Distributed Processing Techniques for Parameter Estimation and Efficient Data Gathering in Wireless Communication and Sensor Networks

by

Nikola Bogdanović

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the

Computer Engineering and Informatics Department
Signal processing & Communications Lab

December, 2014

Κατανεμημένες Τεχνικές Επεξεργασίας για Εκτίμηση Παραμέτρων και Αποδοτική Συλλογή Δεδομένων σε Ασύρματα Δίκτυα Επικοινωνιών και Αισθητήρων

Nikola Bogdanović

Επιβλέπων:

Κωνσταντίνος Μπερμπερίδης, Καθηγητής

Dissertation Title:

Distributed Processing Techniques for Parameter Estimation and Efficient Data-gathering in Wireless Communication and Sensor Networks

Supervisor:

Prof. Kostas Berberidis[1]

Thesis Committee:

Prof. Kostas Berberidis[1]

Associate Prof. Sotiris Nikoletseas[1]

Dr. Athanasios Rontogiannis[2]

Thesis Examination Committee:

Prof. Kostas Berberidis[1]

Associate Prof. Sotiris Nikoletseas[1]

Dr. Athanasios Rontogiannis[2]

Prof. Sergios Theodoridis[3]

Prof. Emmanuel Varvarigos[1]

Assistant Prof. Emmanouil Psarakis[1]

Assistant Prof. Eleftherios Kofidis[4]

[1]Department of Computer Engineering and Informatics, University of Patras
[2]National Observatory of Athens
[3]Department of Informatics and Telecommunications, University of Athens
[4]Department of Statistics and Insurance Science, University of Piraeus

Abstract of the dissertation

Distributed Processing Techniques for Parameter Estimation and Efficient Data Gathering in Wireless Communication and Sensor Networks

by Nikola Bogdanović

Doctor of Philosophy

Signal processing & Communications Lab
Computer Engineering and Informatics Department
University of Patras, Greece

This dissertation deals with the distributed processing techniques for parameter estimation and efficient data-gathering in wireless communication and sensor networks.

The estimation problem consists in inferring a set of parameters from temporal and spatial noisy observations collected by different nodes that monitor an area or field. The objective is to derive an estimate that is as accurate as the one that would be obtained if each node had access to the information across the entire network. With the aim of enabling an energy aware and low-complexity distributed implementation of the estimation task, several useful optimization techniques that generally yield linear estimators were derived in the literature. Up to now, most of the works considered that the nodes are interested in estimating the same vector of global parameters. This scenario can be viewed as a special case of a more general problem where the nodes of the network have overlapped but different estimation interests.

Motivated by this fact, this dissertation states a new Node-Specific Parameter Estimation (NSPE) formulation where the nodes are interested in estimating parameters of local, common and/or global interest. We consider a setting where the NSPE interests are partially overlapping, while the non-overlapping parts can be arbitrarily different. This setting can model several applications, e.g., cooperative spectrum sensing in cognitive radio networks, power system state estimation in smart grids etc. Unsurprisingly, the effectiveness of any distributed adaptive implementation is dependent on the ways cooperation is established at the network level, as well as the processing strategies considered at the node level. At the network level, this dissertation is concerned with the incremental and diffusion cooperation schemes in the NSPE settings. Under the incremental mode, each node communicates with only one neighbor, and the data are processed in a cyclic manner throughout the network at each time instant. On the other hand, in the diffusion mode at each time step each node of the network cooperates with a set of neighboring nodes. Based on Least-Mean Squares (LMS) and Recursive Least-Squares (RLS) learning rules employed at the node level, we derive novel distributed estimation algorithms that undertake distinct but coupled optimization processes in order to obtain adaptive solutions of the considered NSPE setting. The detailed analyses of the mean convergence and the steady-state mean-square performance have been provided. Finally, different performance gains have been illustrated in the context of cooperative spectrum sensing in cognitive radio networks.

Another fundamental problem that has been considered in this dissertation is the data-gathering problem, sometimes also named as the sensor reachback, that arises in Wireless Sensor Networks (WSN). In particular, the problem is related to the transmission of the acquired observations to a data-collecting node, often termed to as sink node, which has increased processing capabilities and more available power as compared

to the other nodes. Here, we focus on WSNs deployed for structural health monitoring. In general, there are several difficulties in the sensor reachback problem arising in such a network. Firstly, the amount of data generated by the sensor nodes may be immense, due to the fact that structural monitoring applications need to transfer relatively large amounts of dynamic response measurement data. Furthermore, the assumption that all sensors have direct, line-of-sight link to the sink does not hold in the case of these structures.

To reduce the amount of data required to be transmitted to the sink node, the correlation among measurements of neighboring nodes can be exploited. A possible approach to exploit spatial data correlation is Distributed Source Coding (DSC). A DSC technique may achieve lossless compression of multiple correlated sensor outputs without establishing any communication links between the nodes. Other approaches employ lossy techniques by taking advantage of the temporal correlations in the data and/or suitable stochastic modeling of the underlying processes. In this dissertation, we present a channel-aware lossless extension of sequential decoding based on cooperation between the nodes. Next, we also present a cooperative communication protocol based on adaptive spatio-temporal prediction. As a more practical approach, it allows a lossy reconstruction of transmitted data, while offering considerable energy savings in terms of transmissions toward the sink.

Περίληψη Διδακτορικής Διατριβής με τίτλο

Κατανεμημένες Τεχνικές Επεξεργασίας για Εκτίμηση Παραμέτρων και Αποδοτική Συλλογή Δεδομένων σε Ασύρματα Δίκτυα Επικοινωνιών και Αισθητήρων

Nikola Bogdanović

Εργαστηρίο Επεξεργασίας Σημάτων και Τηλεπικοινωνιών
Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής
Πανεπιστήμιο Πατρών

Η παρούσα διατριβή ασχολείται με τεχνικές κατανεμημένης επεξεργασίας για εκτίμηση παραμέτρων και για την αποδοτική συλλογή δεδομένων σε ασύρματα δίκτυα επικοινωνιών και αισθητήρων.

Το πρόβλημα της εκτίμησης συνίσταται στην εξαγωγή ενός συνόλου παραμέτρων από χρονικές και χωρικές θορυβώδεις μετρήσεις που συλλέγονται από διαφορετικούς κόμβους οι οποίοι παρακολουθούν μια περιοχή ή ένα πεδίο. Ο στόχος είναι να εξαχθεί μια εκτίμηση που θα είναι τόσο ακριβής όσο αυτή που θα πετυχαίναμε εάν κάθε κόμβος είχε πρόσβαση στην πληροφορία που έχει το σύνολο του δικτύου. Στο πρόσφατο σχετικά παρελθόν έγιναν διάφορες προσπάθειες που είχαν ως σκοπό την ανάπτυξη ενεργειακά αποδοτικών και χαμηλής πολυπλοκότητας κατανεμημένων υλοποίησεων του εκτιμητή. Έτσι, υπάρχουν πλέον στη βιβλιογραφία διάφορες ενδιαφέρουσες τεχνικές βελτιστοποίησης που οδηγούν σε γραμμικούς, κυρίως, εκτιμητές. Μέχρι τώρα, οι περισσότερες εργασίες θεωρούσαν ότι οι κόμβοι ενδιαφέρονται για την εκτίμηση ενός κοινού διανύσματος παραμέτρων, το οποίο είναι ίδιο για όλο το δίκτυο. Αυτό το σενάριο μπορεί να θεωρηθεί ως μια ειδική περίπτωση ενός γενικότερου προβλήματος, όπου οι κόμβοι του δικτύου έχουν επικαλυπτόμενα αλλά διαφορετικά ενδιαφέροντα εκτίμησης.

Παρακινημένη από αυτό το γεγονός, αυτή η Διατριβή ορίζει ένα νέο πλαίσιο της Κόμβο-Ειδικής Εκτίμησης Παραμέτρων (ΚΕΕΠ), όπου οι κόμβοι ενδιαφέρονται για την εκτίμηση των παραμέτρων τοπικού ενδιαφέροντος, των παραμέτρων που είναι κοινές σε ένα υποσύνολο των κόμβων ή/και των παραμέτρων που είναι κοινές σε όλο το δίκτυο. Θεωρούμε ένα περιβάλλον όπου η ΚΕΕΠ αναφέρεται σε ενδιαφέροντα που αλληλεπικαλύπτονται εν μέρει, ενώ τα μη επικαλυπτόμενα τμήματα μπορούν να είναι αυθαίρετα διαφορετικά. Αυτό το πλαίσιο μπορεί να μοντελοποιήσει διάφορες εφαρμογές, π.χ., συνεργατική ανίχνευση φάσματος σε γνωστικά δίκτυα ραδιοεπικοινωνιών, εκτίμηση της κατάστασης ενός δικτύου μεταφοράς ενέργειας κλπ. Όπως αναμένεται, η αποτελεσματικότητα της οποιασδήποτε κατανεμημένης προσαρμοστικής τεχνικής εξαρτάται και από τον συγκεκριμένο τρόπο με τον οποίο πραγματοποιείται η συνεργασία σε επίπεδο δικτύου, καθώς και από τις στρατηγικές επεξεργασίας που χρησιμοποιούνται σε επίπεδο κόμβου. Σε επίπεδο δικτύου, αυτή η διατριβή ασχολείται με τον incremental (κυκλικά εξελισσόμενο) και με τον diffusion (διαχεόμενο) τρόπο συνεργασίας στο πλαίσιο της ΚΕΕΠ. Στον ινκρεμενταλ τρόπο, κάθε κόμβος επικοινωνεί μόνο με ένα γείτονα, και τα δεδομένα από το δίκτυο υποβάλλονται σε επεξεργασία με ένα κυκλικό τρόπο σε κάθε χρονική στιγμή. Από την άλλη πλευρά, στον diffusion τρόπο σε κάθε χρονική στιγμή κάθε κόμβος του δικτύου συνεργάζεται με ένα σύνολο γειτονικών κόμβων. Με βάση τους αλγορίθμους Ελαχίστων Μέσων Τετραγώνων (ΕΜΤ) και Αναδρομικών Ελαχίστων Τετραγώνων (ΑΕΤ) οι οποίοι χρησιμοποιούνται ως κανόνες μάθησης σε επίπεδο κόμβου, αναπτύσσουμε νέους κατανεμημένους αλγόριθμους για την εκτίμηση οι οποίοι αναλαμβάνουν ευδιακριτές, αλλά συνδεδεμένες διαδικασίες βελτιστοποίησης, προκειμένου να αποκτηθούν οι προσαρμοστικές λύσεις της εξεταζόμενης ΚΕΕΠ. Οι λεπτομερείς αναλύσεις για τη σύγκλιση

ως προς τη μέση τιμή και για τη μέση τετραγωνική απόδοση σταθερής κατάστασης έχουν επίσης εξαχθεί στο πλαίσιο αυτής της Διατριβής. Τέλος, όπως αποδεικνύεται, η εφαρμογή των προτεινόμενων τεχνικών εκτίμησης στο πλαίσιο της συνεργατικής ανίχνευσης φάσματος σε γνωστικές ραδιοεπικοινωνίες, οδηγεί σε αισθητά κέρδη απόδοσης.

Ένα άλλο βασικό πρόβλημα που έχει μελετηθεί στην παρούσα εργασία είναι το πρόβλημα συλλογής δεδομένων, επίσης γνωστό ως sensor reachback, το οποίο προκύπτει σε ασύρματα δίκτυα αισθητήρων (ΑΔΑ). Πιο συγκεκριμένα, το πρόβλημα σχετίζεται με την μετάδοση των λαμβανόμενων μετρήσεων σε έναν κόμβο συλλογής δεδομένων, που ονομάζεται sink node, ο οποίος έχει αυξημένες δυνατότητες επεξεργασίας και περισσότερη διαθέσιμη ισχύ σε σύγκριση με τους άλλους κόμβους. Εδώ, έχουμε επικεντρωθεί σε ΑΔΑ που έχουν αναπτυχθεί για την παρακολούθηση της υγείας κατασκευών. Σε γενικές γραμμές, σε ένα τέτοιο δίκτυο προκύπτουν πολλές δυσκολίες σε ότι αφορά το sensor reachback πρόβλημα. Πρώτον, η ποσότητα των δεδομένων που παράγονται από τους αισθητήρες μπορεί να είναι τεράστια, γεγονός που οφείλεται στο ότι για την παρακολούθηση της υγείας κατασκευών είναι απαραίτητο να μεταφερθούν σχετικά μεγάλες ποσότητες μετρήσεων δυναμικής από-κρισης. Επιπλέον, η υπόθεση ότι όλοι οι αισθητήρες έχουν απευθείας μονοπάτι μετάδοσης, με άλλα λόγια ότι βρίσκονται σε οπτική επαφή με τον sink node, δεν ισχύει στην περίπτωση των δομών αυτών.

Για να μειωθεί η ποσότητα των δεδομένων που απαιτούνται για να μεταδοθούν στον sink node, αξιοποιείται η συσχέτιση μεταξύ των μετρήσεων των γειτονικών κόμβων. Μία πιθανή προσέγγιση για την αξιοποίηση της χωρικής συσχέτισης μεταξύ δεδομένων σχετίζεται με την Κατανεμημένη Κωδικοποίηση Πηγής (ΚΚΠ). Η τεχνική ΚΚΠ επιτυγχάνει μη απωλεστική συμπίεση των πολλαπλών συσχετιζόμενων μετρήσεων των κόμβων χωρίς να απαιτεί την οποιαδήποτε επικοινωνία μεταξύ των κόμβων. Άλλες προσεγγίσεις χρησιμοποιούν απ-ωλεστικές τεχνικές συμπίεσης εκμεταλλευόμενες τις χρονικές συσχετίσεις στα δεδομένα ή / και κάνοντας μία κατάλληλη στοχαστική μοντελοποίηση των σχετικών διαδικασιών. Σε αυτή τη Διατριβή, παρουσιάζουμε μία επέκταση της διαδοχικής αποκωδικοποίησης χωρίς απώλειες λαμβάνοντας υπόψιν το κανάλι και βασιζόμενοι σε κατάλληλα σχεδιασμένη συνερ-γασία μεταξύ των κόμβων. Επιπρόσθετα, παρουσιάζουμε ενα συνεργατικό πρωτόκολλο επικοινωνίας που στηρίζεται σε προσαρμοστική χωρο-χρονική πρόβλεψη. Ως μια πιο πρακ-τική προσέγγιση, το πρωτόκολλο επιτρέπει απώλειες στην ανακατασκευή των μεταδιδόμενων δεδομένων, ενώ προσφέρει σημαντική εξοικονόμηση ενέργειας μειώνοντας των αριθμό των απαιτούμενων μεταδόσεων προς τον sink node.

Acknowledgements

I would like to outline that I dedicate all my effort and success to my parents, Jovica and Evica, and my sister Nataša, because of their love and support of all kinds during all these years. I cannot help but remember also my sister's late husband Nikola who tragically lost his life just before finishing his PhD and who had been also giving me great encouragement. I am also thankful to my girlfriend Ivana for her unconditional support, since all this would not have a meaning without her. Last, but not least, I thank all my friends in Serbia, Greece and everywhere, for their friendship and the strength that they gave me throughout these years. Hvala svima!

<div style="text-align: right">

Nikola Bogdanović

13/11/2014

Patras, Greece

</div>

Contents

List of Figures

List of Tables

Abbreviations

NSPE	**N**ode-**S**pecific **P**arameter **E**stimation
LMS	**L**east-**M**ean **S**quares
RLS	**R**ecursive **L**east-**S**quares
MSD	**M**ean-**S**quare **D**eviation
EMSE	**E**xcess **M**ean-**S**quare **E**rror
CTA	**C**ombine-**t**hen-**A**dapt
ATC	**A**dapt-**t**hen-**C**ombine
CR	**C**ognitive **R**adio
PU	**P**rimary **U**sers
SU	**S**econdary **U**sers
LI	**L**ocal **I**nterferer
CI	**C**ommon **I**nterferer
PSD	**P**ower **S**pectral **D**ensity
WSN	**W**ireless **S**ensor **N**etworks
SHM	**S**tructural **H**ealth **M**onitoring
DSC	**D**istributed **S**ource **C**oding
AWGN	**A**dditive **W**hite **G**aussian **N**oise

TDMA   **T**ime **D**ivision **M**ultiple **A**ccess

MISO   **M**ultiple **I**nput **S**ingle **O**utput

DF   **D**ecode and **F**orward

List of Symbols

$(\cdot)^T$	matrix transpose
$(\cdot)^H$	matrix conjugate transpose
$(\cdot)^{-1}$	matrix inverse
$A \succ 0$ or $A \succeq 0$	A is a positive (or positive semi)definite matrix
∇	gradient operator
$a \bmod b$	a modulo b, i.e., remainder after dividing a by b
$\rho(\cdot)$	spectral radius of a matrix
$\lambda_{\max}(\cdot)$	largest eigenvalue of a matrix
$E\{\mathbf{A}\}$	expectation operator of a random matrix $\mathbf{A}$
$R_{\mathbf{A}}$	$= E\{\mathbf{A}^H\mathbf{A}\}$, for any random matrix $\mathbf{A}$
$R_{\mathbf{A},\mathbf{B}}$	$= E\{\mathbf{A}^H\mathbf{B}\}$, for any random matrices $\mathbf{A}$ and $\mathbf{B}$
$r_{\mathbf{A},\mathbf{b}}$	$= E\{\mathbf{A}^H\mathbf{b}\}$, for any random matrix $\mathbf{A}$ and random vector $\mathbf{b}$
$\mathrm{blockdiag}\{A, B, C\}$	block-diagonal matrix with arguments on block-diagonal
$\mathrm{col}\{A, B, C\}$	column operator stacking arguments on top of each other
$\mathrm{diag}\{a\}$	vector to diagonal matrix operator
$\mathrm{diag}\{A\}$	diagonal matrix to vector operator
$\mathrm{vec}\{A\}$	vectorization operator stacking columns of A on top of each other
$\mathrm{Tr}\{A\}$	trace of a matrix A, i.e., sum of diagonal constants
$\in$	belongs to
$\cap$	set intersection
$|\mathcal{X}|$	cardinality, if $\mathcal{X}$ is a set
$\mathcal{X}(j)$	j-th element of $\mathcal{X}$, if the set $\mathcal{X}$ is ordered
$\|x\|$ or $\|x\|_2$	Euclidean norm of a vector x
$\|x\|_\Sigma^2$	$\triangleq x^H \Sigma x$, with a vector x and a Hermitian matrix $\Sigma \succeq 0$
$\|x\|_{b,\infty}$	block maximum norm of a block-vector x

$A \otimes B$ Kronecker product of matrices A and B

$A \odot B$ Khatri-Rao product of matrices A and B

I_L $L \times L$ identity matrix

$0_{L \times M}$ $L \times M$ zero matrix

1_L $L \times 1$ vector of ones

Dedicated to Jovica, Evica and Nataša

Chapter 1

Introduction

Distributed processing over networks relies on in-network signal processing and cooperation among neighboring agents. Each agent performs some signal processing task such as estimation, detection, compression, fitting a model, and so forth. Cooperation among the agents can be established in different manners, e.g., by forming a cyclic cooperation path, or broadcasting information to nearby neighbors, and under different distortion criteria and channel modeling assumptions. Distributed processing techniques are suitable for resolving critical issues in various types of networks, such as cognitive radio networks, power grids, wireless sensor networks, next generation mobile networks, to mention a few. For instance, cognitive radio networks aim to alleviate today's radio spectrum scarcity problem. The wireless spectral detection and estimation techniques for sensing of available spectrum play a key role in implementing cognitive radio networks. However, the performance of spectrum sensing at a single agent may be detrimentally affected by the signal path loss or local shadowing. The effect of these phenomena can be facilitated through collaboration among the agents and thus the probability of detecting the channels occupied by the licensed users can be improved. In future power grids, distributed processing is desirable for performing power system state estimation and tracking, monitoring and prediction of energy consumption and production from renewable energy sources, detection of malicious attacks and anomalies etc. Also, wireless sensor networks have received great attention from the scientific community over the past decade or so, because they hold the key to revolutionize many aspects of the industry and our life. They encompass a wide range of applications, some of which are related to health, military, environmental monitoring, or monitoring of civil infrastructures, such as roads, bridges and buildings. In a practical wireless sensor network, there is typically one or a few nodes responsible for the data fusion, usually termed to as sink node(s). The sink nodes gather the observations taken by the sensor nodes and process

them, in a centralized fashion, so as to produce some decision about the observed phenomena. Instead of sending the raw data to the sink nodes, sensor nodes may exploit their processing capabilities to locally perform simple computational tasks and transmit only the required and partially processed data, thereby saving bandwidth and energy, and extending the network lifetime.

This dissertation deals with two basic problems of in-network collaborative signal processing. The main research emphasis is on parameter estimation in case where there is no statistical model for the underlying processes of interest. Then, the focus has been put on some theoretical and practical aspects of data gathering in wireless sensor networks.

1.1 Parameter estimation

In the last decades there have been considerable research efforts on the estimation of a parameter or signal from noisy, temporal and spatial data measured by a set of nodes deployed over a geographical area. Traditionally, the observations from the nodes would be sent to a central base station for processing. Then, the central processor would perform all computations regarding the estimation tasks and transmit the results back to the individual nodes. However, this approach is not scalable with respect to both communication resources and computational power. Therefore, in many cases a distributed approach is desired, where the nodes rely only on their own data and on interactions with their immediate neighbors. As a result, in addition to an increase of the energy efficiency, an improved robustness and scalability can be achieved when performing the estimation task in a distributed fashion [1–6].

With the aim of enabling an energy-aware and low-complexity distributed implementation of the estimation task, several useful optimization techniques that generally yield linear estimators have been derived. Generally, these techniques rely on some separable structure of a cost function, and most of the existing works in literature belong to the following two categories. The first category consists of the techniques following a consensus approach. In some initial works, for instance [7], the implementation of the consensus strategy is done in two stages. Unfortunately, this kind of implementation is not suitable for real time estimation as required in time-varying environments. Afterwards, motivated by the procedure obtained in [8], alternative implementations of the consensus strategy were presented in the literature (e.g., [9]-[10]) which force agreement among the cooperating nodes in a single time-scale. For instance, in [9], the alternating-direction method of multipliers is used to obtain a distributed solution. For a thorough review of the literature on consensus algorithms, see [11] and references therein. The

second category, motivating the approach suggested in this dissertation, comprises single time-scale distributed estimation algorithms that are based on adaptive filtering techniques. In particular, in this kind of techniques a linear estimator is obtained by distributing a specific stochastic gradient method under a diffusion or an incremental mode of cooperation. In the so-called diffusion mode, considered for instance in [12–18], at each time step each node of the network cooperates with a set of neighboring nodes in order to provide a robust and adaptive solution of the network-wide estimation problem. Under the incremental mode (e.g., [19–21]), each node communicates with only one neighbor, and the data are processed in a cyclic manner throughout the network at each time instant. Determining a cyclic path that covers all nodes is an NP-hard problem [22] and, in addition, cyclic trajectories are more sensitive to node failures and to link failures. On the contrary, since the incremental cooperation requires the minimum amount of power, it constitutes a highly-competitive distributed solution for energy-constrained networks whose size allows to find a cyclic trajectory. Moreover, it is well-known that incremental strategies can achieve the performance of the centralized-like solution, which cannot be achieved by diffusion-based algorithms as long as the nodes are not allowed to share measurements or there is no prior knowledge regarding the quality of the data observed at the different nodes. On the other hand, better reliability can be achieved at the expense of increased energy consumption in the diffusion mode, due to the fact that the cooperation is undertaken in a fully ad-hoc manner.

In most of the distributed estimation problems, it is considered that the nodes have the same interests. This scenario can be viewed as a special case of a more general problem where the nodes of the network have overlapping but different estimation interests. Some examples of this kind of networks can be found in the context of power system state estimation in smart grids, speech enhancement and active noise control in wireless acoustic networks and cooperative spectrum sensing in Cognitive Radio (CR) networks. In the signal estimation case, some of the first works explicitly considering the aforementioned setting can be found in [23]-[24]. In these works, for networks with a fully connected and tree topology, Bertrand *et al.* proposed distributed algorithms that allow to estimate node-specific desired signals sharing a common latent signal subspace.

This dissertation is concerned with the estimation scenarios which can be formulated as Node-Specific Parameter Estimation (NSPE) problems. Within this category, most of the existing works are based on consensus implementations. For instance, the consensus approach presented in [25] is based on optimization techniques that force different nodes to reach an agreement when estimating parameters of common interest. At the same time, the consensus-based technique in [25] allows each node to estimate a vector of parameters that is only of its own interest. In the case of schemes based on a distributed implementation of adaptive filtering techniques, NSPE problems are recently

receiving an increasing attention. In [26], a diffusion-based scheme is used to solve an NSPE problem where the node-specific estimation interests are expressed as the product of a node-specific matrix of basis functions and a vector of global parameters. Since the matrix of basis functions is known by each node, the problem finally reduces to the estimation of a vector of global interest. In [27], the authors use diffusion adaptation and scalarization techniques to obtain a Pareto-optimal solution for the multi-objective cost function that appears in a distributed estimation problem where each node has a different interest. For a network formed by non-overlapping clusters of nodes, each with a different estimation interest, a diffusion-based strategy with an adaptive combination rule is proposed in [28]. However, in the proposed strategy the cooperation is finally limited to nodes that have exactly the same objectives. For the same network, a diffusion-based algorithm with spatial regularization that simultaneously provides biased estimates of the multiple vectors of parameters has been derived in [29]. Unlike previous works, the algorithm allows cooperation among neighboring nodes as long as they have numerically similar parameter estimation interests. Additionally, in [30] the authors analyze the performance of the diffusion-based LMS algorithm derived in [13] when it is run in the NSPE setting considered in [29].

In this dissertation, as thoroughly explained in Section 1.3, we state a new NSPE formulation where the nodes are interested in estimating parameters of local, common and/or global interest. In other words, the NSPE interests are partially overlapping, while the non-overlapping parts can be arbitrarily different.

1.2 Data gathering

Wireless Sensor Networks (WSN), as an alternative to the conventional wired systems, provide accurate and continuous monitoring of some phenomena over some specific territory or structure, see [1–6] and references therein. The application of our interest is related to Structural Health Monitoring (SHM). The SHM systems are widely adopted to monitor the behavior of structures during forced vibration testing or natural excitation (e.g. earthquakes, winds, live loading). Structural monitoring systems are applicable to a number of common structures including buildings, bridges, aircrafts and ships. The monitoring system is primarily responsible for collecting the measurement output from sensors installed in the structure and gathering the measurement data at the central sink node [31]. There are already commercially available sensor platforms that can meet the demands of SHM, such as Imote2 [32].
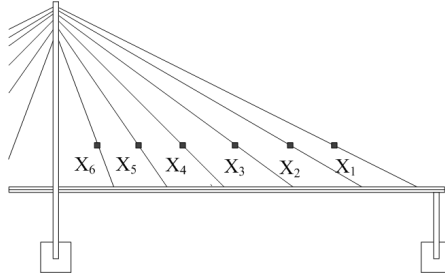
FIGURE 1.1: A span on a bridge.

When accomplishing monitoring tasks such as SHM, one of the most fundamental problems arising in such a network is the data gathering problem [33]. The data gathering problem, or the sensor reachback problem as it is sometimes referred to, is related to the transmission of the acquired observations to a data-collecting node (the sink node) which has increased processing and power consumption capabilities as compared to the sensor nodes. In general, there are several difficulties in the data gathering problem. Firstly, the amount of data generated by the sensor nodes is immense, owing to the fact that structural monitoring applications need to transfer relatively large amounts of dynamic response measurement data with sampling frequencies as high as 1000 Hz [34]. Also, the number of sensor nodes may be very large. Next, the assumption that all sensors have direct, line-of-sight link to the sink does not hold in the case of these structures. Radio communication on and around structures made of concrete or steel components is usually complicated due to radio wave reflection, absorption, and other phenomena that result in poor received signal quality. Moreover, sensor nodes are frequently installed in partially- or completely- obscured areas, such as between girders. As a result, not all sensors may always have a channel to the sink of good enough quality and therefore, direct communication between each sensor node and the sink would consume all the energy stored in the batteries of the sensor nodes very quickly.

The problem of the limited energy that the sensor nodes can afford for data transmission can be alleviated by exploiting recent advances in the field of cooperative communications. To reduce the amount of data required to be transmitted to the sink node, and therefore, to tackle the problem of massive data generated at the sensor nodes, the correlation among the measurements from nearby sensors can be leveraged [35]. For instance, the data collected by the sensors on each span of a bridge are correlated since they are measuring the vibration of the same part of the physical structure (Fig. 1.1). In addition, in some cases of bridge design, two adjacent spans are connected to a common anchorage, resulting in the data across the two spans to be correlated. Similarly, in the case of large buildings, it is natural to group the sensors of the several distinct parts of the building (e.g. floors). In all these cases, data compression approaches exploiting

the correlation of the data offer the potential to greatly reduce the amount of data that needs to be transmitted. An approach which could exploit the spatial data correlation is Distributed Source Coding (DSC). A DSC technique achieves lossless compression of multiple correlated sensor outputs [36] without establishing any communication links between the nodes. A DSC algorithm for the reachback problem, based on pair matching of the nodes, was proposed in [37]. A significantly improved algorithm was proposed in [38], based on application of DSC strategy in a sequential manner. Other approaches employ lossy techniques by taking advantage of the temporal correlation in the data and/or suitable stochastic modeling of the underlying processes. In [39], one may find a comprehensive survey of the research efforts toward the practical data compression techniques for WSNs, e.g., the techniques based on compressed sensing, distributed transform coding, differential pulse code modulation etc.

In this dissertation, first we propose a channel-aware lossless extension of sequential decoding based on cooperation between the nodes. Secondly, we present cooperative communication protocols based on adaptive spatio-temporal prediction. As more practical approaches, they allow a lossy reconstruction of transmitted data, while offering considerable energy savings in terms of transmissions toward the sink.

1.3 Contributions and Outline

In this section, the main contributions of the dissertation are stated. In addition, the organization of the work is provided and the motivation is discussed chapter by chapter.

The contributions of the dissertation are presented in the following:

1) **To formulate a new Node-Specific Parameter Estimation (NSPE) setting,** [40–44]:

   As discussed in Section 1.1, most of the existing incremental/diffusion literature is actually restricted to the cases when all the nodes have the same vectors of parameters to estimate, i.e., $w_k^o = w^o$. Hence, the aim of this dissertation was to remove these restrictions by introducing a new NSPE formulation to this category of solutions where the nodes are interested in estimating parameters of local, common and/or global interest, see Fig. 2.1. We consider a setting where the NSPE interests are partially overlapping, while the non-overlapping parts can be arbitrarily different. One of the facts that motivated us to state this formulation was the applicability of the algorithms that assume the same vector of coefficients w^o modeling all the events in the network. In real world, it may rarely occur that there
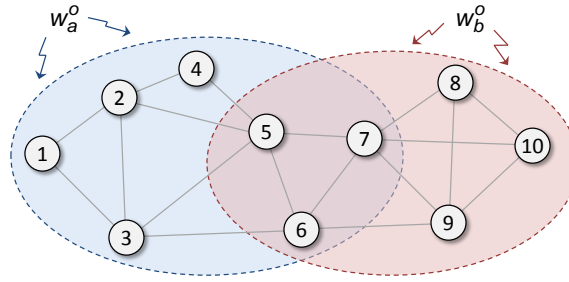
FIGURE 1.2: An example of an NSPE network affected by two overlapping events specified by w_a^o and w_b^o.

is only one phenomenon influencing the whole network or that different events can be modeled only with w^o since they have their influences over exactly the same geographical areas. It is more realistic to assume that the events may affect the nodes in different scales, i.e., global, common and/or local. Note that the algorithm has been designed for the case where parameters of local, common and global interest coexist, however, the derived algorithm can be simplified straightforwardly to scenarios where there might not be all of these parameters. For instance, the proposed NSPE formulation can be used to model a setting where there are two events and where there are some nodes influenced by one of the two events as well other nodes affected by both events, i.e., by their superposition (Fig. 1.2).

2) **To derive distributed adaptive strategies, i.e., Incremental Least Mean Squares, Incremental Recursive Least Squares, Diffusion Least Mean Squares, that solve the formulated NSPE problem, [40–44]:**

The effectiveness of distributed adaptive algorithms that solve the novel NSPE problem depends on the ways cooperation is organized at the network level, as well as the processing strategies adopted at the node level. At the network level, this dissertation derives the incremental and diffusion cooperation schemes in the NSPE settings. Incremental strategies constitute a highly-competitive distributed solution for energy-constrained networks since they can achieve the performance of the centralized-like solution. On the other hand, due to the requirement of finding a Hamiltonian path, they might not scale well with network size. However, the proposed algorithm can be simplified straightforwardly to a scenario where there might not be parameters of local, common or global interest. As a result, the longest Hamiltonian cycle that has to be found for the implementation of I-NSPE LMS depends only on the maximum number of nodes interested in estimating one specific vector of parameters. In the diffusion mode, better reliability can be achieved at the expense of increased energy consumption, due to the fact that the cooperation is carried out in a fully ad-hoc manner where at each time step each

node of the network cooperates with a set of neighboring nodes. Based on Least-Mean Squares (LMS) and Recursive Least-Squares (RLS) learning rules employed at the node level, we derive novel distributed estimation algorithms that undertake distinct but coupled optimization processes so as to obtain adaptive solutions of the considered NSPE setting.

3) ***To undertake the theoretical performance analyses of the Incremental NSPE LMS and Diffusion NSPE LMS and to apply these algorithms to the problem of cooperative spectrum sensing in cognitive radio networks,*** **[43, 44]:**

   The two algorithms mentioned above are analyzed in terms of their mean performance and their mean-square performance in the steady state. Firstly, in Theorem 2.1, we provide conditions for the mean convergence of the Incremental NSPE LMS to the unique centralized batch solution of (2.6). To prove this, it should be emphasized that we do not assume any independence among the regressors corresponding to the global, common and local parameters. Next, for the scalar observation model (2.38) and Gaussian data, the mean-square steady-state performance analysis of the Incremental NSPE LMS is conducted by relying on the weighted energy conservation arguments. For the Diffusion NSPE LMS, Theorem 4.1 establishes conditions for its asymptotic unbiasedness. Then, its mean-square analysis is performed by assuming the more involving vector observation model (4.1) and the regression matrices following a real matrix variate normal distribution where their fourth-order moments are also evaluated. Simulation results confirm the theoretical findings for both algorithms. Furthermore, they are applied to a problem that recently attracted a lot of interest, i.e., cooperative spectrum sensing in cognitive radio networks.

4) ***To analyze some theoretical and practical aspects of data gathering in Wireless Sensor Networks for Structural Health Monitoring and derive suitable data gathering schemes,*** **[45–48]:**

   The final contribution is related to the data gathering task in wireless sensor networks that are applied to monitor some civil structures. To tackle the challenges corresponding to this task, we analyze them from the two essentially different perspectives, i.e., the lossless and the lossy compression perspective. In the former approach, to utilize spatial data correlation, we propose a scheme relied on distributed source coding where we also incorporate channel-awareness. The latter one deals with a more practical approach based on adaptive spatio-temporal

prediction. The proposed protocols mitigate the problem of the energy consumption for sensor nodes, either by reducing the required transmission power or by decreasing the number of transmissions toward the data-gathering node.

