y_{jn} = \sqrt{E_s}(2x_n - 1) + \epsilon_{jn}, \quad x_n = 0, 1 \tag{3.28}$$

where $\epsilon_{jn} \sim \mathcal{N}(0, N_{0,j}/2)$. The conditional probability is

$$p[y_{jn}|x_n] = \frac{1}{\sqrt{\pi N_{0,j}}} \exp\left[-[y_{jn} - \sqrt{E_s}(2x_n - 1)]^2/N_{0,j}\right] \tag{3.29}$$

and the LLR can be explicitly expressed as

$$\gamma_{jn} = -\frac{4\sqrt{E_s}y_{jn}}{N_{0,j}} = \frac{4E_s}{N_{0,j}}(1 - 2x_n) - \frac{4\sqrt{E_s}\epsilon_{jn}}{N_{0,j}}. \tag{3.30}$$

Thus, the LLR γ_{jn} of the AP-sensor channel is a Gaussian random variable with conditional distribution $p(\gamma_n|x_n = 0) = \mathcal{N}(\mu_j, \sigma_j^2)$, where $\mu_j := 4E_s/N_{0,j}$ and $\sigma_j^2 := 8E_s/N_{0,j}$. Substituting the latter into (3.27), and minimizing w.r.t. λ leads to

$$\Delta = \exp\left(-\frac{E_s}{N_{0,j}}\right) \tag{3.31}$$

which coincides with the Bhatacharyya parameter for AWGN channels in [35, Sec. 6.8].

The subsequent steps to derive the BER bound are standard: once the parameter Δ is determined, the error performance can be upper bounded by a polynomial in Δ with coefficients that depend on the specific codeword structure, and are independent of the underlying channel [35, Sec. 6.8].

The next objective is to find similar bounds for the error performance of the local decoder using $\bar{\boldsymbol{\gamma}}_j(k)$, the average LLR estimate available per iteration k.

3.4.2 Error Performance per Iteration

The key to evaluating BER performance per iteration k is to specify the distribution of $\bar{\boldsymbol{\gamma}}_j(k)$ as a function of k. To encompass both CA-SI and CA-MoM algorithms in the subsequent analysis, the following lemma is needed.

Lemma 3.1 *Under (a3.2a), the LLR iterates in (3.11), or (3.18a)-(3.18b) can be commonly expressed as*

$$\bar{\gamma}_{jn}(k) = \sum_{i=1}^{J} c_{ji}(k)\gamma_{in}, \quad \forall j \in \mathcal{J},\ n = 1, \ldots, N \tag{3.32}$$

where the coefficients $c_{ji}(k)$ depend solely on the network topology.

Proof: See Appendix 3.6.5. □

The implication of Lemma 3.1 is that $\{\bar{\boldsymbol{\gamma}}_j(k)\}_{j\in\mathcal{J}}$ maintain the memoryless and symmetry properties from the downlink channel [cf. (a3.2)]. As a result, $\Delta^{(\lambda)}(k)$ in (3.27) becomes

$$\Delta_j^{(\lambda)}(k) := E\left\{e^{-\lambda\bar{\gamma}_{jn}(k)}|x_n = 0\right\} = \prod_{i=1}^{J} \Delta_i^{(\lambda/c_{ji}(k))} \tag{3.33}$$

where $\Delta_j^{(\lambda)}$ corresponds to the Δ parameter for the local channel associated with $\boldsymbol{\gamma}_j$.

Lemma 3.1 asserts a linear relationship for ideal inter-sensor links. Generalization to the noisy links in (a3.2b) is possible given that this will only cause an additive equivalent noise term in (3.32) [cf. Appendix 3.6.3]. Hence, under (a3.2b) equation (3.33) can be modified to include the noise effects accordingly.

To obtain a tight upper bound for the BER at sensor j, let $\Delta_j(k) := \min_{\lambda>0} \Delta_j^{(\lambda)}(k)$ represent the equivalent channel at iteration k and sensor j. The parameter $\Delta_j(k)$ cannot be generally obtained in closed form for an arbitrary pdf $p[\gamma_n|x_n]$. It is however possible for AWGN AP-sensor channels, as demonstrated in the next example.

Example 3.3 *(PEP bound for AWGN AP-sensor channels.)* Consider again BPSK transmission over AWGN AP-sensor channels as in Examples 3.1 and 3.2. Similar to Example

3.2, the local LLR for $x_n = 0$ is Gaussian with mean μ_j and variance σ_j^2. Under (a3.2a), the average LLR estimate $\bar{\gamma}_{jn}(k)$ in (3.32) is a linear combination of Gaussian random variables. Hence, it is Gaussian distributed, with mean and variance given, respectively, by

$$\bar{\mu}_j(k) = \sum_{i=1}^{J} c_{ji}(k)\mu_i, \quad \text{and} \quad \bar{\sigma}_j^2(k) = \sum_{i=1}^{J} c_{ji}^2(k)\sigma_i^2. \tag{3.34}$$

If zero-mean inter-sensor noise is present as in (a3.2b), the average LLR estimate $\bar{\gamma}_{jn}(k)$, in either (3.12) or (3.20a)-(3.20b), remains Gaussian distributed, with mean and variance given respectively by

$$\bar{\mu}_j(k) = \sum_{i=1}^{J} c_{ji}(k)\mu_i, \quad \text{and} \quad \bar{\sigma}_j^2(k) = \sum_{i=1}^{J} c_{ji}^2(k)\sigma_i^2 + \Sigma_j(k) \tag{3.35}$$

where per iteration k, $\Sigma_j(k)$ is the concatenated equivalent noise variance at sensor j and can be obtained from the noise level σ_i^2 at sensor $i \in \mathcal{N}_j$. For example, Appendix 3.6.3 provides its closed-form expression for the CA-MoM algorithm, where $\Sigma_j(k) = [\mathbf{\Sigma}_{\bar{\gamma}_n}(k)]_{jj}$ with $\mathbf{\Sigma}_{\bar{\gamma}_n}(k)$ given in (3.70). Similar argument also holds for the CA-SI algorithm.

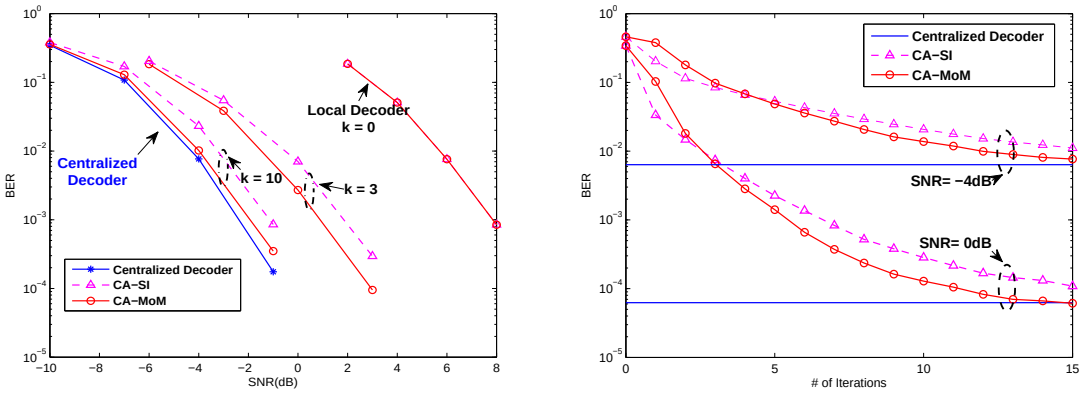
Under both (a3.2a) and (a3.2b) and given the respective mean and variance, (3.33) now becomes [cf. (3.31)]

$$\Delta_j(k) = \exp\left(-\frac{\bar{\mu}_j^2(k)}{2\bar{\sigma}_j^2(k)}\right). \tag{3.36}$$

As a closing note, it is worth mentioning that unlike most existing applications of consensus to distributed detection and estimation problems, which assert estimation/detection performance only asymptotically (as $k \to \infty$), the present section evaluates error performance for a finite number of iterations. This is particularly important for channel decoding because it allows one to quantify the number of iterations needed to attain a prescribed BER value.

3.5 Numerical Tests

The CA-SI and CA-MoM distributed solvers of (P1) and (P2) are tested and compared in this section under various channel settings. The WSN includes $J = 10$ sensors uniformly

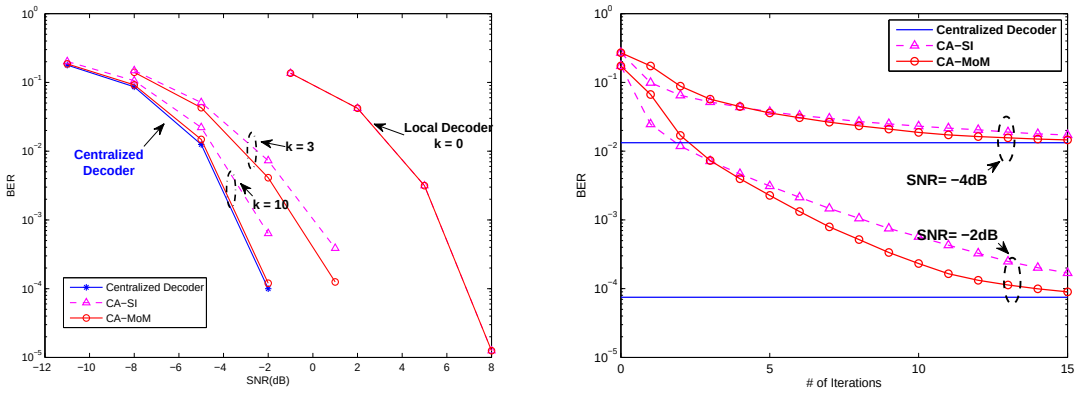


(a) (b)

Figure 3.2: Distributed ML decoder performance with ideal inter-sensor links: (a) BER vs. SNR (in dB) curves with variable number of consensus iterations; (b) BER vs. number of consensus iterations for different AP-sensor SNRs.

distributed over the unit square $[0,1] \times [0,1]$. The communication range of each sensor is $r = 0.5$. Two nodes are connected if their Euclidean distance is less than r. As a result, the number of graph edges here is $|\mathcal{E}| = 18$. The channel between AP and sensor j is modeled as AWGN with SNR $\frac{2E_s}{N_{0,j}}$. For simplicity, the SNR is set to be the same at all sensors; i.e., $N_{0,j} = N_0 \ \forall j$. Matrix $\mathbf{W}$ of the CA-SI in Section 3.3.1 is given by $\mathbf{W} = \mathbf{I} - \xi\mathbf{L}$, where $\xi = 1/(\lambda_2 + \lambda_J)$, and λ_i is the i-th eigenvalue of $\mathbf{L}$, as in [54].

Test Case 3.1 *(Block decoding with ideal inter-sensor links.)* In this case, the average BER of the block ML decoder is simulated for ideal inter-sensor communication links [cf. (a3.2a)]. The AP encodes messages of $K = 40$ bits using a rate 2/3 convolutional code ($N = 60$) with generator registers $[4, 3]$ in octal form. Upon reception, each sensor initiates iterations with its neighbors to estimate the average LLR in (3.7). The iterations in (3.11) are used for the CA-SI algorithm, whereas the iterations in (3.18a)-(3.18b) are used for the CA-MoM algorithm. After a given number of iterations k, each sensor relies on the average LLR $\bar{\gamma}(k)$ collected up to that instant to run the DiVA decoder. Fig. 3.2(a) compares the resulting average BER as a function of the SNR for different values of the iteration index k.

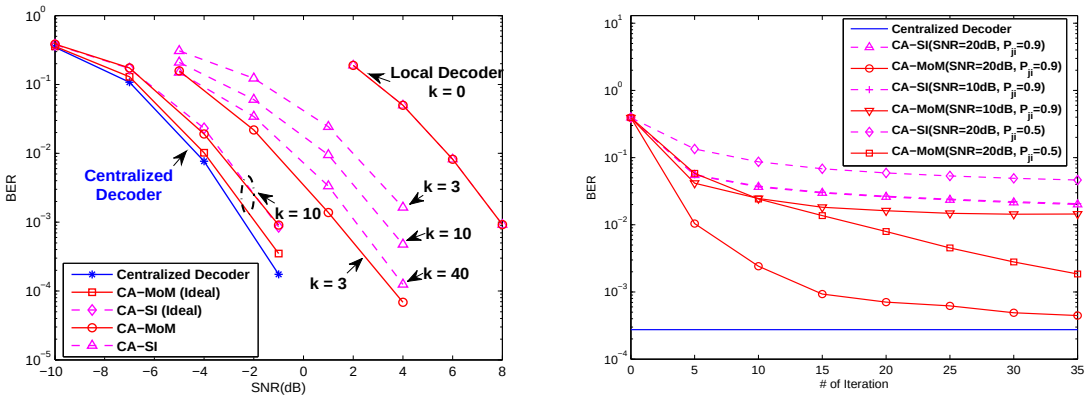


(a) (b)

Figure 3.3: Distributed MAP decoder performance with ideal inter-sensor links: (a) BER vs. SNR (in dB) curves with variable number of consensus iterations; (b) BER vs. number of consensus iterations for different AP-sensor SNRs.

The *centralized* decoder (using $\bar{\gamma}$ in (3.7)) is included as a benchmark. The *local* decoding performance per sensor (corresponding to the distributed decoder after iteration number $k = 0$) is also included for comparison. At each iteration k there are JN transmissions throughout the network when running the CA-SI algorithm. For the CA-MoM algorithm this number is $(J + 2|\mathcal{E}|)N$. Clearly, the average BER sharply reduces as the number of iterations increases, approaching that of the centralized benchmark after a few iterations of both CA-SI and CA-MoM algorithms. This is not surprising since the diversity collected by the decoder increases as the LLR information from neighboring sensors iteratively becomes available. As seen, even for a few iterations, the CA-MoM algorithm outperforms the CA-SI, as predicted by Remark 3.3. Fig. 3.2(b) depicts the decoding performance as a function of the number of consensus iterations for different AP-sensor SNRs. The centralized decoder is shown as a benchmark for the distributed ones. It can be seen that the average BER sharply reduces after a few iterations.

Test Case 3.2 *(Bitwise decoding with ideal inter-sensor links.)* In this test case, the AP encodes messages of $K = 4$ bits using the Hamming (7,4) code. As in Test Case 3.1, each



(a) (b)

Figure 3.4: Distributed ML decoder performance with imperfect inter-sensor links: (a) BER vs. SNR (in dB) curves with variable number of consensus iterations; (b) BER vs. number of consensus iterations with various inter-sensor link SNRs and failure probabilities.

sensor initiates iterations with its neighbors to estimate the average LLR $\bar{\gamma}(k)$ and uses it to find the APP as in (3.10), which it employs subsequently to carry out the bitwise MAP decoder. Both Figs. 3.3(a) and (b) demonstrate similar relative performance of the two CA-SI and CA-MoM algorithms as in Test Case 3.1.

Test Case 3.3 *Block decoding with imperfect inter-sensor links.)* Fig. 3.4(a) depicts the average BER in the presence of both noise and random failures of inter-sensor links. The AP encodes messages as in the Test Case 3.1 and sensors decode using the VA. The additive noise power is 20dB above the signal power for all links, whereas the probability of failure is set to $1 - p_{ji} = 0.1$ for all links. In the simulations, $\alpha(k)$ in (3.12) is set to be $1/k$. Fig. 3.4(a) compares the resulting average BER as a function of the SNR for different values of the iteration index k. The distributed decoders after $k = 10$ iterations with ideal AP-sensor links from Test Case 3.1 are included for comparison. Both centralized and local decoders are also included. The resulting BER of both algorithms degrades w.r.t. the one in Test Case 3.1. This is expected, since the additive noise in the inter-sensor communication links increases the uncertainty in the distributed ML decoding problem. Note, however, that the

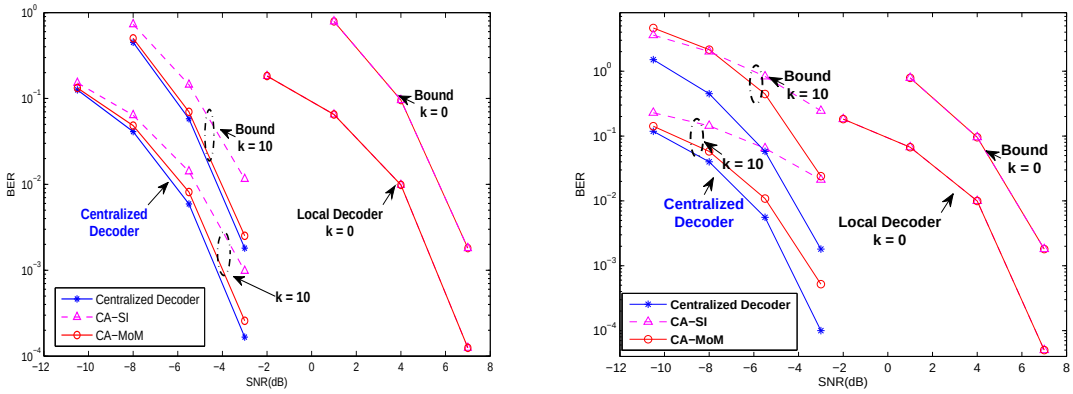


Figure 3.5: Distributed ML decoder performance: BER vs. SNR (in dB) comparison of error bounds with: (a) ideal inter-sensor links; (b) noisy inter-sensor links.

gap between the BER of the CA-MoM algorithm and the CA-SI increases markedly, since the CA-SI has to resort to a vanishing stepsize to ensure convergence even for a relatively low inter-sensor noise level of 20dB. Interestingly, even under these adverse conditions, the CA-MoM algorithm after $k = 10$ iterations exhibits approximately the same performance as the CA-SI one for ideal AP-sensor links, and is still close to the centralized one that is by definition not affected by inter-sensor imperfections with all observations available centrally. Fig. 3.4(b) depicts the performance of various distributed decoders given different inter-sensor link conditions. The AP-sensor SNR level is set to be -1dB. The inter-sensor noise level affects the convergence floor of the distributed decoders; while the link failure probability determines the convergence rate. For example, with $p_{ji} = 0.9$, both distributed CA-MoM decoders reach steady state after 30 iterations approximately. Intuitively, a lower inter-sensor noise level (SNR $=$ 20 dB) will lead to a much better BER performance. In addition, given the same inter-sensor link SNR level of 20 dB, the CA-MoM decoder with $p_{ji} = 0.9$ converges much faster. Intuitively, the larger p_{ji} is, the less probable it is for each link to fail per transmission, and in this sense the CA-MoM decoder operates more efficiently.

Test Case 3.4 *(BER bounds.)* Here, the error performance bounds developed in Section 3.4 are tested for block ML decoding of the Hamming (7,4) code. Both ideal and noisy inter-sensor links are assumed. The theoretical upper bounds on the BER per iteration are computed based on (3.36). Figs. 3.5(a) and 3.5(b) plot the resulting curves for different values of k with ideal and noisy inter-sensor links, respectively. The simulated BER obtained through Monte Carlo runs is also included for comparison. The theoretical upper bounds exhibit the same diversity order as the simulated BER, verifying the validity of $\Delta_j(k)$ in (3.33) as a performance metric of the CA-SI based block ML decoders. In practical distributed decoding problems, these upper bounds offer a useful means of determining the number of iterations needed to guarantee a given level of error performance at any given sensor.

3.6 Appendices

3.6.1 Sufficient statistics for arbitrary alphabets

If x_n takes values from the alphabet $\{0,\ldots,Q-1\}$, then $\log p[y_{jn}|x_n]$ is a piece-wise linear function of x_n [cf. (3.4)]. Specifically, if $x_n \in \{q-1,q\}$, then $\log p[y_{jn}|x_n]$ is given by

$$\log p[y_{jn}|x_n] = -(x_n - q + 1)\log\left(\frac{p[y_{jn}|x_n = q-1]}{p[y_{jn}|x_n = q]}\right) + \log p[y_{jn}|x_n = q-1]. \tag{3.37}$$

Because the sum of piece-wise linear functions is a piece-wise linear function, the ML decoding problem in (3.1) is equivalent to

$$\hat{\mathbf{x}}_{ML} = \underset{\mathbf{x}\in\mathcal{C}}{\operatorname{argmin}} \sum_{n=1}^{N} \gamma_n^{PL}(x_n) \tag{3.38}$$

where

$$\gamma_n^{PL}(x_n) := (x_n - q + 1)\left(\frac{1}{J}\sum_{j=1}^{J}\gamma_{jn}^{(q)}\right) + \sum_{q'=1}^{q-1}\left(\frac{1}{J}\sum_{j=1}^{J}\gamma_{jn}^{(q')}\right) \tag{3.39}$$

for $x_n \in \{q-1,q\}$, and $q = 1,\ldots,Q-1$. With the local LLRs defined as

$$\gamma_{jn}^{(q)} = \left(\frac{p[y_{jn}|x_n = q-1]}{p[y_{jn}|x_n = q]}\right), \quad q = 1,\ldots,Q-1 \tag{3.40}$$

it follows that $(Q-1)$ average LLRs are sufficient statistics for this distributed decoding problem.

3.6.2 Derivation of (3.18a)-(3.18b)

Because the variables $\bar{\gamma}_{jn}$ in (3.16) are not coupled, (3.17a) is equivalent to the following J separable sub-problems, one per sensor j

$$\begin{aligned}\bar{\gamma}_{jn}(k+1) \;=\; \arg\min_{\bar{\gamma}_{jn}} \Bigg\{ &\frac{1}{2}(\gamma_{jn} - \bar{\gamma}_{jn})^2 + \sum_{i\in\mathcal{N}_j}(v_{jin}(k) - v'_{ijn}(k))\bar{\gamma}_{jn} \\ &+ \sum_{i\in\mathcal{N}_j}\frac{c}{2}\{(\bar{\gamma}_{jn} - z_{jin}(k))^2 + (-\bar{\gamma}_{jn} - z'_{ijn}(k))^2\}\Bigg\}. \end{aligned}\tag{3.41}$$

Being linear-quadratic in $\bar{\gamma}_{jn}$, each of these sub-problems can be solved in closed form to obtain

$$\bar{\gamma}_{jn}(k+1) = \frac{1}{1+2c|\mathcal{N}_j|}\left\{\gamma_{jn} - \sum_{i\in\mathcal{N}_j}[v_{jin}(k) - v'_{ijn}(k) - c(z_{jin}(k) - z'_{ijn}(k))]\right\}. \tag{3.42}$$

Similarly, $\boldsymbol{z}_n(k+1)$ in (3.17b) is obtained by solving the following $2|\mathcal{E}|$ sub-problems indexed by $(j,i) \in \mathcal{E}$

$$\{z_{jin}(k+1), z'_{jin}(k+1)\} = \arg \min_{z_{jin}+z'_{jin}=0} \Big\{ -v_{jin}(k) z_{jin} - v'_{jin}(k) z'_{jin} + \tfrac{c}{2}(\bar{\gamma}_{jn}(k+1) - z_{jin})^2 + \tfrac{c}{2}(-\bar{\gamma}_{jn}(k+1) - z'_{jin})^2 \Big\}. \tag{3.43}$$

If $v_{jin}(k)$ and $v'_{jin}(k)$ are initialized as $v_{jin}(0) = v'_{jin}(0)\ \forall (j,i) \in \mathcal{E}$, the solution to (3.43) for $k = 0$ becomes

$$z_{jin}(1) = -z'_{jin}(1) = \frac{1}{2}(\bar{\gamma}_{jn}(1) + \bar{\gamma}_{in}(1)). \tag{3.44}$$

Substituting (3.44) into (3.17c) and (3.17d) yields

$$v_{jin}(1) = v'_{jin}(1) = v_{jin}(0) + \frac{c}{2}(\bar{\gamma}_{jn}(1) - \bar{\gamma}_{in}(1)). \tag{3.45}$$

Proceeding by induction, if $v_{jin}(k) = v'_{jin}(k)$, the solution to (3.43) is

$$z_{jin}(k+1) = -z'_{jin}(k+1) = \frac{1}{2}(\bar{\gamma}_{jn}(k+1) + \bar{\gamma}_{in}(k+1)). \tag{3.46}$$

Substituting (3.46) into (3.17c) and (3.17d) proves that

$$v_{jin}(k+1) = v'_{jin}(k+1) = v_{jin}(k) + \frac{c}{2}(\bar{\gamma}_{jn}(k+1) - \bar{\gamma}_{in}(k+1)) \tag{3.47}$$

which establishes along with (3.45) that $v_{jin}(k) = v'_{jin}(k)\ \forall k > 0$.

Equality (3.47) shows that it is sufficient to update only one set of multipliers $\{v_{jin}(k)\}_{j\in\mathcal{J}}^{i\in\mathcal{N}_j}$ per iteration as in (3.18a). Finally, substituting (3.46) into (3.42) proves the validity of (3.18b).

3.6.3 Proof of Proposition 3.3

To prove convergence in the mean, recall that the noise terms $\eta_{jin}(k)$ and $\bar{\eta}_{jin}(k)$ in (3.20a) and (3.20b) are zero-mean and uncorrelated across time and links. Given the linearity of the expectation operator, taking averages on both sides of (3.20a) and (3.20b) results in iterations similar to (3.18a) and (3.18b), which involve deterministic iterates whose convergence is ensured by Proposition 3.2.

To establish that the iterates in (3.20a)-(3.20b) have bounded variance, define the $2|\mathcal{E}| \times 1$ vectors $\boldsymbol{\eta}_n(k) := [\{\eta_{1in}(k)\}_{i\in\mathcal{N}_1}, \ldots, \{\eta_{Jin}(k)\}_{i\in\mathcal{N}_j}]^T$ and $\bar{\boldsymbol{\eta}}_n(k) := [\{\bar{\eta}_{1in}(k)\}_{i\in\mathcal{N}_1}, \ldots, \{\bar{\eta}_{Jin}(k)\}_{i\in\mathcal{N}_j}]^T$; and the $J \times 2|\mathcal{E}|$ incidence matrix $\mathbf{M}$, see e.g., [51, pg. 56], formed by columns $\{\mathbf{m}_p\}_{p=1}^{2|\mathcal{E}|}$ of size $J \times 1$. With the subscript p corresponding to the $j \to i$ directed edge of the graph, the $\mathbf{m}_{p(j\to i)}$ column has all entries equal to 0 except its jth and ith, for which $[\mathbf{m}_{p(j\to i)}]_j = 1$ and $[\mathbf{m}_{p(j\to i)}]_i = -1$. Using the so defined matrix $\mathbf{M}$, (3.20a) can be expressed in vector form as

$$\mathbf{v}_n(k) = \mathbf{v}_n(k-1) + \frac{c}{2}[\mathbf{M}^T\bar{\boldsymbol{\gamma}}_n(k) - \bar{\boldsymbol{\eta}}_n(k)]. \tag{3.48a}$$

Let now $\mathbf{M}'$ denote the matrix formed by setting to zero all non-positive entries of $\mathbf{M}$; and $\mathbf{N}_c := \mathbf{I}_J + 2c\mathbf{D}$, where $\mathbf{D}$ is the degree matrix defined in the opening paragraph of Section 3.1. Using these definitions, (3.20b) can be likewise expressed in vector form as

$$\bar{\boldsymbol{\gamma}}_n(k+1) = \mathbf{N}_c^{-1}[\bar{\boldsymbol{\gamma}}_n - \mathbf{M}\mathbf{v}_n(k) + c(\mathbf{A}+\mathbf{D})\bar{\boldsymbol{\gamma}}_n(k) + \mathbf{M}'\boldsymbol{\eta}_n(k) + c\mathbf{M}'\bar{\boldsymbol{\eta}}_n(k)]. \tag{3.48b}$$

Upon defining the $(J + 2|\mathcal{E}|) \times 1$ augmented state vector $\mathbf{s}_n^a(k) := [\bar{\boldsymbol{\gamma}}_n^T(k+1), \mathbf{v}_n^T(k)]^T$, the iterations (3.48a) and (3.48b) can be collectively written as

$$\mathbf{s}_n^a(k) = \mathbf{W}^a\mathbf{s}_n^a(k-1) + \mathbf{b}_n^a + \boldsymbol{\eta}_n^a(k) \tag{3.49}$$

where the $(J+2|\mathcal{E}|)\times(J+2|\mathcal{E}|)$ augmented state transition matrix $\mathbf{W}^a$, and the $(J+2|\mathcal{E}|)\times 1$ vector $\mathbf{b}_n^a$ are given, respectively, by

$$\mathbf{W}^a := \begin{bmatrix} 2c\mathbf{N}_c^{-1}\mathbf{A} & -\mathbf{N}_c^{-1}\mathbf{M} \\ \frac{c}{2}\mathbf{M}^T & \mathbf{I}_{2|\mathcal{E}|} \end{bmatrix}, \text{ and } \mathbf{b}_n^a := \begin{bmatrix} \mathbf{N}_c^{-1}\boldsymbol{\gamma}_n \\ \mathbf{0}_{2|\mathcal{E}|} \end{bmatrix} \tag{3.50}$$

whereas the augmented noise term $\boldsymbol{\eta}_n^a(k)$ can be expressed as $\boldsymbol{\eta}_n^a(k) := \mathbf{E}\boldsymbol{\eta}_n(k) + \bar{\mathbf{E}}\bar{\boldsymbol{\eta}}_n(k)$ with

$$\mathbf{E} := \begin{bmatrix} \mathbf{N}_c^{-1}\mathbf{M}' \\ \mathbf{0}_{2|\mathcal{E}|} \end{bmatrix}, \text{ and } \bar{\mathbf{E}} := \begin{bmatrix} \frac{c}{2}\mathbf{N}_c^{-1}(\mathbf{M}+2\mathbf{M}') \\ -c\mathbf{I}_{2|\mathcal{E}|}/2 \end{bmatrix}. \tag{3.51}$$

The next lemma establishes useful properties for the eigenvalues and eigenvectors of $\mathbf{W}^a$.

Lemma 3.2 *If $c > 0$, and the eigenvalues $\{\lambda_i\}_{i=1}^{J+2|\mathcal{E}|}$ of $\mathbf{W}^a$ are ordered so that $|\lambda_1| \leq \ldots \leq |\lambda_{J+2|\mathcal{E}|}|$, then:*

(a) $|\lambda_i| < 1$ for $i = 1, 2, \ldots, 2J-1$, and $\lambda_i = 1$ for $i = 2J, \ldots, J+2|\mathcal{E}|$;

(b) with $\mathbf{u}_{1,i}$ ($\mathbf{w}_{1,i}$) representing the first J entries of the right (left) eigenvector of $\mathbf{W}^a$, and $\mathbf{u}_{2,i}$ ($\mathbf{w}_{2,i}$) the last $2|\mathcal{E}|$ entries, it holds that:

$$\mathbf{u}_{1,i} = \mathbf{w}_{1,i} = \mathbf{M}\mathbf{u}_{2,i} = \mathbf{M}\mathbf{w}_{2,i} = \mathbf{0}_J, \quad i = 2J, \ldots, J+2|\mathcal{E}|. \tag{3.52}$$

Proof: Since $\mathbf{W}^a\mathbf{u}_i = \lambda_i\mathbf{u}_i$, it follows that $\mathbf{W}_1^a\mathbf{u}_i = \lambda_i\mathbf{u}_{1,i}$ and $\mathbf{W}_2^a\mathbf{u}_i = \lambda_i\mathbf{u}_{2,i}$, where $\mathbf{W}_1^a$ ($\mathbf{W}_2^a$) is formed by the first (last) J ($2|\mathcal{E}|$) rows of $\mathbf{W}^a$. Using (3.50) to eliminate $\mathbf{u}_{2,i}$ from these eigenvector-eigenvalue equations, and left multiplying the result with $\mathbf{u}_{1,i}^{\mathcal{H}}$, yields the second-order equation

$$\left(\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}\right)\lambda_i^2 - \left[\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{I} + 2c\mathbf{A} + 2c\mathbf{D})\mathbf{u}_{1,i}\right]\lambda_i + c\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{A}+\mathbf{D})\mathbf{u}_{1,i} = 0, \tag{3.53}$$

whose roots can be expressed as $\lambda_i = \alpha_1 \pm \sqrt{\alpha_1^2 - \alpha_2}$, where

$$\alpha_1 := \frac{\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{I} + 2c\mathbf{A} + 2c\mathbf{D})\mathbf{u}_{1,i}}{2\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}}, \quad \text{and} \quad \alpha_2 := \frac{c\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{A}+\mathbf{D})\mathbf{u}_{1,i}}{\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}}. \tag{3.54}$$

Because $c > 0$, for any non-zero $\mathbf{u}_{1,i}$, it holds that

$$2c\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{A}\mathbf{u}_{1,i} \leq 2c\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{D}\mathbf{u}_{1,i} < \mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i} \tag{3.55}$$

where the first inequality comes from the positive semi-definiteness of the graph Laplacian $\mathbf{L} := \mathbf{D} - \mathbf{A}$ [51, pg. 469], and the second is due to the definition $\mathbf{N}_c := \mathbf{I}_J + 2c\mathbf{D}$.

If $\alpha_1^2 < \alpha_2$, then the complex root λ_i has nonzero imaginary part. Using (3.54) and (3.55), one can deduce that in this case $|\lambda_i| = \alpha_2 < 1$. If $\alpha_1^2 \geq \alpha_2$, then λ_i is real. In this case, (3.55) implies that $\alpha_1 < 1$. Upon defining $\alpha_3 := 1 - \alpha_1 = \frac{\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{I}+2c\mathbf{L})\mathbf{u}_{1,i}}{2\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}}$, the root can be rewritten as

$$\lambda_i = 1 - \alpha_3 \pm \sqrt{\alpha_3^2 - \frac{c\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{L}\mathbf{u}_{1,i}}{\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}}}. \tag{3.56}$$

Using again (3.55), it follows that $0 \leq \alpha_3 < 1$, so the maximum value inside the square root is α_3^2; thus, $|\lambda_i| \leq 1$. Strict equality holds when $\mathbf{L}\mathbf{u}_{1,i} = \mathbf{0}_J$. In this case, the

graph connectivity of (a3.1) implies that $\mathbf{u}_{1,i} = u_{1,i}\mathbf{1}_J$. Substituting into $\mathbf{W}^a\mathbf{u}_i = \lambda_i\mathbf{u}_i$ gives $u_{1,i}\mathbf{1}_J + \mathbf{M}\mathbf{u}_{2,i} = \mathbf{0}_J$; and left multiplying the latter with $\mathbf{1}_J^T$ yields $u_{1,i} = 0$, and thus $\mathbf{M}\mathbf{u}_{2,i} = \mathbf{0}_J$. The connectivity of the graph further implies that $\text{rank}(\mathbf{M}) = J - 1$; hence, the geometric multiplicity of the eigenvalue $\lambda_i = 1$, equal to the nullity of $\mathbf{M}$, is $2|\mathcal{E}| - J + 1$. Similar reasoning applied to the left eigenvectors implies that $\mathbf{w}_{1,i} = \mathbf{M}\mathbf{w}_{2,i} = \mathbf{0}_J$ corresponding to $\lambda_i = 1$.

The remaining step is to show that the algebraic multiplicity of $\lambda_i = 1$ is also $2|\mathcal{E}|-J+1$. We will prove this by contradiction. Suppose that the eigenvalue $\lambda_i = 1$ has at least one Jordan block of size $\ell\times\ell$, where $\ell \geq 2$. Hence, there exists a matrix $[\mathbf{t}_1, \mathbf{t}_2]$ of size $(J+2|\mathcal{E}|)\times 2$ satisfying

$$\mathbf{W}^a[\mathbf{t}_1, \mathbf{t}_2] = [\mathbf{t}_1, \mathbf{t}_2] \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = [\mathbf{t}_1, \mathbf{t}_1 + \mathbf{t}_2]. \tag{3.57}$$

From (3.57), clearly $\mathbf{t}_1$ is the eigenvector of $\mathbf{W}^a$ corresponding to the eigenvalue 1. As in Lemma 3.2 (b), let $\mathbf{t}_{1,i}$ represent the first J entries of $\mathbf{t}_i$, and $\mathbf{t}_{2,i}$ the last $2|\mathcal{E}|$ entries for $i = 1, 2$. It clearly holds that

$$\mathbf{t}_{1,1} = \mathbf{0}_J, \tag{3.58a}$$

$$\mathbf{M}\mathbf{t}_{2,1} = \mathbf{0}_J. \tag{3.58b}$$

A second equality following from (3.57) is that $\mathbf{W}^a\mathbf{t}_2 = \mathbf{t}_1 + \mathbf{t}_2$, which can be expressed using the definition of $\mathbf{W}^a$ in (3.50) as

$$2c\mathbf{A}\mathbf{t}_{1,2} - \mathbf{M}\mathbf{t}_{2,2} = \mathbf{N}^c(\mathbf{t}_{1,1} + \mathbf{t}_{1,2}) = (\mathbf{I}_J + 2c\mathbf{D})\mathbf{t}_{1,2}, \tag{3.59a}$$

$$\frac{c}{2}\mathbf{M}^T\mathbf{t}_{1,2} + \mathbf{t}_{2,2} = \mathbf{t}_{2,1} + \mathbf{t}_{2,2}, \tag{3.59b}$$

where the last equality in (3.59a) comes from both (3.58a) and the definition of $\mathbf{N}_c$. Furthermore, (3.59b) implies that $\mathbf{t}_{2,1} = \frac{c}{2}\mathbf{M}^T\mathbf{t}_{1,2}$, and this equality together with (3.58b) yields

$$\mathbf{M}\left(\frac{c}{2}\mathbf{M}^T\mathbf{t}_{1,2}\right) = c\mathbf{L}\mathbf{t}_{1,2} = \mathbf{M}\mathbf{t}_{2,1} = \mathbf{0}_J, \tag{3.60}$$

where $\mathbf{L}$ denotes the graph Lagrangian. The graph connectivity implies that the only solution to (3.60) is $\mathbf{t}_{1,2} = t_{1,2}\mathbf{1}_J$, where $t_{1,2}$ is some constant. Substituting back into (3.59a) yields

$$-\mathbf{M}\mathbf{t}_{2,2} = t_{1,2}\mathbf{1}_J. \tag{3.61}$$

But this violates the fact that the all-one vector $\mathbf{1}_J$ is always in the null space of the incidence matrix $\mathbf{M}$; hence, the **contradiction,** which concludes the proof of Lemma 3.2 □

Based on Lemma 3.2, we have the Jordan canonical form of $\mathbf{W}^a$ satisfying $\mathbf{W}^a = \mathbf{T}\mathbf{J}\mathbf{T}^{-1}$, where $\mathbf{J}$ denotes the Jordan matrix and the nonsingular matrix $\mathbf{T} := [\mathbf{t}_1, \ldots, \mathbf{t}_{J+2|\mathcal{E}|}]$. Since the Jordan block corresponding to $\lambda_i = 1$, we have $\mathbf{t}_i = \mathbf{u}_i$, $\forall i = 2J, \ldots, J + 2|\mathcal{E}|$. Suppose the Jordan matrix $\mathbf{J}$ contains κ Jordan blocks for those $(2J-1)$ eigenvalues $|\lambda_i| < 1$, and consequently takes the block diagonal form of

$$\mathbf{J} = \text{bdiag}(\mathbf{J}_1, \ldots, \mathbf{J}_\kappa, \mathbf{I}_{2|\mathcal{E}|-J+1}) \tag{3.62}$$

where $\mathbf{J}_i$ denotes the i-th Jordan block corresponding to some $|\lambda_i| < 1$, and $\mathbf{I}_{2|\mathcal{E}|-J+1}$ represents the $2|\mathcal{E}| - J + 1$ size-1 Jordan blocks corresponding to $\lambda_i = 1$.