The dissertation is organized as follows. Chapters 2-4 are concerned with the parameter estimation problem over adaptive networks while Chapters 5 and 6 are devoted to the problem of data gathering in wireless sensor networks for structural health monitoring.

In Chapters 2 and 3, the novel NSPE formulation is stated and analyzed within the framework of incremental strategies.

Specifically, Chapter 2 proposes incremental LMS algorithm for NSPE problem. At each node, the parameters to be estimated can be of local interest, global interest to the whole network and common interest to a subset of nodes. To estimate each set of local, common and global parameters, a least mean squares algorithm is implemented under an incremental mode of cooperation. Coupled with the estimation of the different sets of parameters, the implementation of each LMS algorithm is only undertaken by the nodes of the network interested in a specific set of local, common or global parameters. After the algorithm derivation, the conditions under which it converges in the mean are provided. Firstly, the theoretical analysis of the mean-square steady-state performance for a scenario where there are only parameters of global and local interests has been undertaken, and then, a more general scenario with parameters of local, common and global interest has been analyzed. The theoretical results related to its mean and mean-square performance are validated through computer simulations, and also simulation results are provided in the context of cooperative spectrum sensing in cognitive radio networks.

Chapter 3 deals with the derivation of two distributed RLS-based schemes, under an incremental mode of cooperation, that solve a NSPE problem where each node is interested in a set of parameters of local interest and a set of global parameters. Initially, we propose a scheme which achieves the exact RLS solution of a central unit processing the data of all the nodes. Then, a scheme with lower transmission complexity for applications where the communications and energy resources are scarce has been derived. Additionally, it was shown that this simplified scheme may converge to the exact RLS solution. Finally, by performing computer simulations we showed the effectiveness of the proposed algorithms.

Chapter 4 motivates the diffusion NSPE LMS algorithm. To address the different node-specific parameter estimation problems, the proposed algorithm relies on a

diffusion-based implementation of different least mean squares algorithms, each associated with the estimation of a specific set of local, common or global parameters. The study of convergence in the mean sense reveals that the proposed algorithm is asymptotically unbiased. The closed-form expressions for the excess mean square error and mean square deviation achieved by each node have been provided. Finally, the theoretical analysis is verified via generic simulations and the algorithm is applied to the problem of cooperative spectrum sensing.

Next, the problem of data gathering in WSN for SHM is discussed in Chapters 5 and 6.

Chapter 5 examines an information-theoretic lossless approach based on Distributed Source Coding (DSC). By building upon the fact that the source coding problem can be optimally separated from the transmission problem in the considered setting, we explain the merit of relays in providing power-efficient and channel-aware protocol based on DSC. Under the assumption that some nodes experience deeply faded channels toward the sink comparing to their neighbors, the proposed strategy achieves both a lower peak power constraint, as well as reduced total power consumption.

Chapter 6 considers a more practical approach where a certain information loss is allowed while satisfying a predefined accuracy constraint. In particular, a cooperative communication protocol based on adaptive spatio-temporal prediction is proposed. The spatio-temporal prediction is realized with adaptive filtering techniques such as least mean squares or recursive least squares. The proposed techniques have been tested extensively via real acceleration measurements from the Canton Tower in China. For both kinds of implementations, it turns out that the proposed strategy may offer considerable savings in transmitted energy.

Finally, Chapter 7 draws a conclusion and describes open research issues to be addressed in future work.

Chapter 2

Incremental NSPE LMS

Incremental strategies process the data in a cyclic manner where each node communicates with only one neighbor [19], [49–51]. In fact, these strategies encompass all the information available in the network at each time instant, and therefore, they can achieve the performance of the centralized-like solution. On the other hand, finding a cyclic path for a given network is known to be an NP-hard problem and they are also prone to node and link failures [20–22]. However, these issues could be mitigated by allowing retransmissions and by letting certain nodes to be visited more than once [20]. Thus, incremental strategies establish a highly-competitive distributed solution for energy-constrained networks.

The discussion in Section 1.1 reveals that most research efforts have been focused on the case where the nodes are interested in estimating exactly the same vector of global parameters in both the incremental [19–21] and the diffusion mode of cooperation [13], [14], [16–18]. To the authors knowledge, there is no existing literature dealing with incremental-type adaptive filtering for node-specific parameter estimation problems. On the other hand, some initial attempts in the literature toward solving NSPE problems based on the diffusion cooperation have been thoroughly surveyed in Section 1.1.

Hence, this chapters states a new NSPE formulation where the nodes are interested in estimating parameters of local, common and/or global interest. In considered setting, the NSPE interests are partially overlapping, while the non-overlapping parts can be arbitrarily different. Based on adaptive filtering techniques, we derive a novel distributed estimation algorithm that undertakes distinct but coupled optimization processes in order to obtain unbiased and adaptive solutions of the considered NSPE problem. Each one of them relies on a least mean squares adaptive algorithm that, under an incremental mode of cooperation, is implemented by a subset of nodes of the network interested in
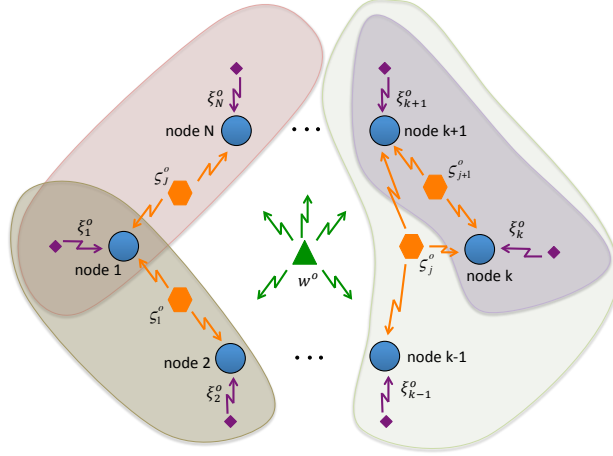
FIGURE 2.1: Network with node-specific parameter estimation interests.

the estimation of a specific set of local, common and/or global parameters. We provide the conditions under which it converges in the mean to the solution of a central unit that processes all the observations of the nodes. The theoretical analysis of the steady-state mean-square performance, for Gaussian data, is provided. The results are verified via generic simulations, and also simulation results are given in the context of cooperative spectrum sensing in cognitive radio networks.

2.1   Problem statement

Let us consider a network consisting of N nodes randomly deployed over some region (see Fig. 2.1). Each node k, at discrete time i, has access to data $\{d_{k,i}, U_{k,i}\}$. These data are related to events that take place in the network area and follow the subsequent model

$$\mathbf{d}_{k,i} = \mathbf{U}_{k,i} w_k^o + \mathbf{v}_{k,i} \tag{2.1}$$

where, for each time instant i,

- w_k^o equals the vector of dimension M_k that gathers all parameters of interest for node k,

- $\mathbf{v}_{k,i}$ denotes measurement and/or model noise with zero mean and covariance matrix $R_{v_k,i}$ of dimensions $L_k \times L_k$,

- $\mathbf{d}_{k,i}$ and $\mathbf{U}_{k,i}$ are zero-mean random variables with dimensions $L_k \times 1$ and $L_k \times M_k$, respectively.

Given the previous observation model, the objective for each node in the network is to use data $\{d_{k,i}, U_{k,i}\}$ in order to estimate its specific unknown vector w_k^o. In particular, we seek the set of linear node-specific estimators $\{w_k\}_{k=1}^N$ that minimize the following global cost function

$$J_{\text{glob}}(\{w_k\}_{k=1}^N) = \sum_{k=1}^N E\left\{\|\mathbf{d}_{k,i} - \mathbf{U}_{k,i}w_k\|^2\right\}. \tag{2.2}$$

In most of the existing papers, e.g., [13, 19, 21], the derived adaptation strategies minimize (2.2) when $w_k^o = w^o$ for all $k \in \{1, 2, \ldots, N\}$. However, note that the formulation considered in this dissertation goes one step further by considering a more general scenario where not all node-specific parameters of interest, i.e. $\{w_k^o\}_{k=1}^N$, are the same. Instead, we allow some of these parameters to differ from one node to another.

As shown in Fig. 2.1, each vector $\{w_k^o\}_{k=1}^N$ might consist of parameters of global interest to the whole network, parameters of common interest to subsets of nodes including node k and parameters of local interest for node k. In particular, each one of the parameters in w_k^o may account for events that affect the nodes in different scales. The global parameters might be related to a phenomenon affecting the whole network. The parameters of common interest might be associated with events measured by a specific subset of nodes. Note that subsets of nodes associated with different events of common interest might be partially or fully overlapped. Finally, the parameters of local interest may reflect an influence of some phenomena that is only present over an area monitored by one node of the network. In this way, considering a scenario where there are J different subsets of common parameters (see Fig. 2.1), the observation model provided in (2.1) can be reformulated as

$$\mathbf{d}_{k,i} = \mathbf{U}_{k_g,i}w^o + \sum_{j \in \mathcal{I}_k} \mathbf{U}_{k_{jc},i}\varsigma_j^o + \mathbf{U}_{k_l,i}\xi_k^o + \mathbf{v}_{k,i} \tag{2.3}$$

where, for $k \in \{1, 2, \ldots, N\}$, $j \in \{1, 2, \ldots, J\}$ and $i \geq 1$,

- w^o is a $M_g \times 1$ sub-vector of w_k^o that consists of all the parameters of global interest,

- ς_j^o is a $M_{j_c} \times 1$ sub-vector of w_k^o formed by the j-th subset of parameters of common interest,

- ξ_k^o is a $M_{k_l} \times 1$ sub-vector of w_k^o that gathers all the parameters of local interest for node k,

- $\mathcal{I}_k$ equals an ordered set of indices j associated with the vectors ς_j^o that are of interest for sensor k,

- $\mathbf{U}_{k_g,i}$, $\mathbf{U}_{k_{jc},i}$ and $\mathbf{U}_{k_l,i}$ are matrices of dimensions $L_k \times M_g$, $L_k \times M_{j_c}$ and $L_k \times M_{k_l}$ that might be correlated, and consist of the columns of $\mathbf{U}_{k,i}$ associated with w^o, ς_j^o and ξ_k^o, respectively.

Thus, according to (2.2) and (2.3), our NSPE problem can be restated as minimizing the following cost

$$\sum_{k=1}^{N} E\left\{\|\mathbf{d}_{k,i} - \mathbf{U}_{k_g,i}w - \sum_{j\in\mathcal{I}_k}\mathbf{U}_{k_{jc},i}\varsigma_j - \mathbf{U}_{k_l,i}\xi_k\|^2\right\} \qquad (2.4)$$

with respect to variables $w, \{\varsigma_j\}_{j=1}^{J}$ and $\{\xi_k\}_{k=1}^{N}$.

2.2 A solution of the new NSPE problem

In this section, first we derive a centralized solution of the optimization problem (2.4), and then we develop a distributed strategy that converges to this centralized solution. For the sake of simplicity and without losing generality, we assume that $M_{k_l} = M_l$, $M_{j_c} = M_c$ and $L_k = L$ for all $k \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, J\}$.

2.2.1 Centralized solution

As it can be seen by inspecting (2.4), in order to solve the considered NSPE problem we have to optimize a scalar real-valued cost function w.r.t. multiple vector variables, i.e., $\{w, \{\varsigma_j\}_{j=1}^{J}, \{\xi_k\}_{k=1}^{N}\}$. After gathering these variables into the following augmented vector

$$\tilde{w} = \begin{bmatrix} w^T & \varsigma_1^T & \varsigma_2^T & \cdots & \varsigma_J^T & \xi_1^T & \xi_2^T & \cdots & \xi_N^T \end{bmatrix}^T \quad (\widetilde{M} \times 1) \qquad (2.5)$$

where $\widetilde{M} = M_g + J \cdot M_c + N \cdot M_l$, we can easily verify that our optimization problem is equivalent to

$$\widehat{\tilde{w}} = \underset{\tilde{w}}{\operatorname{argmin}}\left\{J_{\text{glob}}(\tilde{w})\right\} = \underset{\tilde{w}}{\operatorname{argmin}}\left\{\sum_{k=1}^{N} E\left\{\|\mathbf{d}_{k,i} - \widetilde{\mathbf{U}}_{k,i}\tilde{w}\|^2\right\}\right\} \qquad (2.6)$$

where $\widetilde{\mathbf{U}}_{k,i}$ is defined in (2.7) at the top of the following page with $M_a = (k-1)M_l$ and $M_b = (N-k)M_l$ and

$$\mathbb{1}_{\{\mathcal{X}\in\mathcal{A}\}} = \begin{cases} 1 & \text{if } \mathcal{X} \subseteq \mathcal{A}, \\ 0 & \text{otherwise.} \end{cases} \qquad (2.8)$$

$$\widetilde{\mathbf{U}}_{k,i} = \begin{bmatrix} \mathbf{U}_{k_g,i} & \mathbb{1}_{\{1 \in \mathcal{I}_k\}} \mathbf{U}_{k_{1c},i} & \mathbb{1}_{\{2 \in \mathcal{I}_k\}} \mathbf{U}_{k_{2c},i} & \cdots & \mathbb{1}_{\{J \in \mathcal{I}_k\}} \mathbf{U}_{k_{Jc},i} & 0_{L \times M_a} & \mathbf{U}_{k_l,i} & 0_{L \times M_b} \end{bmatrix} \tag{2.7}$$

$$R_{\widetilde{U}_k} = \begin{bmatrix} R_{U_{k_g}} & \cdots & \mathbb{1}_{\{J \in \mathcal{I}_k\}} R_{U_{k_g}, U_{k_{Jc}}} & 0_{M_g \times M_a} & R_{U_{k_g}, U_{k_l}} & 0_{M_g \times M_b} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \mathbb{1}_{\{J \in \mathcal{I}_k\}} R^H_{U_{k_g}, U_{k_{Jc}}} & \cdots & \mathbb{1}_{\{J \in \mathcal{I}_k\}} R_{U_{k_{1c}}} & 0_{M_c \times M_a} & \mathbb{1}_{\{J \in \mathcal{I}_k\}} R_{U_{k_{Jc}}, U_{k_l}} & 0_{M_c \times M_b} \\ 0_{M_a \times M_g} & \cdots & 0_{M_a \times M_c} & 0_{M_a \times M_a} & 0_{M_a \times M_l} & 0_{M_a \times M_b} \\ R^H_{U_{k_g}, U_{k_l}} & \cdots & \mathbb{1}_{\{J \in \mathcal{I}_k\}} R_{U_{l_g}, U_{k_{Jc}}} & 0_{M_l \times M_a} & R_{U_{k_l}} & 0_{M_l \times M_b} \\ 0_{M_b \times M_g} & \cdots & 0_{M_b \times M_c} & 0_{M_b \times M_a} & 0_{M_b \times M_l} & 0_{M_b \times M_b} \end{bmatrix} \tag{2.10}$$

$$r_{\widetilde{U}_k d_k} = \begin{bmatrix} r^H_{U_{k_g} d_k} & \mathbb{1}_{\{1 \in \mathcal{I}_k\}} r^H_{U_{k_{1c}} d_k} & \cdots & \mathbb{1}_{\{J \in \mathcal{I}_k\}} r^H_{U_{k_{Jc}} d_k} & 0^H_{M_a \times 1} & r^H_{U_{k_l} d_k} & 0^H_{M_b \times 1} \end{bmatrix}^H \tag{2.11}$$

It is well-known that the resulting solutions $\widehat{\widetilde{w}}$ are optimal, if random processes $\{\mathbf{d}_{k,i}, \mathbf{U}_{k,i}\}$ are jointly wide-sense stationary, and are given by the normal equations [52]

$$\left(\sum_{k=1}^{N} R_{\widetilde{U}_k} \right) \cdot \widehat{\widetilde{w}} = \sum_{k=1}^{N} r_{\widetilde{U}_k d_k}, \tag{2.9}$$

where $R_{\widetilde{U}_k}$ and $r_{\widetilde{U}_k d_k}$ are given in (2.10) and (2.11) , respectively.

However, this centralized batch solution requires the inversion of a square matrix whose dimension is actually proportional to the number of nodes N, and hence, a prohibitively high computational cost is needed. To alleviate this problem, different iterative procedures can be followed, e.g., an iterative steepest descent method [52].

2.2.2 Distributed solution

To improve energy efficiency, robustness and scalability of the previously described centralized approach, it is highly desirable to design a distributed and adaptive scheme for the computation of $\widehat{\widetilde{w}}$. To this aim, our starting point is a partly distributed version of the traditional steepest-descent algorithm. Next, we focus on a fully distributed incremental method that generally has better rate of convergence and steady-state performance than its steepest descent counterpart [19].

Taking into account that our global cost function $J_{\text{glob}}(\tilde{w})$ is given as the sum of N local cost functions $\{J_k(\tilde{w})\}_{k=1}^{N}$ with

$$J_k(\tilde{w}) = E\left\{\|\mathbf{d}_{k,i} - \widetilde{\mathbf{U}}_{k,i}\tilde{w}\|^2\right\}, \tag{2.12}$$

a recursion for updating $\tilde{w}^{(i)}$ from $\tilde{w}^{(i-1)}$, where $\tilde{w}^{(i)}$ is an estimate of $\tilde{w}^o$ at iteration i, can be realized in a distributed manner by splitting the update across the network. In particular, by setting $\tilde{w}^{(i)} = \tilde{\psi}_N^{(i)}$, the distributed steepest-descent algorithm can be written as

$$\tilde{\psi}_k^{(i)} = \tilde{\psi}_{(k-1)}^{(i)} - \mu_k \left[\nabla J_k(\tilde{w}^{(i-1)})\right]^H \tag{2.13}$$

with

$$\left[\nabla J_k(\tilde{w}^{(i-1)})\right]^H = R_{\widetilde{U}_k}\tilde{w}^{(i-1)} - r_{\widetilde{U}_k, d_k} \tag{2.14}$$

where $\tilde{\psi}_k^{(i)}$ denotes a local estimate of $\tilde{w}^o$ at node k and iteration i according to (2.6), $\tilde{\psi}_N^{(0)}$ equals an initial guess about $\tilde{w}^o$, and μ_k is a suitably chosen positive step-size. Note that throughout the text, we use the subindex $(k-1)$ according to

$$\tilde{\psi}_{(k-1)}^{(i)} = \begin{cases} \tilde{\psi}_N^{(i-1)} & \text{if } k = 1 \\[2mm] \tilde{\psi}_{k-1}^{(i)} & \text{otherwise.} \end{cases} \tag{2.15}$$

Next, for sufficiently small step-sizes $\{\mu_k\}_{k=1}^{N}$ obeying

$$0 < \mu_k < 2/\lambda_{\max} \tag{2.16}$$

with $\lambda_{\max}$ equal to the largest eigenvalue of the invertible matrix $\sum_{k=1}^{N} R_{\widetilde{U}_k}$ [52], this strategy converges to the centralized solution

$$\lim_{i \to \infty} \tilde{\psi}_k^{(i)} = \widehat{\tilde{w}} \tag{2.17}$$

with $k \in \{1, 2, \ldots, N\}$ and $\widehat{\tilde{w}}$ equal to the solution of (2.9) (see [19] and the references therein).

Note that in (2.13)-(2.14), each node k at time i is required to have access to the global information, i.e., $\tilde{w}^{(i-1)}$. To obtain a fully distributed solution, we apply incremental gradient technique ([19], [49] and [51]), where each node k, at time i, evaluates

the partial gradient $\nabla J_k(\cdot)$ at the local estimate $\tilde{\psi}^{(i)}_{(k-1)}$. The resulting algorithm is

$$\tilde{\psi}^{(i)}_k = \tilde{\psi}^{(i)}_{(k-1)} - \mu_k \left[\nabla J_k(\tilde{\psi}^{(i)}_{(k-1)}) \right]^H \tag{2.18}$$

with

$$\left[\nabla J_k(\tilde{\psi}^{(i)}_{(k-1)}) \right]^H = R_{\widetilde{U}_k} \tilde{\psi}^{(i)}_{(k-1)} - r_{\widetilde{U}_k, d_k}. \tag{2.19}$$

This incremental algorithm establishes a cyclic cooperation where each node of the network transmits only its local estimate $\tilde{\psi}^{(i)}_k$ to an immediate neighbor. Nonetheless, since the dimension of $\tilde{\psi}^{(i)}_k$ depends on the number of nodes, this kind of iterative solutions is still non-scalable with respect to both communication resources and computational power, an issue that will be addressed in the following.

Due to the structure of the augmented regressors $\widetilde{\mathbf{U}}_{k,i}$ defined in (2.7) and the correlation quantities involved in (2.18) and (2.19), we can easily see that, only $2 + |\mathcal{I}_k|$ sub-vectors of $\tilde{\psi}^{(i)}_k$ are updated at each iteration i, when a specific node k performs the update step of (2.18). In particular, according to (4.5) and (2.18), only the sub-vectors associated with the local estimates of w^o, $\{\varsigma^o_{k,j}\}_{j \in \mathcal{I}_k}$ and ξ^o_k at node k and iteration i, denoted as $\psi^{(i)}_k$, $\{\varsigma^{(i)}_{k,j}\}_{j \in \mathcal{I}_k}$ and $\xi^{(i)}_k$, respectively, are updated based on a linear combination of $\psi^{(i)}_{(k-1)}$, $\xi^{(i-1)}_k$ and $\{\varsigma^{(i)}_{f_j(k),j}\}_{j \in \mathcal{I}_k}$ where

$$\varsigma^{(i)}_{f_j(k),j} = \begin{cases} \varsigma^{(i-1)}_{\max\{\mathcal{C}_j\},j} & \text{if } \mathcal{C}_{j,k} = \emptyset, \\[2mm] \varsigma^{(i)}_{\max\{\mathcal{C}_{j,k}\},j} & \text{otherwise,} \end{cases} \tag{2.20}$$

with

$$\mathcal{C}_{j,k} = \{k' \in \mathcal{C}_j : k' < k\}, \tag{2.21}$$

and $\mathcal{C}_j$ denoting the ordered set of indexes associated with those nodes interested in estimating ς^o_j, i.e.,

$$\mathcal{C}_j = \{k : j \in \mathcal{I}_k\}. \tag{2.22}$$

Therefore, without any loss of optimality the previous fact allows for properly modifying (2.18) and (2.19) to obtain an incremental-based NSPE algorithm where, for some

initializations $\psi_N^{(0)}$, $\{\xi_k^{(0)}\}_{k=1}^N$ and $\{\varsigma_{\max\{\mathcal{C}_j\},j}^{(0)}\}_{j=1}^J$, at iteration $i > 0$, each node k executes

$$\begin{bmatrix} \psi_k^{(i)} \\ \tilde{\varsigma}_k^{(i)} \\ \xi_k^{(i)} \end{bmatrix} = \begin{bmatrix} \psi_{(k-1)}^{(i)} \\ \tilde{\varsigma}_{(k-1)}^{(i)} \\ \xi_k^{(i-1)} \end{bmatrix} + \mu_k \left[r_{U_{k,i}d_{k,i}} - R_{U_{k,i}} \begin{bmatrix} \psi_{(k-1)}^{(i)} \\ \tilde{\varsigma}_{(k-1)}^{(i)} \\ \xi_k^{(i-1)} \end{bmatrix} \right] \tag{2.23}$$

where

$$\tilde{\varsigma}_k^{(i)} = \mathrm{col}\left\{ \varsigma_{k,\mathcal{I}_k(1)}^{(i)}, \varsigma_{k,\mathcal{I}_k(2)}^{(i)}, \ldots, \varsigma_{k,\mathcal{I}_k(|\mathcal{I}_k|)}^{(i)} \right\} \tag{2.24}$$

and

$$\tilde{\varsigma}_{(k-1)}^{(i)} = \mathrm{col}\left\{ \varsigma_{f_{\mathcal{I}_k(1)}(k),\mathcal{I}_k(1)}^{(i)}, \ldots, \varsigma_{f_{\mathcal{I}_k(|\mathcal{I}_k|)}(k),\mathcal{I}_k(|\mathcal{I}_k|)}^{(i)} \right\}. \tag{2.25}$$

with $f_j(k)$ defined in (2.20)-(2.22) for $j \in \mathcal{I}_k$.

Unlike the incremental algorithm of (2.18) and (2.19), the above NSPE algorithm is scalable in terms of computational burden and energy resources. On the one hand, regarding the computational complexity, at each iteration, each node k only needs to update $2 + |\mathcal{I}_k|$ vectors whose dimensions are independent of the number of nodes. According to (2.23), these $2 + |\mathcal{I}_k|$ vectors are $\psi_k^{(i)}$, $\{\varsigma_{k,j}^{(i)}\}_{j \in \mathcal{I}_k}$ and $\xi_k^{(i)}$, consequently a total of $M_g + |\mathcal{I}_k| \cdot M_c + M_l$ parameters are updated at node k and any iteration i. On the other hand, decreasing the power consumption, at each iteration i, the NSPE strategy requires $J+1$ cyclic modes of cooperation (see Fig. 2.2). In one of them, all nodes of the network are involved in order to estimate w^o in an incremental fashion. To do so, at each iteration i, each node transmits its local estimate of the vector of global parameters, $\psi_k^{(i)}$, to its immediate neighbor (see Fig. 2.2). It should be realized that the index k assigned to each node sets the order at which nodes cooperate when estimating w^o. Similarly, an incremental mode of cooperation is established to estimate each one of the vectors of common parameters ς_j^o with $j \in \{1, 2, \ldots, J\}$. However, now the cooperation is only established by the nodes interested in estimating ς_j^o, which are gathered in the ordered set $\mathcal{C}_j$. Note that $\mathcal{C}_j$ is a subset of the ordered set of node indexes defining the order of cooperation associated with the estimation of global parameters w^o. In particular, given a specific order of cooperation for the estimation of w^o, (2.20)-(2.22) allow the scheme provided in (2.23) to achieve the same performance as the non-scalable strategy described in (2.18)-(2.19). Nevertheless, other orders of cooperation can be set among the nodes estimating a specific vector ς_j^o.

To proceed further, let us derive a suitable adaptive mechanism that will enable the network to respond to time-variations in the underlying signal statistics. To do so, several approaches may be followed. Among them, in this work we adopt the instantaneous
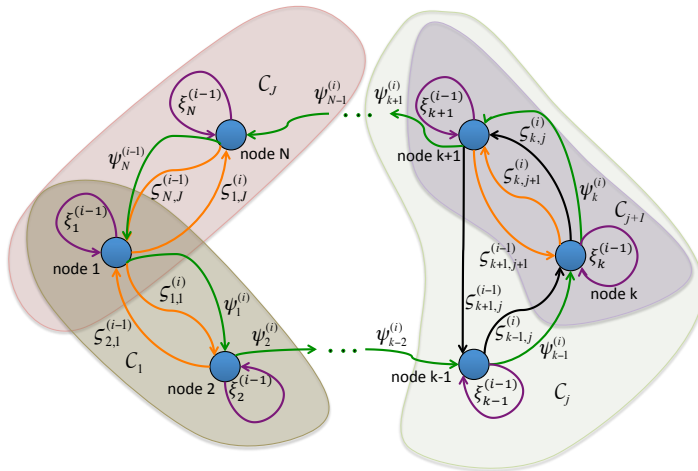
FIGURE 2.2: Order of cooperation among the nodes of the network when implementing the Incremental-based NSPE.

approximations $r_{U_{k,i}d_{k,i}} \approx U_{k,i}^H d_{k,i}$ and $R_{U_{k,i}} \approx U_{k,i}^H U_{k,i}$ in (2.23), where now i denotes time step. This leads to a distributed incremental-based LMS type algorithm depicted in Fig 2.3 and summarized as follows,

---

Incremental-Based NSPE LMS (I-NSPE LMS)

- Start with some initial guess $\psi_N^{(0)}$, $\{\xi_k^{(0)}\}_{k=1}^N$ and $\{\varsigma_{\max\{\mathcal{C}_j\},j}^{(0)}\}_{j=1}^J$ with $\mathcal{C}_j$ defined as in (2.22).

- At each time i, for each node $k \in \{1, 2, \ldots, N\}$, update the vectors $\psi_k^{(i)}$, $\tilde{\varsigma}_k^{(i)}$ and $\xi_k^{(i)}$ by executing

$$
\begin{bmatrix} \psi_k^{(i)} \\ \tilde{\varsigma}_k^{(i)} \\ \xi_k^{(i)} \end{bmatrix} = \begin{bmatrix} \psi_{(k-1)}^{(i)} \\ \tilde{\varsigma}_{(k-1)}^{(i)} \\ \xi_k^{(i-1)} \end{bmatrix} + \mu_k U_{k,i}^H \left[d_{k,i} - U_{k,i} \begin{bmatrix} \psi_{(k-1)}^{(i)} \\ \tilde{\varsigma}_{(k-1)}^{(i)} \\ \xi_k^{(i-1)} \end{bmatrix} \right] \tag{2.26}
$$

with $\tilde{\varsigma}_{(k-1)}^{(i)}$ defined as in (2.25) according to the order of cooperation given in (2.20)-(2.22).

The new algorithm couples, in a fully distributed fashion, a set of $N+J+1$ optimization processes. One of them consists in the estimation of w^o by means of a global LMS algorithm that is implemented at all nodes under an incremental mode of cooperation. Similarly, an incremental-based LMS is implemented to solve each one of the J optimization processes associated with the estimation of the J vectors of common parameters,
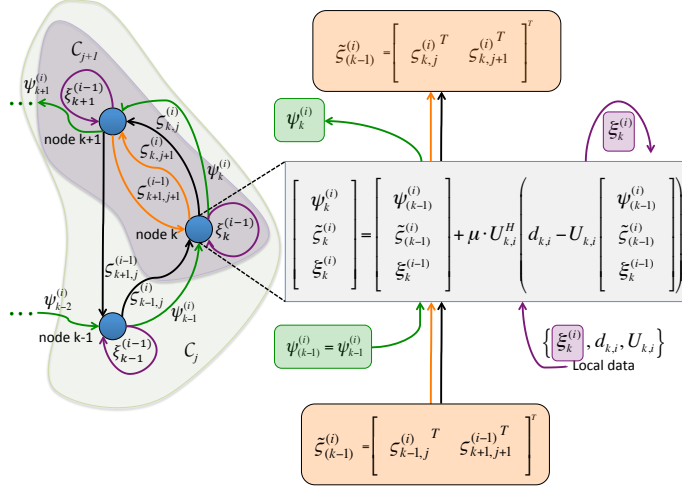
FIGURE 2.3: Structure of the distributed I-NSPE.

$\{\varsigma_j^o\}_{j=1}^J$. In this case, the implementation of the incremental-based LMS estimating ς_j^o is only undertaken by the subset of nodes $\mathcal{C}_j$, which might be overlapped with the subset of nodes $\mathcal{C}_{j'}$ interested in $\varsigma_{j'}^o$ with $j, j' \in \{1, \dots, J\}$ and $j \neq j'$. Unlike the estimation of the global and common vectors of parameters, each one of the N remaining processes is solved by an LMS algorithm that is locally performed at each node in order to estimate its parameters of local interest.

Although the algorithm has been designed for the case where parameters of local, common and global interest coexist, note that the derived algorithm can be simplified straightforwardly to scenarios where there might not be all of these types of parameters. Therefore, according to the order of cooperation established among the nodes to solve the considered NSPE problem (see Fig. 2.2), the longest Hamiltonian cycle that has to be found in the network depends on the maximum number of nodes interested in estimating one specific vector of parameters. As a result, if there are no parameters of global interest, the implementation of the proposed algorithm does not require a Hamiltonian cycle that covers all nodes of the network, and therefore, the proposed incremental algorithm can scale well according the network size. Note that the drawback related to the determination of cyclic paths of cooperation can be significantly alleviated when there are no vectors of global parameters and the sets $\{\mathcal{C}_j\}$ have a cardinality sufficiently smaller than N.

Next, in Theorem 2.1 we provide the conditions under which the I-NSPE algorithm converges to the centralized solution of (2.6). Before stating the theorem formally, we provide a useful property from [53, p.410], that was used in the work of [54] in the context of proving the mean stability of a stand-alone sequential partial update LMS algorithm.

Property 1 - [53, p.410]. Let A be an arbitrary $N \times N$ matrix. Then, $\rho(A) < 1$ if and only if there exists some positive definite $N \times N$ matrix B such that $B - A^H B A$ is positive definite. Here, $\rho(A)$ denotes the spectral radius of A, i.e., $\rho(A) = \max_i |\lambda_i(A)|$.

Theorem 2.1. *Consider any initialization of $\psi_N^{(0)}$, $\{\xi_k^{(0)}\}_{k=1}^N$ and $\{\varsigma_{\max\{C_j\},j}^{(0)}\}_{j=1}^J$ with C_j defined as in* (2.22). *Assume that*

- *noise $\mathbf{v}_{k,i}$ is independent of the regressors $\mathbf{U}_{k',i'}$ for all $k, k' \in \{1, 2, \ldots, N\}$ and $i, i' > 0$,*

- *$\{\mathbf{d}_{k,i}, \mathbf{U}_{k,i}\}$ are jointly wide-sense stationary random processes,*

- *the regressors are spatially and temporally independent.*

Furthermore, assume that

$$\sum_{k=1}^N R_{\widetilde{\mathbf{U}}_k} \succ 0. \tag{2.27}$$

Then, the I-NSPE LMS algorithm asymptotically converges in the mean to the unique centralized batch solution of (2.6), *if every step-size μ_k satisfies*

$$0 < \mu_k < 2/\lambda_{\max}(R_{U_k}). \tag{2.28}$$

Proof. We characterize the mean behavior of I-NSPE LMS over the network by analyzing the mean behavior of the estimates $\tilde{\psi}_k^{(i)}$ of the augmented vector of parameters $\tilde{w}^o$ at node k and time instant i. As previously defined in Section 2.2.2, recall that $\tilde{\psi}_k^{(i)}$ stacks the most recent estimates of each of the sub-vectors of $\tilde{w}^o$ till node k and time i. Firstly, we consider the spatial updates that occur throughout the cycle from time $i - 1$ to i. Throughout this cycle, it can be shown that the expectation of $\tilde{\varphi}_k^{(i)} = \tilde{w}^o - \tilde{\psi}_k^{(i)}$ evolve as

$$E\tilde{\varphi}_k^{(i)} = \prod_{k=1}^N (I_{\widetilde{M}} - \mu_k R_{\widetilde{\mathbf{U}}_k}) E\tilde{\varphi}_k^{(i-1)}, \tag{2.29}$$

From (2.29) note that $\lim_{i \to \infty} \|E\tilde{\varphi}_k^{(i)}\|^2 = 0$ if $\rho\left(\prod_{k=1}^N (I_{\widetilde{M}} - \mu_k R_{\widetilde{\mathbf{U}}_k})\right) < 1$.

Similarly to the argument in [54], we now analyze the adaptation of the squares of the mean errors. For simplicity of presentation, let $\mathbf{x}_{k,i} = E\tilde{\varphi}_k^{(i)}$, where $\mathbf{x}_{N,0}$ is an arbitrary nonzero vector. Initially, we focus on the update at node k from node $k - 1$,

at time i, i.e.,

$$
\begin{aligned}
\mathrm{x}_{k,i}^H \mathrm{x}_{k,i} =& \mathrm{x}_{k-1,i}^H (I_{\widetilde{M}} - \mu_k R_{\widetilde{\mathbf{U}}_k})^H (I_{\widetilde{M}} - \mu_k R_{\widetilde{\mathbf{U}}_k}) \mathrm{x}_{k-1,i} \\
=& \mathrm{x}_{k-1,i}^H \mathrm{x}_{k-1,i} - \alpha_k \mathrm{x}_{k-1,i}^H \mu_k R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{k-1,i} \\
& - \beta_k \mathrm{x}_{k-1,i}^H \mu_k R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{k-1,i} + \mathrm{x}_{k-1,i}^H \mu_k R_{\widetilde{\mathbf{U}}_k} \mu_k R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{k-1,i},
\end{aligned}
\tag{2.30}
$$

where $\beta_k = \lambda_{\max}(\mu_k R_{\widetilde{\mathbf{U}}_k})$ and $\alpha_k = 2 - \beta_k$. Next, in order to understand the influence of the last two terms of the r.h.s in (2.30), it can be verified that the resulting matrix

$$
\beta_k \mu_k R_{\widetilde{\mathbf{U}}_k} - \mu_k R_{\widetilde{\mathbf{U}}_k} \mu_k R_{\widetilde{\mathbf{U}}_k} = \beta_k \mu_k R_{\widetilde{\mathbf{U}}_k} (I - \frac{1}{\beta_k} \mu_k R_{\widetilde{\mathbf{U}}_k})
\tag{2.31}
$$

is a positive semi-definite matrix [55]. This implies that the relation (2.30) can be rewritten as the following inequality

$$
\mathrm{x}_{k,i}^H \mathrm{x}_{k,i} \le \mathrm{x}_{k-1,i}^H \mathrm{x}_{k-1,i} - \alpha_k \mathrm{x}_{k-1,i}^H \mu_k R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{k-1,i}.
\tag{2.32}
$$

Since $\lambda_{\max}(R_{\widetilde{\mathbf{U}}_k}) = \lambda_{\max}(R_{U_k})$, note that $\alpha_k > 0$ if the condition in (2.28) holds.

At this point, let us show by contradiction that the adaption occurs. By iterating (2.32) over the whole network cycle at time $i = 1$, we can see that $\mathrm{x}_{N,1}^H \mathrm{x}_{N,1} < \mathrm{x}_{N,0}^H \mathrm{x}_{N,0}$ if $\mathrm{x}_{k-1,1}^H \mu_k R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{k-1,1} > 0$ for at least one node k. If not, then $\mathrm{x}_{k-1,1}^H \mu_k R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{k-1,1} = 0$, for all k. Note that this last condition implies that $\mu_k R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{k-1,1} = 0$ [56, p.431], and that $\mathrm{x}_{k,1} = \mathrm{x}_{N,0}$ for all k. Hence, we have that $0 = \sum_{k=1}^N \mathrm{x}_{N,0}^H R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{N,0}$ if $\mathrm{x}_{k-1,1}^H \mu_k R_{\widetilde{\mathbf{U}}_k} \mathrm{x}_{k-1,1} > 0$ does not hold for at least one node k. However, this is in contradiction with the assumption of positive definiteness given in (2.27).

Finally, based on the previous argument and relation (2.29), we conclude that

$$
\mathrm{x}_{N,0}^H B \mathrm{x}_{N,0} - \mathrm{x}_{N,0}^H A^H B A \mathrm{x}_{N,0} > 0,
\tag{2.33}
$$

where $B = I_{\widetilde{M}}$ and $A = \prod_{k=1}^N (I_{\widetilde{M}} - \mu_k R_{\widetilde{\mathbf{U}}_k})$. Thus, by applying Property 1, we conclude that

$$
\rho \left(\prod_{k=1}^N (I_{\widetilde{M}} - \mu_k R_{\widetilde{\mathbf{U}}_k}) \right) < 1.
\tag{2.34}
$$

Consequently, under the assumptions in (2.27) and (2.28), the I-NSPE LMS is convergent in the mean and hence, asymptotically unbiased. This concludes the proof. □

Remark 2.1: Note that in Theorem 2.1, we do not assume any independence among $\mathbf{U}_{k_g,i}$, $\mathbf{U}_{k_l,i}$ and $\mathbf{U}_{k_{jc},i}$. Assuming such intra-independence in the regressors at each node

k, the sufficient conditions ensuring strict convexity in (2.6) reduce to that

$$R_{U_{k_l,i}} \succ 0 \tag{2.35}$$

for all $k \in \{1, \ldots, N\}$ and that, for at least one index $k \in \{1, 2, \ldots, N\}$,

$$R_{U_{k_g,i}} \succ 0 \tag{2.36}$$

and

$$R_{U_{k_{jc},i}} \succ 0 \tag{2.37}$$

for all $j \in \{1, 2, \ldots, J\}$. From the previous conditions, we can easily see that the cooperation established by I-NSPE for the estimation of the global and common parameters can allow each node k to estimate w_k^o even when its local cost function is not strictly convex. Consequently, concluding this section we can easily realize that the proposed strategy leverages cooperation to alleviate the problem of local unobservability of the parameters of interest, w_k^o, at a specific set of nodes.

2.3 Performance analysis: A scenario with global and local parameters

In this section, we rely on the weighted energy conservation arguments [19], [52] in order to conduct the mean-square performance analysis of the proposed algorithm. Specifically, we evaluate the steady-state performance at each individual node in terms of Mean-Square Deviation (MSD) and Excess Mean-Square Error (EMSE). For the sake of an easy exposition, we initially consider the performance analysis for the case where a node estimates only the parameters of the global and its local interests. In the following section, we generalize this to the case where there are also different vectors $\{\varsigma_j\}_{j=1}^J$ of common parameters.

2.3.1 Data assumptions

To proceed, we adopt the following independence assumptions on the data, where we analyze, for the sake of clarity, the scalar counterpart ($L_k = 1$) of the observation model defined in (2.1), i.e.,

$$\mathbf{d}_k(i) = \mathbf{u}_{k,i} w_k^o + \boldsymbol{v}_k(i) \tag{2.38}$$

where

- $\boldsymbol{v}_k(i)$ is temporally and spatially white noise whose variance is $\sigma_{v_k}^2$ and which is independent of $\mathbf{d}_{k'}(i')$ for $k \neq k'$ and $i \neq i'$, and $\mathbf{u}_{k',i'}$ for all k' and i', with $k, k' \in \{1, 2, \ldots, N\}$ and $i, i' > 0$;

- $\mathbf{u}_{k,i}$ is independent of $\mathbf{u}_{k',i}$, with $k, k' \in \{1, 2, \ldots, N\}$ and $k \neq k'$ (spatial independence);

- $\mathbf{u}_{k,i}$ is independent of $\mathbf{u}_{k,i'}$, with $i, i' > 0$ and $i \neq i'$ (temporal independence).

2.3.2   Weighted Spatial-Temporal Energy Relation

Now, let us define the local error quantities at each node k:

$$\mathbf{e}_{a,k}(i) \triangleq \mathbf{u}_{k,i} \boldsymbol{\varphi}_{(k-1)}^{(i)} \ (a \ priori \ \text{error}) \tag{2.39}$$

$$\mathbf{e}_{p,k}(i) \triangleq \mathbf{u}_{k,i} \boldsymbol{\varphi}_k^{(i)} \ (a \ posteriori \ \text{error}) \tag{2.40}$$

$$\mathbf{e}_k(i) \triangleq \mathbf{d}_k(i) - \mathbf{u}_{k,i} \begin{bmatrix} \boldsymbol{\psi}_{(k-1)}^{(i)} \\ \boldsymbol{\xi}_k^{(i-1)} \end{bmatrix} \ (\text{output error}) \tag{2.41}$$

where

$$\boldsymbol{\varphi}_k^{(i)} \triangleq \begin{bmatrix} w^o \\ \xi_k^o \end{bmatrix} - \begin{bmatrix} \boldsymbol{\psi}_k^{(i)} \\ \boldsymbol{\xi}_k^{(i)} \end{bmatrix} \ \text{and} \ \boldsymbol{\varphi}_{(k-1)}^{(i)} \triangleq \begin{bmatrix} w^o \\ \xi_k^o \end{bmatrix} - \begin{bmatrix} \boldsymbol{\psi}_{(k-1)}^{(i)} \\ \boldsymbol{\xi}_k^{(i-1)} \end{bmatrix}. \tag{2.42}$$

Note that the output error can also be expressed as

$$\mathbf{e}_k(i) = \mathbf{u}_{k,i} \boldsymbol{\varphi}_{(k-1)}^{(i)} + \boldsymbol{v}_k(i). \tag{2.43}$$

In order to obtain MSD and EMSE at each node k, we need to evaluate the means of the weighted norms of $\boldsymbol{\varphi}_k^{(\infty)}$ and $\boldsymbol{\varphi}_{(k-1)}^{(\infty)}$, as follows:

$$\text{MSD}_k^{tot} \triangleq E\|\boldsymbol{\varphi}_k^{(\infty)}\|_I^2 \tag{2.44}$$

$$\text{EMSE}_k^{tot} \triangleq E|\mathbf{e}_{a,k}(\infty)|^2 = E\|\boldsymbol{\varphi}_{(k-1)}^{(\infty)}\|_{R_{u_k}}^2. \tag{2.45}$$

Thus, to undertake the performance analysis, it firstly suffices to calculate $E\|\boldsymbol{\varphi}_{(k-1)}^{(\infty)}\|_{\Sigma_k}^2$ with $\Sigma_k \succeq 0$, and then particularize the previous performance measure to $\Sigma_k = R_{u_k}$ and

$\Sigma_k = I$. Toward this goal, let us define the weighted *a priori* and *a posteriori* errors at node k [52],

$$\mathbf{e}_{a,k}^{D_k \Sigma_k}(i) \triangleq \mathbf{u}_{k,i} D_k \Sigma_k \boldsymbol{\varphi}_{(k-1)}^{(i)} \tag{2.46}$$

$$\mathbf{e}_{p,k}^{D_k \Sigma_k}(i) \triangleq \mathbf{u}_{k,i} D_k \Sigma_k \boldsymbol{\varphi}_k^{(i)} \tag{2.47}$$

where

$$D_k = \mathrm{diag}(\mu_{\psi_k} I_{M_g}, \mu_{\xi_k} I_{M_l}). \tag{2.48}$$

Next, by particularizing (2.26) for the case where a node k estimates only the global and local parameters (i.e., w^o and ξ_k^o, respectively), and using (2.42), we get the following recursion:

$$\boldsymbol{\varphi}_k^{(i)} = \boldsymbol{\varphi}_{(k-1)}^{(i)} - D_k \mathbf{u}_{k,i}^H \mathbf{e}_k(i). \tag{2.49}$$

Multiplying both sides of (2.49) by $\mathbf{u}_{k,i} D_k \Sigma_k$ from the left, after rearrangement we express the output error vector as

$$\mathbf{e}_k(i) = \frac{\mathbf{e}_{a,k}^{D_k \Sigma_k}(i) - \mathbf{e}_{p,k}^{D_k \Sigma_k}(i)}{\|\mathbf{u}_{k,i}\|_{D_k \Sigma_k D_k}^2}. \tag{2.50}$$

Now, after substituting (2.50) back into (2.49) and equating the weighted norm of both sides, we can write the weighted energy-conservation relation for the analyzed algorithm as

$$\|\boldsymbol{\varphi}_k^{(i)}\|_{\Sigma_k}^2 + \frac{|\mathbf{e}_{a,k}^{D_k \Sigma_k}(i)|^2}{\|\mathbf{u}_{k,i}\|_{D_k \Sigma_k D_k}^2} = \|\boldsymbol{\varphi}_{(k-1)}^{(i)}\|_{\Sigma_k}^2 + \frac{|\mathbf{e}_{p,k}^{D_k \Sigma_k}(i)|^2}{\|\mathbf{u}_{k,i}\|_{D_k \Sigma_k D_k}^2}. \tag{2.51}$$

2.3.3 Weighted Variance Relation

Firstly, note that the weighted *a posteriori* error vector can be expressed from (2.50). Then, substituting it in (2.51) and rearranging, we get

$$\|\boldsymbol{\varphi}_k^{(i)}\|_{\Sigma_k}^2 = \|\boldsymbol{\varphi}_{(k-1)}^{(i)}\|_{\Sigma_k}^2 - \mathbf{e}_{a,k}^{D_k \Sigma_k}(i)^H \mathbf{e}_k(i) - \mathbf{e}_k(i)^H \mathbf{e}_{a,k}^{D_k \Sigma_k}(i) + \|\mathbf{u}_{k,i}\|_{D_k \Sigma_k D_k}^2 |\mathbf{e}_k(i)|^2. \tag{2.52}$$

Now, using (2.43) and taking expectations of both sides, due to the independence of the regression data $\{\mathbf{u}_k\}$ assumed in Subsection 2.3.1, we can rewrite (2.52) into

$$E\|\boldsymbol{\varphi}_k^{(i)}\|_{\Sigma_k}^2 = E\|\boldsymbol{\varphi}_{(k-1)}^{(i)}\|_{\Sigma_k'}^2 + \sigma_{v_k}^2 E\|\mathbf{u}_{k,i}\|_{D_k\Sigma_k D_k}^2 \tag{2.53}$$

where Σ_k' is a deterministic matrix defined as

$$\Sigma_k' = \Sigma_k - \Sigma_k D_k E\left\{\mathbf{u}_{k,i}^H \mathbf{u}_{k,i}\right\} - E\left\{\mathbf{u}_{k,i}^H \mathbf{u}_{k,i}\right\} D_k \Sigma_k + E\left\{\|\mathbf{u}_{k,i}\|_{D_k\Sigma_k D_k}^2 \mathbf{u}_{k,i}^H \mathbf{u}_{k,i}\right\}. \tag{2.54}$$

2.3.4 Gaussian data

In order to evaluate all the terms in the variance relation given in (2.53) and (2.54), we may follow two approaches. One approach is to assume that the regressors come from a circular Gaussian distribution, and then introduce the eigendecomposition and the diagonalization steps [19]. Alternatively, one could use the vectorization operator and the properties of Kronecker product [21], [52]. For simplicity of presentation, we choose the former approach.

To begin with, let us perform the eigendecomposition $R_{u_k} = V_k \Lambda_k V_k^H$ where V_k is unitary and Λ_k is a diagonal matrix with the eigenvalues of R_{u_k}. Also, define the following transformed quantities
$\overline{\boldsymbol{\varphi}}_k \triangleq V_k^H \boldsymbol{\varphi}_k, \ \overline{\boldsymbol{\varphi}}_{(k-1)} \triangleq V_k^H \boldsymbol{\varphi}_{(k-1)}, \ \overline{\mathbf{u}}_k \triangleq \mathbf{u}_k V_k, \ \overline{\Sigma}_k \triangleq V_k^H \Sigma_k V_k,$
$\overline{\Sigma}_k' \triangleq V_k^H \Sigma_k' V_k, \ \overline{D}_k \triangleq V_k^H D_k V_k.$

By following the same arguments as in [19], it is straightforward to show that the variance relation (2.53) can be rewritten as

$$E\|\overline{\boldsymbol{\varphi}}_k^{(i)}\|_{\overline{\Sigma}_k}^2 = E\|\overline{\boldsymbol{\varphi}}_{(k-1)}^{(i)}\|_{\overline{\Sigma}_k'}^2 + \sigma_{v_k}^2 \text{Tr}(\Lambda_k \overline{D}_k \overline{\Sigma}_k \overline{D}_k) \tag{2.55}$$

$$\overline{\Sigma}_k' = \overline{\Sigma}_k - \Lambda_k \overline{D}_k \overline{\Sigma}_k - \overline{\Sigma}_k \overline{D}_k \Lambda_k + \Lambda_k \text{Tr}(\overline{D}_k \overline{\Sigma}_k \overline{D}_k \Lambda_k) + \gamma \Lambda_k \overline{D}_k \overline{\Sigma}_k \overline{D}_k \Lambda_k. \tag{2.56}$$

where $\gamma = 1$ for circular complex data and $\gamma = 2$ for real data.

2.3.5 Steady-State Performance

Next, we focus on obtaining the MSD and EMSE associated with the estimation of the vectors of global and local parameters at each node k. After this derivation, we will be able to get the total MSD and EMSE at each node. To simplify the mean-square analysis in the steady state, we first adopt the following additional assumption:

- $\mathbf{u}_{k_g,i}$ regressor is independent of $\mathbf{u}_{k_l,i}$.

Now, the matrix Λ_k has the following structure

$$\Lambda_k = \text{blockdiag}\{\Lambda_{k,g}, \Lambda_{k,l}\} \tag{2.57}$$

where $\Lambda_{k,g}$ and $\Lambda_{k,l}$ are diagonal matrices with the eigenvalues corresponding to $R_{u_{k_g}}$ and $R_{u_{k_l}}$, respectively, i.e.,

$$R_{u_{k_g}} = V_{k,g}\Lambda_{k,g}V_{k,g}^H \quad \text{and} \quad R_{u_{k_l}} = V_{k,l}\Lambda_{k,l}V_{k,l}^H.$$

Hence, to conduct the performance analysis it suffices to focus on a weighting matrix with a block-diagonal structure, which implies that its transformed counterpart has the structure

$$\overline{\Sigma}_k = \text{blockdiag}\{\overline{\Sigma}_{k,g}, \overline{\Sigma}_{k,l}\} \tag{2.58}$$

where

$$\overline{\Sigma}_{k,g} \triangleq V_{k,g}^H \Sigma_{k,g} V_{k,g} \quad \text{and} \quad \overline{\Sigma}_{k,l} \triangleq V_{k,l}^H \Sigma_{k,l} V_{k,l}. \tag{2.59}$$

At this point, note that $\overline{D}_k = D_k$ and that the term $\text{Tr}(\Lambda_k \overline{D}_k \overline{\Sigma}_k \overline{D}_k)$ can be rewritten as

$$\text{Tr}(\Lambda_k \overline{D}_k \overline{\Sigma}_k \overline{D}_k) = \mu_{\psi_k}^2 \text{Tr}(\Lambda_{k,g}\overline{\Sigma}_{k,g}) + \mu_{\xi_k}^2 \text{Tr}(\Lambda_{k,l}\overline{\Sigma}_{k,l}). \tag{2.60}$$

If we now verify that the transformed quantity $\overline{\boldsymbol{\varphi}}_k^{(i)}$ consists of the global and the local weight error vectors (yet transformed),

$$\overline{\boldsymbol{\psi}}_k^{(i)} \triangleq V_{k,g}^H \boldsymbol{\varphi}_{k,g}^{(i)} \quad \text{and} \quad \overline{\boldsymbol{\xi}}_k^{(i)} \triangleq V_{k,l}^H \boldsymbol{\varphi}_{k,l}^{(i)}$$

where $\boldsymbol{\varphi}_{k,g}^{(i)} \triangleq w^o - \boldsymbol{\psi}_k^{(i)}$ and $\boldsymbol{\varphi}_{k,l}^{(i)} \triangleq \xi_k^o - \boldsymbol{\xi}_k^{(i)}$, respectively, we can write (2.55) and (2.56) for $\overline{\boldsymbol{\psi}}_k^{(i)}$ and $\overline{\boldsymbol{\xi}}_k^{(i)}$ as follows

$$E\|\overline{\boldsymbol{\psi}}_k^{(i)}\|_{\overline{\Sigma}_{k,g}}^2 = E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(i)}\|_{\overline{\Sigma}_{k,g}'}^2 + \mu_{\psi_k}^2 \sigma_{v_k}^2 \text{Tr}(\Lambda_{k,g}\overline{\Sigma}_{k,g}) + \mu_{\psi_k}^2 E\|\overline{\boldsymbol{\xi}}_k^{(i-1)}\|_{\Lambda_{k,l}}^2 \text{Tr}(\Lambda_{k,g}\overline{\Sigma}_{k,g}) \tag{2.61}$$

$$E\|\overline{\boldsymbol{\xi}}_k^{(i)}\|_{\overline{\Sigma}_{k,l}}^2 = E\|\overline{\boldsymbol{\xi}}_k^{(i-1)}\|_{\overline{\Sigma}_{k,l}'}^2 + \mu_{\xi_k}^2 \sigma_{v_k}^2 \text{Tr}(\Lambda_{k,l}\overline{\Sigma}_{k,l}) + \mu_{\xi_k}^2 E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(i)}\|_{\Lambda_{k,g}}^2 \text{Tr}(\Lambda_{k,l}\overline{\Sigma}_{k,l}) \tag{2.62}$$

with

$$\overline{\Sigma}'_{k,g} = \overline{\Sigma}_{k,g} - \mu_{\psi_k} \left(\Lambda_{k,g} \overline{\Sigma}_{k,g} + \overline{\Sigma}_{k,g} \Lambda_{k,g} \right) + \mu_{\psi_k}^2 \left(\Lambda_{k,g} \text{Tr}(\overline{\Sigma}_{k,g} \Lambda_{k,g}) + \gamma \Lambda_{k,g} \overline{\Sigma}_{k,g} \Lambda_{k,g} \right) \tag{2.63}$$

and

$$\overline{\Sigma}'_{k,l} = \overline{\Sigma}_{k,l} - \mu_{\xi_k} \left(\Lambda_{k,l} \overline{\Sigma}_{k,l} + \overline{\Sigma}_{k,l} \Lambda_{k,l} \right) + \mu_{\xi_k}^2 \left(\Lambda_{k,l} \text{Tr}(\overline{\Sigma}_{k,l} \Lambda_{k,l}) + \gamma \Lambda_{k,l} \overline{\Sigma}_{k,l} \Lambda_{k,l} \right). \tag{2.64}$$

It can be easily shown that, when $\mathbf{u}_{k_g,i}$ is independent of $\mathbf{u}_{k_l,i}$ for all $i > 0$ and $k \in \{1, \ldots, N\}$, equation (2.61) represents the weighted variance relation which (instead of using (2.49)) is obtained from the subsequent recursion

$$\boldsymbol{\varphi}_{k,g}^{(i)} = \boldsymbol{\varphi}_{(k-1),g}^{(i)} - \mu_{\psi_k} \mathbf{u}_{k_g,i}^H \mathbf{e}_k(i).$$

Similarly, (2.62) and (2.64) can be obtained from the weighted variance relation that results from considering the update equation corresponding to the estimate of the vector of local parameters at node k and iteration i.

Subsequently, to obtain the closed-form expressions for the mean-square quantities, we proceed by exploiting the diagonal structure of the matrices below

$$\overline{\sigma}_{k,g} \triangleq \text{diag}(\overline{\Sigma}_{k,g}), \quad \overline{\sigma}'_{k,g} \triangleq \text{diag}(\overline{\Sigma}'_{k,g}), \quad \lambda_{k,g} \triangleq \text{diag}(\Lambda_{k,g}),$$
$$\overline{\sigma}_{k,l} \triangleq \text{diag}(\overline{\Sigma}_{k,l}), \quad \overline{\sigma}'_{k,l} \triangleq \text{diag}(\overline{\Sigma}'_{k,l}), \lambda_{k,l} \triangleq \text{diag}(\Lambda_{k,l}),$$

where $a \triangleq \text{diag}(A)$ is a column vector with the diagonal entries of A. Also, we will use $A \triangleq \text{diag}(a)$, where A is a diagonal matrix whose entries are those of the column vector a. In this way, the relation (2.63) is now equivalent to

$$\overline{\sigma}'_{k,g} = \overline{F}_{k,g} \overline{\sigma}_{k,g} \tag{2.65}$$

where the coefficient matrix is

$$\overline{F}_{k,g} \triangleq I - 2\mu_{\psi_k} \Lambda_{k,g} + \gamma \mu_{\psi_k}^2 \Lambda_{k,g}^2 + \mu_{\psi_k}^2 \lambda_{k,g} \lambda_{k,g}^T. \tag{2.66}$$

As a result, the relation (2.61) is

$$E\|\overline{\boldsymbol{\psi}}_k^{(i)}\|^2_{\text{diag}(\overline{\sigma}_{k,g})} = E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(i)}\|^2_{\text{diag}(\overline{F}_{k,g}\overline{\sigma}_{k,g})} + \mu_{\psi_k}^2 \sigma_{v,k}^2 \lambda_{k,g}^T \overline{\sigma}_{k,g}$$
$$+ \mu_{\psi_k}^2 E\|\overline{\boldsymbol{\xi}}_k^{(i-1)}\|^2_{\text{diag}(\lambda_{k,l})} \lambda_{k,g}^T \overline{\sigma}_{k,g}. \tag{2.67}$$

Likewise, the expression (2.62) for evaluating the performance of the local part for each node k is given by

$$
\begin{aligned}
E\|\overline{\boldsymbol{\xi}}_k^{(i)}\|^2_{\text{diag}(\overline{\sigma}_{k,l})} = {} & E\|\overline{\boldsymbol{\xi}}_k^{(i-1)}\|^2_{\text{diag}(\overline{F}_{k,l}\overline{\sigma}_{k,l})} + \mu^2_{\xi_k}\sigma^2_{v,k}\lambda^T_{k,l}\overline{\sigma}_{k,l} \\
& + \mu^2_{\xi_k}E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(i)}\|^2_{\text{diag}(\lambda_{k,g})}\lambda^T_{k,l}\overline{\sigma}_{k,l}
\end{aligned}
\tag{2.68}
$$

with

$$
\overline{F}_{k,l} \triangleq I - 2\mu_{\xi_k}\Lambda_{k,l} + \gamma\mu^2_{\xi_k}\Lambda^2_{k,l} + \mu^2_{\xi_k}\lambda_{k,l}\lambda^T_{k,l}.
\tag{2.69}
$$

For compactness, from now on we drop the $\text{diag}(\cdot)$ notation in (2.67) and (2.68).

We collect now the equations (2.67), (2.68) in steady-state, i.e., as $i \to \infty$, for all the nodes. Thus, we obtain two systems of N equations which are interrelated. By defining

$$
h_{k,g} \triangleq \mu^2_{\psi_k}\lambda_{k,g} \quad \text{and} \quad h_{k,l} \triangleq \mu^2_{\xi_k}\lambda_{k,l},
\tag{2.70}
$$

the $(k-1)^{th}$ equation, out of N equations associated with the parameters of global interest, is written as follows

$$
\begin{aligned}
E\|\overline{\boldsymbol{\psi}}_{k-1}^{(\infty)}\|^2_{\overline{\sigma}_{k-1,g}} = {} & E\|\overline{\boldsymbol{\psi}}_{k-2}^{(\infty)}\|^2_{\overline{F}_{k-1,g}\overline{\sigma}_{k-1,g}} + \sigma^2_{v,k-1}h^T_{k-1,g}\overline{\sigma}_{k-1,g} \\
& + E\|\overline{\boldsymbol{\xi}}_{k-1}^{(\infty)}\|^2_{\lambda_{k-1,l}}h^T_{k-1,g}\overline{\sigma}_{k-1,g}
\end{aligned}
\tag{2.71}
$$

while the $(k-1)^{th}$ equation of the second system, associated with the parameters of local interest, is given by

$$
\begin{aligned}
E\|\overline{\boldsymbol{\xi}}_{k-1}^{(\infty)}\|^2_{\overline{\sigma}_{k-1,l}} = {} & E\|\overline{\boldsymbol{\xi}}_{k-1}^{(\infty)}\|^2_{\overline{F}_{k-1,l}\overline{\sigma}_{k-1,l}} + \sigma^2_{v,(k-1)}h^T_{k-1,l}\overline{\sigma}_{k-1,l} \\
& + E\|\overline{\boldsymbol{\psi}}_{k-2}^{(\infty)}\|^2_{\lambda_{(k-1),g}}h^T_{k-1,l}\overline{\sigma}_{k-1,l}.
\end{aligned}
\tag{2.72}
$$

Observe now that the desired performance measures for each node k can be expressed as follows

$$
\text{MSD}_k^{tot} = \text{MSD}_k^g + \text{MSD}_k^l = E\|\overline{\boldsymbol{\psi}}_k^{(\infty)}\|^2_{1_{M_g}} + E\|\overline{\boldsymbol{\xi}}_k^{(\infty)}\|^2_{1_{M_l}},
\tag{2.73}
$$

$$
\text{EMSE}_k^{tot} = \text{EMSE}_k^g + \text{EMSE}_k^l = E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|^2_{\lambda_{k,g}} + E\|\overline{\boldsymbol{\xi}}_k^{(\infty)}\|^2_{\lambda_{k,l}},
\tag{2.74}
$$

where $k \in \{1, 2, \ldots, N\}$ and 1_M denotes a $M \times 1$ vector of ones.

Subsequently, let us express the EMSE_k^l from the kth equation in (2.72), which can be rewritten as

$$E\|\overline{\boldsymbol{\xi}}_k^{(\infty)}\|_{[I-\overline{F}_{k,l}]\overline{\sigma}_{k,l}}^2 = \sigma_{v,k}^2\, h_{k,l}^T\overline{\sigma}_{k,l} + E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|_{\lambda_{k,g}}^2\, h_{k,l}^T\overline{\sigma}_{k,l}. \tag{2.75}$$

To this end, the weighting vector $\overline{\sigma}_{k,l}$ should satisfy the following relation

$$\left[I - \overline{F}_{k,l}\right]\overline{\sigma}_{k,l} = \lambda_{k,l}. \tag{2.76}$$

In this way, for all $k \in \{1, 2, \ldots, N\}$, it can be verified that

$$\text{EMSE}_k^l = \zeta_{k,l}\left(\sigma_{v,k}^2 + \text{EMSE}_k^g\right) \tag{2.77}$$

with

$$\zeta_{k,l} = h_{k,l}^T\left[I - \overline{F}_{k,l}\right]^{-1}\lambda_{k,l}. \tag{2.78}$$

By substituting the previous relation for each node k into (2.71), the kth equation in (2.71) takes the following form

$$E\|\overline{\boldsymbol{\psi}}_k^{(\infty)}\|_{\overline{\sigma}_{k,g}}^2 = E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|_{B_{k,g}\overline{\sigma}_{k,g}}^2 + b_{k,g}^T\overline{\sigma}_{k,g} \tag{2.79}$$

where, for $k \in \{1, 2, \ldots, N\}$,

$$B_{k,g} = \overline{F}_{k,g} + \zeta_{k,l}\lambda_{k,g}h_{k,g}^T \tag{2.80}$$

and

$$b_{k,g} = \sigma_{v,k}^2(1 + \zeta_{k,l})h_{k,g}. \tag{2.81}$$

By iterating (2.79) over the whole network cycle, and by choosing the proper weighting vectors $\overline{\sigma}_{k,g}$ for each k, it can be verified that

$$\begin{aligned}
E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|_{\overline{\sigma}_{(k-1),g}}^2 = {}& E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|_{B_{k,g}\cdots B_{N,g}B_{1,g}\cdots B_{(k-1),g}\overline{\sigma}_{(k-1),g}}^2 \\
& + b_{k,g}^T B_{k+1,g}\cdots B_{N,g}B_{1,g}\cdots B_{(k-1),g}\overline{\sigma}_{(k-1),g} \\
& + b_{k+1,g}^T B_{k+2,g}\cdots B_{N,g}B_{1,g}\cdots B_{(k-1),g}\overline{\sigma}_{(k-1),g} \\
& + \cdots + b_{k-2,g}^T B_{(k-1),g}\overline{\sigma}_{(k-1),g} + b_{(k-1),g}\overline{\sigma}_{(k-1),g}
\end{aligned} \tag{2.82}$$

where all subscripts are either $\mod N$ or N if the result of the $\mod N$ operation equals 0.

If we now define the transition matrix

$$\Pi_{k,g,\ell} \triangleq B_{k+\ell-1,g} \cdots B_{N,g} B_{1,g} \cdots B_{(k-1),g} \qquad (2.83)$$

with $\ell, k \in \{1, 2, \ldots, N\}$, (2.82) can straightforwardly be rewritten as

$$E\|\overline{\psi}_{(k-1)}^{(\infty)}\|_{[I-\Pi_{k,g,1}]\overline{\sigma}_{(k-1),g}}^2 = a_{k,g}^T \overline{\sigma}_{(k-1),g}. \qquad (2.84)$$

where

$$a_{k,g}^T \triangleq b_{k,g}^T \Pi_{k,g,2} + b_{k+1,g}^T \Pi_{k,g,3} + \cdots + b_{k-2,g}^T \Pi_{k,g,N} + b_{(k-1),g}^T. \qquad (2.85)$$

By choosing $\overline{\sigma}_{(k-1),g}$ as the solutions of the linear equations $[I - \Pi_{k,g,1}]\overline{\sigma}_{(k-1),g} = \lambda_{k,g}$ and $[I - \Pi_{k,g,1}]\overline{\sigma}_{(k-1),g} = 1_{M_g}$, according to the definitions given in (2.73) and (2.74) we obtain the desired EMSE_k^g and MSD_k^g for each node $k \in \{1, 2, \ldots, N\}$, respectively,

$$\text{EMSE}_k^g = a_{k,g}^T [I - \Pi_{k,g,1}]^{-1} \lambda_{k,g} \qquad (2.86)$$

and

$$\text{MSD}_{k-1}^g = a_{k,g}^T [I - \Pi_{k,g,1}]^{-1} 1_{M_g}. \qquad (2.87)$$

Now, we express the estimation performance measures related to the parameter of local interest, at node k. In order to obtain MSD_k^l, from (2.75) we can check that the weighting vector $\overline{\sigma}_{k,l}$ has to satisfy

$$\left[I - \overline{F}_{k,l}\right] \overline{\sigma}_{k,l} = 1_{M_l}. \qquad (2.88)$$

Thus, taking into account the definition of EMSE_k^g provided in (2.74), we can note that

$$\text{MSD}_k^l = \eta_{k,l} \left(\sigma_{v,k}^2 + \text{EMSE}_k^g\right) \qquad (2.89)$$

with

$$\eta_{k,l} = h_{k,l}^T \left[I - \overline{F}_{k,l}\right]^{-1} 1_{M_l}. \qquad (2.90)$$

Finally, from (2.77), (2.86), (2.87) and (2.89), for the definitions in (2.73) and (2.74) we get

$$\text{MSD}_k^{tot} = a_{k+1,g}^T [I - \Pi_{k+1,g,1}]^{-1} 1_{M_g} + \eta_{k,l} \left(\sigma_{v,k}^2 + a_{k,g}^T [I - \Pi_{k,g,1}]^{-1} \lambda_{k,g}\right) \qquad (2.91)$$

and

$$\text{EMSE}_k^{tot} = a_{k,g}^T \left[I - \Pi_{k,g,1} \right]^{-1} \lambda_{k,g} + \zeta_{k,l} \left(\sigma_{v,k}^2 + a_{k,g}^T \left[I - \Pi_{k,g,1} \right]^{-1} \lambda_{k,g} \right). \qquad (2.92)$$

As it is expected from (2.77) and (2.89), the expressions (2.91) and (2.92) indicate a coupling effect between the estimation processes for the global and local parameters at each node.

2.4   Performance analysis: A scenario with global, common and local parameters

The performance analysis steps for the more general case, where local, common and global interests coexist, depend also on the way the sets of nodes $\{\mathcal{C}_j\}_{j=1}^J$ are overlapped. Due to this fact, for the sake of generality, we will show the main steps that have to be followed to obtain the performance analysis once the different node-specific interests have been defined.