With the matrix canonical decomposition and the Jordan matrix in (3.62), the power of $\mathbf{W}^a$ becomes

$$(\mathbf{W}^a)^l = \mathbf{T}\mathbf{J}^l\mathbf{T}^{-1} = \mathbf{T}\text{bdiag}(\mathbf{J}_1^l, \ldots, \mathbf{J}_\kappa^l, \mathbf{I}_{2|\mathcal{E}|-J+1})\mathbf{T}^{-1} \tag{3.63}$$

Furthermore, upon defining the matrix

$$\check{\mathbf{J}} := \text{bdiag}\left(\mathbf{J}_1, \ldots, \mathbf{J}_\kappa, \text{diag}(\mathbf{0}_{2|\mathcal{E}|-J+1})\right) \tag{3.64}$$

which contains all the first κ Jordan blocks of $\mathbf{J}$ in (3.62), the matrix power in (3.63) can be rewritten as

$$(\mathbf{W}^a)^l = \mathbf{T}(\check{\mathbf{J}} + \tilde{\mathbf{J}})^l\mathbf{T}^{-1} = \mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1} + \mathbf{T}\tilde{\mathbf{J}}\mathbf{T}^{-1}, \tag{3.65}$$

where the diagonal residue matrix $\tilde{\mathbf{J}} := \mathbf{J} - \check{\mathbf{J}}$ contains the last $2|\mathcal{E}| - J + 1$ size-1 Jordan blocks in $\mathbf{J}$, which is invariant in (3.63) for any integer l.

To proceed with the proof of Proposition 3.3, note that (3.49) can be rewritten as

$$\mathbf{s}_n^a(k) = (\mathbf{W}^a)^k \mathbf{s}_n^a(0) + \sum_{l=0}^{k-1} (\mathbf{W}^a)^l [\mathbf{b}_n^a + \boldsymbol{\eta}_n^a(k-l)] \tag{3.66}$$

from where one can readily obtain the covariance matrix of $\mathbf{s}_n^a(k)$ as the covariance of the noise term in (3.66)

$$\begin{aligned} \boldsymbol{\Sigma}_{s_n}(k) &:= E\left\{[\mathbf{s}_n^a(k) - E\mathbf{s}_n^a(k)][\mathbf{s}_n^a(k) - E\mathbf{s}_n^a(k)]^T\right\} \\ &= E\left\{\left[\sum_{l=0}^{k-1}(\mathbf{W}^a)^l \boldsymbol{\eta}_n^a(k-l)\right]\left[\sum_{l=0}^{k-1}(\mathbf{W}^a)^l \boldsymbol{\eta}_n^a(k-l)\right]^T\right\}. \end{aligned} \tag{3.67}$$

Using the power $(\mathbf{W}^a)^l$ in (3.65) and the fact that $\boldsymbol{\eta}_n^a(k)$ is uncorrelated across time, (3.67) reduces to

$$\begin{aligned} \boldsymbol{\Sigma}_{s_n}(k) &= \sum_{l=0}^{k-1}\left(\mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1} + \mathbf{T}\tilde{\mathbf{J}}\mathbf{T}^{-1}\right)\boldsymbol{\Sigma}_{\eta_n^a}\left(\mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1} + \mathbf{T}\tilde{\mathbf{J}}\mathbf{T}^{-1}\right)^T \\ &= \sum_{l=0}^{k-1}\Big(\mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1}\boldsymbol{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}(\check{\mathbf{J}}^l)^T\mathbf{T}^T + \mathbf{T}\tilde{\mathbf{J}}\mathbf{T}^{-1}\boldsymbol{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}\tilde{\mathbf{J}}\mathbf{T}^T + \\ &\qquad \mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1}\boldsymbol{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}\tilde{\mathbf{J}}\mathbf{T}^T + \mathbf{T}\tilde{\mathbf{J}}\mathbf{T}^{-1}\boldsymbol{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}(\check{\mathbf{J}}^l)^T\mathbf{T}^T\Big) \end{aligned} \tag{3.68}$$

where $\boldsymbol{\Sigma}_{\eta_n^a}$ is given by

$$\boldsymbol{\Sigma}_{\eta_n^a} := E\left\{\boldsymbol{\eta}_n^a(k)\left(\boldsymbol{\eta}_n^a(k)\right)^T\right\} = \mathbf{E}\boldsymbol{\Sigma}_{\eta_n}\mathbf{E}^T + \bar{\mathbf{E}}\boldsymbol{\Sigma}_{\bar{\eta}_n}\bar{\mathbf{E}}^T \tag{3.69}$$

with $\boldsymbol{\Sigma}_{\eta_n}$ and $\boldsymbol{\Sigma}_{\bar{\eta}_n}$ denoting the covariance matrices of $\boldsymbol{\eta}_n(k)$ and $\bar{\boldsymbol{\eta}}_n(k)$, respectively. Clearly, the diagonal entries of both matrices are σ_j^2 with the subscript j depending on the receive sensor index for the corresponding noise $\eta_{jin}(k)$ in $\boldsymbol{\eta}_n(k)$, or, $\bar{\eta}_{jin}(k)$ in $\bar{\boldsymbol{\eta}}_n(k)$.

Recalling the definition of $\mathbf{s}_n^a(k)$, the covariance $\boldsymbol{\Sigma}_{\bar{\gamma}_n}(k) := E[\bar{\boldsymbol{\gamma}}_n(k)\bar{\boldsymbol{\gamma}}_n^T(k)]$ becomes the $J \times J$ upper-left sub-block of $\boldsymbol{\Sigma}_{s_n}(k-1)$. At the same time, the equations (3.52) implies the $J \times (2|\mathcal{E}| + J - 1)$ upper-right sub-block of $\mathbf{T}$ have all zero entries. Notice from (3.68) that $\boldsymbol{\Sigma}_{s_n}(k)$ contains four partial matrix sums. However, with the special structure of $\tilde{\mathbf{J}}$ and $\mathbf{T}$, the partial sum matrix involving $\mathbf{T}\tilde{\mathbf{J}}\mathbf{T}^{-1}\boldsymbol{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}\tilde{\mathbf{J}}\mathbf{T}^T$ in (3.68) actually has all zero entries in its $J \times J$ upper-left sub-block. As a result, the covariance matrix of local iterates

becomes

$$\begin{aligned}\mathbf{\Sigma}_{\bar{\gamma}_n}(k+1) &= \Bigg[\sum_{l=0}^{k-1}\Big(\mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1}\mathbf{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}(\check{\mathbf{J}}^l)^T\mathbf{T}^T + \mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1}\mathbf{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}\tilde{\mathbf{J}}\mathbf{T}^T \\ &\qquad + \mathbf{T}\tilde{\mathbf{J}}\mathbf{T}^{-1}\mathbf{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}(\check{\mathbf{J}}^l)^T\mathbf{T}^T\Big)\Bigg]_{1:J,1:J}. \qquad (3.70)\end{aligned}$$

Next, we will prove that all the three partial sum matrices in (3.70) will converge as $k \to \infty$. To this end, recall that $\check{\mathbf{J}}$ contains only the eigenvalues $|\lambda_i| < 1$ of $\mathbf{J}$ with other $2|\mathcal{E}| - J + 1$ eigenvalues being 0, hence its spectral radius $\rho(\check{\mathbf{J}}) < 1$. From Lemma 5.6.10 in [Horn-Johnson, 1999, p. 297], we know that given any $\epsilon > 0$, there exists some matrix norm $\|\cdot\|$ such that $\rho(\check{\mathbf{J}}) \leq \|\check{\mathbf{J}}\| \leq \rho(\check{\mathbf{J}}) + \epsilon$. Let us pick the positive number $\epsilon = [1 - \rho(\check{\mathbf{J}})]/2$, then we have

$$\|\check{\mathbf{J}}\| \leq \rho(\check{\mathbf{J}}) + \epsilon = [1 + \rho(\check{\mathbf{J}})]/2 := \beta < 1, \qquad (3.71)$$

which in turn leads to

$$\begin{aligned}\lim_{k\to\infty}\left\|\sum_{l=0}^{k-1}\mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1}\mathbf{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}(\check{\mathbf{J}}^l)^T\mathbf{T}^T\right\| &\leq \lim_{k\to\infty}\sum_{l=0}^{k-1}\left\|\mathbf{T}\check{\mathbf{J}}^l\mathbf{T}^{-1}\mathbf{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}(\check{\mathbf{J}}^l)^T\mathbf{T}^T\right\| \\ &\leq \lim_{k\to\infty}\sum_{l=0}^{k-1}\|\mathbf{T}\|\|\check{\mathbf{J}}\|^l\|\mathbf{T}^{-1}\mathbf{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}\|\|\check{\mathbf{J}}\|^l\|\mathbf{T}^T\| \\ &\leq \frac{1}{1-\beta^2}\left(\|\mathbf{T}\|^2\|\mathbf{T}^{-1}\mathbf{\Sigma}_{\eta_n^a}\mathbf{T}^{-T}\|\right). \qquad (3.72)\end{aligned}$$

Therefore, the above partial sum matrix has bounded norm as $k \to \infty$. Similarly, using the same matrix norm $\|\cdot\|$ introduced earlier, we can prove that all the other two partial sum matrices in (3.70) also converge. Therefore, it is easy to conclude that each entry (including the diagonal ones) of the covariance matrix of noisy CA-MoM iterates, i.e., $\mathbf{\Sigma}_{\bar{\gamma}_n}(k+1)$ in (3.70), will also converge to some bounded value, which completes the proof of Proposition 3.3.

3.6.4 Proof of Proposition 3.4

Convergence in this case will be established through a non-increasing Lyapunov function [2, Sec. 6.10] of the iterates (3.22a)-(3.22c). With this objective in mind, and similar to

Appendix 3.6.3, we first express the iterations (3.22a)-(3.22c) in vector form

$$\mathbf{v}_n(k) = \mathbf{v}_n(k-1) + \frac{c}{2}\mathbf{M}^T(k)\bar{\boldsymbol{\gamma}}_n(k) \tag{3.73a}$$

$$\bar{\boldsymbol{\gamma}}_n(k+1) = \mathbf{N}_c^{-1}\left\{\boldsymbol{\gamma}_n - \bar{\mathbf{M}}\mathbf{v}_n(k) + c[2\mathbf{D} - \mathbf{L}(k)]\bar{\boldsymbol{\gamma}}_n(k)\right\} \tag{3.73b}$$

where $\mathbf{M}(k)$ is the $J\times 2|\mathcal{E}|$ time-varying incidence matrix and $\bar{\mathbf{M}} := E\{\mathbf{M}(k)\}$ is its expected value. Notice that $\bar{\mathbf{M}}$ has the same non-zero entries as $\mathbf{M}$, with each entry replaced by the corresponding link failure probability p_{ji} as prescribed in (a3.2c). Next, define the limits $\bar{\boldsymbol{\gamma}}_n^* := \bar{\gamma}_n\mathbf{1}$ and $\mathbf{v}_n^*$ such that $\bar{\mathbf{M}}\mathbf{v}_n^* := \bar{\boldsymbol{\gamma}}_n - \bar{\boldsymbol{\gamma}}_n^*$. If the network is connected, matrix $\bar{\mathbf{M}}^T$ has rank $J-1$ and its null space is $\{\mu\mathbf{1}_J\}$, $\forall\mu\in\mathbb{R}$ since $\bar{\mathbf{M}}^T\mathbf{1}_J = \mathbf{0}_{2|\mathcal{E}|}$ (note that each column $\bar{\mathbf{m}}_{p(j\to i)}$ of $\bar{\mathbf{M}}$ has only two non-zero entries $\pm p_{ji}$). This ensures the existence of vector $\mathbf{v}_n^*$ since $\bar{\boldsymbol{\gamma}}_n - \bar{\boldsymbol{\gamma}}_n^*$ is guaranteed to be in the range space of $\bar{\mathbf{M}}$. Therefore, the equivalent recursions for the difference vectors $\tilde{\boldsymbol{\gamma}}_n(k) := \bar{\boldsymbol{\gamma}}_n(k) - \bar{\boldsymbol{\gamma}}_n^*$ and $\tilde{\mathbf{v}}_n(k) := \mathbf{v}_n(k) - \mathbf{v}_n^*$ become [cf. (3.73a)-(3.73b)]

$$\tilde{\mathbf{v}}_n(k) = \tilde{\mathbf{v}}_n(k-1) + \frac{c}{2}\mathbf{M}^T(k)\tilde{\boldsymbol{\gamma}}_n(k) \tag{3.74a}$$

$$\tilde{\boldsymbol{\gamma}}_n(k+1) = \mathbf{N}_c^{-1}\left\{-\bar{\mathbf{M}}\tilde{\mathbf{v}}_n(k) + c[2\mathbf{D} - \mathbf{L}(k)]\tilde{\boldsymbol{\gamma}}_n(k)\right\}. \tag{3.74b}$$

We wish to prove that the iterations (3.74a) and (3.74b) reach an equilibrium as $k\to\infty$. To this end, consider the Lyapunov function

$$\begin{aligned}\mathcal{V}(k) &:= 2\|\tilde{\mathbf{v}}_n(k)\|^2 + 2c\tilde{\mathbf{v}}_n^T(k)\bar{\mathbf{M}}\tilde{\boldsymbol{\gamma}}_n(k+1) + 2c^2\tilde{\boldsymbol{\gamma}}_n^T(k+1)\mathbf{D}\tilde{\boldsymbol{\gamma}}_n(k+1)\\ &= 2\|\tilde{\mathbf{v}}_n(k) + \frac{c}{2}\bar{\mathbf{M}}\tilde{\boldsymbol{\gamma}}_n(k+1)\|^2 + \frac{c^2}{2}\tilde{\boldsymbol{\gamma}}_n^T(k+1)(4\mathbf{D} - \bar{\mathbf{M}}^T\bar{\mathbf{M}})\tilde{\boldsymbol{\gamma}}_n(k+1).\end{aligned} \tag{3.75}$$

The positive semi-definiteness of matrix $(4\mathbf{D}-\bar{\mathbf{M}}^T\bar{\mathbf{M}})$ ensures that $\mathcal{V}(k)$ is a valid (positive) candidate Lyapunov function of $(\tilde{\boldsymbol{\gamma}}_n(k+1), \tilde{\mathbf{v}}_n(k))$. Substituting (3.74a) and (3.74b) into (3.75) the following difference of successive Lyapunov functions becomes

$$\begin{aligned}\mathcal{V}(k-1) - \mathcal{V}(k) &= 2c\|\tilde{\boldsymbol{\gamma}}_n(k+1)\|^2 + 2c\tilde{\mathbf{v}}_n^T(k-1)[\bar{\mathbf{M}} - \mathbf{M}(k)]\tilde{\boldsymbol{\gamma}}_n(k)\\ &\quad + c^2(\tilde{\boldsymbol{\gamma}}_n(k+1) - \tilde{\boldsymbol{\gamma}}_n(k))^T(2\mathbf{D} - \mathbf{L}(k))(\tilde{\boldsymbol{\gamma}}_n(k+1) - \tilde{\boldsymbol{\gamma}}_n(k))\\ &\quad + c^2\tilde{\boldsymbol{\gamma}}_n^T(k+1)\mathbf{L}(k)\tilde{\boldsymbol{\gamma}}_n(k+1)\end{aligned} \tag{3.76}$$

Because the zero-mean matrix difference $\bar{\mathbf{M}} - \mathbf{M}(k)$ is uncorrelated with the state $(\tilde{\boldsymbol{\gamma}}_n(k), \tilde{\mathbf{v}}_n(k-1))$ at the previous iteration [cf. (a3.2c)], the expectation of (3.75) satisfies (since the last two quadratic terms in (3.75) are positive semi-definite)

$$E\mathcal{V}(k-1) - E\mathcal{V}(k) \geq 2cE\{||\tilde{\boldsymbol{\gamma}}_n(k+1)||^2\}. \tag{3.77}$$

Summing up the first K of these differences, yields

$$\sum_{k=1}^{K} E\{\|\tilde{\boldsymbol{\gamma}}_n(k)\|^2\} \leq [E\mathcal{V}(0) - E\mathcal{V}(K-1)]/(2c) \leq E\mathcal{V}(0)/(2c) \tag{3.78}$$

where the last inequality comes from the non-negativity of $\mathcal{V}(K-1)$. The partial sum of the non-negative series $\left\{E\{\|\tilde{\boldsymbol{\gamma}}_n(k)\|^2\}\right\}_{k=1}^{\infty}$ in (3.78) is bounded; thus, $E\{\|\tilde{\boldsymbol{\gamma}}_n(k)\|^2\}$ converges to zero as $k \to \infty$, and so does $E\{|\tilde{\gamma}_{jn}(k)|^2\}$, which completes the proof of (3.23).

3.6.5 Proof of Lemma 3.1

Consider first the CA-SI in (3.11), which in vector form can be expressed after backward substitution as $\bar{\boldsymbol{\gamma}}_n(k) = \mathbf{W}^k \bar{\boldsymbol{\gamma}}_n(0)$. Writing the latter entry-wise proves (3.32) after setting $c_{ji}(k) = [\mathbf{W}^k]_{ji}$.