2.4.1   Weighted Variance Relation for Gaussian data

For a scenario where some subsets of nodes are interested in estimating several vectors of common parameters, let us firstly make the subsequent redefinitions

$$\boldsymbol{\varphi}_k^{(i)} \triangleq \begin{bmatrix} w^o \\ \tilde{\varsigma}_k^o \\ \xi_k^o \end{bmatrix} - \begin{bmatrix} \psi_k^{(i)} \\ \tilde{\varsigma}_k^{(i)} \\ \xi_k^{(i)} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\varphi}_{(k-1)}^{(i)} \triangleq \begin{bmatrix} w^o \\ \tilde{\varsigma}_k^o \\ \xi_k^o \end{bmatrix} - \begin{bmatrix} \psi_{(k-1)}^{(i)} \\ \tilde{\varsigma}_{(k-1)}^{(i)} \\ \xi_k^{(i-1)} \end{bmatrix} \qquad (2.93)$$

and $D_k = \text{diag}\left(\mu_{\psi_k} I_{M_g}, D_{\tilde{\varsigma}_k}, \mu_{\xi_k} I_{M_l} \right)$, with

$$\tilde{\varsigma}_k^o = \text{col}\left\{ \varsigma_{\mathcal{I}_k(1)}^o, \varsigma_{\mathcal{I}_k(2)}^o, \ldots, \varsigma_{\mathcal{I}_k(|\mathcal{I}_k|)}^o \right\} \qquad (2.94)$$

and $D_{\tilde{\varsigma}_k} = \text{diag}\left(\mu_{\varsigma_{k,\mathcal{I}_k(1)}} I_{M_c}, \ldots, \mu_{\varsigma_{k,\mathcal{I}_k(|\mathcal{I}_k|)}} I_{M_c} \right).$

At this point, under the assumptions stated in Subsection 2.3.1 it can be straightforwardly shown that the weighted variance relation for Gaussian data follows the same expression as the one provided in (2.55) and (2.56) for a scenario where there are only parameters of global and local interest. Consequently, assuming that $\mathbf{u}_{k_g,i}$, $\mathbf{u}_{k_l,i}$ and $\mathbf{u}_{k_{jc},i}$ are independent for any $j \in \{1, 2, \ldots, J\}$, by following the same reasoning as in

Subsection 2.3.5 we can particularize (2.55) and (2.56) to

$$
\begin{aligned}
E\|\overline{\boldsymbol{\psi}}_k^{(i)}\|_{\overline{\Sigma}_{k,g}}^2 &= E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(i)}\|_{\overline{\Sigma}'_{k,g}}^2 + \mu_{\psi_k}^2 \sigma_{v_k}^2 \mathrm{Tr}(\Lambda_{k,g}\overline{\Sigma}_{k,g}) \\
&\quad + \mu_{\psi_k}^2 E\|\overline{\boldsymbol{\xi}}_k^{(i-1)}\|_{\Lambda_{k,l}}^2 \mathrm{Tr}(\Lambda_{k,g}\overline{\Sigma}_{k,g}) + \mu_{\psi_k}^2 \sum_{j\in\mathcal{I}_k} E\|\overline{\boldsymbol{\varsigma}}_{f_j(k),j}^{(i)}\|_{\Lambda_{k,j}}^2 \mathrm{Tr}(\Lambda_{k,g}\overline{\Sigma}_{k,g})
\end{aligned}
\tag{2.95}
$$

$$
\begin{aligned}
E\|\overline{\boldsymbol{\varsigma}}_{k,j}^{(i)}\|_{\overline{\Sigma}_{k,j}}^2 &= E\|\overline{\boldsymbol{\varsigma}}_{f_j(k),j}^{(i)}\|_{\overline{\Sigma}'_{k,j}}^2 + \mu_{\varsigma_{k,j}}^2 \sigma_{v,k}^2 \mathrm{Tr}(\Lambda_{k,j}\overline{\Sigma}_{k,j}) + \mu_{\varsigma_{k,j}}^2 E\|\overline{\boldsymbol{\xi}}_k^{(i-1)}\|_{\Lambda_{k,l}}^2 \mathrm{Tr}(\Lambda_{k,j}\overline{\Sigma}_{k,j}) \\
&\quad + \mu_{\varsigma_{k,j}}^2 E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(i)}\|_{\Lambda_{k,g}}^2 \mathrm{Tr}(\Lambda_{k,j}\overline{\Sigma}_{k,j}) + \mu_{\varsigma_{k,j}}^2 \sum_{\substack{j'\in\mathcal{I}_k \\ j'\neq j}} E\|\overline{\boldsymbol{\varsigma}}_{f_{j'}(k),j'}^{(i)}\|_{\Lambda_{k,j'}}^2 \mathrm{Tr}(\Lambda_{k,j}\overline{\Sigma}_{k,j})
\end{aligned}
\tag{2.96}
$$

and

$$
\begin{aligned}
E\|\overline{\boldsymbol{\xi}}_k^{(i)}\|_{\overline{\Sigma}_{k,l}}^2 &= E\|\overline{\boldsymbol{\xi}}_k^{(i-1)}\|_{\overline{\Sigma}'_{k,l}}^2 + \mu_{\xi_k}^2 \sigma_{v_k}^2 \mathrm{Tr}(\Lambda_{k,l}\overline{\Sigma}_{k,l}) \\
&\quad + \mu_{\xi_k}^2 E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(i)}\|_{\Lambda_{k,g}}^2 \mathrm{Tr}(\Lambda_{k,l}\overline{\Sigma}_{k,l}) + \mu_{\xi_k}^2 \sum_{j\in\mathcal{I}_k} E\|\overline{\boldsymbol{\varsigma}}_{f_j(k),j}^{(i)}\|_{\Lambda_{k_j}}^2 \mathrm{Tr}(\Lambda_{k,l}\overline{\Sigma}_{k,l})
\end{aligned}
\tag{2.97}
$$

where, for $k \in \{1,\dots,N\}$, $\overline{\Sigma}_{k,g}$, $\overline{\Sigma}_{k,l}$, $\overline{\Sigma}'_{k,g}$ and $\overline{\Sigma}'_{k,l}$ are given in (2.59), (2.63) and (2.64), respectively. In (2.96), for any $j \in \mathcal{I}_k$,

$$
\overline{\Sigma}'_{k,j} = \overline{\Sigma}_{k,j} - \mu_{\varsigma_{k,j}} \left(\Lambda_{k,j}\overline{\Sigma}_{k,j} + \overline{\Sigma}_{k,j}\Lambda_{k,j}\right) + \mu_{\varsigma_{k,j}}^2 \left(\Lambda_{k,j}\mathrm{Tr}(\overline{\Sigma}_{k,j}\Lambda_{k,j}) + \gamma\Lambda_{k,j}\overline{\Sigma}_{k,j}\Lambda_{k,j}\right)
\tag{2.98}
$$

where $\Lambda_{k,j}$ equals to a diagonal matrix whose entries are the eigenvalues corresponding to $R_{u_{k_j}}$, i.e., $R_{u_{k_j}} = V_{k,j}\Lambda_{k,j}V_{k,j}^H$, and where $\overline{\Sigma}_{k,j} \triangleq V_{k,j}^H \Sigma V_{k,j}$ and $\overline{\boldsymbol{\varsigma}}_{k,j}^{(i)} \triangleq V_{k,j}^H \boldsymbol{\varphi}_{k,j}^{(i)}$ with $\boldsymbol{\varphi}_{k,j}^{(i)} = \varsigma_j^o - \boldsymbol{\varsigma}_{k,j}^{(i)}$.

2.4.2 Diagonalization and Steady-State Performance

In order to leverage the diagonal structure of $\overline{\Sigma}_{k,g}$, $\overline{\Sigma}_{k,l}$ and $\overline{\Sigma}_{k,j}$ with $j \in \mathcal{I}_k$, we additionally define

$$
\overline{\sigma}_{k,j} = \mathrm{diag}(\overline{\Sigma}_{k,j}), \ \lambda_{k,j} \triangleq \mathrm{diag}(\Lambda_{k,j})
$$

and $\overline{F}_{k,j} \triangleq I - 2\mu_{\varsigma_{k,j}}\Lambda_{k,j} + \gamma\mu_{\varsigma_{k,j}}^2\Lambda_{k,j}^2 + \mu_{\varsigma_{k,j}}^2\lambda_{k,j}\lambda_{k,j}^T$.

Next, we proceed with the derivation of the steady state equations associated with the vectors of global, common and local parameters at node k, i.e., $\overline{\boldsymbol{\psi}}_k^{(\infty)}$, $\overline{\boldsymbol{\varsigma}}_{k,j}^{(\infty)}$ and $\overline{\boldsymbol{\xi}}_k^{(\infty)}$,

respectively, with $k \in \{1, 2, \ldots, N\}$ and $j \in \mathcal{I}_k$. In particular, after defining

$$h_{k,j} \triangleq \mu^2_{\varsigma_{k,j}} \lambda_{k,j} \tag{2.99}$$

for $k \in \{1, 2, \ldots, N\}$ and $j \in \mathcal{I}_k$, from the evaluation of (2.95)-(2.97) at $i = \infty$ we obtain

$$\begin{aligned}
E\|\overline{\boldsymbol{\psi}}_k^{(\infty)}\|^2_{\overline{\sigma}_{k,g}} = E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|^2_{\overline{F}_{k,g}\overline{\sigma}_{k,g}} &+ h^T_{k,g}\overline{\sigma}_{k,g}\sigma^2_{v,k} \\
&+ h^T_{k,g}\overline{\sigma}_{k,g}\mathrm{EMSE}^l_k + h^T_{k,g}\overline{\sigma}_{k,g}\sum_{j\in\mathcal{I}_k}\mathrm{EMSE}^{c_j}_k
\end{aligned} \tag{2.100}$$

$$\begin{aligned}
E\|\overline{\boldsymbol{\varsigma}}_{k,j}^{(\infty)}\|^2_{\overline{\sigma}_{k_j}} = E\|\overline{\boldsymbol{\varsigma}}_{f_j(k),j}^{(\infty)}\|^2_{\overline{F}_{k,j}\overline{\sigma}_{k,j}} &+ h^T_{k,j}\overline{\sigma}_{k,j}\,\sigma^2_{v,k} \\
&+ h^T_{k,j}\overline{\sigma}_{k,j}\mathrm{EMSE}^l_k + h^T_{k,j}\overline{\sigma}_{k,j}\mathrm{EMSE}^g_k + h^T_{k,j}\overline{\sigma}_{k,j}\sum_{\substack{j'\in\mathcal{I}_k \\ j'\neq j}}\mathrm{EMSE}^{c_{j'}}_k
\end{aligned} \tag{2.101}$$

and

$$E\|\overline{\boldsymbol{\xi}}_k^{(\infty)}\|^2_{\overline{\sigma}_{k,l}} = E\|\overline{\boldsymbol{\xi}}_k^{(\infty)}\|^2_{\overline{F}_{k,l}\overline{\sigma}_{k,l}} + h^T_{k,l}\overline{\sigma}_{k,l}\sigma^2_{v,k} + h^T_{k,l}\overline{\sigma}_{k,l}\mathrm{EMSE}^g_k + h^T_{k,l}\overline{\sigma}_{k,l}\sum_{j\in\mathcal{I}_k}\mathrm{EMSE}^{c_j}_k \tag{2.102}$$

where, for $k \in \{1, 2, \ldots, N\}$, $h_{k,g}$, $h_{k,l}$, EMSE^g_k and EMSE^l_k are given in (2.70) and (2.74), respectively, and where

$$\mathrm{EMSE}^{c_j}_k \triangleq E\|\overline{\boldsymbol{\varsigma}}_{f_j(k),j}^{(\infty)}\|^2_{\lambda_{k,j}} \tag{2.103}$$

for $j \in \mathcal{I}_k$.

If we now take into account the definition of EMSE^l_k provided in (2.74) and follow the same steps as in Subsection 2.3.5, we can verify that, when $\overline{\sigma}_{k,l}$ satisfies (2.76), then (2.102) leads to

$$\mathrm{EMSE}^l_k = \zeta_{k,l}\left[\sigma^2_{v,k} + \mathrm{EMSE}^g_k + \sum_{j\in\mathcal{I}_k}\mathrm{EMSE}^{c_j}_k\right] \tag{2.104}$$

with $\zeta_{k,l}$ defined in (2.78) and $k \in \{1, 2, \ldots, N\}$. Analogously,

$$\mathrm{MSD}^l_k = \eta_{k,l}\left[\sigma^2_{v,k} + \mathrm{EMSE}^g_k + \sum_{j\in\mathcal{I}_k}\mathrm{EMSE}^{c_j}_k\right] \tag{2.105}$$

with $\eta_{k,l}$ defined in (2.90) and $k \in \{1, 2, \ldots, N\}$.

From (2.104) and (2.105) we can check that, for each node $k \in \{1, 2, \ldots, N\}$, both

EMSE_k^l and MSD_k^l are dependent on EMSE_k^g and $\{\text{EMSE}_k^{c_j}\}_{j \in \mathcal{I}_k}$. Thus, to obtain an analytical expression for the aforementioned performance measures, we need to derive closed-form expressions for the mean-square measures associated with the estimation of w^o and $\{\varsigma_j^o\}_{j \in \mathcal{I}_k}$. Toward this goal, let us first substitute the right-hand side of (2.104) into the right-hand side of (2.101). As a result of this substitution, after some algebraic manipulations, we get

$$
\begin{aligned}
E\|\overline{\varsigma}_{k,j}^{(\infty)}\|_{\overline{\sigma}_{k,j}}^2 = {} & E\|\overline{\varsigma}_{f_j(k),j}^{(\infty)}\|_{B_{k,j}\overline{\sigma}_{k,j}}^2 + b_{k,j}^T \overline{\sigma}_{k,j} \\
& + h_{k,j}^T \overline{\sigma}_{k,j}(1 + \zeta_{k,l}) \text{EMSE}_k^g + h_{k,j}^T \overline{\sigma}_{k,j}(1 + \zeta_{k,l}) \sum_{\substack{j' \in \mathcal{I}_k \\ j' \neq j}} \text{EMSE}_k^{c_{j'}}
\end{aligned}
\tag{2.106}
$$

with

$$
B_{k,j} = \overline{F}_{k,j} + \zeta_{k,l}\lambda_{k,j}h_{k,j}^T
\tag{2.107}
$$

and

$$
b_{k,j} = \sigma_{v,k}^2(1 + \zeta_{k,l})h_{k,j}.
\tag{2.108}
$$

Subsequently, similarly to the steps undertaken in Subsection 2.3.5, by only focusing on the nodes interested in estimating ς_j^o we can construct a set of $n_j = |\mathcal{C}_j|$ equations, whose k-th member is given by (2.106). After defining $f_j^\ell(k)$ as the repeated composition of $f_j(k)$ with itself $\ell - 1$ times, i.e.,

$$
f_j^\ell(k) = f_j(k) \circ \underbrace{f_j(k) \circ f_j(k) \circ \cdots \circ f_j(k)}_{\ell - 1 \text{ times}}
$$

with $\ell, k \in \{1, 2, \ldots, n_j\}$ and $j \in \mathcal{I}_k$, we now iterate (2.106) over the set $\mathcal{C}_j$ and choose the proper weighting vectors $\overline{\sigma}_{k,j}$ for each $k \in \mathcal{C}_j$. Thus, it can be verified that

$$
\begin{aligned}
E\|\overline{\varsigma}_{f_j(k),j}^{(\infty)}\|_{\overline{\sigma}_{f_j(k),j}}^2 = {} & E\|\overline{\varsigma}_{f_j(k),j}^{(\infty)}\|_{\Pi_{k,j,1}\overline{\sigma}_{f_j(k),j}}^2 + a_{k,j}^T \overline{\sigma}_{f_j(k),j} \\
& + \left[h_{k,j}^g \right]^T \overline{\sigma}_{f_j(k),j} + \left[h_{k,j}^c \right]^T \overline{\sigma}_{f_j(k),j}
\end{aligned}
\tag{2.109}
$$

where

$$
a_{k,j}^T \triangleq b_{k,j}^T \Pi_{k,j,2} + b_{f_j^{n_j-1}(k),j}^T \Pi_{k,j,3} + \cdots + b_{f_j^2(k),j}^T \Pi_{k,j,n_j} + b_{f_j(k),j}^T,
\tag{2.110}
$$

$$
\Pi_{k,j,\ell} \triangleq B_{f_j^{n_j-\ell+1}(k),j} \cdots B_{\max\{\mathcal{C}_j\},j} B_{\mathcal{C}_j(1),j} \cdots B_{f_j(k),j},
\tag{2.111}
$$

with

$$
\begin{aligned}
\left[\, h_{k,j}^g \,\right]^T &\triangleq h_{k,j}^T\, \Pi_{k,j,2}\,(1+\zeta_{k,l})\mathrm{EMSE}_k^g + h_{f_j^{n_j-1}(k),j}^T \Pi_{k,j,3}(1+\zeta_{f_j^{n_j-1}(k),l})\mathrm{EMSE}_{f_j^{n_j-1}(k)}^g \\
&\quad + \cdots + h_{f_j^2(k),j}^T\Pi_{k,j,n_j}(1+\zeta_{f_j^2(k),l})\mathrm{EMSE}_{f_j^2(k)}^g + h_{f_j(k),j}^T(1+\zeta_{f_j(k),l})\mathrm{EMSE}_{f_j(k)}^g
\end{aligned}
\tag{2.112}
$$

and

$$
\begin{aligned}
\left[\, h_{k,j}^c \,\right]^T &\triangleq h_{k,j}^T\, \Pi_{k,j,2}\,(1+\zeta_{k,l}) \sum_{j'\in\mathcal{I}_k, j'\neq j} \mathrm{EMSE}_k^{c_{j'}} + \cdots \\
&\quad + h_{f_j^2(k),j}^T\Pi_{k,j,n_j}(1+\zeta_{f_j^2(k),l}) \sum_{j'\in\mathcal{I}_{f_j^2(k)}, j'\neq j} \mathrm{EMSE}_{f_j^2(k)}^{c_{j'}} \\
&\quad + h_{f_j(k),j}^T(1+\zeta_{f_j(k),l}) \sum_{j'\in\mathcal{I}_{f_j(k)}, j'\neq j} \mathrm{EMSE}_{f_j(k)}^{c_{j'}}.
\end{aligned}
\tag{2.113}
$$

Now, according to (2.103) for $\overline{\sigma}_{f_j(k),j} = [I_{M_c} - \Pi_{k,j,1}]^{-1}\lambda_{k,j}$, the expression provided in (2.109) leads to

$$
\mathrm{EMSE}_k^{c_j} = a_{k,j}^T\,[I_{M_c} - \Pi_{k,j,1}]^{-1}\lambda_{k,j} + \left[h_{k,j}^c + h_{k,j}^g\right]^T [I_{M_c} - \Pi_{k,j,1}]^{-1}\lambda_{k,j}
\tag{2.114}
$$

for $k \in \{1, 2, \ldots, N\}$ and $j \in \mathcal{I}_k$. Similarly, according to the subsequent definition

$$
\mathrm{MSD}_k^{c_j} \triangleq E\|\overline{\varsigma}_{k,j}^{(\infty)}\|_{1_{M_c}}^2,
\tag{2.115}
$$

the expression (2.109) yields

$$
\mathrm{MSD}_{f_j(k)}^{c_j} = a_{k,j}^T\,[I_{M_c} - \Pi_{k,j,1}]^{-1}\,1_{M_c} + \left[h_{k,j}^c + h_{k,j}^g\right]^T [I_{M_c} - \Pi_{k,j,1}]^{-1}\,1_{M_c}.
\tag{2.116}
$$

with $k \in \{1, 2, \ldots, N\}$ and $j \in \mathcal{I}_k$.

An analysis of (2.114) and (2.116) reveals that both $\mathrm{EMSE}_k^{c_j}$ and $\mathrm{MSD}_{f_j(k)}^{c_j}$ are dependent on $\{\mathrm{EMSE}_{k'}^g\}_{k'\in\mathcal{C}_j}$ and

$$
\{\mathrm{EMSE}_{k'}^{c_{j'}}\}_{k'\in\mathcal{C}_j, j'\in\mathcal{I}_{k'}\setminus j}.
$$

As a result, to derive closed-form expressions of $\mathrm{EMSE}_k^{c_j}$ and $\mathrm{MSD}_{f_j(k)}^{c_j}$ that are only dependent on the performance measures associated with the estimation of the vector of global interest, we have to solve a linear system of equations. Each equation of this system is given by the expressions provided in (2.114) for $\mathrm{EMSE}_{k'}^{c_{j'}}$, where $j' \in \mathcal{I}_{k'}$ and $k' \in \{\Theta_j\}$ with Θ_j equal to the set that results from ordering the connected component of

$$
\cup_{j'=1}^J \mathcal{G}_{j'}
\tag{2.117}
$$
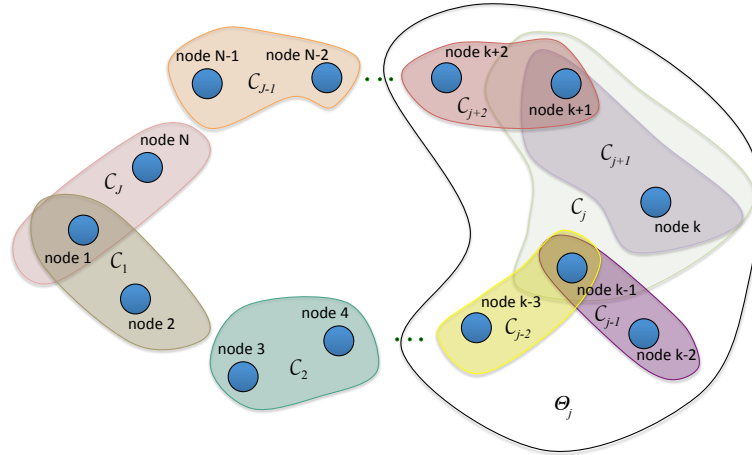
FIGURE 2.4: Connected component associated with the vector of common parameters ς_j^o is denoted Θ_j.

that contains the set $\mathcal{C}_j$ (see Fig. 2.4).

In (2.117), the graph $\mathcal{G}_{j'}$ denotes a directed graph whose nodes are $\mathcal{C}_{j'}$ and whose directed edges are given by $\{(f_{j'}(k), k)\}_{k \in \mathcal{C}_{j'}}$. Hence, the aforementioned linear system has the following form

$$A_{\Theta_j}^c \cdot EMSE_{\Theta_j}^c = b_{\Theta_j}^c \qquad (2.118)$$

where $EMSE_{\Theta_j}^c$ is the vector formed by stacking $EMSE_{k'}^{c_j'}$, with $k' \in \Theta_j$ and $j' \in \mathcal{I}_{k'}$, according to the bijective function

$$f_{\Theta_j}(k', j') = \sum_{m=1}^{|\Theta_{j,k'}|} |\mathcal{I}_{\Theta_j(m)}| + |\mathcal{I}_{k',j'}| + 1 \qquad (2.119)$$

defined on

$$\mathcal{D}_j = \left\{ (k', j') \in \Theta_j \times \{1, 2, \ldots, J\} \mid \forall k' \in \Theta_j, j' \in \mathcal{I}_{k'} \right\} \qquad (2.120)$$

where $\mathcal{I}_{k,j'} = \{j'' \in \mathcal{I}_k | j'' < j'\}$, and $\Theta_{j,k'} = \{k'' \in \Theta_j | k'' < k'\}$. Accordingly, for $k', k'' \in \Theta_j$, $j' \in \mathcal{I}_{k'}$ and $j'' \in \mathcal{I}_{k''}$,

$$b_{\Theta_j}^c(f_{\Theta_j}(k', j')) = \left[a_{k',j'} + h_{k',j'}^g \right]^T \left[I_{M_c} - \Pi_{k',j',1} \right]^{-1} \lambda_{k',j'} \qquad (2.121)$$

and

$$
A^c_{\Theta_j}(f_{\Theta_j}(k',j'), f_{\Theta_j}(k'',j'')) = \begin{cases} 1 & \text{if } (k',j') = (k'',j'') \\ -(1+\zeta_{k'',l})h^T_{k'',j'} & \text{if } j' \neq j'' \text{ and} \\ \times \, \Pi_{k',j',\rho^c(k',k'',j')} & \mathcal{I}_{k'} \cap \mathcal{I}_{k''} \neq \emptyset, \\ \times \left[I_{M_c} - \Pi_{k',j',1} \right]^{-1} \lambda_{k',j'} & \\ 0 & \text{otherwise} \end{cases}
$$

$$(2.122)$$

where $\rho^c(k',k'',j') = \text{mod}(\Delta^c_{k'\to k'',j'} + 2, n_{j'} + 1)$, and where $\Delta^c_{k'\to k'',j'}$ equals the number of sub-iterations from node k' to node k'' when estimating the common vector of parameters $\varsigma^o_{j'}$ with $\Delta^c_{k'\to k',j'} = 0$ and $\Pi_{k',j',0} = I_{M_c}$.

Due to the fact that the elements of $A^c_{\Theta_j}$ do not depend on $\{\text{EMSE}^g_{k'}\}^N_{k'=1}$, the solution of (2.118) is given by a linear combination of $\{\text{EMSE}^g_{k'}\}_{k'\in\Theta_j}$. Consequently, after solving as many linear systems of equations as the number of connected components in $\cup^J_{j'=1}\mathcal{G}_{j'}$, from (2.114) and (2.116) we are able to express $\text{EMSE}^{c_j}_k$ and $\text{MSD}^{c_j}_k$ as a linear combination of $\{\text{EMSE}^g_{k'}\}^N_{k'=1}$ with $k \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, J\}$. At the same time, after solving the aforementioned linear system of equations, from (2.104) and (2.105) we are able to derive closed-form solutions for EMSE^l_k and MSD^l_k that only depend on EMSE^g_k with $k \in \{1, 2, \ldots, N\}$.

For scenarios where there is no vector w^o of global parameters in the network, we can easily check that the entries of the vector $b^c_{\Theta_j}$ defined in (2.121) are not dependent on $\{\text{EMSE}^g_{k'}\}^N_{k'=1}$ for any $j \in \{1, 2 \ldots, J\}$. In this case, by solving (2.118) for each connected component of the graph defined in (2.117), we are able to obtain closed-form solutions of $\{\text{EMSE}^{c_j}_k\}_{k\in\mathcal{C}_j}$ for any $j \in \{1, 2, \ldots, J\}$. Afterwards, since EMSE^l_k, MSD^l_k and $\text{MSD}^{c_j}_{f_j(k)}$ do not depend on $\{\text{EMSE}^g_{k'}\}^N_{k'=1}$, closed-form expressions for these mean-square measures can be obtained from the substitution of the solution of (2.118) into (2.104), (2.105) and (2.116). Unlike the scenario where there are only parameters of local and global interest, due to the overlapping among the areas of influence $\{\mathcal{C}_j\}^J_{j=1}$, note that the closed-form expressions are now obtained after solving linear systems of equations of the form given in (2.118).

To conclude the derivation of all the mean-square measures that quantify the estimation performance at each node for a scenario where there is a vector of global parameters, we only need to obtain closed-form expressions for $\{\text{EMSE}^g_{k'}, \text{MSD}^g_{k'}\}^N_{k'=1}$. Toward this goal, we firstly substitute the right-hand side of (2.104) into the right-hand

side of (2.100) to get the subsequent expression

$$E\|\overline{\boldsymbol{\psi}}_k^{(\infty)}\|_{\overline{\sigma}_{k,g}}^2 = E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|_{B_{k,g}\overline{\sigma}_{k,g}}^2 + b_{k,g}^T\overline{\sigma}_{k,g} + h_{k,g}^T\overline{\sigma}_{k,g}(1+\zeta_{k,l})\sum_{j'\in\mathcal{I}_k}\mathrm{EMSE}_k^{c_{j'}} \quad (2.123)$$

where, for $k \in \{1, 2, \ldots, N\}$, $B_{k,g}$ and $b_{k,g}$ are provided in (2.80) and (2.81) respectively. Note that there is a total of N equations with the same form as (2.123), each one per node k. Consequently, if we take the equation corresponding to node $k - 1$ and iterate over the other $N - 1$ other equations by choosing the proper vectors $\overline{\sigma}_{k,g}$, then similarly to Subsection 2.3.5, we obtain

$$E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|_{\overline{\sigma}_{(k-1),g}}^2 = E\|\overline{\boldsymbol{\psi}}_{(k-1)}^{(\infty)}\|_{\Pi_{k,g,1}\overline{\sigma}_{(k-1),g}}^2 + a_{k,g}^T\overline{\sigma}_{(k-1),g} + \big[\, h_{k,g}^c \,\big]^T \overline{\sigma}_{(k-1),g} \quad (2.124)$$

with $a_{k,g}^T$ given in (2.85) and

$$
\begin{aligned}
\big[\, h_{k,g}^c \,\big]^T \triangleq\ & h_{k,g}^T \Pi_{k,g,2}\,(1+\zeta_{k,l})\sum_{j'\in\mathcal{I}_k}\mathrm{EMSE}_k^{c_{j'}} \\
& + h_{k+1,g}^T\Pi_{k,g,3}(1+\zeta_{k+1,l})\sum_{j'\in\mathcal{I}_{k+1}}\mathrm{EMSE}_{k+1}^{c_{j'}} \\
& + \cdots + h_{k-2,g}^T\Pi_{k,g,N}(1+\zeta_{k-2,l})\sum_{j'\in\mathcal{I}_{k-2}}\mathrm{EMSE}_{k-2}^{c_{j'}} \\
& + h_{(k-1),g}^T(1+\zeta_{(k-1),l})\sum_{j'\in\mathcal{I}_{(k-1)}}\mathrm{EMSE}_{(k-1)}^{c_{j'}}
\end{aligned}
\quad (2.125)
$$

According to the definition of EMSE_k^g given in (2.74), (2.124) yields

$$\mathrm{EMSE}_k^g = \big[a_{k,g} + h_{k,g}^c\big]^T \big[I_{M_g} - \Pi_{k,g,1}\big]^{-1} \lambda_{k,g} \quad (2.126)$$

where $\overline{\sigma}_{(k-1),g} = \big[I_{M_g} - \Pi_{k,g,1}\big]^{-1} \lambda_{k,g}$. After substituting into (2.125) the solutions of (2.118) obtained for each connected component in $\cup_{j'=1}^{J}\mathcal{G}_{j'}$, the evaluation of (2.126) at each node $k \in \{1, 2, \ldots, N\}$ yields a linear system of N equations whose solution provides closed-form expressions for $\{\mathrm{EMSE}_{k'}^g\}_{k'=1}^N$. Alternatively, closed-form expressions for $\{\mathrm{EMSE}_{k'}^g\}_{k'=1}^N$ can be obtained together with closed-form expressions for $\{\mathrm{EMSE}_k^{c_j}\}_{k=1,j\in\mathcal{I}_k}^N$ by solving the subsequent linear system of equations

$$A^{g,c}\,\mathrm{EMSE}^{g,c} = b^{g,c} \quad (2.127)$$

where $\mathrm{EMSE}^{g,c}$ is the vector resulting from stacking $\{\mathrm{EMSE}_{k'}^g\}_{k'=1}^N$ and $\{\mathrm{EMSE}_k^{c_j}\}_{k=1,j\in\mathcal{I}_k}^N$. Note that the entries of $A^{g,c}$ and $b^{g,c}$ in (2.127) can be easily deduced from (2.114), (2.126) and the order under which $\{\mathrm{EMSE}_{k'}^g\}_{k'=1}^N$ and $\{\mathrm{EMSE}_k^{c_j}\}_{k=1,j\in\mathcal{I}_k}^N$ have been stacked in $\mathrm{EMSE}^{g,c}$.

Similarly to the reasoning used when deriving (2.126), and according to the definition of MSD_k^g given in (2.73), if we choose $\overline{\sigma}_{(k-1),g} = \left[I_{M_g} - \Pi_{k,g,1}\right]^{-1} 1_{M_g}$, equation (2.124) yields

$$\text{MSD}_{k-1}^g = \left[a_{k,g} + h_{k,g}^c\right]^T \left[I_{M_g} - \Pi_{k,g,1}\right]^{-1} 1_{M_g}. \tag{2.128}$$

Therefore, to conclude, from (2.128) note that closed-form expressions for $\{\text{MSD}_{k'}^g\}_{k'=1}^N$ are straightforwardly obtained once closed-form expressions

$$\{\text{EMSE}_{k'}^{c_{j'}}\}_{k'\in\{1,\dots,N\},j'\in\mathcal{I}_{k'}}$$

are available, which occurs after substituting the solutions for $\{\text{EMSE}_{k'}^g\}_{k'=1}^N$ into the solutions of (2.118) for each connected component in $\cup_{j'=1}^J \mathcal{G}_{j'}$.

2.5   Simulation results

In this section, first, some generic simulations verifying the theoretical expressions (2.91) and (2.92) are discussed, and then, the proposed algorithm is tested for a specific application, namely, a spectrum sensing task in a Cognitive Radio Network.

2.5.1   Validation of mean-square theoretical results

First, to verify the theoretical expressions (2.91) and (2.92), we assume a network with $N = 20$ nodes where the measurements follow the observation model (2.38) with $M_g = 8$ and $M_{k_l} = 6$ for all $k = \{1, \dots, N\}$. We have also considered that the background noise $\boldsymbol{v}_k(i)$ has variance $\sigma_{v_k}^2 = \sigma_v^2 = 10^{-3}$ across the network. In addition, the regressors $\mathbf{u}_{k_g,i}$ and $\mathbf{u}_{k_l,i}$ are independently generated according to a time-correlated spatially independent Gaussian distribution. In particular, both $\mathbf{u}_{k_g,i}$ and $\mathbf{u}_{k_l,i}$ follow a stationary first-order autoregressive (AR) model with correlation functions $r_{k_g}(i) = \sigma_{u,k_g}^2 \alpha_{k_g}^{|i|}$ and $r_{k_l}(i) = \sigma_{u,k_l}^2 \alpha_{k_l}^{|i|}$, respectively. In the previous AR models, the parameters $\{\alpha_{k_g}, \alpha_{k_l}\}$ and $\{\sigma_{u,k_g}^2, \sigma_{u,k_l}^2\}$ have been randomly chosen in (0,1) so that the signal-to-noise-ratio at each node is different and varies from -2 dB to 27 dB.

The experimental values in Fig. 2.5 are obtained by running I-NSPE LMS for 10 000 iterations and then averaged over 100 independent experiments to generate the ensemble-average curves. It is clear to see that there is a good match between simulations and theory.
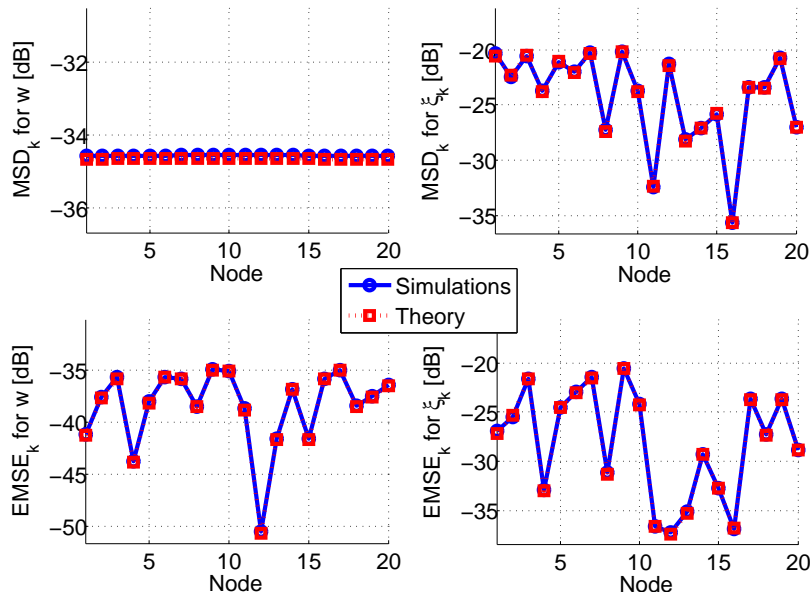
FIGURE 2.5: Steady-state MSD and EMSE per node for the parameters of global and local interest.

In order to confirm the theoretical results obtained in Section 2.4, similar experiments are plotted in Figs. 2.6-2.7 for a scenario with $N = 10$ nodes and two vectors of parameters, ς_1^o and ς_2^o. We consider that the nodes in $\mathcal{C}_1 \in \{2, 3, 4, 5, 6\}$ and $\mathcal{C}_2 \in \{5, 6, 7, 8\}$ are interested in estimating ς_1^o and ς_2^o of dimensions $M_{c_1} = 5$ and $M_{c_1} = 4$, respectively. Therefore, the areas of influence $\mathcal{C}_1$ and $\mathcal{C}_2$ are overlapped. Moreover, we still assume that each node is interested in estimating a vector of global parameters and a vector of local parameters with dimension $M_g = 10$ and $M_l = 6$, respectively. To solve the different estimation tasks, each node of the network performs observations of the environment according to the model (2.1) with $L_k = 1$. As in the previous experiments, we consider that the variance of the background noise is $\sigma_{v_k}^2 = 10^{-3}$ and that the regressors $\mathbf{u}_{k_g,i}$, $\mathbf{u}_{k_l,i}$ and $\mathbf{u}_{k_{jc},i}$, with $j \in \mathcal{I}_k$, are independently generated according to stationary first-order autoregressive (AR) processes. In these AR models, the parameters characterizing the autocorrelation function are again randomly chosen so that the SNR at each nodes varies from 6 to 20 dB and the regressors $\mathbf{u}_{k_g,i}$, $\mathbf{u}_{k_l,i}$ and $\mathbf{u}_{k_{jc},i}$, with $j \in \mathcal{I}_k$, have different temporal correlation. Despite the temporal correlation of the regressors, we can again see a good match between the theoretical expressions obtained from (2.127) and the simulated curves that result from averaging the mean-square measures over 100 independent experiments where I-NSPE LMS is run for 10 000 iterations.
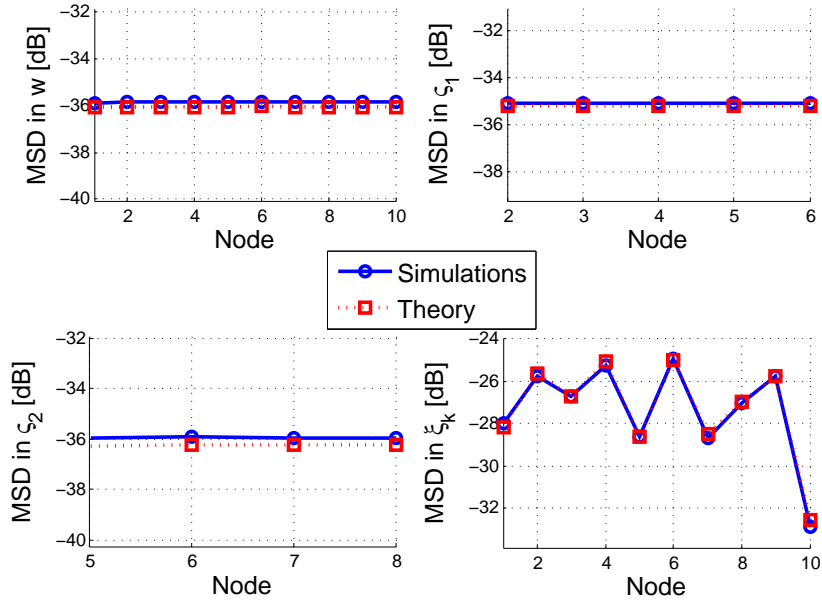
FIGURE 2.6: Steady-state MSD per node.
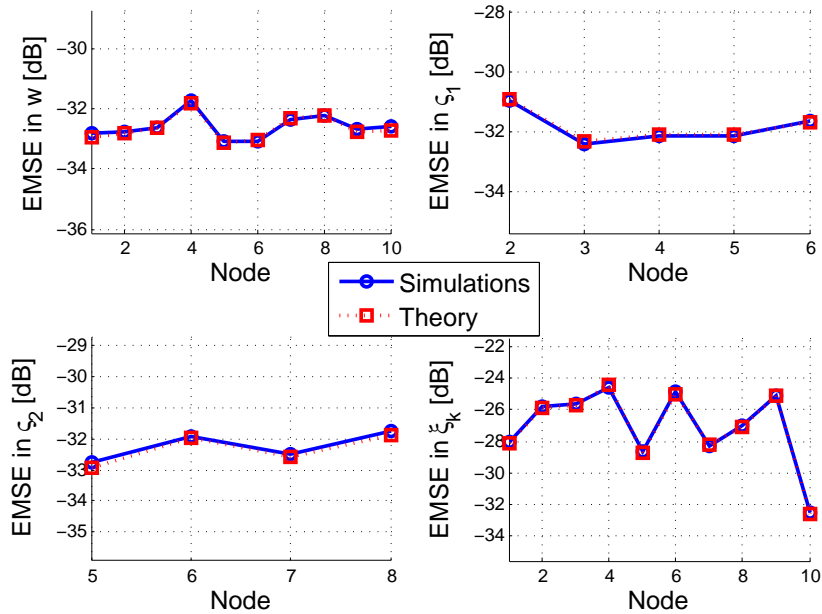
FIGURE 2.7: Steady-state EMSE per node.

2.5.2 Illustrative application

In the following, we will also demonstrate the performance of the proposed algorithm when used for cooperative spectrum sensing in cognitive radio networks (see [57–60], [14, Section 2.4]). In brief, there are Q primary users (PU) transmitting and N secondary users (SU) sensing the power spectrum. In addition to PUs, for each SU we also assume two types of low-power interference sources, i.e., local interferer (LI) and common
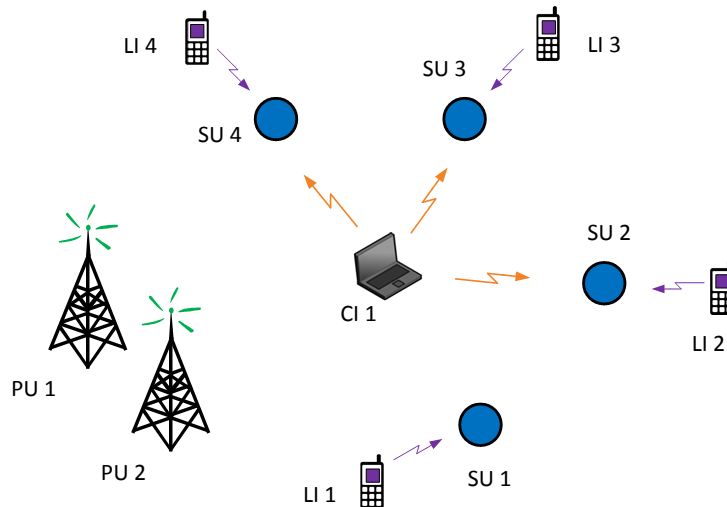
FIGURE 2.8: An illustrative cognitive radio scenario. Each of 4 secondary users (SU) has the scope to estimate the aggregated spectrum transmitted by the primary users and its interferer(s). Apart from PUs, SU 1 is influenced only by its local interferer (LI 1) while SU 2, 3 and 4 are influenced by their local as well as their common interferer (CI 1).

interferers (CI). The former is affecting only one SU, while the latter are influencing several SUs. Therefore, the aim for each SU is to estimate the aggregated spectrum transmitted by all the PUs as well as the spectrum of its own LI and CI. An example of such a scenario is given in Fig. 2.8.

Next, the power spectral density (PSD) of the signal transmitted by the q-th PU, denoted by $\Phi_q^t(f)$, can be approximated by using the subsequent model of X basis functions

$$\Phi_q^t(f) = \sum_{x=1}^{X} b_x(f)\breve{w}_{qx}^o = b_0^T(f)\breve{w}_q^o \qquad (2.129)$$

where $b_0(f) = [b_1(f), \ldots, b_X(f)]^T \in \mathbb{R}^X$ is a vector of basis functions evaluated at frequency f and $\breve{w}_q^o = [\breve{w}_{q1}^o, \ldots, \breve{w}_{qX}^o]^T \in \mathbb{R}^X$ is a vector of weighting coefficients representing the power transmitted by the q-th PU over each basis. For illustration purposes, we use Gaussian pulses as basis functions and if X is chosen large enough, the model in (2.129) can well approximate the transmitted spectrum [57],[59].

Due to the fact that the transmitted signals in cognitive radio are usually wideband, the propagation medium introduces frequency-selective channel gains. Let $p_{tk,i}(f) = |H_{tk}(f,i)|^2$ be the frequency-dependent attenuation coefficient, where $H_{tk}(f,i)$ is the channel frequency response between the t-th transmitter and k-th receiver [60]. For each instant i and frequency f, let us now define the following quantities

- $p_{qk,i}(f)$ denotes the attenuation coefficient between the q-th PU and the k-th SU,

- $p_{Ik,i}(f)$ refers to the attenuation coefficient between the local interferer and the k-th SU,

- $p_{jk,i}(f)$ is the attenuation coefficient between the j-th common interferer and the k-th SU, where $j \in \mathcal{I}_k$.

Then, under the assumption of spatial uncorrelatedness among the channels, the signal received by the k-th SU at time instant i can be expressed as

$$\Phi^r_{k,i}(f) = b^T_{k,i}(f) w^o_k + z_{k,i}, \tag{2.130}$$

where $w^o_k = \mathrm{col}\left\{\breve{w}^o_1, \ldots, \breve{w}^o_Q, \varsigma^o_{\mathcal{I}_k(1)}, \ldots, \varsigma^o_{\mathcal{I}_k(|\mathcal{I}_k|)}, \xi^o_k\right\} \in \mathbb{R}^{(Q+|\mathcal{I}_k|+1)X}$ with ξ^o_k and ς^o_j equal to the vectors of weighting coefficients representing the power transmitted by the LI and j-th CI associated with the k-th SU, respectively. Also, $b_{k,i}(f) = p_{k,i}(f) \otimes b_0(f) \in \mathbb{R}^{(Q+|\mathcal{I}_k|+1)X}$ with $\otimes$ standing for the Kronecker product, and

$$p_{k,i}(f) = [p_{1k,i}, \ldots, p_{qk,i}, p_{\mathcal{I}_k(1),i}, \ldots, p_{\mathcal{I}_k(|\mathcal{I}_k|),i}, p_{Ik,i}]^T, \tag{2.131}$$

while $z_{k,i}$ is the measurement and/or model noise. In the above expression, we dropped the frequency index for compactness of notation. Also note that, in practice, the attenuation factors $p_{tk,i}$ cannot be estimated accurately, so we assume access only to noisy estimates $\hat{p}_{tk,i}$ hereafter.

Considering that, at discrete time i, each node k observes the received PSD in (4.91) over L frequency samples $\{f_m\}^L_{m=1}$, the subsequent vector linear model is obtained

$$\mathbf{d}_{k,i} = \mathbf{U}_{k,i} w^o_k + \mathbf{v}_{k,i} \tag{2.132}$$

where $\mathbf{v}_{k,i}$ denotes noise with zero mean and covariance matrix $R_{v_k,i}$ of dimension $L \times L$ and $\mathbf{U}_{k,i} = [b_{k,i}(f_1) \ldots b_{k,i}(f_L)]^T$ is of dimension $L \times (Q + |\mathcal{I}_k| + 1)X$ with $L > (Q + |\mathcal{I}_k| + 1)X$. Note that the temporal index i in the regressors $\mathbf{U}_{k,i}$ allows to account for possible variations in the channel conditions over time.

For the computer simulations presented here, we compare the I-NSPE scheme with an LMS-based non-cooperative strategy. At this point, it should be emphasized that we do not compare the I-NSPE algorithm with the incremental strategy in [19], or diffusion strategies in [13, 17, 18], [14], designed for a scenario where $w^o_k = w^o$ for all $k \in \{1, 2, \ldots, N\}$. Notice that such a comparison would not be fair since the latter strategies were not designed to estimate parameters of local and common interest, and therefore, they would experience the terms $\sum_{j \in \mathcal{I}_k} \mathbf{U}_{k_{jc},i} \varsigma^o_j$ and $\mathbf{U}_{k_l,i} \xi^o_k$ in (4.3) as additional noise at each node. Furthermore, since the I-NSPE scheme undertakes N updates for the

estimate of w^o, $|\mathcal{C}_j|$ updates for each ς_j^o and one update for the estimate of each ξ_k^o per time step, we assume that

$$\mu_{\xi_k}^{I-NSPE} = \mu_{\xi_k}^{nc} = \mu_w^{nc} = \mu_{\varsigma_j}^{nc}$$

$$\mu_{\xi_k}^{I-NSPE} = |\mathcal{C}_j|\mu_{\varsigma_j}^{I-NSPE} = N\mu_w^{I-NSPE}$$

where μ_a^{I-NSPE} and μ_a^{nc} stands for the step-size used by the I-NSPE and the non-cooperative schemes to estimate the vector a, respectively. This way, we get a fair comparison between both strategies.

For a scenario where there is only one common interferer whose PSD can be sensed by nodes in $\mathcal{C}_1 = \{1,4,5,9,10\}$, Figure 2.9 depicts the learning behavior of the two schemes in terms of the network MSD associated with the estimation of w^o, ς_1^o and ξ_k^o. Each network MSD is the result of averaging the local MSDs associated with the estimation of w^o and ξ_k^o at each node, except for the network MSD associated with the estimation of ς_1^o, which is averaged over the nodes belonging to the set $\mathcal{C}_1$. To generate each plot, we have averaged the results over 100 independent experiments where we assumed $Q = 2$ PUs, $N = 10$ SUs and $X = 16$ Gaussian basis functions, of amplitude normalized to one and standard deviation $\sigma_b = 0.05$, to model the expansion of the transmitted spectrum. Furthermore, we have considered that each SU scans $L = 80$ channels over the normalized frequency axis between 0 and 1, whereas the noise $z_{k,i}$ in (2.130) is zero-mean Gaussian with standard deviation varying between 0.04 and 0.16 for different k.

Each attenuation coefficient follows $\hat{p}_{tk,i}(f) = p_{tk,i}(f) + n_{tk}$, where n_{tk} denotes a zero-mean Gaussian variable with standard deviation in the range between 0.3 and 1.25, while $p_{tk,i}(f)$ is related to the frequency response of the channel modeled as a static 3-tap FIR filter. Each tap is assumed to be a zero-mean complex Gaussian random variable with variance $\sigma_h^2 = 0.25$. Under this setting, due to the cooperation between the nodes, we observe that the proposed scheme outperforms the non-cooperative one, especially when estimating w^o and ς_1^o. Although there is no exchange of estimates of ξ_k^o throughout the network, the I-NSPE scheme has enhanced performance in comparison with the non-cooperative strategy. This is a consequence of the coupling between the three estimation tasks undertaken by I-NSPE.

Finally, to illustrate the asymptotic unbiasedness of the proposed technique, in Fig. 2.10 we plot its mean weight behavior under the previously described setting. The figure indicates the mean weight evolution of some vector coefficients related to the global, common and local parameters at randomly selected nodes, whereas the optimal

values from (2.9) are indicated by the black lines. As expected by Theorem 2.1, I-NSPE LMS has estimated the optimum weight vectors without bias.
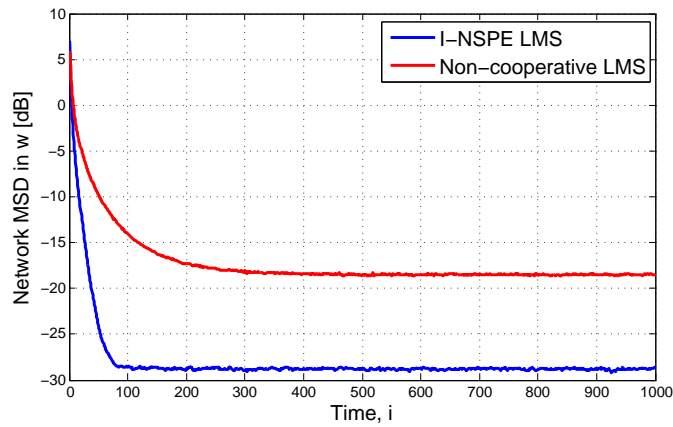
(a)
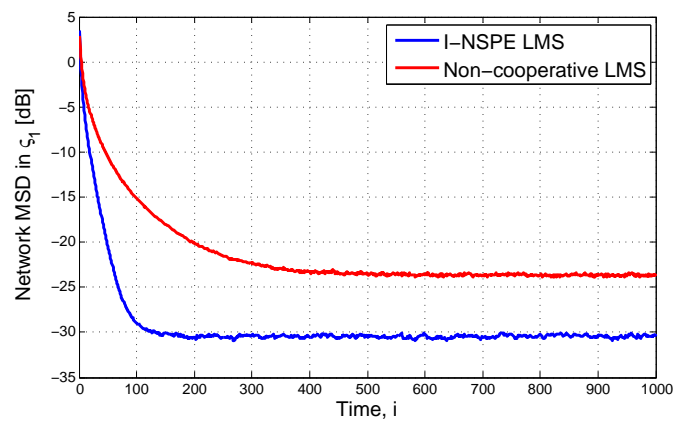
(b)
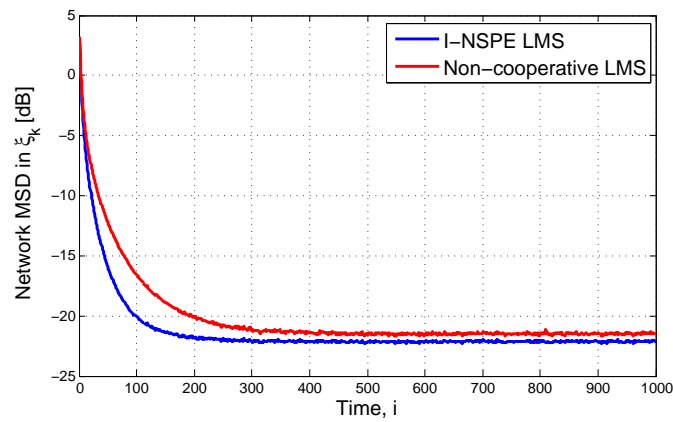
(c)

FIGURE 2.9: Learning behavior of network MSD with respect to the parameters of global interest (a), common interest (b) and for the parameters of local interest (c).
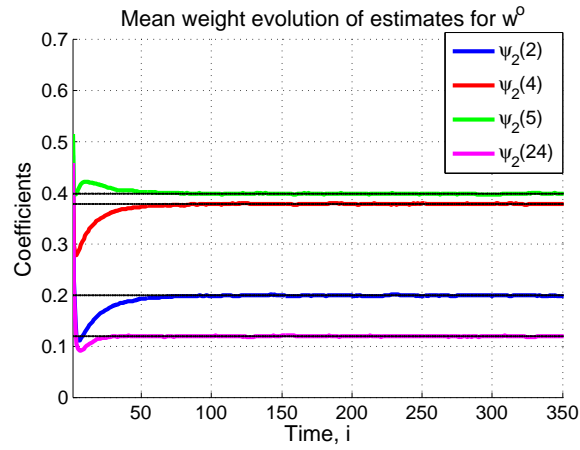
(a)
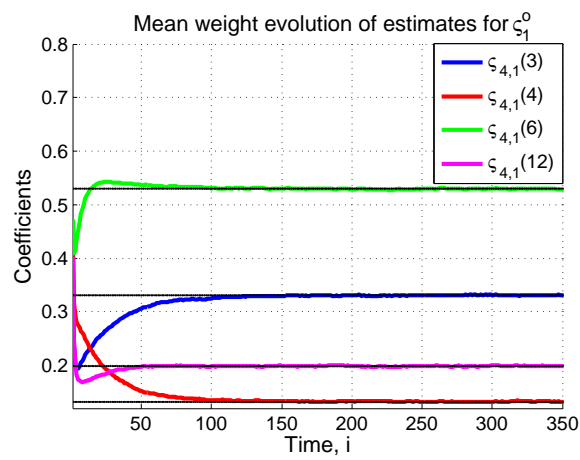
(b)
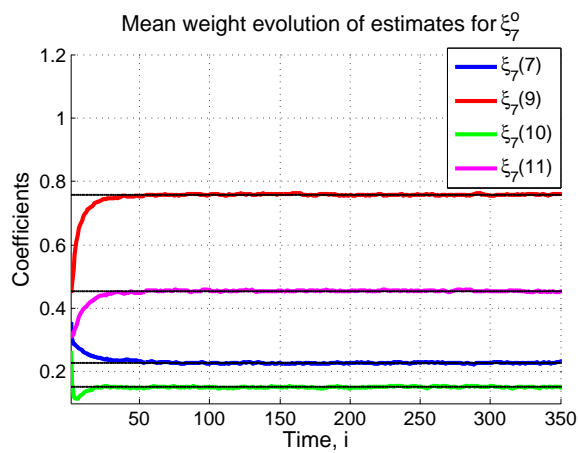
(c)

FIGURE 2.10: The mean trajectories of some vector coefficients related to the global (a), common (b) and local parameters (c) at randomly selected nodes.

2.6   Concluding Remarks

This chapter has addressed a novel NSPE problem where the estimation interests of the nodes consist of a set of local parameters, network-wide global parameters as well as common parameters to a subset of nodes. More specifically, we have proposed a distributed adaptive scheme where a local LMS is run at each node in order to estimate each set of local parameters. Coupled among themselves and with all these local estimation processes, the parameters of global and common interests are estimated by LMS-based schemes implemented under an incremental mode of cooperation.

In the following chapter, under the incremental cooperation strategy, two algorithms based on RLS adaptive rule will be derived for the NSPE problem with global and local interests.

Chapter 3

Incremental NSPE RLS

This chapter explores the use of a recursive least squares learning rule for solving a node specific parameter estimation problem in a distributed way.

It is well-known that, for stationary processes, the stand-alone adaptive filters based on RLS algorithm converge faster than those based on LMS [61],[52]. A main drawback of the RLS algorithm is its computational cost which is an order of magnitude higher than that of LMS. However, available processors continuously decrease in cost and have more powerful batteries as well as computational capabilities. Therefore, to obtain a better estimation performance at the expense of increased computational complexity and energy consumption, RLS-type distributed algorithms have already been derived [15, 62, 63]. However, as explained in Section 1.1, these studies are restricted to the scenarios where there is only a global vector of parameters affecting all nodes.

In this chapter, it is of interest to examine the incremental NSPE approach presented in the previous chapter by adopting the RLS adaptive rule, instead of the LMS one, at each node. To that end, it suffices to consider a scenario with global and local interests, which is a special case of the one assumed in the previous chapter. More specifically, we introduce an adaptive incremental-based technique that attains the exact recursive least squares solution of the aforementioned NSPE problem. The resulting algorithm outperforms the corresponding LMS solution [40] at the expense of increased transmission requirements. Later, since the involved communications cost may be prohibitive in some applications [1], we modify the initial scheme to reduce the communication burden. Finally, the performance of the proposed schemes is illustrated via indicative computer simulations.
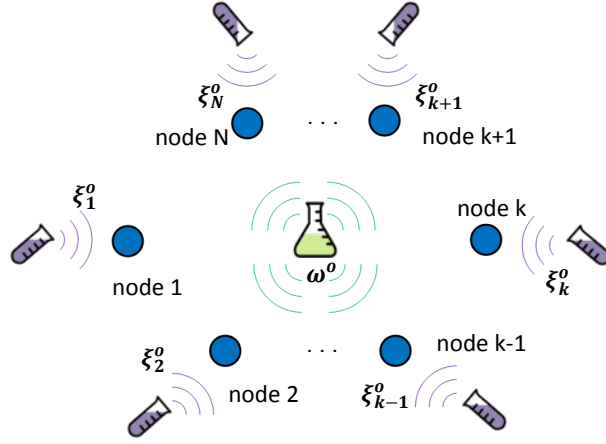
FIGURE 3.1: Network with node-specific parameter estimation interests.

3.1 Problem statement

To begin with, let us consider that a network of N nodes is randomly deployed over some region in order to estimate some unknown vectors of parameters (see Fig. 3.1). At every time instant i, each node k has access to data $\{d_{k,i}, U_{k,i}\}$. These data are assumed to be related to the unknown vectors of parameters by the following model

$$\mathbf{d}_{k,i} = \mathbf{U}_{k,i} w_k^o + \mathbf{v}_{k,i} = \begin{bmatrix} \mathbf{U}_{k_g,i} & \mathbf{U}_{k_l,i} \end{bmatrix} \begin{bmatrix} w^o \\ \xi_k^o \end{bmatrix} + \mathbf{v}_{k,i} \tag{3.1}$$

where, for each time instant i,

- w_k^o equals the vector of dimension M_k that contains all the parameters of interest for node k. This vector is formed by w^o, which is a sub-vector of dimension $M_g \times 1$ consisting of all the parameters of global interest, and by ξ_k^o, which is a sub-vector of dimension $M_{k_l} \times 1$ that gathers all the parameters of local interest,

- $\mathbf{v}_{k,i}$ is measurement and/or model noise with zero mean and covariance matrix $R_{v_k,i}$ of dimensions $L_k \times L_k$,

- $\mathbf{d}_{k,i}$ and $\mathbf{U}_{k,i}$ are zero-mean random variables with dimensions $L_k \times 1$ and $L_k \times M_k$, respectively. Forming the matrix $\mathbf{U}_{k,i}$, the matrices $\mathbf{U}_{k_g,i}$ and $\mathbf{U}_{k_l,i}$, of dimensions $L_k \times M_g$ and $L_k \times M_{k_l}$, might be correlated, and consist of the columns $\mathbf{U}_{k,i}$ associated with w^o and ξ_k^o, respectively.

In most existing works, e.g., [15] and [63], the derived adaptation schemes assume $w_k^o = w^o$, for all $k \in \{1, 2, \ldots, N\}$. As in our previous work [40], we consider a scenario where the node-specific parameters of interest, i.e., $\{w_k^o\}_{k=1}^N$, may, in general, be different.

Hence, the objective for each node k is to estimate its specific unknown vector w_k^o from the data $\{d_{k,i}, U_{k,i}\}$, $k \in \{1, 2, \ldots, N\}$. In particular, as shown in Fig. 3.1, each vector $\{w_k^o\}_{k=1}^N$ consists of globally common components as well as components of local interest for sensor k. The parameters of global interest in the network may account for an event common to all nodes. In contrast, the parameters of local interest for each node k may represent an influence of some local phenomena that is different for each node.

Given the previous observation model, the goal for each node k is to collect the measurements and regressors from time 0 up to time i, i.e.,

$$\mathcal{Y}_{k,i} = \text{col}\{d_{k,0}, d_{k,1}, \ldots, d_{k,i}\}$$

and

$$\mathcal{H}_{k,i} = \text{col}\{U_{k,0}, U_{k,1}, \ldots, U_{k,i}\},$$

respectively, in order to obtain the node-specific estimators $\{w_k\}_{k=1}^N$ that minimize the associated weighted, regularized, least-squares cost

$$\sum_{k=1}^N \left(\lambda^{i+1} \|w_k\|_{\Pi_k}^2 + \|\mathcal{Y}_{k,i} - \mathcal{H}_{k,i} w_k\|_{\mathcal{W}_{k,i}}^2 \right), \tag{3.2}$$

where $\Pi_k = \delta^{-1} I_{M_k}$ and

$$\mathcal{W}_{k,i} = \text{diag}\{\lambda^i \Gamma_k, \lambda^{i-1}\Gamma_k, \ldots, \lambda\Gamma_k, \Gamma_k\}$$

with $\delta > 0$ equal to a large constant, $\Gamma_k = R_{v_k,i}^{-1}$ and forgetting factor $0 \ll \lambda \leq 1$. Thus, after particularizing for global and local vector of parameters, our NSPE problem can be casted as

$$\{\hat{w}, \{\hat{\xi}_k\}_{k=1}^N\} = \underset{w, \{\xi_k\}_{k=1}^N}{\text{argmin}} \left\{ \sum_{k=1}^N \lambda^{i+1} \begin{bmatrix} w^H & \xi_k^H \end{bmatrix} \begin{bmatrix} \Pi_{k,g} & 0 \\ 0 & \Pi_{k,l} \end{bmatrix} \begin{bmatrix} w \\ \xi_k \end{bmatrix} \right.$$
$$\left. + \sum_{k=1}^N \|\mathcal{Y}_{k,i} - \mathcal{H}_{k,i}^g w - \mathcal{H}_{k,i}^l \xi_k\|_{\mathcal{W}_{k,i}}^2 \right\} \tag{3.3}$$

where

$$\mathcal{H}_{k,i}^g = \text{col}\{U_{k_g,0}, U_{k_g,1}, \ldots, U_{k_g,i}\}$$

and

$$\mathcal{H}_{k,i}^l = \text{col}\{U_{k_l,0}, U_{k_l,1}, \ldots, U_{k_l,i}\}.$$

3.2 A Solution of the NSPE problem

In this section, we firstly derive a centralized solution of the optimization problem (3.3), and then we develop a distributed strategy that converges to this centralized solution. For the sake of simplicity and without losing generality, we assume that $M_k = M$, $M_g = M_g$, $M_{k_l} = M_l$ and $L_k = L$ for all $k \in \{1, 2, \ldots, N\}$.

3.2.1 Centralized solution

To solve the considered NSPE problem in (3.3), we have to optimize a scalar real-valued cost function with respect to (w.r.t.) multiple vector variables, i.e., $\{w, \{\xi_k\}_{k=1}^N\}$. After defining the following augmented vector

$$\tilde{w} = \begin{bmatrix} w^T & \xi_1^T & \xi_2^T & \cdots & \xi_N^T \end{bmatrix}^T \quad (\,\widetilde{M} \times 1\,)$$

and gathering all the data up to time i into

$$\mathcal{Y}_i = \mathrm{col}\{y_0, y_1, \ldots, y_i\} \quad (\,N \cdot L \cdot (i+1) \times 1\,)$$

and

$$\widetilde{\mathcal{H}}_i = \mathrm{col}\{\widetilde{H}_0, \widetilde{H}_1, \ldots, \widetilde{H}_i\} \quad (\,N \cdot L \cdot (i+1) \times \widetilde{M}\,) \tag{3.4}$$

where $\widetilde{M} = M_g + N \cdot M_l$. In the definition of $\mathcal{Y}_i$ and $\widetilde{\mathcal{H}}_i$

$$y_i = \mathrm{col}\{d_{1,i}, d_{2,i}, \ldots, d_{N,i}\} \quad (N \cdot L \times 1)$$

$$\widetilde{H}_i = \mathrm{col}\{\widetilde{U}_{1,i}, \widetilde{U}_{2,i}, \ldots, \widetilde{U}_{N,i}\} \quad (N \cdot L \times \widetilde{M})$$

and the augmented regressor is expressed as

$$\widetilde{U}_{k,i} = \begin{bmatrix} U_{k_g,i} & 0_{L \times M_a} & U_{k_l,i} & 0_{L \times M_b} \end{bmatrix} \tag{3.5}$$

with $M_a = (k-1)M_l$ and $M_b = (N-k)M_l$. Now, we can easily verify that our optimization problem is equivalent to

$$\widehat{\tilde{w}} = \underset{\tilde{w}}{\mathrm{argmin}} \left\{ \lambda^{i+1} \|\tilde{w}\|_{\widetilde{\Pi}}^2 + \|\mathcal{Y}_i - \widetilde{\mathcal{H}}_i \tilde{w}\|_{\mathcal{W}_i}^2 \right\} \tag{3.6}$$

where

$$\mathcal{W}_i = \mathrm{diag}\{\lambda^i D, \lambda^{i-1} D, \ldots, \lambda D, D\}$$

and

$$\widetilde{\Pi} = \text{diag}\left\{ \sum_{k=1}^{N} \Pi_{k,g}, \Pi_{1,l}, \Pi_{2,l}, \ldots, \Pi_{N,l} \right\},$$

with

$$D = \text{diag}\{\Gamma_1, \Gamma_2, \ldots, \Gamma_N\}.$$

It is well-known that the solution $\widehat{\widetilde{w}}_i$ is given by [52]:

$$\widehat{\widetilde{w}}_i = \widetilde{P}_i \widetilde{\mathcal{H}}_i^H \mathcal{W}_i \mathcal{Y}_i, \tag{3.7}$$

where

$$\widetilde{P}_i = \left(\lambda^{i+1} \widetilde{\Pi} + \widetilde{\mathcal{H}}_i^H \mathcal{W}_i \widetilde{\mathcal{H}}_i \right)^{-1}. \tag{3.8}$$

However, this centralized batch solution requires the inversion of a square matrix whose dimension is actually proportional to the number of nodes N. In addition, it requires that we store in memory all data available until time i. Hence, a prohibitively high computational and memory-wise cost is needed.

3.2.2 Distributed solution

With the aim of increasing energy efficiency and improving robustness and scalability it is highly desirable to design a distributed and adaptive scheme in order to update $\widehat{\widetilde{w}}_{i-1}$ to $\widehat{\widetilde{w}}_i$. Toward this goal, we firstly develop a distributed recursion for $\widetilde{P}_i$.

By following the approach described in [15], we firstly express the relation (3.8) as follows

$$\begin{aligned}
\widetilde{P}_i^{-1} &= \lambda^{i+1} \widetilde{\Pi} + \widetilde{\mathcal{H}}_i^H \mathcal{W}_i \widetilde{\mathcal{H}}_i \\
&= \lambda \left(\lambda^i \widetilde{\Pi} + \widetilde{\mathcal{H}}_{i-1}^H \mathcal{W}_{i-1} \widetilde{\mathcal{H}}_{i-1} \right) + \widetilde{H}_i^H D \widetilde{H}_i \\
&= \lambda \widetilde{P}_{i-1}^{-1} + \widetilde{H}_i^H D \widetilde{H}_i.
\end{aligned} \tag{3.9}$$

After noting that (3.9) can be rewritten as a sequence of rank-L updates, we can apply the matrix inversion lemma so that the following distributed recursion for $\widetilde{P}_i$ is obtained

$$\begin{cases} \widetilde{P}_{0,i} \leftarrow \lambda^{-1}\widetilde{P}_{N,i-1} \\ \text{for } k = 1 : N \\ \qquad \widetilde{G}_{k,i} = \left(\Gamma_k^{-1} + \widetilde{U}_{k,i}\widetilde{P}_{k-1,i}\widetilde{U}_{k,i}^H\right)^{-1} \\ \qquad \widetilde{P}_{k,i} = \widetilde{P}_{k-1,i} \\ \qquad\qquad + \widetilde{P}_{k-1,i}\widetilde{U}_{k,i}^H\widetilde{G}_{k,i}\widetilde{U}_{k,i}\widetilde{P}_{k-1,i} \\ \text{end} \end{cases} \tag{3.10}$$

where $\widetilde{P}_{k,i}$ denotes the local estimate of $\widetilde{P}_i$ at node k at some time instant i.