To prove (3.32) for the CA-MoM iterations in (3.18a)-(3.18b), consider as in Appendix 3.6.3 the augmented vector recursion

$$\mathbf{s}_n^a(k) = \mathbf{W}^a \mathbf{s}_n^a(k-1) + \mathbf{b}_n^a \tag{3.79}$$

where $\mathbf{W}^a$ and $\mathbf{b}_n^a$ are as in (3.50). Initializing with $\mathbf{s}_n^a(0) := [\boldsymbol{\gamma}_n^T \ \mathbf{0}_{2|\mathcal{E}|}^T]^T$, backward substitution yields

$$\begin{aligned} \mathbf{s}_n^a(k) &= (\mathbf{W}^a)^k \, \mathbf{s}_n^a(0) + \sum_{l=0}^{k-1} (\mathbf{W}^a)^l \mathbf{b}_n^a \\ &= \sum_{i=1}^{J+2|\mathcal{E}|} \lambda_i^k s_i \mathbf{u}_i + \sum_{l=0}^{k-1} \sum_{i=1}^{J+2|\mathcal{E}|} \lambda_i^l b_i \mathbf{u}_i \end{aligned} \tag{3.80}$$

where

$$\begin{aligned} &s_i := \mathbf{w}_{i,1}^T \boldsymbol{\gamma}_n, \ b_i = \mathbf{w}_{i,1}^T \mathbf{N}_c^{-1} \boldsymbol{\gamma}_n, \ \ i = 1, 2, \ldots, , 2J-1 \\ &s_i = b_i = 0, \ \ i = 2J, \ldots, 2J + |\mathcal{E}|. \end{aligned} \tag{3.81}$$

With these definitions, the upper part of (3.80) is given by

$$\begin{aligned}\bar{\boldsymbol{\gamma}}_n(k) &= \sum_{i=1}^{2J-1} \lambda_i^k \mathbf{u}_{i,1}\mathbf{w}_{i,1}^T\boldsymbol{\gamma}_n + \sum_{l=0}^{k-1}\sum_{i=1}^{2J-1} \lambda_i^l \mathbf{u}_{i,1}\mathbf{w}_{i,1}^T\mathbf{N}_c^{-1}\boldsymbol{\gamma}_n \\ &= \sum_{i=1}^{2J-1} \left(\lambda_i^k \mathbf{u}_{i,1}\mathbf{w}_{i,1}^T + \frac{1-\lambda_i^k}{1-\lambda_i}\mathbf{u}_{i,1}\mathbf{w}_{i,1}^T\mathbf{N}_c^{-1}\right)\boldsymbol{\gamma}_n. \end{aligned} \tag{3.82}$$

Written entry-wise, (3.82) can be expressed as in (3.32), which proves the lemma for the CA-MoM iterates, provided that one initializes (3.18a) and (3.18b) with $[\boldsymbol{\gamma}_n^T, \mathbf{0}^T]^T$.

Chapter 4

Distributed In-Network Demodulation

This chapter deals with distributed demodulation of space-time transmissions from a multi-antenna access point to a wireless sensor network. Based on local message exchanges with single-hop neighboring sensors, two algorithms are developed for distributed demodulation in the broadcast link. The first algorithm has sensors to consent on the demodulated symbols. By relaxing the finite-alphabet constraints on the symbols, the demodulation task is formulated as a distributed convex optimization problem that is solved iteratively using the method of multipliers. Distributed versions of the centralized zero-forcing (ZF) and minimum mean-square error (MMSE) demodulators follow as special cases. The second algorithm leads sensors to iteratively reach consensus on the average of (cross-) covariances of locally available per-sensor data vectors with the channel matrices, which constitute sufficient statistics for maximum likelihood demodulation. Distributed versions of the sphere decoding algorithm and the ZF/MMSE demodulators are also developed. These algorithms offer distinct merits in terms of error performance and resilience to non-ideal inter-sensor links. In both cases, the per-iteration error performance is analyzed, and the approximate number of iterations needed to attain a prescribed error rate are quantified. Simulated tests verify the analytical claims. Interestingly, only a few consensus iterations suffice for the novel distributed demodulators to approach the performance of their centralized counterparts.

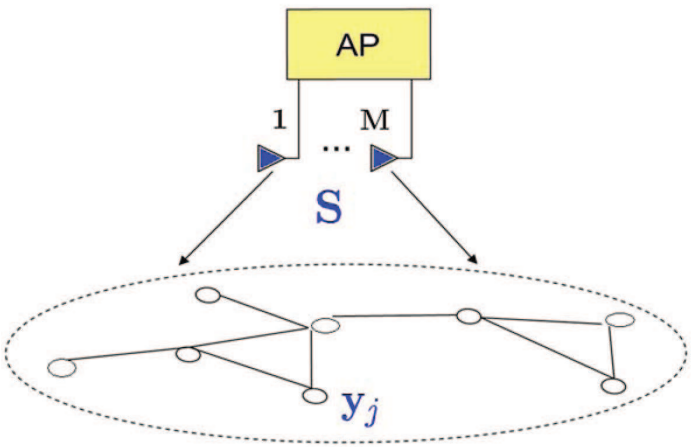


Figure 4.1: Cooperative demodulation system model.

4.1 Modeling and Problem Statement

Consider a (possibly mobile) access point (AP) equipped with M antennas as depicted in Fig. 4.1. Constellation symbols at the AP are mapped to an $M \times N$ space-time matrix $\mathbf{S}$ belonging to a finite alphabet $\mathcal{A}$, where N is the number of time slots. The AP broadcasts $\mathbf{S}$ to a connected *ad hoc* WSN with J sensors. The WSN is modeled as a graph $\mathcal{G} := \{\mathcal{E}, \mathcal{J}\}$, where $\mathcal{J} := \{1, \ldots, J\}$ denotes the set of sensors, and $\mathcal{E} \subset \mathcal{J} \times \mathcal{J}$ the set of available communication links (graph edges). The set of neighbors of sensor j is denoted by $\mathcal{N}_j \subseteq \mathcal{J}$.

With reference to Fig. 4.1, the $N \times 1$ received block $\mathbf{y}_j$ at the j-th sensor is given by the following input/output (I/O) relationship

$$\mathbf{y}_j = \mathbf{S}^T \mathbf{h}_j + \boldsymbol{\epsilon}_j \tag{4.1}$$

where $\mathbf{h}_j$ denotes the $M \times 1$ AP-to-sensor j fading channel, and $\boldsymbol{\epsilon}_j \sim \mathcal{N}(\mathbf{0}_N, \mathbf{I}_N)$ stands for additive white Gaussian noise (AWGN) that is assumed uncorrelated across sensors, and without loss of generality (w.l.o.g.) of zero mean and unit variance. Each sensor j acquires $\mathbf{h}_j$ through training. For notational brevity, but also w.l.o.g., focus will be placed on real baseband equivalent models instead of complex ones. Indeed, by concatenating the real and imaginary parts of a vector, the complex model can be readily converted to a real model; see e.g., [17, Sec 3.5.6].

With the definitions $\mathbf{H}_j := \mathbf{I}_N \otimes \mathbf{h}_j^T$ and $\mathbf{s} := \text{vec}(\mathbf{S})$, the I/O relationship in (4.1) can be rewritten as $\mathbf{y}_j = \mathbf{H}_j\mathbf{s} + \boldsymbol{\epsilon}_j$. And with $\mathbf{y} := [\mathbf{y}_1^T, \ldots, \mathbf{y}_J^T]^T$ collecting the received blocks across sensors, the $NJ \times 1$ vector $\mathbf{y}$ can be compactly expressed as

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \boldsymbol{\epsilon} \tag{4.2}$$

where $\mathbf{H} := [\mathbf{H}_1^T, \ldots, \mathbf{H}_J^T]^T$, and $\boldsymbol{\epsilon} := [\boldsymbol{\epsilon}_1^T, \ldots, \boldsymbol{\epsilon}_J^T]^T$. Since the noise $\boldsymbol{\epsilon}$ has uncorrelated entries of unit variance, the *centralized* ML demodulator for (4.2) amounts to finding

$$\hat{\mathbf{s}}_{ML} = \arg \max_{\mathbf{s} \in \mathcal{A}^{NM}} -\|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 = \arg \max_{\mathbf{s} \in \mathcal{A}^{NM}} -\sum_{j=1}^{J} \|\mathbf{y}_j - \mathbf{H}_j\mathbf{s}\|^2. \tag{4.3}$$

Clearly, if $\{\mathbf{y}_j, \mathbf{H}_j\}_{j=1}^J$ were available at a central location (or at all sensors), then $\hat{\mathbf{s}}_{ML}$ in (4.3) could be centrally (locally) found. However, $\{\mathbf{y}_j, \mathbf{H}_j\}_{j=1}^J$ are distributed across the network. The objective of this chapter is to find suitable single-hop inter-sensor exchanges such that each sensor is able to solve the centralized demodulation problem (or a relaxed version thereof) in a *distributed* fashion.

Remark 4.1 The centralized ML demodulation problem in (4.3) can be viewed as a multiple hypotheses testing problem, where each hypothesis is a block $\mathbf{S}$ with entries drawn from $\mathcal{A}$. Therefore, this problem can in principle be solved by the distributed hypotheses testing (DHT) algorithms developed for sensor networks in e.g., [23, 30], and [43]. However, these schemes - designed primarily for detection problems entailing a few hypotheses - require implementation of a distributed algorithm for each hypothesis. When it comes to demodulating wireless transmissions, their complexity grows exponentially with the constellation size and, for the present space-time broadcast setup, with the product NM. Using knowledge of the alphabet and the modulation scheme, the distinct contribution of this chapter is twofold: (i) reduce the complexity of the DHT algorithms in [23, 30, 43]; and (ii) provide thorough error analysis for distributed demodulation and equalization. These tasks were not addressed in Chapter 3, which dealt with in-network hard and soft channel decoding.

4.2 Distributed Linear Demodulators

This section introduces a linear distributed demodulation algorithm for solving (4.3). To this end, the centralized linear demodulation task is formulated as a convex optimization problem that can be distributed across sensors. Two popular linear demodulators will be considered jointly: the zero-forcing (ZF) and the minimum mean-square error (MMSE) demodulators.

Recall first that the centralized ZF demodulator is given by [cf. (4.3)]

$$\hat{\mathbf{s}}_{ZF} = \arg\min_{\mathbf{s}} \frac{1}{2}\sum_{j=1}^{J} \|\mathbf{y}_j - \mathbf{H}_j\mathbf{s}\|^2 = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y}. \tag{4.4}$$

The centralized MMSE demodulator, namely $\hat{\mathbf{s}}_{MMSE} = E\{\mathbf{s}\mathbf{y}^T\}E^{-1}\{\mathbf{y}\mathbf{y}^T\}\mathbf{y}$, can also be expressed in closed form as

$$\hat{\mathbf{s}}_{MMSE} = \sigma_s^2\mathbf{H}^T(\sigma_s^2\mathbf{H}\mathbf{H}^T + \mathbf{I}_{NJ})^{-1}\mathbf{y} = (\mathbf{H}^T\mathbf{H} + \sigma_s^{-2}\mathbf{I}_{NM})^{-1}\mathbf{H}^T\mathbf{y} \tag{4.5}$$

where $\sigma_s^2 := E\{s_\ell^2\}$, $\ell = 1, \ldots, NM$ denotes the average symbol energy; and the second equality comes from the matrix inversion lemma. Similar to the ZF demodulator, (4.5) can be viewed as the solution of an unconstrained LS problem as follows

$$\hat{\mathbf{s}}_{MMSE} = \left(\sum_{j=1}^{J}\mathbf{H}_j^T\mathbf{H}_j + \sigma_s^{-2}\mathbf{I}_{NM}\right)^{-1}\left(\sum_{j=1}^{J}\mathbf{H}_j^T\mathbf{y}_j\right) = \arg\min_{\mathbf{s}}\frac{1}{2}\sum_{j=1}^{J}\|\mathbf{y}_j' - \mathbf{H}_j'\mathbf{s}\|^2 \tag{4.6}$$

where $\mathbf{y}_j' := [\mathbf{y}_j^T, \mathbf{0}_{NM}^T]^T$, and $\mathbf{H}_j' := [\mathbf{H}_j^T, (\sigma_s\sqrt{J})^{-1}\mathbf{I}_{NM}]^T$.

Equation (4.6) shows that the centralized MMSE demodulator is obtained from an LS minimization problem similar to the one in (4.4). Hence, all the ensuing results developed for distributed ZF demodulators carry over to the MMSE ones by simply substituting $\mathbf{H}_j'$ for $\mathbf{H}_j$ and $\mathbf{y}_j'$ for $\mathbf{y}_j$.

4.2.1 Distributed Consensus on Demodulated Symbols

The objective of this section is to solve to (4.4) in a distributed fashion through message exchanges among single-hop neighboring sensors. This task will be accomplished using the

method of multipliers (MoM) [7, Sec. 3.4.4]. The MoM can afford distributed implementation through *local auxiliary* variables $\mathbf{s}_j$, which represent the wanted ZF solution per sensor j. All these auxiliary variables $\boldsymbol{s} := \{\mathbf{s}_j\}_{j=1}^J$ can be obtained by minimizing the following consensus-constrained quadratic cost function

$$\min_{\boldsymbol{s}} \ \frac{1}{2}\sum_{j=1}^{J} \|\mathbf{y}_j - \mathbf{H}_j\mathbf{s}_j\|^2 \quad \text{s.t.} \ \ \mathbf{s}_j - \mathbf{s}_i = \mathbf{0}_{NM}, \quad j \in \mathcal{J},\ i \in \mathcal{N}_j. \tag{4.7}$$

Thanks to $\mathbf{s}_j$, the sum-cost can be decoupled and each summand can be minimized separately per sensor j. On the other hand, the neighborhood consensus constraint in (4.7) ensures that consensus is achieved over the entire network, which was assumed connected. Network connectivity provides a sufficient condition to guarantee that the optimum of (4.7) is achieved at the solution $\hat{\mathbf{s}}_{ZF}$ of (4.4).

After introducing an additional set of variables $\boldsymbol{z} := \left\{\{\mathbf{z}_{ji}, \mathbf{z}'_{ji}\}_{i\in\mathcal{N}_j}\right\}_{j\in\mathcal{J}}$, the problem (4.7) can be equivalently written as

$$\begin{aligned} \min_{\boldsymbol{s},\boldsymbol{z}} \quad & \frac{1}{2}\sum_{j=1}^{J} \|\mathbf{y}_j - \mathbf{H}_j\mathbf{s}_j\|^2 \\ \text{s.t.} \quad & \mathbf{s}_j - \mathbf{z}_{ji} = \mathbf{0}_{NM},\ \mathbf{s}_i + \mathbf{z}'_{ji} = \mathbf{0}_{NM},\ \mathbf{z}_{ji} + \mathbf{z}'_{ji} = \mathbf{0}_{NM} \quad j \in \mathcal{J},\ i \in \mathcal{N}_j \end{aligned} \tag{4.8}$$

Let $\mathbf{v}_{ji}$ and $\mathbf{v}'_{ji}$ denote the Lagrange multipliers associated with the constraints $\mathbf{s}_j - \mathbf{z}_{ji} = \mathbf{0}_{NM}$ and $\mathbf{s}_i + \mathbf{z}'_{ji} = \mathbf{0}_{NM}$, respectively. Likewise, define the set $\mathcal{C}_z := \{\boldsymbol{z} : \mathbf{z}_{ji} + \mathbf{z}'_{ji} = \mathbf{0}_{NM},\ \forall j \in \mathcal{J},\ i \in \mathcal{N}_j\}$ that represents the constraints on the entries of $\boldsymbol{z}$. With $\alpha > 0$ denoting a penalty coefficient, consider the augmented Lagrangian function of (4.8), namely

$$\begin{aligned} \mathcal{L}_a(\boldsymbol{s},\boldsymbol{z},\boldsymbol{v},\boldsymbol{v}') \ = \ & \frac{1}{2}\sum_{j=1}^{J} \|\mathbf{y}_j - \mathbf{H}_j\mathbf{s}_j\|^2 + \sum_{j=1}^{J}\sum_{i\in\mathcal{N}_j} \left\{\mathbf{v}_{ji}^T(\mathbf{s}_j - \mathbf{z}_{ji}) + (\mathbf{v}'_{ji})^T(-\mathbf{s}_i - \mathbf{z}'_{ji})\right\} \\ & + \frac{\alpha}{2}\sum_{j=1}^{J}\sum_{i\in\mathcal{N}_j} \left\{\|\mathbf{s}_j - \mathbf{z}_{ji}\|^2 + \| - \mathbf{s}_i - \mathbf{z}'_{ji}\|^2\right\} \end{aligned} \tag{4.9}$$

where the set $\boldsymbol{v} := \left\{\{\mathbf{v}_{ji}\}_{i\in\mathcal{N}_j}\right\}_{j\in\mathcal{J}}$, and likewise for $\boldsymbol{v}'$.

The alternating-direction MoM operates by minimizing $\mathcal{L}_a$ in (4.9) cyclically with respect to (w.r.t.) one set of variables given the other variables, considering the constraint set $\mathcal{C}_z$. Appendix 3.6.2 shows that under proper initialization, the variables $\boldsymbol{v}'$

and $\left\{\{\mathbf{z}'_{ji}\}_{i\in\mathcal{N}_j}\right\}_{j\in\mathcal{J}}$ can be eliminated, and the k-th iteration of the MoM solver becomes

$$\mathbf{v}_{ji}(k)=\mathbf{v}_{ji}(k-1)+\frac{\alpha}{2}(\mathbf{s}_j(k)-\mathbf{s}_i(k)),\ \ j\in\mathcal{J},\ i\in\mathcal{N}_j \tag{4.10a}$$

$$\mathbf{s}_j(k+1)=(\mathbf{H}_j^T\mathbf{H}_j+2\alpha|\mathcal{N}_j|\mathbf{I}_{NM})^{-1}\left\{\mathbf{H}_j^T\mathbf{y}_j-\sum_{i\in\mathcal{N}_j}[\mathbf{v}_{ji}(k)-\mathbf{v}_{ij}(k)-\alpha(\mathbf{s}_j(k)+\mathbf{s}_i(k))]\right\},\ \ j\in\mathcal{J}. \tag{4.10b}$$

Iterations (4.10a) and (4.10b) constitute the DC-DS algorithm. Sensor $j\in\mathcal{J}$ maintains the local estimate of the ZF solution $\mathbf{s}_j(k)$ and all the multipliers $\{\mathbf{v}_{ji}(k)\}_{i\in\mathcal{N}_j}$. During the k-th iteration, sensor j receives the broadcasted estimates $\mathbf{s}_i(k)$ from all its neighboring sensors $i\in\mathcal{N}_j$, and updates the corresponding multipliers via (4.10a). It then transmits back the updated multiplier $\mathbf{v}_{ji}(k)$ to each of its neighboring sensors $i\in\mathcal{N}_j$, based on which each sensor j is able to determine $\mathbf{s}_j(k+1)$ via (4.10b). Subsequently, all sensors $j\in\mathcal{J}$ broadcast their updated estimates $\mathbf{s}_j(k+1)$ to their neighbors, thus completing the k-th iteration and initializing the next one.

Notice that the overall number of scalars required to consent on is NM, regardless of the number of hypotheses $|\mathcal{A}|^{NM}$. This presents considerable communication savings compared to the DHT solvers of (4.3) in [30] and [43]. Equally important, the iterates in (4.10a)-(4.10b) are provably convergent, as asserted in the following proposition.

Proposition 4.1 *(DC-DS with ideal inter-sensor links.) The iterations (4.10a) and (4.10b) with arbitrary initialization of* $\mathbf{s}_j(1)$ *and* $\mathbf{v}_{ji}(0)$, $\forall(j,i)\in\mathcal{E}$ *and* $\alpha>0$, *reach consensus to the centralized ZF demodulation* $\hat{\mathbf{s}}_{ZF}$ *in (4.4) as* $k\to\infty$; *i.e.,*

$$\lim_{k\to\infty}\mathbf{s}_j(k)=\hat{\mathbf{s}}_{ZF},\quad\forall j\in\mathcal{J}. \tag{4.11}$$

Proof: Appendix 4.5.1 shows that iterations (4.10a)-(4.10b) are equivalent to the MoM approach in [7, pg. 255]. As the cost function in (4.8) is convex and the problem constraints comply with [7, Assumption 4.1, pg. 255], the iterates converge to the optimal solution to (4.8) as established by [7, Proposition 4.2, pg. 256]. □

Remark 4.2 *(Imperfect inter-sensor links.)* Supposing that sufficiently powerful error control codes are employed, the inter-sensor messages involved in the DC-DS iterations have been assumed ideal; i.e., local exchanges are received error free. However, (4.10a) and (4.10b) can be modified to also accommodate non-ideal inter-sensor links affected by additive noise and random link failures. Chapter 3 provides the details on these modified iterations. Mimicking Chapter 3, it can be shown that if the inter-sensor links are corrupted with additive noises, the DC-DS will converge in the mean, i.e., $\lim_{k\to\infty} E\{\mathbf{s}_j(k) - \hat{\mathbf{s}}_{ZF}\} = \mathbf{0}_{NM}$, with a bounded variance. Moreover, if the inter-sensor links fail randomly and the failures follow a Bernoulli process, then DC-DS will converge in the mean-square sense (m.s.s.); i.e., $\lim_{k\to\infty} E\left\{\|\mathbf{s}_j(k) - \hat{\mathbf{s}}_{ZF}\|^2\right\} = 0$.

Remark 4.3 *(Comparison with [45].)* The DC-DS algorithm of this section is related to the consensus-based distributed best linear unbiased estimators in [45]. Compared to [45], the algorithm here offers three distinct novelties: (i) it does not require a bridge sensor set with which sensors need to communicate, thus offering a *fully* distributed approach; (ii) it is provably convergent in the presence of inter-sensor link failures; and (iii) it is possible to analyze the error performance per iteration, which is the topic of the next subsection.

4.2.2 Performance Analysis

The key to evaluating error performance per iteration k is to specify the relationship between $\mathbf{s}_j(k)$ and $\mathbf{s}$ as a function of k. To this end, the following lemma is instrumental.

Lemma 4.1 *The consensus-based ZF iterates in (4.10a)-(4.10b) can be expressed as*

$$\mathbf{s}_j(k) = \sum_{i=1}^{J} \mathbf{C}_{ji}(k)\mathbf{y}_i, \quad \forall j \in \mathcal{J} \tag{4.12}$$

where the coefficient matrix $\mathbf{C}_{ji}(k)$ depends solely on the network topology and α.

Proof: See Appendix 4.5.2. □

Using (4.12) into (4.11), Proposition 4.1 and (4.4) imply that $\lim_{k\to\infty} \mathbf{C}_{ji}(k) = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}_i^T$, $\forall j$. Substituting $\mathbf{y}_i$ from (4.1) into (4.12), it is further possible to express

$\mathbf{s}_j(k)$ as

$$\mathbf{s}_j(k) = \mathbf{G}_j(k)\mathbf{s} + \mathbf{w}_j(k) \tag{4.13}$$

where $\mathbf{G}_j(k)$ and $\mathbf{w}_j(k)$ are defined, respectively, as

$$\mathbf{G}_j(k) := \sum_{i=1}^{J} \mathbf{C}_{ji}(k)\mathbf{H}_i, \quad \text{and} \quad \mathbf{w}_j(k) := \sum_{i=1}^{J} \mathbf{C}_{ji}(k)\boldsymbol{\epsilon}_i. \tag{4.14}$$

Vector $\mathbf{w}_j(k)$ denotes zero-mean colored Gaussian noise with covariance matrix $\mathbf{\Sigma}_j(k) := \sum_{i=1}^{J} \mathbf{C}_{ji}(k)\mathbf{C}_{ji}^T(k)$. Using these definitions, the ℓ-th entry of $\mathbf{s}_j(k)$ satisfies [cf. (4.13)]

$$s_{j,\ell}(k) = g_{j,\ell\ell}(k)s_\ell + \sum_{\substack{\ell=1 \\ \ell' \neq \ell}}^{NM} g_{j,\ell\ell'}(k)s_{\ell'} + w_{j,\ell}(k) \tag{4.15}$$

where $g_{j,\ell\ell'}(k)$ and $w_{j,\ell}(k)$ are the (ℓ,ℓ')-th and ℓ-th entries of $\mathbf{G}_j(k)$ and $\mathbf{w}_j(k)$, respectively. The last two terms in the right-hand side of (4.15) capture interference-plus-noise effects with variance

$$\tilde{\sigma}_{j,\ell}^2(k) := \text{var}\left\{ \sum_{\substack{\ell=1 \\ \ell' \neq \ell}}^{NM} g_{j,\ell\ell'}(k)s_{\ell'} + w_{j,\ell}(k) \right\} = \sum_{\substack{\ell=1 \\ \ell' \neq \ell}}^{NM} g_{j,\ell\ell'}^2(k)\sigma_s^2 + \Sigma_{j,\ell\ell}(k) \tag{4.16}$$

where $\Sigma_{j,\ell\ell}(k)$ is the (ℓ,ℓ)-th entry of $\mathbf{\Sigma}_j(k)$. Using (4.16), the SNR in (4.15) becomes

$$\rho_{j,\ell}(k) = E\{[g_{j,\ell\ell}(k)s_\ell]^2\}/\tilde{\sigma}_{j,\ell}^2(k) = g_{j,\ell\ell}^2(k)\sigma_s^2/\tilde{\sigma}_{j,\ell}^2(k). \tag{4.17}$$

Based on (4.17), the symbol error rate (SER) can be readily obtained in closed form or bounded as in e.g., [35, Sec. 4.3] for all popular one- or two-dimensional constellation points $s_\ell \in \mathcal{A}$ transmitted over an AWGN channel with SNR $\rho_{j,\ell}(k)$. As an example, if s_ℓ is drawn from a $q-$ary PAM constellation, the optimal SER per sensor j at iteration k is given by

$$P_{e,\ell}^j(k) = 2\left(1 - \frac{1}{q}\right) Q\left(\sqrt{\frac{3}{q^2-1}\rho_{j,\ell}(k)}\right) \tag{4.18}$$

where $Q(x) := (1/\sqrt{2\pi}) \int_x^\infty \exp(-t^2/2)dt$ denotes the Gaussian tail function.

Such SER expressions can be easily obtained also when inter-sensor links are corrupted by zero-mean additive noise. Because the iterations are linear, the iterates $\mathbf{s}_j(k)$ in (4.12)

will now include an extra colored noise term with covariance determined by the inter-sensor noise level and the network topology. To obtain the SNR in this case, it suffices to incorporate this extra noise term into $\mathbf{w}_j(k)$, update the variance term $\tilde{\sigma}^2_{j,\ell}$ in (4.16), and plug it into (4.17).

4.3 Distributed Consensus on Sufficient Statistics

An alternative approach for solving (4.3) in a distributed fashion is to have all sensors agree on minimal sufficient statistics for the demodulation problem. The motivation behind this approach is twofold: (i) reduce the communication overhead per sensor for a prescribed SER; and (ii) allow for more general (possibly non-linear) (near-) optimal demodulators.

Bearing these goals in mind, the ML demodulator in (4.3) can be re-expressed as

$$\hat{\mathbf{s}}_{ML} = \arg \max_{\mathbf{s}\in\mathcal{A}^{NM}} \left\{ 2 \left(\sum_{j=1}^{J} \mathbf{H}_j^T \mathbf{y}_j \right)^T \mathbf{s} - \mathbf{s}^T \left(\sum_{j=1}^{J} \mathbf{H}_j^T \mathbf{H}_j \right) \mathbf{s} \right\}. \tag{4.19}$$

Upon defining the sample cross-covariance between the received block $\mathbf{y}_j$ and the channel $\mathbf{H}_j$ as $\boldsymbol{\varphi}_j := \mathbf{H}_j^T \mathbf{y}_j$, and the channel's sample covariance matrix as $\boldsymbol{\Gamma}_j := \mathbf{H}_j^T \mathbf{H}_j$, the ML demodulator in (4.19) is equivalent to

$$\hat{\mathbf{s}}_{ML} = \arg \max_{\mathbf{s}\in\mathcal{A}^{NM}} \left\{ 2 \left(\frac{1}{J} \sum_{j=1}^{J} \boldsymbol{\varphi}_j \right)^T \mathbf{s} - \mathbf{s}^T \left(\frac{1}{J} \sum_{j=1}^{J} \boldsymbol{\Gamma}_j \right) \mathbf{s} \right\}. \tag{4.20}$$

Thus, in order to solve (4.20) locally, it suffices for each sensor to acquire the following averages:

$$\bar{\boldsymbol{\varphi}} := \frac{1}{J} \sum_{j=1}^{J} \boldsymbol{\varphi}_j, \quad \text{and} \quad \bar{\boldsymbol{\Gamma}} := \frac{1}{J} \sum_{j=1}^{J} \boldsymbol{\Gamma}_j. \tag{4.21}$$

Given $\bar{\boldsymbol{\varphi}}$ and $\bar{\boldsymbol{\Gamma}}$, the optimal $\hat{\mathbf{s}}_{ML}$ in (4.20) depends only on the symbol constellation $\mathcal{A}$, that is assumed available at all sensors. The main insight is summarized in the following proposition:

Proposition 4.2 *(Sufficient statistics for demodulation.) In order to solve the ML demodulation problem (4.3), it suffices for all sensors to consent on $\bar{\boldsymbol{\varphi}}$ and $\bar{\boldsymbol{\Gamma}}$ in (4.21), which are known to constitute minimal sufficient statistics for the centralized demodulator in (4.3).*

The averages $\bar{\boldsymbol{\varphi}}$ and $\bar{\boldsymbol{\Gamma}}$ are also sufficient statistics for sub-optimal (non-ML) and (near-) optimal demodulation algorithms such as:

- *Distributed ZF and MMSE demodulators,* which from (4.4) and (4.5) take the form

$$\hat{\mathbf{s}}_{ZF} = \bar{\boldsymbol{\Gamma}}^{-1}\bar{\boldsymbol{\varphi}} \tag{4.22}$$

$$\hat{\mathbf{s}}_{MMSE} = \left(\bar{\boldsymbol{\Gamma}} + \frac{1}{J\sigma_s^2}\mathbf{I}_{NM}\right)^{-1}\bar{\boldsymbol{\varphi}}. \tag{4.23}$$

- *Distributed sphere decoder (SD),* which uses the fact that $\bar{\boldsymbol{\Gamma}}$ is Hermitian positive definite, and thus admits a Cholesky decomposition of the form $\bar{\boldsymbol{\Gamma}} = \bar{\boldsymbol{\Gamma}}^{1/2}\left(\bar{\boldsymbol{\Gamma}}^{1/2}\right)^T$ [18, pp. 163]. Based on the latter, the ML demodulator in (4.20) can be re-expressed as

$$\begin{aligned}\hat{\mathbf{s}}_{ML} &= \arg\max_{\mathbf{s}\in\mathcal{A}^{NM}}\left\{2\bar{\boldsymbol{\varphi}}^T\bar{\boldsymbol{\Gamma}}^{-1/2}\bar{\boldsymbol{\Gamma}}^{1/2}\mathbf{s} - \mathbf{s}^T\bar{\boldsymbol{\Gamma}}^{1/2}\left(\bar{\boldsymbol{\Gamma}}^{1/2}\right)^T\mathbf{s} - \bar{\boldsymbol{\varphi}}^T\bar{\boldsymbol{\Gamma}}^{-1/2}\left(\bar{\boldsymbol{\Gamma}}^{-1/2}\right)^T\bar{\boldsymbol{\varphi}}\right\}\\ &= \arg\max_{\mathbf{s}\in\mathcal{A}^{NM}}\left\{-\left\|\tilde{\mathbf{y}} - (\bar{\boldsymbol{\Gamma}}^{1/2})^T\mathbf{s}\right\|^2\right\}\end{aligned} \tag{4.24}$$

  where $\tilde{\mathbf{y}} := (\bar{\boldsymbol{\Gamma}}^{-1/2})^T\bar{\boldsymbol{\varphi}}$. Notice that $\tilde{\mathbf{y}}$ and $(\bar{\boldsymbol{\Gamma}}^{1/2})^T$ can be viewed as the receive vector and channel matrix of an equivalent channel whose demodulation objective as a function of $\mathbf{s}$ in (4.24) coincides with that of (4.20). Hence, the SD algorithm can be readily applied in a distributed fashion so long as each sensor j acquires the sufficient statistics $\bar{\boldsymbol{\varphi}}$ and $\bar{\boldsymbol{\Gamma}}$.

Following similar arguments, other demodulation algorithms such as decision-feedback [35, Ch. 10.3], or semi-definite relaxation [48] whose objective function is expressible in terms of (cross-) covariance matrices, can also be written in terms of $\bar{\boldsymbol{\gamma}}$ and $\bar{\boldsymbol{\varphi}}$.

The remaining task is to derive algorithms that reach consensus on each entry of $\bar{\boldsymbol{\Gamma}}$ and $\bar{\boldsymbol{\varphi}}$ in a distributed fashion. This is the theme of the ensuing subsection.

4.3.1 Consensus Averaging Algorithms

The averages $\bar{\boldsymbol{\varphi}}$ and $\bar{\boldsymbol{\Gamma}}$ can be found in a distributed fashion by applying the CA algorithm of e.g., [54] per entry. However, when inter-sensor links suffer from noise these algorithms experience slow convergence. For this reason, we will use a CA variant based on the MoM (CA-MoM), in which the average value is found iteratively as the solution of a distributed

optimization problem; see Section 3.3. In either CA or CA-MoM, if the algorithm stops before finding the exact sufficient statistics for (4.3), each sensor will be allowed to demodulate the signal using the information collected up to that point.

To simplify notation, the scalar variable x_j will be used to denote any entry of $\boldsymbol{\varphi}_j$ or $\boldsymbol{\Gamma}_j$ we wish to consent on at sensor j. The k-th iterate at sensor j will be denoted as $\bar{x}_j(k)$. When consensus is achieved, $\lim_{k\to\infty} \bar{x}_j(k) = \bar{x} := (1/J)\sum_{j=1}^{J} x_j$, which is equal to the corresponding entry in $\bar{\boldsymbol{\varphi}}$ or $\bar{\boldsymbol{\Gamma}}$ [cf. (4.21)].

The CA-MoM approach originates from the well-known fact that the sample average can be viewed as the solution of an LS cost, namely $\bar{x} = \arg\min_\theta \frac{1}{2}\sum_{j=1}^{J}(x_j-\theta)^2 = \arg\min_\theta \frac{1}{2}\|\mathbf{x}-\mathbf{1}_J\theta\|^2$. The CA-MoM iterations for this problem are given by

$$v_{ji}(k) = v_{ji}(k-1) + \frac{\alpha}{2}(\bar{x}_j(k) - \bar{x}_i(k)), \quad j \in \mathcal{J},\ i \in \mathcal{N}_j \tag{4.25a}$$

$$\bar{x}_j(k+1) = \frac{1}{1+2\alpha|\mathcal{N}_j|}\left\{x_j - \sum_{i\in\mathcal{N}_j}\left[v_{ji}(k) - v_{ij}(k) - \alpha(\bar{x}_j(k)+\bar{x}_i(k))\right]\right\}, \quad j \in \mathcal{J} \tag{4.25b}$$

for any $\alpha > 0$. Indeed, upon setting $\mathbf{H} = 1$ and substituting $\mathbf{y}_j$ for x_j, the DC-DS iterations (4.10a)-(4.10b) boil down to (4.25a)-(4.25b). The iterates in (4.25a)-(4.25b) are provably convergent under noisy or random failing inter-sensor links; see Section 3.3.

Remark 4.4 *(Communication savings.)* Recalling that $\mathbf{H}_j := \mathbf{I}_N \otimes \mathbf{h}_j^T$, and due to the fact that $\boldsymbol{\Gamma}_j$ is symmetric, the problem of consenting on each entry of the $NM \times NM$ matrix can be reduced to consenting on $M(M+1)/2$ scalar entries. Thus, including $\boldsymbol{\varphi}_j$, the overall number of scalars required to consent on is $NM + M(M+1)/2$, which is quadratic in M and linear in N but not dependent on the number of hypotheses $|\mathcal{A}|^{NM}$.

4.3.2 Performance analysis

This section derives bounds for the SER performance of the DC-SS algorithm as a function of the number of iterations k, even before consensus on the *exact* (cross-) covariance is achieved. We will first focus on the ML detector to benchmark the (near-) ML solution provided by the SD in (4.20). Subsequently, we will consider the performance of the simpler ZF and MMSE demodulators in (4.22) and (4.23), respectively.