Now, let us focus on the distributed update of $\widehat{\widetilde{w}}_{i-1}$ to $\widehat{\widetilde{w}}_i$. Toward this goal, let us define the intermediate global matrices $\mathcal{Y}_i^k$ and $\widetilde{\mathcal{H}}_i^k$ that stack $\mathcal{Y}_{i-1}$ and $\widetilde{\mathcal{H}}_{i-1}$ in addition to the measurements and regressors collected across the network at time i up to node k, respectively,

$$\mathcal{Y}_i^k = \begin{bmatrix} \mathcal{Y}_{i-1} \\ \hline d_{1,i} \\ d_{2,i} \\ \vdots \\ d_{k,i} \end{bmatrix} \quad \text{and} \quad \widetilde{\mathcal{H}}_i^k = \begin{bmatrix} \widetilde{\mathcal{H}}_{i-1} \\ \hline \widetilde{U}_{1,i} \\ \widetilde{U}_{2,i} \\ \vdots \\ \widetilde{U}_{k,i} \end{bmatrix}.$$

Given the previous definitions, we can easily check that the local estimate $\tilde{w}_i$ at node k, i.e., $\tilde{\psi}_k^{(i)}$, at any time instant i is equal to the solution of the following LS problem

$$\tilde{\psi}_k^{(i)} = \underset{\tilde{w}}{\arg\min} \left\{ \lambda^{i+1}\|\tilde{w}\|_{\widetilde{\Pi}}^2 + \|\mathcal{Y}_i^k - \widetilde{\mathcal{H}}_i^k\tilde{w}\|_{\mathcal{W}_i^k}^2 \right\} \tag{3.11}$$

where

$$\mathcal{W}_i^k = \begin{bmatrix} \lambda\mathcal{W}_{i-1} & 0 \\ 0 & D_k \end{bmatrix}$$

and $D_k = \text{diag}\{\Gamma_1, \ldots, \Gamma_k\}$. Next, if we take into account that the solution of (3.11) is equal to

$$\tilde{\psi}_k^{(i)} = \widetilde{P}_{k,i}[\widetilde{\mathcal{H}}_i^k]^H\mathcal{W}_i^k\mathcal{Y}_i^k, \tag{3.12}$$

by using (3.10) and noting that

$$\mathcal{W}_i^k = \begin{bmatrix} \mathcal{W}_i^{k-1} & 0 \\ 0 & \Gamma_k \end{bmatrix}$$

and

$$[\widetilde{\mathcal{H}}_i^k]^H\mathcal{W}_i^k\mathcal{Y}_i^k = [\widetilde{\mathcal{H}}_i^{k-1}]^H\mathcal{W}_i^{k-1}\mathcal{Y}_i^{k-1} + \widetilde{U}_{k,i}^H\Gamma_k d_{k,i},$$
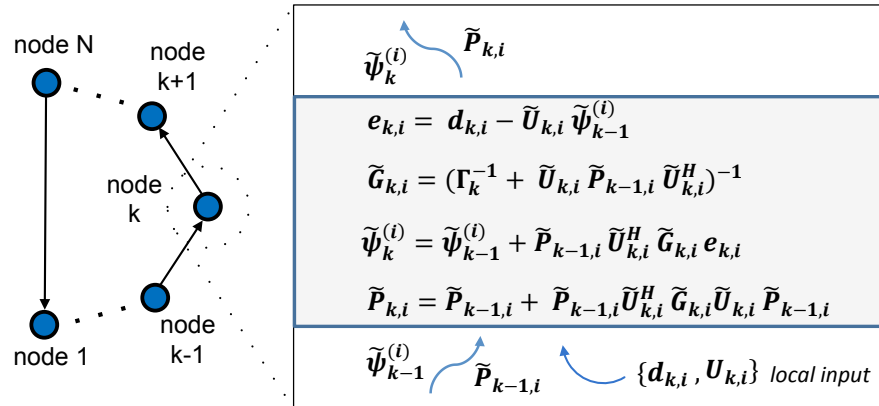
FIGURE 3.2: Data processing in the Incremental-based NSPE RLS.

several algebraic manipulations similar to the ones undertaken in [15] yield the following distribute recursion for $\tilde{\psi}_k^{(i)}$

$$\tilde{\psi}_k^{(i)} = \tilde{\psi}_{k-1}^{(i)} + \widetilde{P}_{k-1,i}\widetilde{U}_{k,i}^H\widetilde{G}_{k,i}\left(d_{k,i} - \widetilde{U}_{k,i}\tilde{\psi}_{k-1}^{(i)}\right). \tag{3.13}$$

Finally, if we group the recursions (3.10) and (3.13), we obtain a distributed incremental-based RLS algorithm that provides the exact solution to the centralized NSPE problem (3.6). The new algorithm is summarized as follows

Incremental-Based NSPE RLS (I-NSPE RLS)

- Initialization: $\psi_N^{(-1)} = 0, \quad \widetilde{P}_{N,-1} = \widetilde{\Pi}^{-1}$.

- At each time $i \geq 0$, for each $k \in \{1, \dots, N\}$ execute

$$
\begin{aligned}
\tilde{\psi}_0^{(i)} &\leftarrow \tilde{\psi}_N^{(i-1)}; \qquad \widetilde{P}_{0,i} \leftarrow \lambda^{-1}\widetilde{P}_{N,i-1} \\
e_{k,i} &= d_{k,i} - \widetilde{U}_{k,i}\tilde{\psi}_{k-1}^{(i)} \\
\widetilde{G}_{k,i} &= \left(\Gamma_k^{-1} + \widetilde{U}_{k,i}\widetilde{P}_{k-1,i}\widetilde{U}_{k,i}^H\right)^{-1} \\
\tilde{\psi}_k^{(i)} &= \tilde{\psi}_{k-1}^{(i)} + \widetilde{P}_{k-1,i}\widetilde{U}_{k,i}^H\widetilde{G}_{k,i}e_{k,i} \\
\widetilde{P}_{k,i} &= \widetilde{P}_{k-1,i} + \widetilde{P}_{k-1,i}\widetilde{U}_{k,i}^H\widetilde{G}_{k,i}\widetilde{U}_{k,i}\widetilde{P}_{k-1,i}
\end{aligned} \tag{3.14}
$$

Figure 3.2 depicts the structure of proposed algorithm, in which estimates $\tilde{\psi}_k^{(i)}$ and matrices $\widetilde{P}_{k,i}$ are shared along the cyclic path.
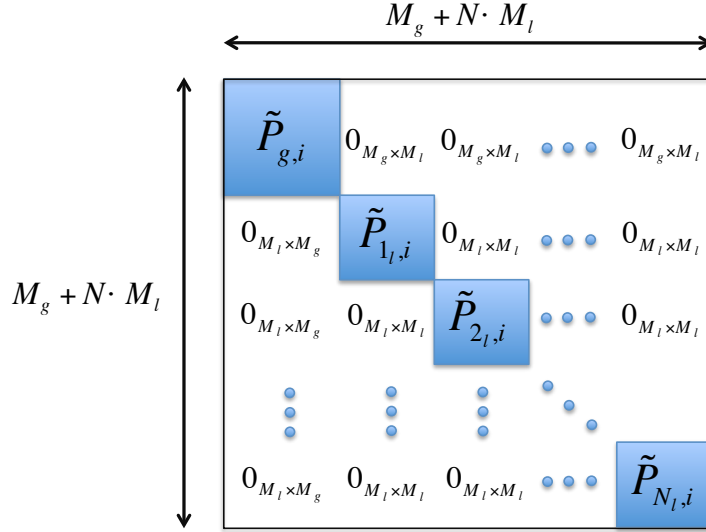
FIGURE 3.3: A block diagonal matrix that approximates $\widetilde{P}_{k,i}$.

3.2.3   Low-communication distributed RLS

Although the previously developed distributed algorithm (3.14) provides the exact RLS solution to the NSPE problem (3.6), it needs $O(\widetilde{M}^2)$ transmission complexity which can be prohibitive for applications with strict energy constraints. Motivated by this fact, at the expense of some performance degradation, we propose a simplified implementation that approximates $\widetilde{P}_{k,i}$ by a block diagonal matrix (see Fig. 3.3) to reduce the number of transmitted parameters between two neighboring nodes.

A careful inspection of (3.5) together with (3.14) reveals that, under a block diagonal approximation for $\widetilde{P}_{k,i}$, only two subvectors of $\tilde{\psi}_k^{(i)}$ and two submatrices of $\widetilde{P}_{k,i}$ are updated at each node and at a some specific time instant i. In particular, according to the recursions (3.10) and (3.13), only the subvectors associated with the local estimates of w^o and ξ_k^o at node k and time instant i, denoted as $\psi_k^{(i)}$ and $\xi_k^{(i)}$, respectively, are updated from $\psi_{k-1}^{(i)}, \xi_k^{(i-1)}$,

$$\widetilde{P}_{k-1,i}(1:M_g) = P_{(k-1)_g,i}$$

and

$$\widetilde{P}_{k,i-1}(M_g + M_a + 1 : M_g + M_l + M_a) = P_{k_l,i-1}$$

where $A(l_a : l_b)$ equals a square submatrix defined by the rows and columns $l_a, l_a + 1, l_a + 2, \ldots, l_b$ with $l_a \leq l_b$. Similarly, at each time instant i each node k updates the submatrices $P_{k_g,i}$ and $P_{k_l,i}$ based on $P_{(k-1)_g,i}$ and $P_{k_l,i-1}$. Therefore, without any loss of optimality w.r.t. the steady state performance of I-NSPE RLS when $\mathbf{U}_{k_g,i}$ and $\mathbf{U}_{k_l,i}$
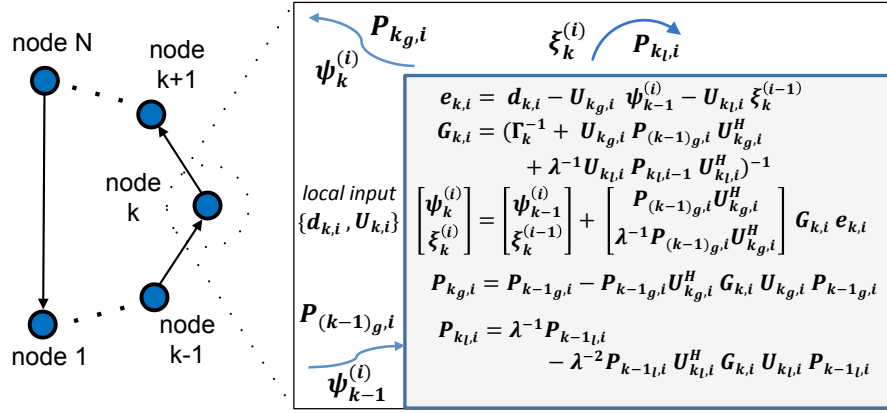
FIGURE 3.4: Data processing in the Incremental-based NSPE low-communication RLS.

are independent, the previous facts allow to properly modify (3.14) in order to obtain the subsequent incremental-based algorithm

$$
\begin{cases}
\psi_0^{(i)} \leftarrow \psi_N^{(i-1)}; \qquad P_{0_g,i} \leftarrow \lambda^{-1} P_{N_g,i-1} \\
\text{for } k = 1 : N \\
e_{k,i} = d_{k,i} - U_{k,i} \begin{bmatrix} \psi_{k-1}^{(i)} \\ \xi_k^{(i-1)} \end{bmatrix} \\
G_{k,i} = \left(\Gamma_k^{-1} + U_{k_g,i} P_{(k-1)_g,i} U_{k_g,i}^H \right. \\
\qquad \left. + \lambda^{-1} U_{k_l,i} P_{k_l,i-1} U_{k_l,i}^H \right)^{-1} \\
\begin{bmatrix} \psi_k^{(i)} \\ \xi_k^{(i)} \end{bmatrix} = \begin{bmatrix} \psi_{k-1}^{(i)} \\ \xi_k^{(i-1)} \end{bmatrix} + \begin{bmatrix} P_{(k-1)_g,i} U_{k_g,i}^H \\ \lambda^{-1} P_{k_l,i-1} U_{k_l,i}^H \end{bmatrix} G_{k,i} e_{k,i} \\
P_{k_g,i} = P_{k-1_g,i} - P_{(k-1)_g,i} U_{k_g,i}^H G_{k,i} U_{k_g,i} P_{(k-1)_g,i} \\
P_{k_l,i} = \lambda^{-1} P_{k_l,i-1} - \lambda^{-2} P_{k_l,i-1} U_{k_l,i}^H G_{k,i} U_{k_l,i} P_{k_l,i-1} \\
\text{end}
\end{cases}
\tag{3.15}
$$

Note that, under the above strategy, each node only needs to transmit the blocks of $\widetilde{P}_{k,i}$ and $\tilde{\psi}_k^{(i)}$ corresponding to the global vector of parameters, i.e., $P_{k_g,i}$ and $\psi_k^{(i)}$ respectively. This is illustrated in Fig 3.4. Since the dimension of the vector of global parameters equals M_g, the scheme in (3.15) requires only $O(M_g^2)$ transmission complexity, which is not dependent on the network size.

3.3 Simulations

We assume a network with $N = 10$ nodes where the measurements follow the observation model (3.1) with $M_{l_k} = 8$, $M_g = 10$ and $L_k = 1$ for all $k = \{1, \ldots, N\}$. We have also considered a forgetting factor $\lambda = 1$ and that the background noise $\mathbf{v_{k,i}}$ has variance
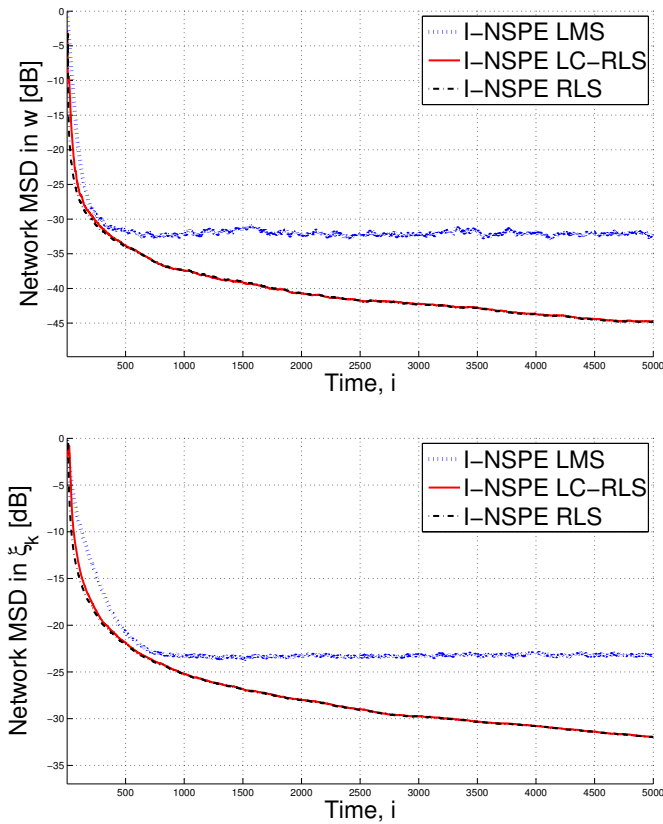
FIGURE 3.5: Learning behavior of network MSD.

$\sigma_{v,k}^2 = \sigma_v^2 = 10^{-3}$ across the network. Additionally, we have assumed that the regressors $\mathbf{u}_{k_g,i}$ and $\mathbf{u}_{k_l,i}$ are independently generated according to a time-correlated spatially independent Gaussian distribution. In particular, both $\mathbf{u}_{k_g,i}$ and $\mathbf{u}_{k_l,i}$ follow a stationary first-order autoregressive (AR) model with correlation functions $r_{k_g}(i) = \sigma_{u,k_g}^2 \alpha_g^{|i|}$ and $r_{k_l}(i) = \sigma_{u,k_l}^2 \alpha_l^{|i|}$, respectively. In the previous AR models, the parameters $\{\alpha_g, \alpha_l\}$ and $\{\sigma_{u,k_g}^2, \sigma_{u,k_l}^2\}$ have been randomly chosen in [0,1) so that the signal-to-noise-ratio at each node is different and varies from 14 dB to 17 dB.

We compare the performance of the I-NSPE RLS strategy provided in (3.14), its low-communication version summarized in (3.15) and denoted as I-NSPE LC-RLS, as well as the I-NSPE LMS algorithm derived in [40]. Specifically, for each one of these I-NSPE algorithms, Figure 3.5 depicts the learning behaviour of the network mean-square deviation (MSD) associated with the estimation of w^o and ξ_k^o. The curves have been generated by averaging 50 independent experiments. We can note that, at the expense of increased computational complexity, both I-NSPE RLS and I-NSPE LC-RLS outperform I-NSPE LMS in terms of steady state floor and rate of convergence. Since the processes of estimating w^o and ξ_k^o are coupled though the observation model, the improved performance appears in both estimation tasks. Furthermore, as it was

expected during the derivations of the proposed schemes, under the considered scenario, I-NSPE LC-RLS achieves identical steady state performance with I-NSPE RLS. In fact, as a result of reducing the transmission complexity of I-NSPE RLS, in the considered setting the I-NSPE LC-RLS scheme suffers only a small performance loss in the rate of convergence.

3.4   Concluding Remarks

This chapter has initially proposed an incremental-type scheme that, in a distributed fashion, implements the exact RLS solution of a central unit processing the data of all the nodes. Next, a scheme with lower transmission complexity for applications where the communications and energy resources are scarce has been derived. Additionally, it was shown that this simplified scheme may converge to the exact RLS solution. Finally, by performing computer simulations we showed the effectiveness of the proposed algorithms. The theoretical analysis concerning the performance of the proposed techniques can be undertaken similarly to the one related the incremental NSPE LMS presented in the previous chapter. A complete analysis is currently under investigation. Also, the distributed RLS schemes which have been proposed in this chapter can be readily extended for operation under the diffusion mode of cooperation. However, in the following chapter, we will focus back on LMS adaptive rules, and develop a distributed algorithm for the general NSPE problem under the diffusion strategy.

Chapter 4

Diffusion NSPE LMS

This chapter studies a distributed adaptive solution to the general NSPE problem, stated in Chapter 2, under the diffusion cooperation strategy.

In the diffusion-based schemes, at each time instant, each node of the network cooperates with a subset of nodes; however, if a network is connected, each node may still experience the effect of the entire network. In other words, at each node, estimates are being exchanged with neighboring nodes, then properly fused and fed into the local adaptive filter [12]. Compared to the incremental strategies [19], this approach achieves better scalability and robustness at the expense of increased energy consumption.

Although there are many published works related to the distributed adaptive algorithms of the diffusion type, only very few papers have recently considered the settings which are not assuming that there is only a global vector of parameters to be estimated. For instance, the authors in [26] assume an NSPE setting, however, the different parameters to be estimated using diffusion strategy are expressed through the same global parameter vector. In [27], the authors use diffusion adaption and scalarization techniques to solve the multi-objective cost function that appears in an NSPE problem and obtain a Pareto-optimal solution. For a network formed by non-overlapping clusters of nodes, a diffusion-based algorithm with spatial regularization that simultaneously provides biased estimates of the multiple vectors of parameters has been derived in [29]. Unlike previous works, the algorithm allows cooperation among neighboring nodes as long as they have numerically similar parameter estimation interests. For an extensive literature review, see Section 1.1.

This chapter derives diffusion-based LMS techniques in order to solve the NSPE problem formulated in Chapter 2. We adopt two peer-to-peer diffusion protocols, Combine-then-Adapt (CTA) and Adapt-then-Combine (ATC). Under the both CTA and ATC
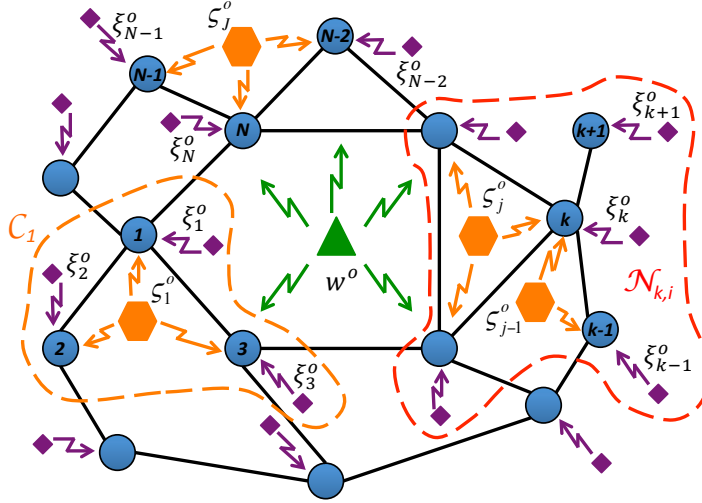
FIGURE 4.1: A network of N nodes with node-specific parameter estimation interests.

schemes, each node undertakes an adaptive filtering task where its local observations are fused with an estimate of its parameters of local interest as well as estimates of the parameters of global and common interest, which have been exchanged with its neighbors. As a result, the network is able to adapt in real time to variations of the data statistics related to parameters of local, common and global interest in the network. Moreover, as a detailed performance analysis of the resulting adaptive network reveals, the proposed NSPE techniques are asymptotically unbiased in the mean sense. Afterwards, a spatial-temporal energy conservation relation is provided to evaluate the steady-state performance at each node in the mean-square sense. In the simulation section, the theoretical results for both the mean and mean-square performance analysis are compared to simulation results. Finally, simulation results are provided in the context of cooperative power spectrum sensing.

4.1   Problem statement

Let us consider a connected network consisting of N nodes that are randomly deployed over some geographical region. Nodes that are able to share information with each other are said to be neighbors. The neighborhood of any particular node k, including also node k, is denoted as $\mathcal{N}_k$. Since the network is connected, as shown in Fig. 4.1, the neighborhoods are set so that there is a path between any pair of the nodes in the network.

At discrete time i, each node k has access to data $\{d_{k,i}, U_{k,i}\}$, corresponding to time realizations of zero-mean random processes $\{\mathbf{d}_{k,i}, \mathbf{U}_{k,i}\}$, with dimensions $L_k \times 1$ and $L_k \times M_k$, respectively. These data are related to events that take place in the monitored

area through the subsequent model

$$\mathbf{d}_{k,i} = \mathbf{U}_{k,i}w_k^o + \mathbf{v}_{k,i} \tag{4.1}$$

where, for each time instant i,

- w_k^o equals the deterministic but unknown vector of dimension M_k that gathers all parameters of interest for node k,

- $\mathbf{v}_{k,i}$ denotes the random noise vector with zero mean and covariance matrix $R_{v_{k,i}}$ of dimensions $L_k \times L_k$,

- $\mathbf{d}_{k,i}$ and $\mathbf{U}_{k,i}$ are zero-mean random variables with dimensions $L_k \times 1$ and $L_k \times M_k$, respectively.

Given the previous observation model, the objective of the network consists in estimating the node-specific vector of parameters $\{w_k\}_{k=1}^N$ that minimize the subsequent cost function

$$J_{\text{glob}}(\{w_k\}_{k=1}^N) = \sum_{k=1}^N E\left\{\|\mathbf{d}_{k,i} - \mathbf{U}_{k,i}w_k\|^2\right\}. \tag{4.2}$$

The vast majority of works dealing with distributed estimation algorithms in the context of adaptive filtering (e.g., [19], [21], [13]), considered the case where the nodes' interests are the same, i.e., $w_k^o = w^o$ for all $k \in \{1, 2, \ldots, N\}$. However, similarly to [40, 41], [43], the formulation of this dissertation goes further by considering that the node-specific interests are different but overlapping.

As depicted in Fig. 4.1, each node-specific vector w_k^o might consist of:

a) a sub-vector w^o of parameters of global interest to the whole network,

b) sub-vectors $\{\varsigma_j^o\}$ of parameters of common interest to subsets of nodes including node k, and

c) a sub-vector ξ_k^o of local parameters for node k.

In particular, the global parameters w^o ($M_g \times 1$) might be related to a phenomenon that can be monitored by all the nodes. In contrast, a set of parameters of common interest ς_j^o ($M_{j_c} \times 1$) might be related to a phenomenon that can be observed by a subset of nodes in the network. The ordered set of indices k associated with the connected subset of nodes interested in ς_j^o is denoted as $\mathcal{C}_j$. For instance, in Fig. 4.1, $\mathcal{C}_1 = \{1, 2, 3\}$. Depending on the areas of influence associated with the events of common interest, a node might be

interested in more than one set of common parameters. As a result, subsets of nodes $\mathcal{C}_j$ and $\mathcal{C}_{j'}$, with $j \neq j'$, might be partially or fully overlapped. For instance, Figure 4.1 indicates that node k is interested in estimating both vectors of common parameters ς_j^o and ς_{j-1}^o, i.e. $\mathcal{C}_{j-1} \cap \mathcal{C}_j = \{k\}$. Finally, each $M_{l_k} \times 1$ vector of local parameters ξ_k^o may represent the influence of some local phenomena that only affects the area monitored by node k. In this way, considering a scenario where there are J different subsets of common parameters (see Fig. 4.1), the observation model provided in (4.1) can be rewritten as

$$\mathbf{d}_{k,i} = \mathbf{U}_{k_g,i}w^o + \sum_{j \in \mathcal{I}_k} \mathbf{U}_{k_{jc},i}\varsigma_j^o + \mathbf{U}_{k_l,i}\xi_k^o + \mathbf{v}_{k,i} \tag{4.3}$$

where, for $k \in \{1, 2, \ldots, N\}$, $j \in \{1, 2, \ldots, J\}$ and $i \geq 1$,

- $\mathcal{I}_k$ equals an ordered set of indices j associated with the vectors ς_j^o that are of interest for sensor k,

- $\mathbf{U}_{k_g,i}$, $\mathbf{U}_{k_{jc},i}$ and $\mathbf{U}_{k_l,i}$ are matrices of dimensions $L_k \times M_g$, $L_k \times M_{j_c}$ and $L_k \times M_{k_l}$ that might be correlated, and consist of the columns of $\mathbf{U}_{k,i}$ associated with w^o, ς_j^o and ξ_k^o, respectively.

Thus, according to (4.2) and (4.3), our NSPE problem can be restated as minimizing the following cost

$$\sum_{k=1}^{N} E \left\{ \|\mathbf{d}_{k,i} - \mathbf{U}_{k_g,i}w - \sum_{j \in \mathcal{I}_k} \mathbf{U}_{k_{jc},i}\varsigma_j - \mathbf{U}_{k_l,i}\xi_k\|^2 \right\} \tag{4.4}$$

with respect to the variables $w, \{\varsigma_j\}_{j=1}^{J}$ and $\{\xi_k\}_{k=1}^{N}$.

4.2 A solution of the new NSPE problem

In this section, acting as a starting point for the derivation of the distributed algorithms and allowing us to introduce some useful notation, we briefly describe the centralized solution provided in [43] to the NSPE problem stated in the previous section. Later, via a diffusion-based approach, we focus on the derivation of distributed algorithms that approximate the centralized solution. For the sake of simplicity and without losing generality, we assume that $M_{l_k} = M_l$, $M_{j_c} = M_c$ and $L_k = L$ for all $k \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, J\}$.

4.2.1 Centralized solution

An inspection of (4.4) reveals that the solution of the considered NSPE problem entails the optimization of a scalar real-valued cost function with respect to the multiple vector variables, i.e., $\{w, \{\varsigma_j\}_{j=1}^{J}, \{\xi_k\}_{k=1}^{N}\}$. If we gather these variables into the following augmented vector

$$\bar{w} = \begin{bmatrix} w^T & \varsigma_1^T & \varsigma_2^T & \cdots & \varsigma_J^T & \xi_1^T & \xi_2^T & \cdots & \xi_N^T \end{bmatrix}^T \quad (\bar{M} \times 1) \tag{4.5}$$

where $\bar{M} = M_g + J \cdot M_c + N \cdot M_l$, from [43] we know that our optimization problem can be cast as

$$\widehat{w} = \underset{\tilde{w}}{\operatorname{argmin}} \{J_{\text{glob}}(\bar{w})\} = \underset{\bar{w}}{\operatorname{argmin}} \left\{ \sum_{k=1}^{N} E\left\{ \|\mathbf{d}_{k,i} - \bar{\mathbf{U}}_{k,i}\bar{w}\|^2 \right\} \right\} \tag{4.6}$$

where

$$\bar{\mathbf{U}}_{k,i} = \begin{bmatrix} \mathbf{U}_{k_g,i} & \mathbb{1}_{\{1 \in \mathcal{I}_k\}}\mathbf{U}_{k_{1c},i} & \mathbb{1}_{\{2 \in \mathcal{I}_k\}}\mathbf{U}_{k_{2c},i} & \cdots & \mathbb{1}_{\{J \in \mathcal{I}_k\}}\mathbf{U}_{k_{Jc},i} & 0_{L \times M_{a_k}} & \mathbf{U}_{k_l,i} & 0_{L \times M_{b_k}} \end{bmatrix} \tag{4.7}$$

with $M_{a_k} = (k-1)M_l$, $M_{b_k} = (N-k)M_l$ and

$$\mathbb{1}_{\{\mathcal{X} \in \mathcal{A}\}} = \begin{cases} 1 & \text{if } \mathcal{X} \subseteq \mathcal{A}, \\ 0 & \text{otherwise.} \end{cases} \tag{4.8}$$

From [52], we know that, if the random processes $\{\mathbf{d}_{k,i}, \mathbf{U}_{k,i}\}$ are jointly wide-sense stationary, the optimal solution is given by the normal equations

$$\left(\sum_{k=1}^{N} R_{\bar{U}_k} \right) \cdot \widehat{w} = \sum_{k=1}^{N} r_{\bar{U}_k d_k}. \tag{4.9}$$

Notice that the solution of the previous system of equations requires the transmission of all sensor observations to the fusion center and the inversion of a square matrix whose dimension is proportional to the network size. As a result, for large networks, the centralized solution in (4.9) is not scalable with respect to both computational power and communication resources, which motivates the derivation of distributed solutions.

4.2.2   Diffusion-based NSPE solutions

By relying on in-network processing of the data $\{d_{k,i}, U_{k,i}\}$, the incremental-based algorithms proposed in [40] and [43] converge to the centralized solution in the mean sense with an increase of the energy efficiency and an improved scalability. Attaining more robustness to link or node failures than the incremental strategies, an alternative mode of cooperation to process the data $\{d_{k,i}, U_{k,i}\}$ in a distributed fashion is based on diffusion strategies, e.g., Combine-then-Adapt (CTA) and Adapt-then-Combine (ATC). In the case where the nodes are interested in estimating the same vector of global parameters, the aforementioned strategies are known to well approximate the centralized solution [13]. In this work, we extend them so as to be applicable to the NSPE problem described in Section 4.1.

First, let us define $\bar{\psi}_k^{(i)}$ as the local estimate of $\bar{w}^o$ at time instant i and node k. Note that $\bar{\psi}_k^{(i)}$ is generally a noisy version of the optimal augmented vector $\bar{w}^o$. By using a diffusion mode of cooperation, at each time instant $i-1$, each node k has access to the set of local estimates of its neighbors, i.e., $\mathcal{N}_k$. Thus, node k can fuse its local estimate with the local estimates of its neighbors as follows

$$\bar{\phi}_k^{(i-1)} = f_k \left(\{\bar{\psi}_\ell^{(i-1)}\}_{\ell \in \mathcal{N}_k} \right) \tag{4.10}$$

where f_k is a local combiner function. In this work, we will focus on linear combiners of the form

$$\bar{\phi}_k^{(i-1)} = \sum_{\ell \in \mathcal{N}_k} \bar{C}_{k,\ell} \, \bar{\psi}_\ell^{(i-1)} \tag{4.11}$$

where

$$\bar{C}_{k,\ell} = \mathrm{diag}\{c_{k,\ell}^w I_{M_g}, c_{k,\ell}^{\varsigma_1} I_{M_c}, \dots, c_{k,\ell}^{\varsigma_J} I_{M_c} c_{k,\ell}^{\xi_1} I_{M_l}, \dots, c_{k,\ell}^{\xi_N} I_{M_l}\}. \tag{4.12}$$

In (4.12), $c_{k,\ell}^w$ equals the weight coefficient used by node k when combining the local estimate of the global vector w^o from node ℓ. Similarly, for $\ell, m \in \{1, \dots, N\}$, $c_{k,\ell}^{\varsigma_j}$ and $c_{k,\ell}^{\xi_m}$ denote the combination coefficients employed by node k when fusing the local estimates of ς_j^o and ξ_m^o, from node ℓ with its corresponding local estimates, respectively. Since the contribution of each node to the different estimation tasks might be different depending on the statistics of its observations as well as its own estimation interests, note that we allow each node to have different coefficients when combining the local estimates of each vector of global, common or local parameters performed by a neighbor node ℓ.

To determine the combination coefficients at each node k, we can interpret (4.11) as a weighted least squares estimate of the augmented vector of parameters $\bar{w}^o$ given its local estimate as well as the local estimates at the neighbor nodes [12]. This way, by collecting the local estimates of the augmented vector $\bar{w}^o$ at the neighbor nodes

$$\bar{\psi}_{\mathcal{N}_k} = \text{col}\left\{\{\bar{\psi}_\ell^{(i-1)}\}_{\ell \in \mathcal{N}_k}\right\} \tag{4.13}$$

and defining

$$B = \text{col}\{I_{\bar{M}}, I_{\bar{M}}, \ldots, I_{\bar{M}}\} \qquad (n_k \cdot \bar{M} \times \bar{M}) \tag{4.14}$$

and

$$\bar{C}_k = \text{diag}\left\{\{\bar{C}_{k,\ell},\}_{\ell \in \mathcal{N}_k}\right\} \tag{4.15}$$

with $n_k = |\mathcal{N}_k|$, we can formulate the subsequent local weighted least-squares problem

$$\underset{\bar{\phi}_k}{\text{argmin}}\left\{\|\bar{\psi}_{\mathcal{N}_k} - B\bar{\phi}_k\|_{\bar{C}_k}^2\right\}, \tag{4.16}$$

whose solution is given by

$$\bar{\phi}_k^{(i-1)} = \left[B^H \bar{C}_k B\right]^{-1} B^H \bar{C}_k \bar{\psi}_{\mathcal{N}_k}. \tag{4.17}$$

In more detail, focusing on the different sub-vectors that form $\bar{\phi}_k^{(i-1)}$, for $k, m \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, J\}$ the solution provided in (4.17) can be rewritten as

$$\phi_{k,w}^{(i-1)} = \sum_{\ell \in \mathcal{N}_k} \frac{c_{k,\ell}^w}{\sum_{k' \in \mathcal{N}_k} c_{k,k'}^w} \psi_{\ell,w}^{(i-1)} \tag{4.18}$$

$$\phi_{k,\varsigma_j}^{(i-1)} = \sum_{\ell \in \mathcal{N}_k} \frac{c_{k,\ell}^{\varsigma_j}}{\sum_{k' \in \mathcal{N}_k} c_{k,k'}^{\varsigma_j}} \psi_{\ell,\varsigma_j}^{(i-1)} \tag{4.19}$$

and

$$\phi_{k,\xi_m}^{(i-1)} = \sum_{\ell \in \mathcal{N}_k} \frac{c_{k,\ell}^{\xi_m}}{\sum_{k' \in \mathcal{N}_k} c_{k,k'}^{\xi_m}} \psi_{\ell,\xi_m}^{(i-1)} \tag{4.20}$$

where $\phi_{k,w}^{(i-1)}$, $\phi_{k,\varsigma_j}^{(i-1)}$ and $\phi_{k,\xi_m}^{(i-1)}$ denote the subvectors of combiner $\bar{\phi}_k^{(i-1)}$ associated with the local estimation of w^o, ς_j^o and ξ_m at node k and time instant $i-1$, respectively. Analogously, $\psi_{k,w}^{(i-1)}$, $\psi_{k,\varsigma_j}^{(i-1)}$ and $\psi_{k,\xi_m}^{(i-1)}$ denote the sub-vectors of the local estimate $\bar{\psi}_k^{(i-1)}$

associated with the local estimation of w^o, ς_j^o and ξ_m at node j and time instant $i-1$, respectively.