1) Pairwise error probability bound for distributed ML demodulation. Invoking the union bound, the error probability of the ML demodulator in (4.3) conditioned on the channel, namely $P_{e|h}$, can be bounded by the pairwise error probability (PEP), which is the probability of erroneously detecting $\mathbf{s}$ as $\mathbf{s}' \in \mathcal{A}^{NM}$ with $\mathbf{s}' \neq \mathbf{s}$. Letting $P^j_{\mathbf{s}\to\mathbf{s}'|h}(k)$ denote the conditional PEP at sensor j and iteration k, it holds that $P^j_{e|h}(k) \leq \sum_{\mathbf{s}\in\mathcal{A}^{NM}} P_{\mathbf{s}} \sum_{\mathbf{s}'\neq\mathbf{s}} P^j_{\mathbf{s}\to\mathbf{s}'|h}(k)$, where $P_{\mathbf{s}}$ is the probability of transmitting $\mathbf{s}$. Given $\bar{\boldsymbol{\varphi}}_j(k)$ and $\bar{\boldsymbol{\Gamma}}_j(k)$ per sensor j at iteration k, the local ML estimate of $\mathbf{s}$, namely $\hat{\mathbf{s}}_j(k)$, becomes

$$\hat{\mathbf{s}}_j(k) = \arg \max_{\mathbf{s}\in\mathcal{A}^{NM}} \left\{ 2\bar{\boldsymbol{\varphi}}_j^T(k)\mathbf{s} - \mathbf{s}^T\bar{\boldsymbol{\Gamma}}_j(k)\mathbf{s} \right\}. \tag{4.26}$$

The PEP for the symbols detected as in (4.26) is clearly

$$P^j_{\mathbf{s}\to\mathbf{s}'|h}(k) = \Pr\left[2\bar{\boldsymbol{\varphi}}_j^T(k)\mathbf{s} - \mathbf{s}^T\bar{\boldsymbol{\Gamma}}_j(k)\mathbf{s} < 2\bar{\boldsymbol{\varphi}}_j^T(k)\mathbf{s}' - \mathbf{s}'^T\bar{\boldsymbol{\Gamma}}_j(k)\mathbf{s}' \right]. \tag{4.27}$$

Using Lemma 4.1, the running averages can be expressed as

$$\bar{\boldsymbol{\varphi}}_j(k) = \frac{1}{J}\sum_{i=1}^{J} c_{ij}(k)\boldsymbol{\varphi}_i, \quad \text{and} \quad \bar{\boldsymbol{\Gamma}}_j(k) := \frac{1}{J}\sum_{i=1}^{J} c_{ij}(k)\boldsymbol{\Gamma}_i \tag{4.28}$$

where the weights $c_{ij}(k)$ are uniquely characterized by the network topology and by the coefficient α in (4.25a)-(4.25b). In the limit, it holds that $\lim_{k\to\infty} c_{ij}(k) = 1$. Based on (4.28) and after substituting $\boldsymbol{\varphi}_i = \mathbf{H}_i^T\mathbf{H}_i\mathbf{s} + \mathbf{H}_i^T\boldsymbol{\epsilon}_i$ into (4.27), we find

$$\begin{aligned} P^j_{\mathbf{s}\to\mathbf{s}'|h}(k) &= \Pr\left[(\mathbf{s}-\mathbf{s}')^T\bar{\boldsymbol{\Gamma}}_j(k)(\mathbf{s}-\mathbf{s}') - 2\left(\frac{1}{J}\sum_{i=1}^{J} c_{ij}(k)\mathbf{H}_i^T\boldsymbol{\epsilon}_i\right)^T(\mathbf{s}-\mathbf{s}') < 0 \right] \\ &= Q\left(\frac{(\mathbf{s}-\mathbf{s}')^T\bar{\boldsymbol{\Gamma}}_j(k)(\mathbf{s}-\mathbf{s}')}{\sqrt{2(\mathbf{s}-\mathbf{s}')^T\tilde{\boldsymbol{\Gamma}}_j(k)(\mathbf{s}-\mathbf{s}')}} \right) \end{aligned} \tag{4.29}$$

where $\tilde{\boldsymbol{\Gamma}}_j(k) := (1/J^2)\sum_{i=1}^{J} c_{ij}^2(k)\boldsymbol{\Gamma}_i$, and the second equality follows from the Gaussianity of $\boldsymbol{\epsilon}_i$. Notice that when consensus is achieved, $c_{ij}(k) = 1$ and so $\tilde{\boldsymbol{\Gamma}}_j(k) = (1/J)\bar{\boldsymbol{\Gamma}}$. In this case, the PEP becomes

$$\lim_{k\to\infty} P^j_{\mathbf{s}\to\mathbf{s}'|h}(k) = Q\left(\sqrt{\frac{1}{2}(\mathbf{s}-\mathbf{s}')^T\left(\sum_{j=1}^{J}\mathbf{H}_j^T\mathbf{H}_j\right)(\mathbf{s}-\mathbf{s}')} \right) \tag{4.30}$$

which equals the PEP obtained by the centralized demodulator in (4.3); see e.g., [17, Ch. 2]. In other words, PEP performance of distributed demodulation can be rendered asymptotically equivalent to the one corresponding to a multiple-input multiple-output (MIMO) system with M transmit antennas and J receive antennas [17, Ch. 2].

A relevant performance measure is the *diversity* order achieved by each sensor j at any given iteration k, defined as [17, Ch. 2.6]

$$G_d^j(k) = \lim_{\sigma_h^2 \to \infty} \frac{\log E_h\{P_{e|h}^j(k)\}}{\log(\sigma_s^2 \sigma_h^2)} \tag{4.31}$$

where σ_h^2 is the variance of the channel coefficients. Interestingly, it is not necessary to let $k \to \infty$ to achieve the diversity of the centralized system distributedly, as established in the following proposition (see Appendix 4.5.3 for the proof).

Proposition 4.3 *(Diversity order after a finite number of iterations) The distributed ML demodulator in (4.26) achieves the same diversity order as the centralized ML of (4.3) in a finite number of iterations.*

Intuitively, after a number of iterations, $\bar{\mathbf{\Gamma}}_j(k)$ achieves the same rank as $\bar{\mathbf{\Gamma}}$. Because the diversity order is characterized by the rank of $\bar{\mathbf{\Gamma}}$ [17, Ch. 2], both systems achieve the same diversity order.

We will wrap up the performance analysis of our distributed ML demodulator by considering noisy inter-sensor links. In this case, $\hat{\boldsymbol{\varphi}}_j(k)$ and $\hat{\mathbf{\Gamma}}_j(k)$ can be written as $\hat{\boldsymbol{\varphi}}_j(k) = \bar{\boldsymbol{\varphi}}_j(k) + \boldsymbol{\eta}_j(k)$ and $\hat{\mathbf{\Gamma}}_j(k) = \bar{\mathbf{\Gamma}}_j(k) + \mathbf{\Xi}_j(k)$, where $\bar{\boldsymbol{\varphi}}_j(k)$ and $\bar{\mathbf{\Gamma}}_j(k)$ are defined in (4.28), $\boldsymbol{\eta}_j(k) \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_N \sigma_j^2(k))$, and $\text{vec}(\mathbf{\Xi}_j(k)) \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{N^2M^2} \sigma_j^2(k))$ with $\sigma_j(k)$ bounded. Notice that consensus is carried on a per-entry basis and the noise terms $\boldsymbol{\eta}_j(k)$ and $\mathbf{\Xi}_j(k)$ are uncorrelated across entries. Therefore, the PEP now becomes [cf. (4.29)]

$$\begin{aligned} P_{\mathbf{s}\to\mathbf{s}'|h}^j(k) &= \Pr\Bigg[(\mathbf{s}-\mathbf{s}')^T \bar{\mathbf{\Gamma}}_j(k)(\mathbf{s}-\mathbf{s}') + \mathbf{s}^T \mathbf{\Xi}_j(k)\mathbf{s} - (\mathbf{s}')^T \mathbf{\Xi}_j(k)\mathbf{s}' \\ &\qquad -2\boldsymbol{\eta}_j^T(k)(\mathbf{s}-\mathbf{s}') - 2\left(\frac{1}{J}\sum_{i=1}^{J} c_{ij}(k)\mathbf{H}_i^T \boldsymbol{\epsilon}_i\right)^T (\mathbf{s}-\mathbf{s}') < 0\Bigg] \\ &= Q\left(\frac{(\mathbf{s}-\mathbf{s}')^T \bar{\mathbf{\Gamma}}_j(k)(\mathbf{s}-\mathbf{s}')}{\sqrt{2(\mathbf{s}-\mathbf{s}')^T[\tilde{\mathbf{\Gamma}}_j(k) + \mathbf{I}_{NN}\sigma_j^2(k)](\mathbf{s}-\mathbf{s}') + \sigma_{\Xi}^2(k)}}\right) \end{aligned} \tag{4.32}$$

where $\sigma^2_{\Xi}(k) := \text{var}\left\{\mathbf{s}^T\boldsymbol{\Xi}_j(k)\mathbf{s} - (\mathbf{s}')^T\boldsymbol{\Xi}_j(k)\mathbf{s}'\right\}$. Because the denominator in (4.32) is strictly larger than that of (4.29), the overall system performance is worse when compared to the noise-free case. As expected, the presence of noise in inter-sensor communications reduces the equivalent overall SNR. As a sanity check, observe that when $\sigma^2_j(k) \to 0$, equation (4.32) reduces to (4.27). Because diversity is defined as σ^2_h goes to infinity, $\sigma^2_j(k)$ will vanish and both systems with and without inter-sensor noise will achieve the same diversity order. This argument can be made rigorous, albeit after employing tedious manipulations. Furthermore, using the same logic as in Appendix 4.5.3, it is not difficult to deduce that full diversity is likewise achievable after a finite number of iterations even for non-ideal inter-sensor links.

2) Linear demodulation. Consider the distributed ZF demodulator in (4.22). Substituting $\bar{\boldsymbol{\Gamma}}$ and $\bar{\boldsymbol{\varphi}}$ for $\bar{\boldsymbol{\Gamma}}_j(k)$ and $\bar{\boldsymbol{\varphi}}_j(k)$, the estimate $\mathbf{s}_j(k)$ of $\hat{\mathbf{s}}_{ZF}$ at sensor j and iteration k is given by

$$\mathbf{s}_j(k) = \bar{\boldsymbol{\Gamma}}_j(k)^{-1}\bar{\boldsymbol{\varphi}}_j(k) = \frac{1}{J}\sum_{i=1}^{J} c_{ij}(k)\bar{\boldsymbol{\Gamma}}_j^{-1}(k)\boldsymbol{\varphi}_i \tag{4.33}$$

where the second inequality comes from (4.28). Since $\boldsymbol{\varphi}_i = \mathbf{H}_i^T\mathbf{H}_i\mathbf{s} + \mathbf{H}_i^T\boldsymbol{\epsilon}_i$, it follows that

$$\mathbf{s}_j(k) = \tilde{\mathbf{G}}_j(k)\mathbf{s} + \tilde{\mathbf{w}}_j(k) \tag{4.34}$$

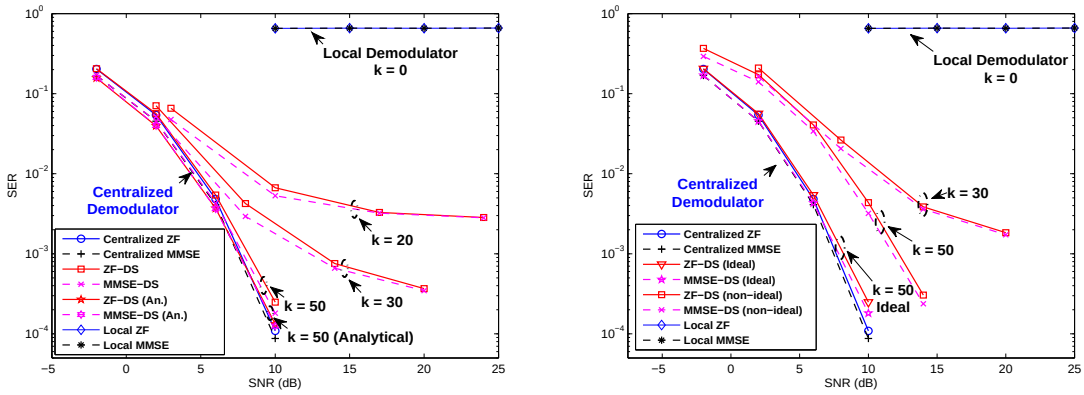
where now $\tilde{\mathbf{G}}_j(k) := (1/J)\sum_{i=1}^{J} c_{ij}(k)\bar{\boldsymbol{\Gamma}}_j^{-1}(k)\mathbf{H}_i^T\mathbf{H}_i$ and $\tilde{\mathbf{w}}_j(k) := (1/J)\sum_{i=1}^{J} c_{ij}(k)\bar{\boldsymbol{\Gamma}}_j^{-1}(k)\mathbf{H}_i^T\boldsymbol{\epsilon}_i$. The rest of the steps are identical to those in Section 4.2.2: (i) write the ℓ-th entry of $\mathbf{s}_j(k)$ as in (4.15); (ii) find the variance of the interference-plus-noise term in (4.16); and (iii) and formulate an equivalent SNR for the ℓ-th entry of $\mathbf{s}_j(k)$ as in (4.17). Recall also that similar performance results can be obtained for the MMSE demodulator by substituting $\bar{\boldsymbol{\Gamma}}_j^{-1}(k)$ for $(\bar{\boldsymbol{\Gamma}}_j(k) + \frac{1}{J\sigma_s^2}\mathbf{I}_{NM})^{-1}$ in (4.33). Performance analysis in the presence of inter-sensor noise is far more challenging in this case, since it involves computing expectations over inverted Gaussian matrices. For this reason, our performance analysis in the presence of noise will rely on simulations.

4.4 Numerical Tests

The distributed demodulators of Sections 4.2 and 4.3 are tested and compared in this section. The WSN has $J = 10$ sensors uniformly placed over the unit square. The communication range of each sensor is $r = 0.5$. Two nodes are connected if their Euclidean distance is less than r. As a result, the number of graph edges here is $|\mathcal{E}| = 18$. Symbols are drawn from a 4-PAM constellation and directly mapped to the entries of matrix $\mathbf{S}$. The AP has $M = 4$ antennas, and the block duration is $N = 2$ time slots. The channels $\mathbf{h}_j \sim \mathcal{N}(\mathbf{0}_M, \sigma_h^2 \mathbf{I}_M)$ are independently and identically distributed. The average AP-sensor SNR in dB is $SNR := 10 \log_{10}(\sigma_s^2 \sigma_h^2)$.

Test Case 4.1 *(Distributed linear demodulators.)* In this case, the average SER of the DC-DS in Section 4.2 is simulated for ideal and non-ideal inter-sensor communication links. Both distributed ZF and MMSE demodulators based on the DC-DS iterations (4.10a)-(4.10b) are considered (tagged as ZF-DS and MMSE-DS, respectively). Each sensor j quantizes the iterate $\mathbf{s}_j(k)$ to obtain the demodulation result at iteration k.

Fig. 4.2(a) plots SER as a function of the AP-to-sensor SNR assuming ideal inter-sensor links for different values of the iteration index k. The *centralized* ZF and MMSE demodulators (using $\mathbf{y}$ and $\mathbf{H}$ in (4.4) and (4.5)) are included as a benchmark. The *local* demodulators using only $\mathbf{y}_j$ and $\mathbf{H}_j$ are provided by the distributed initialization at iteration number $k = 0$. The analytical SER obtained as detailed in (4.18) is also included. Clearly, the average SER improves as the number of iterations increases, approaching that of the centralized benchmark after approximately 50 iterations of the ZF-DS and MMSE-DS demodulators. In the centralized setting, the MMSE is known to outperform the ZF, but their diversity orders are the same. This is also observed for the distributed setting after a finite number of iterations. Notice that local demodulators perform poorly. This is because each sensor locally faces an under-determined scenario (the number of transmit antennas is larger than the number of receive antennas). As the number of consensus iterations increases, the *equivalent* number of receive antennas increases and the system becomes identifiable.

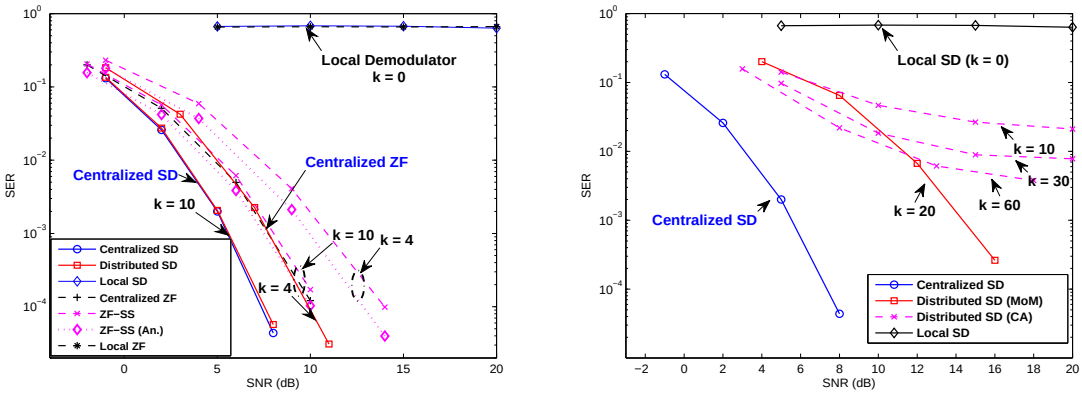


(a) (b)

Figure 4.2: SER vs. SNR (in dB) curves for the DC-DS algorithm with: (a) ideal; (b) non-ideal inter-sensor links.

Fig. 4.2(b) depicts the error performance when considering non-ideal inter-sensor links. Single-hop exchanges are corrupted by AWGN at equivalent $SNR' = SNR + 10$. Also, links are assumed to fail independently with probability (w.p.) 0.1. The noise-free (ideal) case from Fig. 4.2(a) after $k = 50$ iterations is also included here for comparison. Clearly, non-ideal inter-sensor links degrade the error performance. Notwithstanding this performance loss, the slope of the SER curve for both ideal and non-ideal cases is the same; i.e., the same diversity order is obtained in both cases.

Test Case 4.2 *(Distributed demodulation using consensus on sufficient statistics.)* Here, the performance of the algorithm based on the DC-SS approach of Section 4.3 is simulated. As discussed in Section 4.3, this approach allows for a broader class of demodulation methods. Distributed SD and ZF (tagged as ZF-SS) are tested by consenting on sufficient statistics through iterations (4.25a)-(4.25b). After a given number of iterations k, each sensor relies on the averages $\bar{\mathbf{\Gamma}}(k)$ and $\bar{\boldsymbol{\varphi}}(k)$ collected up to that instant to run the demodulators using either (4.22) or (4.20). Fig. 4.3(a) plots the resulting SER vs. SNR curves under ideal inter-sensor links for the SD and the ZF-SS demodulators. Analytical results obtained in Section 4.3.2 for ZF-SS are also included. The average SER improves as the number of

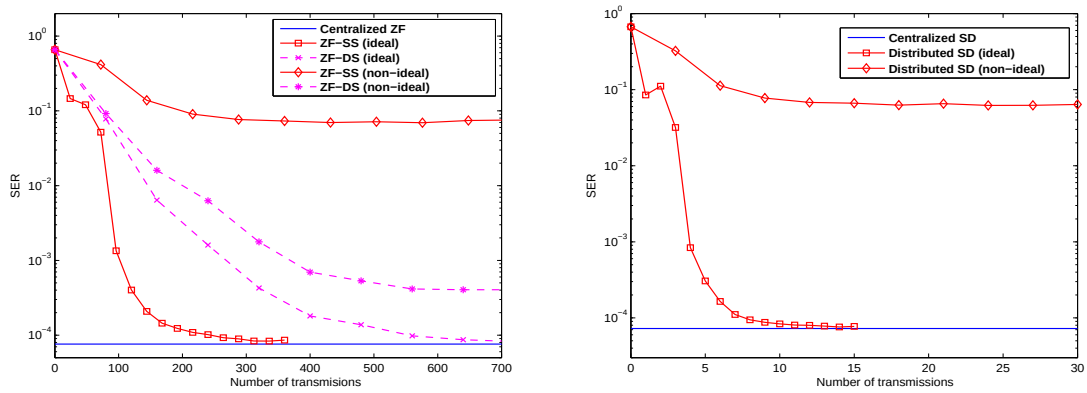


(a) (b)

Figure 4.3: SER vs. SNR (in dB) curves for the DC-SS algorithm with: (a) ideal; (b) non-ideal inter-sensor links.

iterations increases much more rapidly compared to Fig. 4.2(a), approaching that of the centralized benchmark after about $k = 10$ iterations. (See Test Case 4.3 for more detailed comparisons.) This is not surprising since the diversity collected by the demodulator increases as statistical information from neighboring sensors becomes available as iterations proceed. Fig. 4.3(b) shows the performance of distributed SD when imperfect inter-sensor links are present, where now the inter-sensor $SNR' = SNR + 15$ and links fail w.p. 0.1, as before. For comparison, the CA algorithm with the vanishing step size proposed in [20] to cope with noise, is also implemented. The resulting SER of both algorithms degrades w.r.t. the one in Fig. 4.3(a), as expected. As discussed in Section 4.3.2, even under non-ideal links, the CA-MoM algorithm after $k = 20$ iterations exhibits approximately the same diversity as the centralized SD. However, the larger the iteration number k is, the slower the CA iterations progress, especially at high SNR.

Test Case 4.3 *(Performance comparisons.)* Here, we compare the DC-DS and DC-SS demodulators of Sections 4.2 and 4.3. The key to comparing both algorithms is to find the SER for the same number of transmissions (that might be different from the number of iterations of the algorithm). At each iteration of the DC-DS, the number of transmissions



(a) (b)

Figure 4.4: SER vs. number of transmissions curves under both ideal and non-ideal inter-sensor links for (a) distributed ZF demodulators; and (b) distributed SD demodulator using DC-SS.

is proportional to $NM = 8$, which is the length of $\hat{\mathbf{s}}_j(k)$; while the number of transmissions for DC-SS is proportional to $NM + M(M+1)/2 = 18$, as pointed out in Remark 4.4. We set $SNR = 10$ dB, and for the non-ideal inter-sensor links $SNR' = 25$ dB with link failure probability equal to 0.1. Fig. 4.4(a) compares the average SER of both ZF-DS and ZF-SS algorithms as a function of the number of transmissions. For ideal inter-sensor links, the ZF-SS algorithm exhibits fast convergence to the centralized benchmark. However, under non-ideal inter-sensor links, its performance degrades considerably. The ZF-DS algorithm exhibits slower convergence rate compared to the ZF-SS, but it is more robust to non-ideal links. Fig. 4.4(b) plots the average SER of the SD algorithm. Clearly, the distributed SD algorithm converges faster than either ZF-DS or ZF-SS under ideal links at the price of increased demodulation complexity. If links are non-ideal, however, its performance also degrades severely.

4.5 Appendices

4.5.1 Derivation of (4.10a)-(4.10b)

The k-th iteration of the MoM solver of (4.9) is [7, pg. 255]

$$\boldsymbol{s}(k+1) = \arg\min_{\boldsymbol{s}} \ \mathcal{L}_a\left(\boldsymbol{s}, \boldsymbol{z}(k), \boldsymbol{v}(k), \boldsymbol{v}'(k)\right) \tag{4.35a}$$

$$\boldsymbol{z}(k+1) = \arg\min_{\boldsymbol{z}\in\mathcal{C}_z} \mathcal{L}_a\left(\boldsymbol{s}(k+1), \boldsymbol{z}, \boldsymbol{v}(k), \boldsymbol{v}'(k)\right) \tag{4.35b}$$

$$\mathbf{v}_{ji}(k+1) = \mathbf{v}_{ji}(k) + \alpha(\mathbf{s}_j(k+1) - \mathbf{z}_{ji}(k+1)), \quad j \in \mathcal{J},\ i \in \mathcal{N}_j \tag{4.35c}$$

$$\mathbf{v}'_{ji}(k+1) = \mathbf{v}'_{ji}(k) + \alpha(-\mathbf{s}_i(k+1) - \mathbf{z}'_{ji}(k+1)), \quad j \in \mathcal{J},\ i \in \mathcal{N}_j. \tag{4.35d}$$

Because the variables $\mathbf{s}_j$ in (4.9) are not coupled, (4.35a) is equivalent to the following J separable sub-problems, one per sensor j

$$\begin{aligned}\mathbf{s}_j(k+1) \ = \ \arg\min_{\mathbf{s}_j} \Bigg\{ &\frac{1}{2}\|\mathbf{y}_j - \mathbf{H}_j\mathbf{s}_j\|^2 + \sum_{i\in\mathcal{N}_j} (\mathbf{v}_{ji}(k) - \mathbf{v}'_{ij}(k))^T\mathbf{s}_j \\ &+ \sum_{i\in\mathcal{N}_j} \frac{\alpha}{2}\{\|\mathbf{s}_j - \mathbf{z}_{ji}(k)\|^2 + \| - \mathbf{s}_j - \mathbf{z}'_{ij}(k)\|^2\} \Bigg\}. \end{aligned} \tag{4.36}$$

Being linear-quadratic in $\mathbf{s}_j$, each of these sub-problems can be solved in closed form to obtain

$$\mathbf{s}_j(k+1) = \left(\mathbf{H}_j^T\mathbf{H}_j + 2\alpha|\mathcal{N}_j|\mathbf{I}\right)^{-1}\left\{\mathbf{H}_j^T\mathbf{y}_j - \sum_{i\in\mathcal{N}_j}[\mathbf{v}_{ji}(k) - \mathbf{v}'_{ij}(k) - \alpha(\mathbf{z}_{ji}(k) - \mathbf{z}'_{ij}(k))]\right\}. \tag{4.37}$$

Similarly, $\boldsymbol{z}(k+1)$ in (4.35b) is obtained by solving the following $2|\mathcal{E}|$ sub-problems indexed by $(j,i) \in \mathcal{E}$

$$\begin{aligned}\{\mathbf{z}_{ji}(k+1), \mathbf{z}'_{ji}(k+1)\} = \arg\min_{\mathbf{z}_{ji}+\mathbf{z}'_{ji}=\mathbf{0}} \ \Big\{ &-\mathbf{v}_{ji}^T(k)\mathbf{z}_{ji} - [\mathbf{v}'_{ji}(k)]^T\mathbf{z}'_{ji} + \tfrac{\alpha}{2}\|\mathbf{s}_j(k+1) - \mathbf{z}_{ji}\|^2 \\ &+ \tfrac{\alpha}{2}\| - \mathbf{s}_j(k+1) - \mathbf{z}'_{ji}\|^2\Big\}. \end{aligned} \tag{4.38}$$

If $\mathbf{v}_{ji}(k)$ and $\mathbf{v}'_{ji}(k)$ are initialized as $\mathbf{v}_{ji}(0) = \mathbf{v}'_{ji}(0)\ \forall (j,i) \in \mathcal{E}$, the solution to (4.38) for $k = 0$ becomes $\mathbf{z}_{ji}(1) = -\mathbf{z}'_{ji}(1) = \frac{1}{2}(\mathbf{s}_j(1) + \mathbf{s}_i(1))$. Substituting this expression into (4.35c) and (4.35d) yields

$$\mathbf{v}_{ji}(1) = \mathbf{v}'_{ji}(1) = \mathbf{v}_{ji}(0) + \frac{\alpha}{2}(\mathbf{s}_j(1) - \mathbf{s}_i(1)). \tag{4.39}$$

Proceeding by induction, if $\mathbf{v}_{ji}(k) = \mathbf{v}'_{ji}(k)$, the solution to (4.38) is

$$\mathbf{z}_{ji}(k+1) = -\mathbf{z}'_{ji}(k+1) = \frac{1}{2}(\mathbf{s}_j(k+1) + \mathbf{s}_i(k+1)). \tag{4.40}$$

Substituting (4.40) into (4.35c) and (4.35d) proves that

$$\mathbf{v}_{ji}(k+1) = \mathbf{v}'_{ji}(k+1) = \mathbf{v}_{ji}(k) + \frac{\alpha}{2}(\mathbf{s}_j(k+1) - \mathbf{s}_i(k+1)) \tag{4.41}$$

which establishes along with (4.39) that $\mathbf{v}_{ji}(k) = \mathbf{v}'_{ji}(k)\ \forall k > 0$.

Equation (4.41) shows that it is sufficient to update only one set of multipliers $\left\{\{\mathbf{v}_{ji}(k)\}_{i\in\mathcal{N}_j}\right\}_{j\in\mathcal{J}}$ per iteration as in (4.10a). Finally, substituting (4.40) into (4.37) proves the validity of (4.10b).

4.5.2 Proof of Lemma 4.1

Substituting (4.10a) into (4.10b) leads to

$$\mathbf{s}_j(k+1) = (\mathbf{H}_j^T\mathbf{H}_j + 2\alpha|\mathcal{N}_j|\mathbf{I}_{NM})^{-1}\left\{\mathbf{H}_j^T\mathbf{y}_j - \sum_{i\in\mathcal{N}_j}[\mathbf{v}_{ji}(k-1) - \mathbf{v}_{ij}(k-1) - 2\alpha\mathbf{s}_i(k)]\right\}. \tag{4.42}$$

Subtracting (4.42) from its counterpart written for $\mathbf{s}_j(k)$ leads to the multiplier-free recursion

$$\mathbf{s}_j(k+1) = \mathbf{s}_j(k) + 2\alpha(\mathbf{H}_j^T\mathbf{H}_j + 2\alpha|\mathcal{N}_j|\mathbf{I}_{NM})^{-1}\sum_{i\in\mathcal{N}_j}[\mathbf{s}_i(k) - \mathbf{s}_i(k-1)]. \tag{4.43}$$

Hence, to show that $\mathbf{s}_j(k) = \mathbf{C}_j(k)\mathbf{y}$, with $\mathbf{C}_j(k) := [\mathbf{C}_{j1}(k), \ldots, \mathbf{C}_{jJ}(k)]$, it suffices to establish this linear relationship for $k = 1, 2$ and proceed by induction. Wlog initialize (4.10a)-(4.10b) with $\mathbf{v}_{ji}(0) = \mathbf{0}_{NM}$, $\forall(j,i) \in \mathcal{E}$, and let $\mathbf{s}_j(1)$ equal the local ZF demodulator, namely

$$\mathbf{s}_j(1) = \left(\mathbf{H}_j^T\mathbf{H}_j\right)^\dagger \mathbf{H}_j^T\mathbf{y}_j = \mathbf{C}_{jj}(1)\mathbf{y}_j \tag{4.44}$$

where $(\cdot)^\dagger$ denotes matrix pseudo-inverse. Clearly $\mathbf{C}_j(1)$ has all zero entries except for the

j-th block. Furthermore, for $k = 2$ it holds that [cf. (4.42)]

$$\mathbf{s}_j(2) = (\mathbf{H}_j^T\mathbf{H}_j + 2\alpha|\mathcal{N}_j|\mathbf{I}_{NM})^{-1}\left\{\mathbf{H}_j^T\mathbf{y}_j + 2\alpha\sum_{i\in\mathcal{N}_j}\mathbf{s}_i(1)\right\} = \mathbf{C}_{jj}(2)\mathbf{y}_j + \sum_{i\in\mathcal{N}_j}\mathbf{C}_{ji}(2)\mathbf{y}_i \tag{4.45}$$

where the weights are given by

$$\begin{aligned}\mathbf{C}_{jj}(2) &= (\mathbf{H}_j^T\mathbf{H}_j + 2\alpha|\mathcal{N}_j|\mathbf{I}_{NM})^{-1}\mathbf{H}_j^T \\ \mathbf{C}_{ji}(2) &= 2\alpha(\mathbf{H}_j^T\mathbf{H}_j + 2\alpha|\mathcal{N}_j|\mathbf{I}_{NM})^{-1}\mathbf{C}_{ii}(1) \quad \forall i \in \mathcal{N}_j.\end{aligned} \tag{4.46}$$

Using (4.43), it can be readily proved by induction that the linear relationship $\mathbf{s}_j(k) = \mathbf{C}_j(k)\mathbf{y}$ holds for any $k \geq 3$, and the weights can be obtained iteratively as well.

4.5.3 Proof of Proposition 4.3

We first prove that there exists a finite k' for which $c_{ij}(k) > 0$ for all $k \geq k'$. For that matter, observe that $c_{ij}(k)$ converges exponentially to 1; hence, each entry of $c_{ij}(k)$ obeys $|c_{ij}(k) - 1| \leq C\lambda^k$, $\forall k$, for some $0 < C < \infty$ and $0 < \lambda < 1$ [cf. Appendix 3.6.5]. Thus, to guarantee that $c_{ij}(k) > 0$, one can choose any k' such that

$$k' \geq \frac{\log C}{\log(1/\lambda)} < \infty. \tag{4.47}$$

If $c_{ij}(k) > 0\ \forall i$, it is possible to bound the PEP in (4.29) by

$$\begin{aligned}P^j_{\mathbf{s}\to\mathbf{s}'}(k) &\leq Q\left(\frac{(\mathbf{s}-\mathbf{s}')^T\left(\sum_{i=1}^J c_j^{\min}(k)\mathbf{H}_i^T\mathbf{H}_i\right)(\mathbf{s}-\mathbf{s}')}{\sqrt{2(\mathbf{s}-\mathbf{s}')^T\left(\sum_{i=1}^J (c_j^{\max}(k))^2\mathbf{H}_i^T\mathbf{H}_i\right)(\mathbf{s}-\mathbf{s}')}}\right) \\ &\leq Q\left(\kappa_j(k)\sqrt{\frac{1}{2}(\mathbf{s}-\mathbf{s}')^T\left(\sum_{i=1}^J \mathbf{H}_i^T\mathbf{H}_i\right)(\mathbf{s}-\mathbf{s}')}\right)\end{aligned} \tag{4.48}$$

where $c_j^{\min}(k) := \min_i\{c_{ij}(k)\}$, $c_j^{\max}(k) := \max_i\{c_{ij}(k)\}$ and $\kappa_j(k) := c_j^{\min}(k)/c_j^{\max}(k) \leq 1$. The inequality in (4.48) implies that the performance of the system is bounded by an equivalent centralized system with SNR reduced by $\kappa_j^2(k)$. Then, the diversity achieved is *at least* the same as the one achieved by the centralized system. On the other hand, the diversity order per local sensor can only be *at most* that of the centralized demodulator. This implies that, so long as $c_{ij}(k) > 0$, (4.26) achieves the same diversity order as (4.3).

Chapter 5

Conclusions and Future Work

Although the gamut of WSN-driven applications is not yet fully delineated, it is clear that the deployment of WSNs should be task-specific. Moreover, the major challenges in the application design are the constraints of stringent power and bandwidth available at distributed sensors, as well as the robustness to the consequent poor communications. The present thesis addressed those challenging tasks by advancing the development of two fundamental signal processing tools for use by WSNs: *distributed* tracking of nonstationary processes, and *distributed* detection of messages drawn from finite alphabets.

In this final chapter, we provide a summary of the main results in the thesis, and point out possible directions for future research.

5.1 Summarizing Conclusions

Chapter 2 dealt with tracking nonstationary state processes based on reduced-dimensionality data collected from power-limited wireless sensors through (non-)ideal channels. The distributed tracking algorithms were developed in both FC-based WSNs, where sensors are linked with an FC, as well as in ad hoc WSNs, where each sensor is responsible for both data acquisition and processing.

The linear dimensionality-reducing matrices introduced allow the FC to account for both the sensors' limited power and noisy links, as well as to minimize the estimation

MSE. In the single sensor case, MSE-optimal schemes were found in closed form, while an algorithm relying on block coordinate descent iterations was developed for multi-sensor tracking. Power savings result when allowing the FC to feed back state predictions to the sensors. The resultant KF tracker is channel-aware, and optimally allocates the available transmission power among the reduced-dimensionality components in a water-filling like manner. Further, distributed KF schemes using ad hoc WSNs were developed to gain robustness to FC failures, and save transmit-power. Both ideal links (feasible when powerful error control codes are employed), and non-ideal inter-sensor links were considered. When sensor links can be assumed ideal, each broadcasting sensor compresses the innovation sequence of its observations, thus transmitting the minimal required information to perform tracking across sensors. This was not the case, however, for the non-ideal links where sensors must reduce the dimensionality of their raw observations.

Corroborated by numerical examples, the tracking schemes for FC-based and ad hoc WSN topologies offer complementary strengths. The FC-based ones yield higher estimation accuracy, while their ad hoc counterparts gain in robustness and power savings.

As the FC-based WSN applications are limited to the worse scalability of network coverage and lack robustness in case of FC failures, we were encouraged to consider only the ad hoc WSNs for distributed cooperative communication tasks. The ad hoc WSN topology allows each sensor to communicate only with nearby neighbors, where the limited power envisioned at each sensor leads to possible non-ideal inter-sensor links. The problem becomes for sensors to consent on the message broadcasted from the AP through in-network processing. Two types of message constraint sets, popular in the communications context, were considered for this detection application. Specifically, Chapter 3 considered the detection of a coded symbol sequence, which boils down to distributed block- and bitwise-decoding; while in Chapter 4 the message is instead modulated using signal constellations, where the problem amounts to performing distributed demodulation subject to constellation constraints. Under both scenarios, sensors rely on in-network (possibly imperfect) communications with neighboring sensors to achieve the centralized performance.

More specifically, for the distributed decoding task in Chapter 3, sufficient statistics for

decoding were expressed in terms of the average of local LLRs. Based on the latter, two algorithms were presented to obtain such sufficient statistics in a distributed fashion: (i) the CA-SI scheme, which relies on weighted iterations of local averages; and (ii) the CA-MoM scheme, which solves a distributed optimization problem with the alternating direction MoM. Unlike distributed detection alternatives that incur exponential complexity, both have complexity that scales linearly with the problem dimension. With suitable modifications, both algorithms were made robust to random link failures, and additive noise present in the inter-sensor links. Error performance bounds were also derived to determine the number of iterations needed to guarantee a given level of average decoding errors at any given sensor. Numerical tests confirmed that after a few iterations, the error rate of both CA-SI and CA-MoM algorithms comes very close to that of their centralized benchmark.

In addition to decoding, two iterative algorithms were developed to obtain demodulated symbols distributedly in Chapter 4. The DC-DS one resulted after viewing the linear demodulation task as an unconstrained optimization problem solved with the MoM to obtain the optimal solution in a distributed fashion. For ML optimal demodulation and various sub-optimal alternatives, the DC-SS scheme aimed at consensus on the average local sample (cross-) covariance terms, which are sufficient statistics for the general ML, SD and ZF/MMSE demodulators. Both DC-DS and DC-SS algorithms entail affordable communication complexity, irrespective of the constellation size. Similarly, both algorithms can be modified to ensure robustness to random link failures and additive noise present in the inter-sensor links. The per-iteration error performance was analyzed for both algorithms, and the number of iterations needed to attain a prescribed error rate was quantified. Numerical examples demonstrated the effectiveness of the novel algorithms along with the associated complexity versus error performance trade-offs.