At this point, after a suitable redefinition of the combination coefficients that appear in (4.18), (4.19) and (4.20), we can now verify that the combination coefficients in (4.11) and (4.12) must satisfy

$$c_{k,\ell}^w = 0 \text{ if } \ell \notin \mathcal{N}_k; \quad \sum_{\ell \in \mathcal{N}_k} c_{k,\ell}^w = 1 \tag{4.21}$$

$$c_{k,\ell}^{\varsigma_j} = 0 \text{ if } \ell \notin \mathcal{N}_k; \quad \sum_{\ell \in \mathcal{N}_k} c_{k,\ell}^{\varsigma_j} = 1 \tag{4.22}$$

and

$$c_{k,\ell}^{\xi_m} = 0 \text{ if } \ell \notin \mathcal{N}_k; \quad \sum_{\ell \in \mathcal{N}_k} c_{k,\ell}^{\xi_m} = 1 \tag{4.23}$$

for $j \in \{1, 2, \ldots, J\}$ and $k, m \in \{1, 2, \ldots, N\}$.

Next, in order to estimate $\bar{w}^o$ at each node k in an adaptive fashion, the corresponding local aggregate estimate $\bar{\phi}_k^{(i-1)}$ is fed into the local LMS-type adaptive algorithm that minimizes the cost associated with node k in (4.6). This way, the resulting diffusion based strategy can be described as

$$\begin{cases} \text{Combination step:} \\ \bar{\phi}_k^{(i-1)} = \sum_{\ell \in \mathcal{N}_k} \bar{C}_{k,\ell} \, \bar{\psi}_\ell^{(i-1)} \\ \\ \text{Adaptation step:} \\ \bar{\psi}_k^{(i)} = \bar{\phi}_k^{(i-1)} - \mu_k \bar{\mathbf{U}}_{k,i}^H \left[\mathbf{d}_{k,i} - \bar{\mathbf{U}}_{k,i} \, \bar{\phi}_k^{(i-1)} \right] \end{cases} \tag{4.24}$$

with $i \geq 1$, $\{\bar{\psi}_\ell^{(0)}\}_{\ell \in \mathcal{N}_k}$ equal to some initial guess, $\bar{C}_{k,\ell}$ defined in (4.12) and $\mu_k > 0$ equal to a suitably chosen positive step-size parameter.

Due to the structure of the augmented regressors $\bar{\mathbf{U}}_{k,i}$ defined in (4.7), only $2 + |\mathcal{I}_k|$ sub-vectors of $\bar{\psi}_k^{(i)}$ are updated when a specific node k performs the adaptation step at each time instant i (see (4.24)). According to (4.5) and (4.7), based on $\{\mathbf{d}_{k,i}, \mathbf{U}_{k,i}\}$ and the aggregate estimates $\phi_{k,w}^{(i-1)}$, $\{\phi_{k,\varsigma_j}^{(i-1)}\}_{j \in \mathcal{I}_k}$ and $\phi_{k,\xi_k}^{(i-1)}$, the updated sub-vectors correspond to the local estimates of w^o, $\{\varsigma_{k,j}^o\}_{j \in \mathcal{I}_k}$ and ξ_k^o at node k and time i, respectively. Therefore, each node updates only the sub-vectors that are within its interest, which will be now denoted as $\psi_k^{(i)}$, $\{\varsigma_{k,j}^{(i)}\}_{j \in \mathcal{I}_k}$ and $\xi_k^{(i)}$ for the sake of simplicity. The previous

fact allows to set the subsequent equalities in the combination coefficients as,

$$\begin{cases} c_{k,\ell}^{\xi_m} = 0 & \text{if } k \neq \ell \text{ or } k \neq m \\ c_{k,\ell}^{\varsigma_j} = 0 & \text{if } j \notin \mathcal{I}_k \text{ or } j \notin \mathcal{I}_\ell. \end{cases} \tag{4.25}$$

The first set of equalities together with (4.23) show that $c_{k,k}^{\xi_k} = 1$ for each node k. Hence, the vector of local parameters ξ_k^o is only estimated by node k, which is the only node of the network observing ξ_k^o. Continuing the analysis of (4.25), from the second set of equalities we can verify that node k only cooperate to estimate the vectors of common parameters that are within its interests, i.e., $\{\varsigma_j^o\}_{j \in \mathcal{I}_k}$. Then, taking into account (4.22) we can easily show that

$$c_{k,\ell}^{\varsigma_j} = 0 \text{ if } \ell \notin \mathcal{N}_k \cap \mathcal{C}_j; \quad \sum_{\ell \in \mathcal{N}_k \cap \mathcal{C}_j} c_{k,\ell}^{\varsigma_j} = 1. \tag{4.26}$$

As a result, when a node k estimates a specific vector of common parameters that is within its interest, i.e., ς_j^o with $j \in \mathcal{I}_k$, it will only cooperate with the subset of neighbor nodes $\mathcal{N}_k \cap \mathcal{C}_j$, which is composed of the neighbor nodes whose measurements are dependent on ς_j^o. Note that there are several ways by which the combination coefficients can be selected. On one hand, the combination rule can be static, e.g., as in uniform, Metropolis, relative-degree rule [14]. On the other hand, the coefficients can be adapted over time (see, for instance, [28]).

At this point, from (4.24) together with (4.21) and (4.25)-(4.26), we can obtain the Combine-then-Adapt (CTA) diffusion-based LMS algorithm summarized on the following page.

CTA Diffusion-based LMS for NSPE (CTA D-NSPE)

- Start with some initial guesses $\psi_k^{(0)}$, $\{\varsigma_j^{(0)}\}_{j \in \mathcal{I}_k}$ and $\xi_k^{(0)}$ at each node $k \in \{1, 2, \ldots, N\}$.

- For the estimation of w^o and any ς_j^o, choose $N \times N$ combination matrices C^w and C^{ς_j} whose elements in each row k, i.e., $\{c_{k,\ell}^w\}_{\ell=1}^N$ and $\{c_{k,\ell}^{\varsigma_j}\}_{\ell=1}^N$, satisfy (4.21) and (4.26), respectively.

- At each time i, for each $k \in \{1, 2, \ldots, N\}$, execute

  - Combination step:

$$\phi_{k,w}^{(i-1)} = \sum_{\ell \in \mathcal{N}_k} c_{k,\ell}^w \psi_\ell^{(i-1)} \tag{4.27}$$

  and

$$\phi_{k,\varsigma_j}^{(i-1)} = \sum_{\ell \in \mathcal{N}_k \cap \mathcal{C}_j} c_{k,\ell}^{\varsigma_j} \varsigma_{\ell,j}^{(i-1)} \tag{4.28}$$

  for each $j \in \mathcal{I}_k$.

  - Adaptation step:

$$\begin{bmatrix} \psi_k^{(i)} \\ \varsigma_k^{(i)} \\ \xi_k^{(i)} \end{bmatrix} = \begin{bmatrix} \phi_{k,w}^{(i-1)} \\ \phi_{k,\varsigma}^{(i-1)} \\ \xi_k^{(i-1)} \end{bmatrix} + \mu_k U_{k,i}^H \left[d_{k,i} - U_{k,i} \begin{bmatrix} \phi_{k,w}^{(i-1)} \\ \phi_{k,\varsigma}^{(i-1)} \\ \xi_k^{(i-1)} \end{bmatrix} \right] \tag{4.29}$$

  with $\varsigma_k^{(i)} = \text{col}\left\{ \{\varsigma_{k,j}^{(i)}\}_{j \in \mathcal{I}_k} \right\}$ and $\phi_{k,\varsigma}^{(i)} = \text{col}\left\{ \{\phi_{k,\varsigma_j}^{(i)}\}_{j \in \mathcal{I}_k} \right\}$.

Now, let us consider that each node k firstly performs the adaptation step and afterwards, it solves its local weighted least squares problem given in (4.16). Then, by following a derivation that is analogous to the one undertaken for the CTA D-NSPE scheme and that has been omitted for the sake of brevity, we can obtain the Adapt-then-Combine (ATC) diffusion-based LMS algorithm. Basically, as it is summarized in the table on the next page, the new NSPE algorithm consists in reversing the order under which the adaptation and combination steps are performed for each node k according to the CTA D-NSPE strategy.

ATC Diffusion-based LMS for NSPE (ATC D-NSPE)

- Start with some initial guesses $\phi_{k,w}^{(0)}$, $\{\phi_{k,\varsigma_j}^{(0)}\}_{j\in\mathcal{I}_k}$ and $\xi_k^{(0)}$ at each node $k \in \{1, 2, \ldots, N\}$.

- For the estimation of w^o and any ς_j^o, choose $N \times N$ combination matrices C^w and C^{ς_j} whose elements in each row k, i.e., $\{c_{k,\ell}^w\}_{\ell=1}^N$ and $\{c_{k,\ell}^{\varsigma_j}\}_{\ell=1}^N$, satisfy (4.21) and (4.26), respectively.

- At each time i, for each $k \in \{1, 2, \ldots, N\}$, execute

  - Adaptation step:

$$
\begin{bmatrix} \psi_k^{(i)} \\ \varsigma_k^{(i)} \\ \xi_k^{(i)} \end{bmatrix} = \begin{bmatrix} \phi_{k,w}^{(i-1)} \\ \phi_{k,\varsigma}^{(i-1)} \\ \xi_k^{(i-1)} \end{bmatrix} + \mu_k\, U_{k,i}^H \left[d_{k,i} - U_{k,i} \begin{bmatrix} \phi_{k,w}^{(i-1)} \\ \phi_{k,\varsigma}^{(i-1)} \\ \xi_k^{(i-1)} \end{bmatrix} \right]
\tag{4.30}
$$

  with $\varsigma_k^{(i)} = \mathrm{col}\Big\{ \{\varsigma_{k,j}^{(i)}\}_{j\in\mathcal{I}_k} \Big\}$ and $\phi_{k,\varsigma}^{(i)} = \mathrm{col}\Big\{ \{\phi_{k,\varsigma_j}^{(i)}\}_{j\in\mathcal{I}_k} \Big\}$.

  - Combination step:

$$
\phi_{k,w}^{(i)} = \sum_{\ell\in\mathcal{N}_k} c_{k,\ell}^w\, \psi_\ell^{(i)}
\tag{4.31}
$$

  and

$$
\phi_{k,\varsigma_j}^{(i)} = \sum_{\ell\in\mathcal{N}_k\cap\mathcal{C}_j} c_{k,\ell}^{\varsigma_j}\, \varsigma_{\ell,j}^{(i)}.
\tag{4.32}
$$

  for each $j \in \mathcal{I}_k$.

Although the algorithms have been designed for the case where the parameters of local, common and global interest coexist, note that the derived algorithms can be simplified straightforwardly to any other NSPE setting. For instance, the derived algorithms can be easily simplified to a setting where there are no parameters of global interest or where some of the nodes do not have parameters of local interest. Nevertheless, independently of the considered NSPE setting, we can check that both diffusion-based NSPE algorithms are scalable in terms of computational burden and energy resources. On the one hand, regarding the computational complexity, at each time instant, each node k only needs to update a maximum of $1 + 2\,(1 + |\mathcal{I}_k|)$ vectors whose dimensions are

independent of the number of nodes. On the other hand, at each time instant i, in both algorithms, each node k is required to transmit a maximum of $1 + |\mathcal{I}_k|$ vectors, whose dimensions are again independent of the number of nodes.

4.3   Performance analysis

This section is concerned with the performance analysis of CTA D-NSPE and ATC D-NSPE algorithms proposed in Section 4.2. We start by considering a general recursion that includes both algorithms and that captures the behavior of individual nodes across the network. We then study the convergence in the mean of the general model. Finally, we characterize its mean-square performance in the steady-state in terms of Mean-Square Deviation (MSD) and Excess Mean-Square Error (EMSE).

4.3.1   Network-wide recursion

In this subsection, we derive a general algorithmic form that includes CTA D-NSPE and ATC D-NSPE as special cases. In particular, let us write the first combination step as

$$\phi_{k,w}^{(i-1)} = \sum_{\ell \in \mathcal{N}_k} c_{k,\ell}^w \, q_{\ell,w}^{(i-1)} \tag{4.33}$$

and

$$\phi_{k,\varsigma_j}^{(i-1)} = \sum_{\ell \in \mathcal{N}_k} c_{k,\ell}^{\varsigma_j} \, q_{\ell,\varsigma_j}^{(i-1)} \tag{4.34}$$

for each j belonging to $\mathcal{I}_k$. Moreover, the adaptation step is expressed by the following equation

$$\begin{bmatrix} \psi_k^{(i)} \\ \varsigma_k^{(i)} \\ \xi_k^{(i)} \end{bmatrix} = \begin{bmatrix} \phi_{k,w}^{(i-1)} \\ \phi_{k,\varsigma}^{(i-1)} \\ \xi_k^{(i-1)} \end{bmatrix} + \mu_k \, U_{k,i}^H \left[d_{k,i} - U_{k,i} \begin{bmatrix} \phi_{k,w}^{(i-1)} \\ \phi_{k,\varsigma}^{(i-1)} \\ \xi_k^{(i-1)} \end{bmatrix} \right] \tag{4.35}$$

where, with a slight abuse of notation, $\varsigma_k^{(i)} = \mathrm{col}\left\{ \{\varsigma_{k,j}^{(i)}\}_{j \in \mathcal{I}_k} \right\}$ and $\phi_{k,\varsigma}^{(i-1)} = \mathrm{col}\left\{ \{\phi_{k,\varsigma_j}^{(i-1)}\}_{j \in \mathcal{I}_k} \right\}$. Then, at each iteration of the general algorithmic form the second combination step takes place, i.e.,

$$q_{k,w}^{(i)} = \sum_{\ell \in \mathcal{N}_k} a_{k,\ell}^w \, \psi_\ell^{(i)} \tag{4.36}$$

and

$$q_{k,\varsigma_j}^{(i)} = \sum_{\ell \in \mathcal{N}_k} a_{k,\ell}^{\varsigma_j} \varsigma_{\ell,j}^{(i)} \tag{4.37}$$

for each j belonging to $\mathcal{I}_k$. In (4.33), the non-negative real coefficient $c_{k,\ell}^w$ corresponds to the (k,ℓ)-th entries of the $(N \times N)$ combination matrix C^w, which satisfies $C^w 1_N = 1_N$. Moreover, in (4.34), the non-negative real coefficient $c_{k,\ell}^{\varsigma_j}$ corresponds to the $(|\mathcal{C}_{j,k}|, |\mathcal{C}_{j,\ell}|)$ entry of a $(|\mathcal{C}_j| \times |\mathcal{C}_j|)$ combination matrix C^{ς_j}, which satisfies

$$C^{\varsigma_j} 1_{|\mathcal{C}_j|} = 1_{|\mathcal{C}_j|}$$

with

$$\mathcal{C}_{j,k} = \{k' \in \mathcal{C}_j : k' \le k\}. \tag{4.38}$$

and $k, \ell \in \mathcal{C}_j$ for any $j \in \{1, 2, \ldots, J\}$. Similarly, in (4.36)-(4.37) the non-negative real coefficients $\{a_{k,\ell}^w\}$ and $\{a_{k,\ell}^{\varsigma_j}\}$ correspond to the (k,ℓ)-th and the $(|\mathcal{C}_{j,k}|, |\mathcal{C}_{j,\ell}|)$-th entries of the $(N \times N)$ and $(|\mathcal{C}_j| \times |\mathcal{C}_j|)$ combination matrices A^w and A^{ς_j}, respectively, which satisfy

$$A^w 1_N = 1_N, \qquad A^{\varsigma_j} 1_{|\mathcal{C}_j|} = 1_{|\mathcal{C}_j|}$$

for any $j \in \{1, 2, \ldots, J\}$.

Also, note that if we set $C^w = I_N$, $C^{\varsigma_j} = I_{|\mathcal{C}_j|}$ for $j \in \{1, 2, \ldots, J\}$, equations (4.33)-(4.37) represent the ATC D-NSPE algorithm. On the other hand, its CTA counterpart corresponds to selecting $A^w = I_N$, $A^{\varsigma_j} = I_{|\mathcal{C}_j|}$ for $j \in \{1, 2, \ldots, J\}$.

Now, let us interpret data as random variables. Associated with the quantities in the general form in (4.33)-(4.37), we define the weight-error vectors, for $k = \{1, \ldots, N\}$ and $j = \{1, \ldots, J\}$, as follows

$$\begin{aligned}
\tilde{\phi}_{k,w}^{(i)} &= w^o - \phi_{k,w}^{(i)}, \ \tilde{p}_{k,w}^{(i)} = w^o - \psi_k^{(i)}, \ \tilde{q}_{k,w}^{(i)} = w^o - q_{k,w}^{(i)} \\
\tilde{\phi}_{k,\varsigma_j}^{(i)} &= \varsigma_j^o - \phi_{k,\varsigma_j}^{(i)}, \ \tilde{p}_{k,\varsigma_j}^{(i)} = \varsigma_j^o - s_{k,j}^{(i)}, \ \tilde{q}_{k,\varsigma_j}^{(i)} = \varsigma_j^o - q_{k,\varsigma_j}^{(i)} \\
\tilde{\phi}_{k,\xi_k}^{(i)} &= \xi_k^o - \xi_k^{(i)}, \ \tilde{p}_{k,\xi_k}^{(i)} = \xi_k^o - \xi_k^{(i)}, \ \tilde{q}_{k,\xi_k}^{(i)} = \xi_k^o - \xi_k^{(i)}.
\end{aligned} \tag{4.39}$$

Next, we collect these quantities across all agents into the corresponding $(\sum_{k=1}^N M_k) \times 1$ block vectors, i.e., network weight-error vectors,

$$\tilde{\phi}_i = \mathrm{col}\left\{ \left\{ \tilde{\phi}_{k,w}^{(i)}, \{\tilde{\phi}_{k,\varsigma_j}^{(i)}\}_{j \in \mathcal{I}_k}, \tilde{\phi}_{k,\xi_k}^{(i)}, \right\}_{k=1}^N \right\} \tag{4.40}$$

In the same vein, the network vectors $\tilde{\boldsymbol{p}}_i$ and $\tilde{\boldsymbol{q}}_i$ are formed, by stacking the corresponding weight-error vectors. For notational convenience, hereafter we use

$$\breve{M} = \sum_{k=1}^{N} M_k \,.$$

To proceed, let us introduce the diagonal matrix

$$\mathcal{M} = \mathrm{diag}\{\mu_1 I_{M_1}, \dots, \mu_N I_{M_N}\} \qquad (\breve{M} \times \breve{M}), \tag{4.41}$$

the block-diagonal matrix

$$\boldsymbol{\mathcal{D}}_i = \mathrm{diag}\{\mathbf{U}_{1,i}^H \mathbf{U}_{1,i}, \dots, \mathbf{U}_{N,i}^H \mathbf{U}_{N,i}\} \qquad (\breve{M} \times \breve{M}), \tag{4.42}$$

and the vector

$$\boldsymbol{\mathcal{V}}_i = \mathrm{col}\{\mathbf{U}_{1,i}^H \mathbf{v}_{1,i}, \dots, \mathbf{U}_{N,i}^H \mathbf{v}_{N,i}\} \qquad (\breve{M} \times 1). \tag{4.43}$$

Finally, the network-wide behavior can be characterized by these relations for the block quantities:

$$\tilde{\boldsymbol{\phi}}_{i-1} = \breve{\mathcal{C}} \, \tilde{\boldsymbol{q}}_{i-1} \tag{4.44}$$

$$\tilde{\boldsymbol{p}}_i = (I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)\tilde{\boldsymbol{\phi}}_{i-1} - \mathcal{M}\boldsymbol{\mathcal{V}}_i \tag{4.45}$$

$$\tilde{\boldsymbol{q}}_i = \breve{\mathcal{A}} \, \tilde{\boldsymbol{p}}_i \tag{4.46}$$

where the structure of the extended weighting matrices $\breve{\mathcal{C}}$ and $\breve{\mathcal{A}}$ is explained in the following subsection.

To sum up, note that (4.44)-(4.46) can be expressed in the following equivalent form

$$\tilde{\boldsymbol{q}}_i = \breve{\mathcal{A}} \, (I - \mathcal{M}\boldsymbol{\mathcal{D}}_i) \, \breve{\mathcal{C}} \, \tilde{\boldsymbol{q}}_{i-1} - \breve{\mathcal{A}} \, \mathcal{M}\boldsymbol{\mathcal{V}}_i, \tag{4.47}$$

which will be used in the Subsections 4.3.4 and 4.3.5 to study the mean stability and to perform the mean-square steady-state analysis, respectively.

4.3.2 Structure of the extended weighting matrices

The extended weighting matrices $\breve{\mathcal{A}}$ and $\breve{\mathcal{C}}$ have the same form, only the weights can be different. Therefore, in order to define them, let us consider, for instance, the $\breve{\mathcal{A}}$ matrix,

$$\breve{\mathcal{A}} = \text{col} \left\{ A_1^w, \{A_1^{\varsigma_j}\}_{j \in \mathcal{I}_1}, A_1^{\xi_1}, \ldots, A_N^w, \{A_N^{\varsigma_j}\}_{j \in \mathcal{I}_N}, A_N^{\xi_N} \right\}, \tag{4.48}$$

where the blocks being stacked are defined as

$$A_k^w = \begin{bmatrix} a_{k,1}^w I_{M_g} & 0_{M_g \times (M_c|\mathcal{I}_1|+M_l)} & a_{k,2}^w I_{M_g} & 0_{M_g \times (M_c|\mathcal{I}_2|+M_l)} & \cdots & a_{k,N}^w I_{M_g} & 0_{M_g \times (M_c|\mathcal{I}_N|+M_l)} \end{bmatrix} \tag{4.49}$$

$$A_k^{\xi_k} = \begin{bmatrix} 0_{M_l \times (\sum_{\ell=1}^{k-1} M_\ell + M_g + M_c|\mathcal{I}_k|)} & I_{M_l} & 0_{M_l \times (\sum_{\ell=k+1}^{N} M_\ell)} \end{bmatrix} \tag{4.50}$$

and

$$A_k^{\varsigma_j} = \begin{bmatrix} A_{k1}^{\varsigma_j} & A_{k2}^{\varsigma_j} & \cdots & A_{kN}^{\varsigma_j} \end{bmatrix} \tag{4.51}$$

with

$$A_{k\ell}^{\varsigma_j} = \begin{cases} \begin{bmatrix} 0_{M_c \times (M_g + M_c|\mathcal{I}_{j\ell}|)} & a_{k,\ell}^{\varsigma_j} I_{M_c} & 0_{M_c \times (M_c[|\mathcal{I}_\ell|-|\mathcal{I}_{j\ell}|-1]+M_l)} \end{bmatrix} & \text{if } \ell \in \mathcal{C}_j, \\ 0_{M_c \times M_\ell} & \text{if } \ell \notin \mathcal{C}_j \end{cases} \tag{4.52}$$

and $\mathcal{I}_{jk} = \{j' \in \mathcal{I}_k : j' < j\}$.

An alternative way to define $\breve{\mathcal{A}}$ is the following relation

$$\breve{\mathcal{A}} = \mathcal{P} \, \breve{\mathcal{A}}^{\text{blkd}} \, \mathcal{P}^T \tag{4.53}$$

where the block-diagonal matrix $\breve{\mathcal{A}}^{\text{blkd}}$ is given by

$$\breve{\mathcal{A}}^{\text{blkd}} = \text{blockdiag} \left\{ A^w \otimes I_{M_g}, \{A^{\varsigma_j} \otimes I_{M_c}\}_{j=1}^{J}, I_N \otimes I_{M_l} \right\}, \tag{4.54}$$

while $\otimes$ stands for the Kronecker product, and $\mathcal{P}$ is the $\breve{M} \times \breve{M}$ permutation matrix that stacks appropriately chosen $1 \times \breve{M}$ row basis vectors. In particular, a basis vector e_k has the unity at the kth position and zeros elsewhere. The structure of the permutation matrix $\mathcal{P}$ in (4.53) is specified by N blocks, each corresponding to a specific node, i.e., $\mathcal{P} = \text{col}\{E_1, \ldots, E_N\}$, where the kth block E_k, of dimensions $M_k \times \breve{M}$, takes the following

form

$$
\begin{aligned}
E_k = \mathrm{col}\{ & e_{f_g(k,1)}, \ldots, e_{f_g(k,M_g)}, e_{f_c(k,\mathcal{I}_k(1),1)}, \\
& \ldots, e_{f_c(k,\mathcal{I}_k(1),M_c)}, \ldots, e_{f_c(k,\mathcal{I}_k(1),M_c)}, \\
& \ldots, e_{f_c(k,\mathcal{I}_k(|\mathcal{I}_k|),1)}, \ldots, e_{f_c(k,\mathcal{I}_k(|\mathcal{I}_k|),M_c)}, \\
& e_{f_l(k,1)}, \ldots, e_{f_l(k,M_l)} \}
\end{aligned}
\tag{4.55}
$$

with the three counter functions, specifying the position of the unity in the basis vectors $e_{(.)}$, defined by

$$
f_g(k,c) = (k-1) \cdot M_g + c \,,
\tag{4.56}
$$

$$
f_c(k,j,c) = N \cdot M_g + \sum_{j'=1}^{j-1} |\mathcal{C}_{j'}| \cdot M_c + (|\mathcal{C}_{j,k}| - 1) \cdot M_c + c
\tag{4.57}
$$

and

$$
f_l(k,c) = N \cdot M_g + \sum_{j=1}^{J} |\mathcal{C}_j| \cdot M_c + (k-1) \cdot M_l + c
\tag{4.58}
$$

with $\mathcal{C}_{j,k}$ given in (4.38).

4.3.3 Data assumptions

To proceed, we state the following independence assumptions on the data:

A1) $\mathbf{v}_{k,i}$ is temporally and spatially white noise whose covariance matrix is $R_{v_k,i}$ and which is independent of $\mathbf{U}_{k',i'}$ for all k' and i', with $k,k' \in \{1,2,\ldots,N\}$ and $i,i' > 0$;

A2) $\mathbf{U}_{k,i}$ is independent of $\mathbf{U}_{k,i'}$, with $i,i' > 0$ and $i \neq i'$ (temporal independence).

A3) $\mathbf{U}_{k,i}$ is independent of $\mathbf{U}_{k',i}$, with $k,k' \in \{1,2,\ldots,N\}$ and $k \neq k'$ (spatial independence),

A4) $\mathbf{U}_{k_g,i}$, $\mathbf{U}_{k_{jc},i}$ and $\mathbf{U}_{k_l,i}$ are independent for all $k \in \{1,2,\ldots,N\}$ and $j \in \{1,2,\ldots,J\}$;

In order to evaluate the fourth-order moment of the matrix-valued regression data in Subsection 4.3.5, we further assume:

A5) $\mathbf{U}_{k,i}$ ($L_k \times M_k$) has a real matrix variate normal distribution specified by mean $0_{L_k \times M_k}$ and positive-semidefinite matrices Ψ_k ($M_k \times M_k$) and Ω_k ($L_k \times L_k$) (see [64,

Chapter 2]). Equivalently, using standard notation for multivariate normal distribution, the distribution of $\mathbf{U}_{k,i}$ can be defined as $\text{vec}(\mathbf{U}_{k,i}) \sim \mathcal{N}_{M_k L_k}\big(\text{vec}(0_{L_k \times M_k}),$ $\Psi_k \otimes \Omega_k\big)$ where $\otimes$ stands for the Kronecker product.

Remark 4.1: Note that even for the vector-valued regression data, in order to evaluate the fourth-order moment, the Gaussian assumption is required (e.g. see [12] and [13]). The results of the fourth-order moment of the matrix-valued regression data appear to be more challenging than those on its vector counterpart, due to the extra dimension involved. Therefore, the assumption A5 seems well-justified.

4.3.4 Mean stability

By taking the expectation of (4.47) and using Assumptions A1-A3, we obtain

$$E\tilde{\boldsymbol{q}}_i = \check{\mathcal{A}}\,(I - \mathcal{M}\mathcal{R}_U)\,\check{\mathcal{C}}\,E\tilde{\boldsymbol{q}}_{i-1}, \tag{4.59}$$

where

$$\mathcal{R}_U = E\boldsymbol{\mathcal{D}}_i = \text{blockdiag}\{R_{U_1}, R_{U_2}, \ldots, R_{U_N}\}, \tag{4.60}$$

and

$$R_{U_k} = E\,\mathbf{U}_{k,i}^H \mathbf{U}_{k,i} = \text{blockdiag}\left\{R_{U_{k_g}}, \{R_{U_{k_{jc}}}\}_{j \in \mathcal{I}_k}, R_{U_{k_l}}\right\}. \tag{4.61}$$

The algorithm in (4.47) is asymptotically unbiased, i.e, $E\tilde{\boldsymbol{q}}_i \to 0_{\check{M} \times 1}$ as $i \to \infty$, if the matrix $\check{\mathcal{A}}(I - \mathcal{M}\mathcal{R}_U)\check{\mathcal{C}}$ is stable. In order to prove its stability, we will build upon the approaches taken in [14] and [65], by selecting a convenient matrix norm $\|\cdot\|$ and exploit its submultiplicativity property, i.e., $\|AB\| \le \|A\|\|B\|$.

Here, we use the induced block maximum matrix norm [65], [14], however, defined over a block matrix with different block sizes. In particular, let x be a $\check{M} \times 1$ vector consisting of $\check{N}$ blocks, where $\check{N} = N + \sum_{k=1}^n |\mathcal{I}_k| + N$, given as

$$x = \text{col}\left\{\left\{x_k^w, \{x_k^{\varsigma_j}\}_{j \in \mathcal{I}_k}, x_k^{\xi_k}, \right\}_{k=1}^N\right\}.$$

For the previous partition of x, the block maximum norm is defined as

$$\|x\|_{b,\infty} = \max_{1 \le k \le N}\left\{\{\|x_k^w\|, \{\|x_k^{\varsigma_j}\|\}_{j \in \mathcal{I}_k}, \|x_k^{\xi_k}\|\right\}$$

where $\|\cdot\|$ denotes the Euclidean norm of its argument. Moreover, the matrix norm induced from the block maximum norm is given by

$$\|A\|_{b,\infty} = \max_{\|x\|_{b,\infty}=1} \|Ax\|_{b,\infty}$$

where A is $\breve{M} \times \breve{M}$ matrix. As in [65], it can be straightforwardly shown that the block maximum norm has the unitary invariance property of the Euclidean norm under properly defined block-wise transformation.

Next, by evaluating the block maximum norm of (4.59) and by applying its submultiplicativity property, we can obtain the following relation

$$\|E\tilde{\boldsymbol{q}}_i\|_{b,\infty} \leq \|\breve{\mathcal{A}}\|_{b,\infty} \|I - \mathcal{M}\mathcal{R}_U\|_{b,\infty} \|\breve{\mathcal{C}}\|_{b,\infty} \|E\tilde{\boldsymbol{q}}_{i-1}\|_{b,\infty}. \tag{4.62}$$

Let us now evaluate the block maximum norms of the extended combination matrices $\breve{\mathcal{A}}$ and $\breve{\mathcal{C}}$. To do so, we will focus on $\breve{\mathcal{A}}$ given in (4.48), while the same holds for $\|\breve{\mathcal{C}}\|_{b,\infty}$. Since $\breve{\mathcal{A}}$ is a row-stochastic matrix, we can bound $\|\breve{\mathcal{A}}\|_{b,\infty}$ as follows

$$
\begin{aligned}
\|\breve{\mathcal{A}}x\|_{b,\infty} &= \max_{1\leq k\leq N} \left\{ \left\|\sum_{\ell\in\mathcal{N}_k} a_{k,\ell}^w x_\ell^w\right\|, \left\{\left\|\sum_{\ell\in\mathcal{N}_k\cap\mathcal{C}_j} a_{k,\ell}^{\varsigma_j} x_\ell^{\varsigma_j}\right\|\right\}_{j\in\mathcal{I}_k}, \left\|x_k^{\xi_k}\right\| \right\} \\
&\leq \max_{1\leq k\leq N} \left\{ \sum_{\ell\in\mathcal{N}_k} |a_{k,\ell}^w|, \max_{j\in\mathcal{I}_k} \sum_{\ell\in\mathcal{N}_k\cap\mathcal{C}_j} |a_{k,\ell}^{\varsigma_j}|, 1 \right\} \|x\|_{b,\infty} = \|x\|_{b,\infty}.
\end{aligned}
\tag{4.63}
$$

From the previous bound, we can easily verify that $\|\breve{\mathcal{A}}\|_{b,\infty} < 1$ given that A^w, A^{ς_j} are row-stochastic, i.e., $A^w 1_N = 1_N$ and $A^{\varsigma_j} 1_{|\mathcal{C}_j|} = 1_{|\mathcal{C}_j|}$, for $j = \{1,\ldots,J\}$.

At this point, we only need to find the conditions that ensure

$$\|I - \mathcal{M}\mathcal{R}_U\|_{b,\infty} < 1.$$

Under Assumption A4, due to the unitary invariance of the block maximum norm these conditions correspond to the mean stability conditions of stand-alone LMS filters and can be easily realized to be

$$\mu_k < \frac{2}{\lambda_{\max}\left(R_{U_{kg}}, \{R_{U_{k_{jc}}}\}_{j\in\mathcal{I}_k}, R_{U_{k_l}}\right)} \qquad \text{for each } k,$$

where $k = \{1,\ldots,N\}$ and $\lambda_{\max}(X,Y,Z)$ denotes the maximum of the maximum eigenvalues of the Hermitian matrix arguments X, Y and Z.

The above discussion is summarized in the subsequent theorem.