5.2 Future research

The results in this thesis open up interesting directions for a number of future research topics. Next, we outline two of them that we are currently pursuing. The first one is a direct extension of the link robustness analysis carried out for the CA-MoM algorithm of

Chapter 3. The second one pertains to distributed tracking of *sparse* signals using reduced dimensionality data.

5.2.1 Link-Robustness Analysis for CA-MoM

In Chapter 3, the imperfection of inter-sensor links arises from the power and bandwidth constraints envisioned at distributed sensors. Under these constraints, convergence analysis was carried for the CA-MoM algorithm in face of either the random link failures, or, additive noise present in the inter-sensor links. The results are quite interesting and different approaches are used to establish robustness. For noisy links, convergence to the true average was established in the mean sense, while the estimator's variance was ensured to remain bounded. This was derived by utilizing the eigen-decomposition of the linear state transition system. In contrast, the link failures are related to the random state transition matrices across time, where the eigen-decomposition concerning the product of those matrices becomes difficult to analyze. Instead, the non-increasing Lyapunov function associated with this random linear system was pursued, and allowed to establish convergence in the mean-square sense (m.s.s.).

Now the problem arises once both cases are present in the inter-sensor links. Basically, the aforementioned analysis demonstrates that the random link failures influence the convergence rate while the additive noise present degrades the steady-state performance. It is reasonable to conjecture that the presence of both will lead to both degradations. This means that the local estimate should be able to have the mean converge to the true average in the m.s.s., with bounded variance dependent on the power of the noise added at the sensors. To verify this, the CA-MoM algorithm was tested numerically with both random link failures and additive noise present in the inter-sensor links in Section 3.5, and the results fit very well with our conjecture.

Nevertheless, to prove this claim will require a combination of both derivation approaches, which is not easy to deal with. There are two sources of difficulty: (i) the product of random matrices is generally not diminishing even though each factor has all eigenvalues less than 1; and (ii) each of the random augmented state transition matrices in

our setup has only some of its eigenvalues equal 1; hence, convergence only holds for a part of the augmented state, namely the local estimators but not the multipliers. Following the Lyapunov arguments of Section 3.6.4, the interesting part of the random product matrix is guaranteed to diminish. However, this neither leads to a diminishing rate for the converging part, nor it describes the behavior of the other part. Actually, these two properties are essential to bound the variance of local estimators, as shown in Section 3.6.3. Therefore, a closer look is needed in order to discover some special features of the matrix, and/or some properties established in random matrix theory. Although we have solid corroboration from the numerical tests, it is still not clear at this stage how to prove the convergence when both link imperfections are present.

5.2.2 Reduced-Dimensionality Estimation of *Sparse* Signals

In Chapter 2 we assumed that the signals of interest are nonstationary but not necessarily sparse. Recently, with the popularity of compressive sampling, there is an increasing interest to account for the sparsity of signals in a broad range of applications. For example, sparsity has been exploited to solve under-determined linear systems of equations with applications to sub-Nyquist sampling [10]; and also to select variables in linear regression applications using e.g., the least-absolute shrinkage and selection operator (Lasso) [50]. At the same time, sparsity is currently employed for data compression, e.g., through principal component analysis (PCA) [22], where sparse variable selection is utilized to determine the "instrumental variables" describing the data parsimoniously.

More recently, efficient dimensionality reduction methods have been developed by exploiting sparsity in the covariance domain [47]. Specifically, the data vector $\mathbf{s}$, whose dimensionality needs to be reduced, has an unknown covariance matrix $\mathbf{\Sigma}_{ss}$. Motivated by the practical scenarios such as field measurement models described in Chapter 2, $\mathbf{\Sigma}_{ss}$ is assumed to have a sparse eigen-space. Since $\mathbf{\Sigma}_{ss}$ is not available, using a set of training data $\{\mathbf{s}_t\}_{t=1}^n$, the MSE cost is replaced by its 'sample-average' version as

$$(\hat{\mathbf{B}}, \hat{\mathbf{C}}) = \arg\min_{\mathbf{B},\mathbf{C}} \frac{1}{n}\|\mathbf{S} - \mathbf{BCS}\|_F^2 \tag{5.1}$$

where $\mathbf{S} := [\mathbf{s}_1 \dots \mathbf{s}_n] \in \mathbb{R}^{p\times n}$. The *sparse* compression and reconstruction matrices $\mathbf{B}$ and

$\mathbf{C}$ become available with an extra ℓ_1 regularization term, which enforces sparsity on these two matrices in (5.1).

Clearly, the same problem may arise when seeking to estimate a *sparse* signal $\mathbf{s}$ from a correlated observation $\mathbf{x}$. This problem is further extended once we allow the signal to be nonstationary. For the *sparse* estimation using the reduced-dimensionality data, it is important to establish the assumption regarding the space under which $\mathbf{s}$ and $\mathbf{x}$ are sparse. The same question should also be asked once the signal of interest is assumed to be time-varying, since it is possible that sparsity may also evolve with time, and Kalman-filtering like updates may no longer be optimal. In a nutshell, sparsity-related distributed estimation and tracking opens a fresh perspective for WSNs to meet the stringent power constraints.

Bibliography

[1] B. D. O. Anderson and J. B. Moore, *Optimal Filtering.* Englewood Cliffs, NJ: Prentice-Hall, 1979.

[2] P. J. Antsaklis and A. N. Michel, *Linear Systems.* New York: McGraw-Hill, 1997.

[3] T. C. Aysal and K. E. Barner, "On the Convergence of Perturbed and Non-Stationary Consensus Algorithms," in *Proc. of IEEE INFOCOM*, Rio de Janeiro, Brazil, April 19–25, 2009.

[4] W. U. Bajwa, J. D. Haupt, A. M. Sayeed, and R. D. Nowak, "Joint Source-Channel Communication for Distributed Estimation in Sensor Networks," *IEEE Trans. on Info. Theory,* vol. 53, no. 10, pp. 3629–3653, Oct. 2007.

[5] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation.* New York: John Wiley and Sons, 2001.

[6] S. Benedetto and E. Biglieri, *Principles of Digital Transmission with Wireless Application.* New Yorks: Springer-Verlag, 1999.

[7] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods.* Belmont, MA: Athena Scientific, 1997.

[8] D. P. Bertsekas, *Nonlinear Programming,* 2nd edition. Belmont, MA: Athena Scientific, 1999.

[9] D. R. Brillinger, *Time Series: Data Analysis and Theory,* 2nd edition. San Francisco, CA: Holden-Day, 1981.

[10] E. Candès, J. Romberg, and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information," *IEEE Trans. on Info. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[11] H. Chen, K. Zhang, and X. R. Li, "Optimal Data Compression for Multisensor Target Tracking with Communication Constraints," in *Proc. of 43rd Conf. on Dec. and Control*, pp. 2650–2655, Atlantis, Bahamas, Dec. 2004.

[12] T. M. Cover and J. A. Thomas, *Elements of Information Theory,* 2nd edition. Hoboken, NJ: Wiley-Interscience, 2006.

[13] R. Dabora and S. D. Servetto, "Broadcast Channels with Cooperating Decoders," *IEEE Trans. on Information Theory*, vol. 52, pp. 5438–5454, Dec. 2006.

[14] P. M. Djuric, M. Vemula, and M. F. Bugallo, "Tracking with Particle Filtering in Tertiary Wireless Sensor Networks," in *Proc. of IEEE Intl. Conf. Acoustics, Speech, and Signal Processing,* vol. 4, pp. 757–760, Philadelphia, PA, March, 2005.

[15] S. C. Draper, B. J. Frey, and F. R. Kschischang, "Interactive Decoding of a Broadcast Message," in *Proc. of 41st Annual Allterton Conf. Commun., Contr. and Computing*, Monticello, IL, Oct. 2003.

[16] J. Feldman, M. J. Wainwright, and D. R. Karger, "Using Linear Programming to Decode Binary Linear Codes," *IEEE Trans. on Information Theory*, vol. 51, pp. 954–972, Mar. 2005.

[17] G. B. Giannakis, Z. Liu, X. Ma, and S. Zhou, *Space-Time Coding for Broadband Wireless Communications.* Hoboken, NJ: Wiley-Interscience, 2007.

[18] G. H. Golub and C. F. Van Loan, *Matrix Computations,* 3rd edition. Baltimore, MD: The John Hopkins University Press, 1996.

[19] V. Gupta, T. Chung, B. Hassibi, and R. M. Murray, "On a Stochastic Sensor Selection Algorithm with Applications in Sensor Scheduling and Sensor Coverage," *Automatica*, vol. 42, pp. 251–260, Feb. 2006.

[20] Y. Hatano, A. K. Das, and M. Mesbahi, "Agreement in Presence of Noise: Pseudogradients on Random Geometric Networks," in *Proc. of 44th Conf. on Decision and Control and the European Control Conf.*, Seville, Spain, pp. 6382–6387, Dec. 2005.

[21] A. Jadbabaie, J. Lin, and S. Morse, "Coordination of Groups of Mobile Autonomous Agents using Nearest Neighbor Rules," *IEEE Trans. on Auto. Control,* vol. 48, pp. 988-1001, June 2003.

[22] I. Jolliffe, N. Tendafilov, and M. Uddin, "A Modified Principal Component Technique Based on the Lasso," *J. of Comp. and Graph. Statistics*, vol. 12, no. 3, pp. 531–547, 2003.

[23] S. Kar and J. M. F. Moura, "Consensus Based Detection in Sensor Networks: Topology Optimization Under Practical Constraints," in *Proc. of First Intl. Wrkshp. on Info. Theory in Sensor Networks*, Santa Fe, NM, June 2007.

[24] S. Kar and J. M. F. Moura, "Distributed Consensus Algorithms in Sensor Networks with Communication Channel Noise and Random Link Failures," in *Proc. of 41st Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, CA, Oct. 2007.

[25] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory.* Englewood Cliffs, NJ: Prentice Hall, 1993.

[26] M. Kisialiou and Z. -Q. Luo, "Reducing Power Consumption in a Sensor Network by Information Feedback," in *Proc. of European Signal Processing Conference*, Florence, Italy, Sep. 2006.

[27] Z. -Q. Luo, G. B. Giannakis, and S. Zhang, "Optimal Linear Decentralized Estimation in a Bandwidth Constrained Sensor Network," in *Proc. of the Intl. Symp. on Info. Theory*, pp. 1441–1445, Adelaide, Australia, Sep. 2005.

[28] G. Mateos, *Distributed Adaptive Estimation and Tracking using Ad Hoc Wireless Sensor Networks.* Master of Sicence Thesis, University of Minnesota, July 2009.

[29] P. S. Maybeck, *Stochastic Models, Estimation and Control.* New York: Academic Press, 1979.

[30] R. Olfati-Saber, E. Frazzoli, E. Franco, and J. S. Shamma, "Belief Consensus and Distributed Hypothesis Testing in Sensor Networks," in *Network Embedded Sensing and Control*, P. J. Antsaklis and P. Tabuada, Eds. Berlin/Heidelberg: Springer, pp. 169-182, 2006.

[31] R. Olfati-Saber and R. Murray, "Consensus Problems in Networks of Agents with Switching Topology and Time-Delays," *IEEE Trans. on Auto. Control,* vol. 49, pp. 1520-1533, Sept. 2004.

[32] R. Olfati-Saber, "Distributed Kalman Filter with Embedded Consensus Filters," in *Proc. of the 44th Conf. on Dec., and the Eur. Contr. Conf.*, pp. 8179–8184, Seville, Spain, Dec. 2005.

[33] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal, "Locating the Nodes: Cooperative Localization in Wireless Sensor Networks," *IEEE Signal Processing Magazine*, Vol. 22, pp. 54–69, July 2005.

[34] B. T. Polyak and Y. Z. Tsypkin, "Pseudogradient Adaptation and Training Algorithms," *Automation and Remote Control*, no. 3, pp. 45-68, 1973.

[35] J. G. Proakis and M. Salehi, *Digital Communications,* 5th ed. New York, NY: McGraw-Hill, 2008.

[36] M. G. Rabbat, R. D. Nowak, and J. A. Bucklew, "Generalized Consensus Algorithms in Networked Systems with Erasure Links," in *Proc. of Signal Process. Advances in Wireless Commun.*, New York, June 2005.

[37] M. Rabbat, J. Haupt, A. Singh, and R. Nowak, "Decentralized Compression and Predistribution via Randomized Gossiping," in *Proc. of IPSN,* pp. 51–59, Nashville, TN, Apr. 2006.

[38] R. Rajagopal and M. J. Wainwright, "Network-Based Consensus Averaging with General Noisy Channels," in *Proc. of 45th Allerton Conf.*, Univ. of Illinois at U-C, Monticello, IL, Sept. 26-28, 2007.

[39] B. Rao and H. Durrant-Whyte, "Fully Decentralized Algorithm for Multisensor Kalman Filtering," *IEE Proceedings-Control Theory and Applications*, vol. 138, pp. 413–420, Sep. 1991.

[40] A. Ribeiro, G. B. Giannakis, and S. I. Roumeliotis, "SOI-KF: Distributed Kalman Filtering with Low-Cost Communications Using the Sign of Innovations," *IEEE Transactions on Signal Processing*, vol. 54, pp. 4782-4795, December 2006.

[41] A. Ribeiro, I. D. Schizas, J.-J. Xiao, G. B. Giannakis, and Z.-Q. Luo, "Distributed Estimation under Bandwidth and Energy Constraints," Chapter 7 in *Wireless Sensor Networks: Signal Process. and Commun. Perspectives*, A. Swami, Q. Zhao, Y. Hong, and L. Tong, Eds. New York: John Wiley and Sons, pp. 149-184, Oct. 2007.

[42] T. Richardson and R. Urbanke, *Modern Coding Theory.* Cambridge, New York: Cambridge University Press, 2008.

[43] V. Saligrama, M. Alanyali, and O. Savas, "Distributed Detection in Sensor Networks with Packet Losses and Finite Capacity Links," *IEEE Trans. on Signal Processing*, vol. 54, pp. 4118 – 4132, Nov. 2006.

[44] I. D. Schizas, G. B. Giannakis, and Z.Q. Luo, "Distributed Estimation Using Reduced-Dimensionality Sensor Observations," *IEEE Trans. on Signal Processing*, vol. 55, pp. 4284–4299, August 2007.

[45] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc wsns with noisy links - Part I: Distributed estimation of deterministic signals," *IEEE Trans. on Signal Processing*, vol. 56, pp. 342–356, Jan. 2008.

[46] I. D. Schizas, G. B. Giannakis, and S. I. Roumeliotis, and A. Ribeiro, "Consensus in Ad Hoc WSNs with Noisy Links- Part II: Distributed Estimation and Smoothing or Random Signals," *IEEE Trans. on Sig. Proc.*, vol. 56, pp. 1650–1666, April 2008.

[47] I. D. Schizas, G. B. Giannakis, and N. D. Sidiropoulos, "Exploiting Covariance-domain Sparsity for Dimensionality Reduction," to appear in *Proc. of 3rd Intl. Workshop on Comp. Advances in Multi-Sensor Adapt. Proc.*, Aruba Island, Dec. 13–16, 2009.

[48] N. D. Sidiropoulos and Z.-Q. Luo, "A Semidefinite Relaxation Approach to MIMO Detection for High-Order QAM Constellations," *IEEE Signal Processing Letters*, vol. 13, pp. 525-528, Sept. 2006.

[49] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms: Stability and Performance.* Englewood Cliffs, NJ: Prentice Hall, 1995.

[50] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *J. Royal Stat. Soc., Series B,* vol. 58, no. 1, pp. 267–288, 1996.

[51] D. B. West, *Introduction to Graph Theory,* 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 2000.

[52] K. M. Wong, Z.-Q. Luo, Q. Jin, and E. Bosse, "Data Compression, Data Fusion and Kalman Filtering in Wavelet Packet Sub-bands of a Multisensor Tracking System," *IEE Proc. on Radar, Sonar and Navigation,* , vol. 145, pp. 100–108, April 1998.

[53] J.-J. Xiao, A. Ribeiro, Z.-Q. Luo, and G. B. Giannakis, "Distributed Compression-Estimation using Wireless Sensor Networks," *IEEE Signal Processing Magazine*, Vol. 23, pp. 27–41, July 2006.

[54] L. Xiao and S. Boyd, "Fast Linear Iterations for Distributed Averaging," *System and Control Letters,* vol. 53, pp. 65-78, Sept. 2004.

[55] S. Zenios and Y. Censor, *Parallel Optimization: Theory, Algorithms and Applications.* New York: Oxford University Press, 1997.

[56] K. Zhang, X. R. Li, P. Zhang, and H. Li, "Optimal Linear Estimation Fusion–Part VI: Sensor Data Compression," in *Proc. of the Intl. Conf. on Info. Fusion*, Vol. 23, pp. 221–228. Queensland, Australia, 2003.

[57] F. Zhao, J. Shin, and J. Reich, "Information-Driven Dynamic Sensor Collaboration," *IEEE Signal Processing Magazine,* vol. 19, no. 2, pp. 61–72, March 2002.

[58] H. Zhu, I. D. Schizas, and G. B. Giannakis, "Power-Efficient Dimensionality Reduction for Distributed Channel-Aware Kalman Tracking Using Wireless Sensor Networks," in *Proc. of Wrkshp. on Statistical Signal Processing,* Madison, WI, August 26-29, 2007.

[59] H. Zhu, A. Cano, and G. B. Giannakis, "Consensus-Based Distributed MIMO Decoding Using Semi-Definite Relaxation," in *Proc. of 2nd Intl. Workshop on Comp. Advances in Multi-Sensor Adapt. Proc.*, St. Thomas, U.S. Virgin Islands, Dec. 12-14, 2007.

[60] H. Zhu, A. Cano, and G. B. Giannakis, "Distributed In-Network Channel Decoding Using Consensus on Log-Likelihood Ratio Averages," in *Proc. of Conf. on Info. Sciences and Systems,* Princeton Univ., NJ, March 19-21, 2008.

[61] H. Zhu, A. Cano, and G. B. Giannakis, "Distributed Equalization and Decoding using Wireless Sensor Networks," in *Proc. of 42nd Asilomar Conf. on Signals, Systems, and Computers,* Pacific Grove, CA, Oct. 26-29, 2008.

[62] H. Zhu, I. D. Schizas, and G. B. Giannakis, "Power-Efficient Dimensionality Reduction for Distributed Channel-Aware Kalman Tracking Using Wireless Sensor Networks," *IEEE Trans. on Signal Proc.*, vol. 57, no. 8, pp. 3193 - 3207, August 2009.

[63] H. Zhu, G. B. Giannakis, and A. Cano, "Distributed In-Network Channel Decoding," *IEEE Trans. on Signal Proc.*, vol. 57, no. 10, pp. 3970-3983, October 2009.

[64] H. Zhu, A. Cano, and G. B. Giannakis, "Distributed Consensus-Based Demodulation: Algorithms and Error Analysis," *IEEE Transactions on Wireless Communications*, 2009 (accepted).