Theorem 4.1. *For any initial conditions, under the Assumptions A1-A4 made in Subsection 4.3.3, if the positive step-size of each node satisfies*

$$\mu_k < \frac{2}{\lambda_{\max}\left(R_{U_{k_g}}, \{R_{U_{k_{j_c}}}\}_{j \in \mathcal{I}_k}, R_{U_{k_l}}\right)} \ ,$$

then the estimates generated by ATC (or CTA) D-NSPE algorithm converge in the mean, i.e.,

$$\lim_{i \to \infty} E\tilde{\boldsymbol{q}}_i = 0_{\breve{M} \times 1}, \tag{4.64}$$

if the combination matrices related to the estimates of global and common parameters are row-stochastic.

4.3.5 Mean-square steady-state performance

At this point, we aim to evaluate the mean-square performance of the general diffusion model in (4.47). In particular, we will examine the performance in the steady-state in terms of MSD and EMSE.

To this end, we use the energy conservation arguments [52], [14]. Specifically, after equating the weighted norm of (4.47) and taking the expectation under Assumptions A1-A3, we obtain the subsequent variance relation

$$E\|\tilde{\boldsymbol{q}}_i\|_\Sigma^2 = E\|\tilde{\boldsymbol{q}}_{i-1}\|_{\Sigma'}^2 + E\left\{\boldsymbol{\mathcal{V}}_i^H \mathcal{M}\breve{\mathcal{A}}^T \Sigma \breve{\mathcal{A}} \mathcal{M}\boldsymbol{\mathcal{V}}_i\right\} \tag{4.65}$$

where Σ is an arbitrary $(\breve{M} \times \breve{M})$ Hermitian nonnegative-definite matrix that we are free to choose, and

$$\Sigma' = E\left\{\breve{\mathcal{C}}^T (I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)^H \breve{\mathcal{A}}^T \Sigma \breve{\mathcal{A}} (I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)\breve{\mathcal{C}}\right\}. \tag{4.66}$$

To proceed, we have to extract Σ from r.h.s. of (4.66) and from the second term on r.h.s. in (4.65). To do so, we will use vectorization operator and exploit some useful properties of the trace operator and Kronecker product, i.e.,

$$\text{vec}(ABC) = (C^T \otimes A)\text{vec}(B) \tag{4.67}$$

and

$$\text{Tr}(AB) = \text{vec}(A^T)^T \text{vec}(B). \tag{4.68}$$

Furthermore, in addition to Assumptions A1-A3, here we also use Assumption A5, stated in Subsection 4.3.3.

Thus, after defining $V = E\,\boldsymbol{\mathcal{V}}_i\boldsymbol{\mathcal{V}}_i^H = \text{blockdiag}\left(\{(\text{Tr}\,R_{v,k}\Omega_k)\Psi_k\}_{k=1}^N\right)$ (see [66]), we get

$$E\|\tilde{\boldsymbol{q}}_i\|_{\Sigma}^2 = E\|\tilde{\boldsymbol{q}}_{i-1}\|_{\Sigma'}^2 + \text{Tr}\left(\breve{\mathcal{A}}\,\mathcal{M}\,V\,\mathcal{M}\breve{\mathcal{A}}^T\,\Sigma\right). \tag{4.69}$$

Next, we introduce $\sigma = \text{vec}(\Sigma)$. In order to extract Σ from Σ', we take the following steps

$$\sigma' = \text{vec}(\Sigma') = F\sigma \tag{4.70}$$

where F is a matrix, of dimensions $\breve{M}^2 \times \breve{M}^2$, given by

$$\begin{aligned} F &= E\left\{\left(\breve{\mathcal{A}}\,(I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)\,\breve{\mathcal{C}}\right)^T \otimes \left(\breve{\mathcal{C}}^T\,(I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)^T\,\breve{\mathcal{A}}^T\right)\right\} \\ &= (\breve{\mathcal{C}}^T \otimes \breve{\mathcal{C}}^T)E\left\{(I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)^T\breve{\mathcal{A}}^T \otimes ((I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)^T\breve{\mathcal{A}}^T\right\} \\ &= (\breve{\mathcal{C}}^T \otimes \breve{\mathcal{C}}^T)\,G\,(\breve{\mathcal{A}}^T \otimes \breve{\mathcal{A}}^T) \end{aligned} \tag{4.71}$$

with

$$\begin{aligned} G &= E\left\{(I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)^T \otimes (I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)^T\right\} \\ &= I \otimes I - \mathcal{R}_U\mathcal{M} \otimes I - I \otimes \mathcal{R}_U\mathcal{M} + E\{\boldsymbol{\mathcal{D}}_i^T\mathcal{M} \otimes \boldsymbol{\mathcal{D}}_i^T\mathcal{M}\} \end{aligned} \tag{4.72}$$

and $\mathcal{R}_U = \text{blockdiag}\left\{\{\text{Tr}(\Omega_k)\Psi_k\}_{k=1}^N\right\}$ (see (4.60)).

For sufficiently small step sizes, the forth-order moment of regressors, i.e., the right-most term in (4.72), can be discarded. However, under the data assumptions of Subsection 4.3.3, this term can be evaluated as follows

$$E\{\boldsymbol{\mathcal{D}}_i^T\mathcal{M} \otimes \boldsymbol{\mathcal{D}}_i^T\mathcal{M}\} = S \cdot (\mathcal{M} \otimes \mathcal{M}) \tag{4.73}$$

where

$$S = E\{\boldsymbol{\mathcal{D}}_i^T \otimes \boldsymbol{\mathcal{D}}_i^T\} = \text{blockdiag}\left\{\left\{E\{\mathbf{U}_{k,i}^H\mathbf{U}_{k,i} \otimes \boldsymbol{\mathcal{D}}_i\}\right\}_{k=1}^N\right\} \tag{4.74}$$

with

$$E\{\mathbf{U}_{k,i}^H\mathbf{U}_{k,i} \otimes \boldsymbol{\mathcal{D}}_i\} = K_{(M_k,\breve{M})}E\{\boldsymbol{\mathcal{D}}_i \otimes \mathbf{U}_{k,i}^H\mathbf{U}_{k,i}\}K_{(\breve{M},M_k)} \tag{4.75}$$

and $K_{(m,n)}$ denoting the $mn \times mn$ commutation matrix that satisfies

$$K_{(m,n)}\text{vec}(A) = \text{vec}(A^T)$$

for any $m \times n$ matrix A [67]. In (4.75), it can be shown that

$$E\{\boldsymbol{\mathcal{D}}_i \otimes \mathbf{U}_{k,i}^H \mathbf{U}_{k,i}\} = \text{blockdiag}\left\{\left\{E\{\mathbf{U}_{\ell,i}^H \mathbf{U}_{\ell,i} \otimes \mathbf{U}_{k,i}^H \mathbf{U}_{k,i}\}\right\}_{\ell=1}^N\right\}. \tag{4.76}$$

Moreover, from [66] and [68], we can obtain closed-form expressions for the expectations that appear in (4.76). In particular, we can verify that

$$\begin{aligned}E\{\mathbf{U}_{k,i}^H \mathbf{U}_{k,i} \otimes \mathbf{U}_{k,i}^H \mathbf{U}_{k,i}\} =&\text{Tr}(\Omega_k)\text{Tr}(\Omega_k)\Psi_k \otimes \Psi_k + \text{Tr}(\Omega_k\Omega_k)\text{vec}(\Psi)\text{vec}(\Psi)^T \\ &+ \text{Tr}(\Omega_k\Omega_k)K_{(M_k,M_k)}(\Psi_k \otimes \Psi_k)\end{aligned} \tag{4.77}$$

and

$$E\{\mathbf{U}_{\ell,i}^H \mathbf{U}_{\ell,i} \otimes \mathbf{U}_{k,i}^H \mathbf{U}_{k,i}\} = [\text{Tr}(\Omega_\ell)\Psi_\ell] \otimes [\text{Tr}(\Omega_k)\Psi_k] \tag{4.78}$$

for any $k, \ell \in \{1, 2, \ldots, N\}$ with $k \neq \ell$.

To evaluate the performance measures in the steady state, i.e., $i \to \infty$, by using (4.68), we first rewrite (4.69) as

$$E\|\tilde{\boldsymbol{q}}_\infty\|_\sigma^2 = E\|\tilde{\boldsymbol{q}}_\infty\|_{F\sigma}^2 + \left[\text{vec}(\breve{\mathcal{A}}\mathcal{M}V^T\mathcal{M}\breve{\mathcal{A}}^T)\right]^T \sigma. \tag{4.79}$$

After rearranging, we obtain the following relation

$$E\|\tilde{\boldsymbol{q}}_\infty\|_{(I-F)\sigma}^2 = \left[\text{vec}(\breve{\mathcal{A}}\mathcal{M}V^T\mathcal{M}\breve{\mathcal{A}}^T)\right]^T \cdot \sigma. \tag{4.80}$$

Now, to evaluate the MSD averaged across the whole network defined by

$$\text{MSD}^{\text{net}} = \frac{1}{N}E\|\tilde{\boldsymbol{q}}_\infty\|_I^2, \tag{4.81}$$

we select $\sigma = (I - F)^{-1}\frac{1}{N}\text{vec}(I_{\breve{M}})$. In this way, we get

$$\text{MSD}^{\text{net}} = \frac{1}{N}\left[\text{vec}(\breve{\mathcal{A}}\mathcal{M}V^T\mathcal{M}\breve{\mathcal{A}}^T)\right]^T (I - F)^{-1}\text{vec}(I_{\breve{M}}). \tag{4.82}$$

In order to evaluate MSD at each node k, let us first define the *Khatri-Rao* matrix product.

Definition 2: Consider matrices A and B of dimensions $m \times n$ and $p \times q$, respectively. Let $A = (A_{ij})$ be partitioned with A_{ij} of dimensions $m_i \times n_j$ as the (i, j)-th block submatrix and let $B = (B_{ij})$ be partitioned with B_{ij} as the (i, j)-th block submatrix of

dimensions $p_i \times q_j$ ($\sum m_i = m$, $\sum n_j = n$, $\sum p_i = p$ and $\sum q_j = q$). The Khatri-Rao matrix product is defined as

$$A \odot B = (A_{ij} \otimes B_{ij})_{ij}$$

where $A_{ij} \otimes B_{ij}$ is of dimensions $m_i p_i \times n_j q_j$, while $A \odot B$ is of dimensions $(\sum m_i p_i) \times (\sum n_j q_j)$, (see [69]).

Based on the previous definition, the MSD at node k is

$$\text{MSD}_k = \left[\text{vec}(\breve{\mathcal{A}} \mathcal{M} V^T \mathcal{M} \breve{\mathcal{A}}^T) \right]^T (I - F)^{-1} m_k, \tag{4.83}$$

where the vector m_k is given by

$$m_k = \text{vec}\left(\text{diag}(e_k) \odot Y\right) \tag{4.84}$$

with the Khatri-Rao matrix product of two $N \times N$ partitioned matrices, i.e., $Y = \text{blockdiag}\{I_{M_1}, \ldots, I_{M_N}\}$ and diagonal matrix made of the elements of a $1 \times N$ vector e_k with the unity at the kth position and zeros elsewhere.

On the other hand, the MSD related to the estimation of the global, some specific common or the local vector of parameter at node k, can be evaluated by redefining Y as a $(2N + \sum_{k=1}^{N} |\mathcal{I}_k|) \times (2N + \sum_{k=1}^{N} |\mathcal{I}_k|)$ partitioned matrix, i.e.,

$$Y = \text{blockdiag}\{I_{M_g}, I_{|\mathcal{I}_1|M_c}, I_{M_l}, \ldots, I_{M_g}, I_{|\mathcal{I}_N|M_c}, I_{M_l}\}, \tag{4.85}$$

and by taking $1 \times 2N + \sum_{k=1}^{N} |\mathcal{I}_k|$ vector e_k with the unity at the appropriate position and zeros elsewhere.

Similarly, since the EMSE averaged across the whole network is defined as

$$\text{EMSE}^{\text{net}} = \frac{1}{N} E \|\tilde{\boldsymbol{q}}_\infty\|^2_{\mathcal{R}_U}, \tag{4.86}$$

with $\mathcal{R}_U$ given in (4.60), we have that

$$\text{EMSE}^{\text{net}} = \frac{\left[\text{vec}(\breve{\mathcal{A}} \mathcal{M} V^T \mathcal{M} \breve{\mathcal{A}}^T) \right]^T (I - F)^{-1} \text{vec}(\mathcal{R}_U)}{N}. \tag{4.87}$$

Additionally, the EMSE at each node k is

$$\text{EMSE}_k = \left[\text{vec}(\breve{\mathcal{A}} \mathcal{M} V^T \mathcal{M} \breve{\mathcal{A}}^T) \right]^T (I - F)^{-1} p_k, \tag{4.88}$$

where we select a node k by

$$p_k = \text{vec}\left(\text{diag}(e_k) \odot \mathcal{R}_U\right) \tag{4.89}$$

with $\mathcal{R}_U$ defined as $N \times N$ partitioned matrix as in (4.60). Under the independence of $\mathbf{U}_{k_g,i}$, $\mathbf{U}_{k_{jc},i}$ and $\mathbf{U}_{k_l,i}$, we can evaluate EMSE performance measure related to the global, specific common or local parameter at some node k. To do so, we need to properly redefine the partitions of R_U and the size of vector e_k.

4.4 Simulation results

In this section, we initially discuss some generic simulations that verify the mean-square theoretical results (see Section 4.3.5). Afterwards, the effectiveness of the proposed algorithms is illustrated in the context of cooperative spectrum sensing in CR networks.

4.4.1 Validation of mean-square theoretical results

We assume a network with $N = 10$ nodes where the measurements follow the observation model provided in (4.3) with $L_k = 2$ for all k. In the considered setting, two different vectors of common parameters coexist, i.e., ς_1^o and ς_2^o. The vector ς_1^o is composed of 3 parameters, while ς_2^o consists of 2 parameters. Moreover, we consider that the area of influence of ς_1^o and ς_2^o is formed by $\mathcal{C}_1 \in \{2, 3, 4, 5, 6\}$ and $\mathcal{C}_2 \in \{5, 6, 7, 8\}$, respectively. As a result, there are nodes that are interested in estimating either zero, or one or two different vectors of common parameters. In addition, each node k is interested in estimating a vector of global parameters and a vector of local parameters, each one of length equal to $M_g = 2$ and $M_{l_k} = 3$, respectively.

The data observed by each node, i.e., $\{d_{k,i}, U_{k,i}\}$, have been generated under the assumption of a background noise $\mathbf{v}_{k,i}$ with covariance $\sigma_{v_k}^2 I_2$, where $\sigma_{v_k}^2 = \sigma_v^2 = 10^{-3}$ across the network. Furthermore, each one of the L_k rows of the regressor

$$U_{k,i} = \text{col}\left\{ U_{k_g,i}^T \{U_{k_{jc},i}^T\}_{j \in \mathcal{I}_k} U_{k_l,i}^T \right\}^T$$

has been independently drawn from a time-correlated spatially independent Gaussian distribution. In particular, the c-th row of $U_{k,i}$ is generated according to a first-order autoregressive (AR) model with correlation function $r_{k,c}(i) = \sigma_{u_k}^2 \alpha_k^{|i|}$, where the pair of parameters $\{\sigma_{u_k}, \alpha_k\}$ are randomly chosen in (0,1) so that the Signal-to-Noise-Ratio (SNR) at each node ranges from 10 dB to 20 dB. Hence, $\mathbf{U}_{k,i}$ follows a real matrix variate
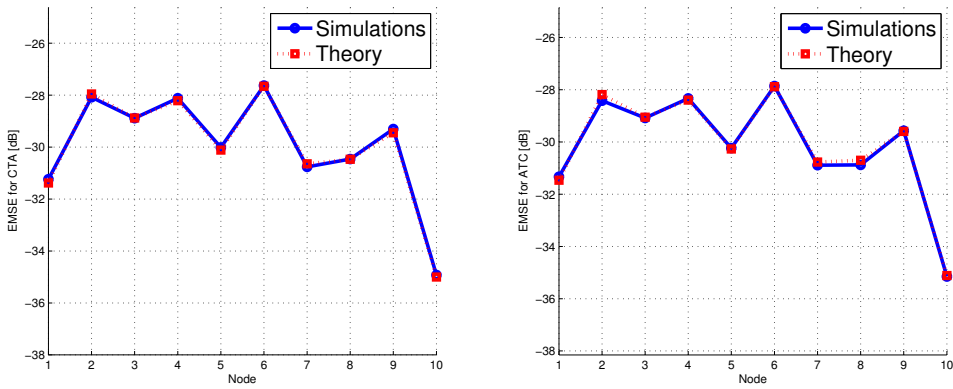
FIGURE 4.2: Steady-state EMSE per node for CTA D-NSPE and ATC D-NSPE.

normal distribution specified by the mean matrix $0_{2 \times M_k}$ and the positive-semidefinite matrices $\Omega_k = I_2$ and $\Psi_k = \text{toeplitz}\{[\sigma_{u_k}^2 \ \sigma_{u_k}^2 \alpha_k \ \ldots \ \sigma_{u_k}^2 \alpha_k^{M_k-1}]^T\}$.

When implementing both CTA D-NSPE and ATC D-NSPE, static uniform combination weights have been assumed, i.e., $a_{k,\ell}^w = 1/|\mathcal{N}_k|$ for all $k \in \{1, 2, \ldots, 10\}$ with $\ell \in \mathcal{N}_k$, and $a_{k,\ell}^{c_j} = 1/|\mathcal{N}_k \cap \mathcal{C}_j|$ for all $k \in \mathcal{C}_j$ and $\ell \in \mathcal{N}_k \cap \mathcal{C}_j$ and $j \in \{1, 2\}$. The neighborhood of each node k has been set so that the network graph as well as the subsets $\mathcal{C}_1$ and $\mathcal{C}_2$ are connected. Moreover, in order to validate the theoretical expressions for non-fully connected networks and non-fully connected subsets $\mathcal{C}_j$, we have assumed that $\max_{1 \leq k \leq N}\{|\mathcal{N}_k|\} \leq 4$ and that $\max_{1 \leq k \leq N}\{|\mathcal{N}_k \cap \mathcal{C}_j|\} \leq 3$ for any $j \in \{1, 2\}$.

The experimental values in Figs. 4.2-4.4 result by averaging the mean-square measures over 100 independent experiments where both CTA D-NSPE LMS and ATC D-NSPE LMS are run for 10 000 iterations. Despite the temporal correlation of the regressors as well as the correlation among $\mathbf{U}_{k_g,i}$, $\mathbf{U}_{k_{jc},i}$ and $\mathbf{U}_{k_l,i}$, which were not assumed for the derivation of the theoretical results, all figures show a good match between the simulated curves and the theoretical expressions for the MSD and EMSE at each node k.

4.4.2 Application to spectrum sensing in cognitive networks

In the following, we will also demonstrate the performance of the proposed algorithm when used for cooperative spectrum sensing in CR networks (see [14, Section 2.4] and [58]-[60]). In brief, there are Q primary users (PU) transmitting and N secondary users (SU) sensing the power spectrum. In addition to PUs, for each SU we also assume two types of low-power interference sources, i.e., local interferer (LI) and common interferers (CI). The former is affecting only one SU, while the latter are influencing several SUs. Therefore, the aim for each SU is to estimate the aggregated spectrum

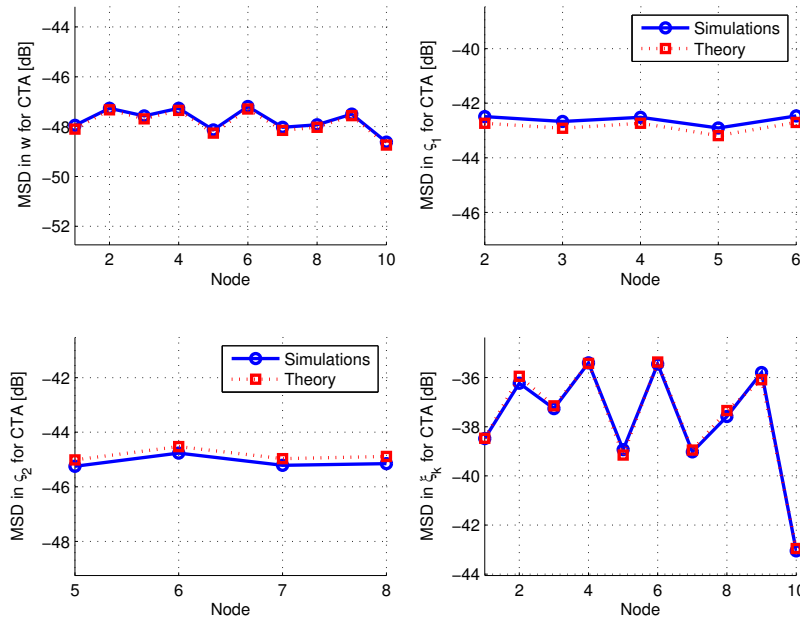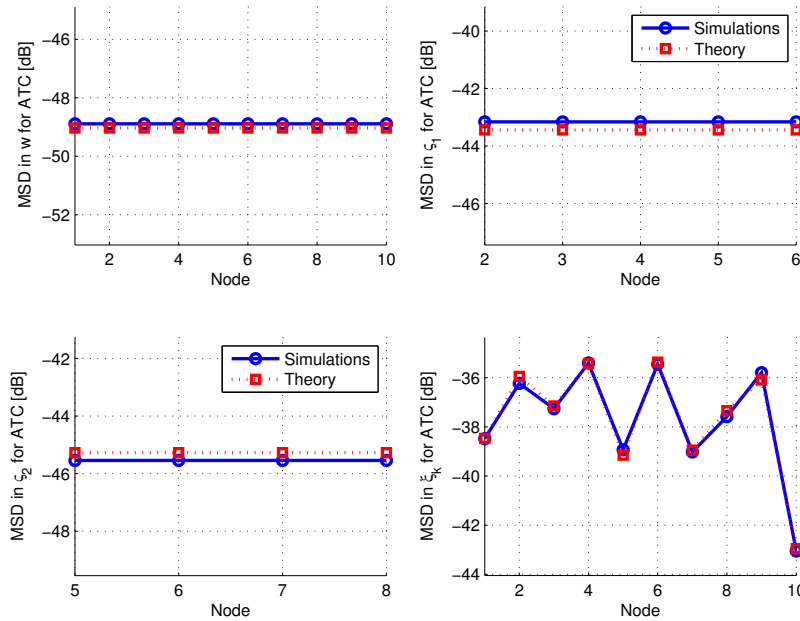FIGURE 4.3: Steady-state MSD per node for CTA D-NSPE LMS.

FIGURE 4.4: Steady-state MSD per node for ATC D-NSPE LMS.

transmitted by all the PUs as well as the spectrum of its own LI and CI. An example of such a scenario is given in Fig. 4.5.

Next, the power spectral density (PSD) of the signal transmitted by the q-th PU, denoted by $\Phi_q^t(f)$, can be approximated by using the subsequent model of X basis
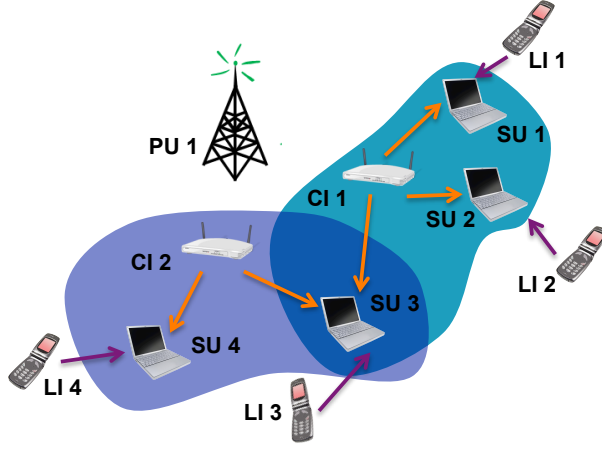
FIGURE 4.5: An illustrative CR scenario. Each one of the 4 secondary users (SU) aims at estimating the aggregated spectrum transmitted by the Primary User (PU) and its interferer(s). Apart from the PU and its Local Interferer (LI), each SU is influenced by one or two Common Interferers (CI). SU 3 is influenced by CI 1 and CI 2, while SUs 1, 2 and 4 are only influenced by one CI.

functions

$$\Phi_q^t(f) = \sum_{x=1}^{X} b_x(f)\breve{w}_{qx}^o = b_0^T(f)\breve{w}_q^o \tag{4.90}$$

where $b_0(f) = [b_1(f), \ldots, b_X(f)]^T \in \mathbb{R}^X$ is a vector of basis functions evaluated at frequency f and $\breve{w}_q^o = [\breve{w}_{q1}^o, \ldots, \breve{w}_{qX}^o]^T \in \mathbb{R}^X$ is a vector of weighting coefficients representing the power transmitted by the q-th PU over each basis.

Let $p_{tk,i}(f) = |H_{tk}(f,i)|^2$ be the frequency-dependent attenuation coefficient, where $H_{tk}(f,i)$ is the channel frequency response between the t-th transmitter and k-th receiver [60]. For each time i and frequency f, we define

- $p_{qk,i}(f)$ denoting the attenuation coefficient between the q-th PU and the k-th SU,

- $p_{Ik,i}(f)$ refering to the attenuation coefficient between the local interferer and the k-th SU,

- $p_{jk,i}(f)$ being the attenuation coefficient between the j-th common interferer and the k-th SU, where $j \in \mathcal{I}_k$.

Then, under the assumption of spatial uncorrelation among the channels, the signal received by the k-th SU at time instant i can be expressed as

$$\Phi_{k,i}^r(f) = b_{k,i}^T(f)w_k^o + z_{k,i}, \tag{4.91}$$

where $w_k^o = \text{col}\, \{\breve{w}_1^o, \dots, \breve{w}_Q^o, \varsigma_{\mathcal{I}_k(1)}^o, \dots, \varsigma_{\mathcal{I}_k(|\mathcal{I}_k|)}^o, \xi_k^o\} \in \mathbb{R}^{(Q+|\mathcal{I}_k|+1)X}$, with ξ_k^o and ς_j^o equal to the vectors of weighting coefficients representing the power transmitted by the LI and j-th CI associated with the k-th SU, respectively. Also, $b_{k,i}(f) = p_{k,i}(f) \otimes b_0(f) \in \mathbb{R}^{(Q+|\mathcal{I}_k|+1)X}$, and

$$p_{k,i}(f) = [p_{1k,i}, \dots, p_{qk,i}, p_{\mathcal{I}_k(1),i}, \dots, p_{\mathcal{I}_k(|\mathcal{I}_k|),i}, p_{Ik,i}]^T, \qquad (4.92)$$

while $z_{k,i}$ is the measurement and/or model noise. In the above expression, we dropped the frequency index for compactness of notation. Also note that, in practice, the attenuation factors $p_{tk,i}$ cannot be estimated accurately, so hereafter we assume access only to noisy estimates $\hat{p}_{tk,i}$.

Considering that, at discrete time i, each node k observes the received PSD in (4.91) over L frequency samples $\{f_m\}_{m=1}^L$, the subsequent vector linear model is obtained

$$\mathbf{d}_{k,i} = \mathbf{U}_{k,i} w_k^o + \mathbf{v}_{k,i} \qquad (4.93)$$

where $\mathbf{v}_{k,i}$ denotes noise with zero mean and covariance matrix R_{v_k} of dimension $L \times L$ and $\mathbf{U}_{k,i} = [b_{k,i}(f_1) \dots b_{k,i}(f_L)]^T$ is of dimension $L \times (Q + |\mathcal{I}_k| + 1)X$ with $L > (Q + |\mathcal{I}_k| + 1)X$.

For the computer simulations presented here, we consider a scenario where there is only one common interferer whose PSD can be sensed by nodes in $\mathcal{C}_1 = \{2, 4, 7, 9\}$. Furthermore, we analyze the ATC D-NSPE LMS scheme for several different combining strategies and degrees of connectivity. In particular, we consider the ATC D-NSPE LMS algorithms with:

a) the same neighborhood size at all the nodes, i.e., $|\mathcal{N}_k| = 5$, while $|\mathcal{N}_k \cap \mathcal{C}_1| = 3$ for all $k \in C_1$. In this scenario, we employ the static uniform combination weights, i.e., $a_{k,\ell}^w = 1/5$ and $a_{k,\ell}^{\varsigma_1} = 1/3$,

b) the clique topology, i.e., $|\mathcal{N}_k| = N$ and $|\mathcal{N}_k \cap \mathcal{C}_1| = |\mathcal{C}_1|$ for all $k \in C_1$, with corresponding static uniform combination weights,

c) the topology set as in a), while the combination weights are adaptive. Specifically, the weights corresponding to both global and common parameter estimation processes are being adapted according to the adaptive combination mechanism proposed in [28]. For instance, the weights $a_{k,\ell}^w$, for $\ell \in \mathcal{N}_k$, evolve as

$$a_{k,\ell}^w(i) = \frac{\gamma_{k,\ell}^{-2}(i)}{\sum_{m \in \mathcal{N}_k} \gamma_{k,m}^{-2}(i)}$$

with

$$\gamma_{k,\ell}^2(i) = (1-\nu)\gamma_{k,\ell}^2(i-1) + \nu||\psi_\ell^{(i)} - \phi_{k,w}^{(i-1)}||^2.$$

We also compare these schemes with an LMS-based non-cooperative strategy as well as with the incremental-based NSPE LMS (I-NSPE LMS), developed in [43], that is used as a benchmark.

The step-size of the LMS adaptation at each node is set equal to $\mu_k = 0.04$ for all the algorithms, except for I-NSPE LMS where μ_k is the step-size for estimating the local parameters only. In I-NSPE LMS, the step-sizes for estimating global and common parameters are set to $\mu_w^{I-NSPE} = \mu_k/N$ and $\mu_{\varsigma_j}^{I-NSPE} = \mu_k/|\mathcal{C}_j|$, respectively, thus assuring a fair comparison among the strategies.

Figure 4.6 depicts the learning behavior of the schemes in terms of the network MSD associated with the estimation of w^o, ς_1^o and ξ_k^o. Each network MSD is the result of averaging the local MSDs associated with the estimation of w^o and ξ_k^o at each node, except for the network MSD associated with the estimation of ς_1^o, which is averaged over the nodes belonging to the set $\mathcal{C}_1$. To generate each plot, we have averaged the results over 100 independent experiments where we assumed $Q = 2$ PUs, $N = 10$ SUs and $X = 16$ Gaussian basis functions, of amplitude normalized to one and standard deviation $\sigma_b = 0.05$. Furthermore, we have considered that each SU scans $L = 80$ channels over the normalized frequency axis between 0 and 1, whereas the noise $z_{k,i}$ in (4.91) is zero-mean Gaussian with standard deviation varying between 0.04 and 0.16 for different k.

Each attenuation coefficient follows $\hat{p}_{tk,i}(f) = p_{tk,i}(f) + n_{tk}$, where n_{tk} denotes a zero-mean Gaussian variable with standard deviation in the range between 0.3 and 1.25, while $p_{tk,i}(f)$ is related to the frequency response of the channel modeled as a static 3-tap FIR filter. Each tap is assumed to be a zero-mean complex Gaussian random variable with variance $\sigma_h^2 = 0.25$. Under this setting, we observe that all the proposed D-NSPE schemes outperform the non-cooperative one, especially when estimating w^o and ς_1^o. Note that D-NSPE(a) and D-NSPE(b) well-approximate the centralized-like performance of the incremental strategy. Finally, due to the fact that the adaptive combiners integrate some additional knowledge regarding the quality of the estimates at the different nodes, D-NSPE(c) outperforms all other schemes including the incremental.

Finally, to illustrate the asymptotic unbiasedness of the proposed technique, in Fig. 4.7 we plot its mean weight error behavior under the previously described setting. In particular, the figure indicates the mean weight error evolution of some vector coefficients related to the global, common and local parameters at randomly selected
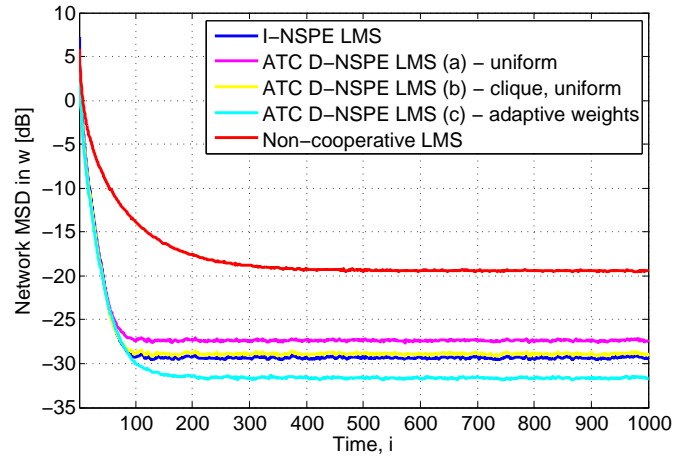
(a)
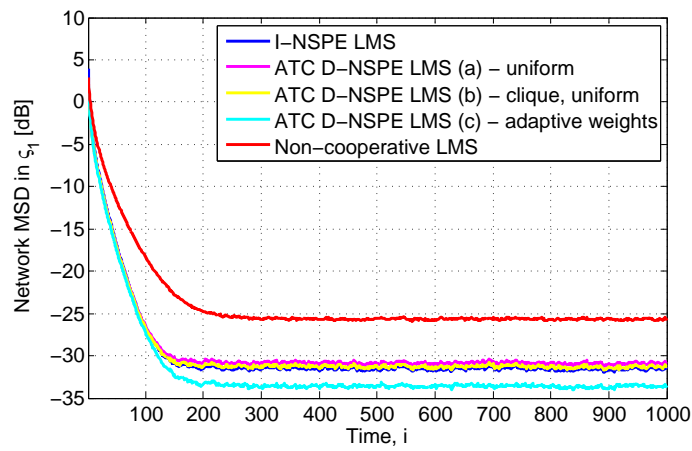
(b)
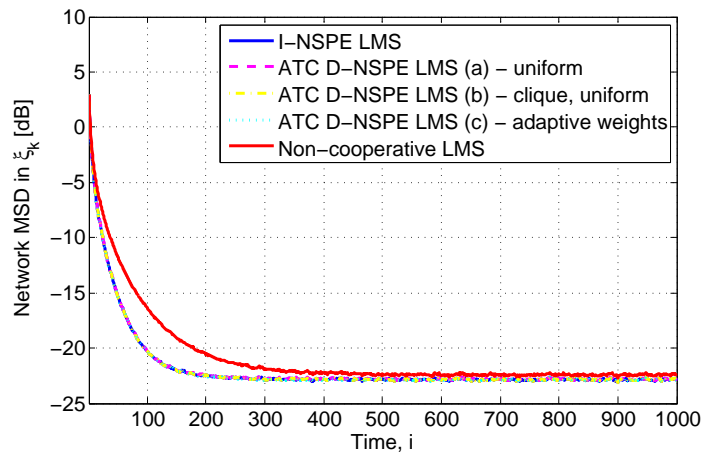
(c)

FIGURE 4.6: Learning behavior of network MSD with respect to the parameters of global interest (a), common interest (b) and for the parameters of local interest (c).

nodes. As expected by Theorem 4.1, D-NSPE LMS has estimated the optimum weight vectors without bias.
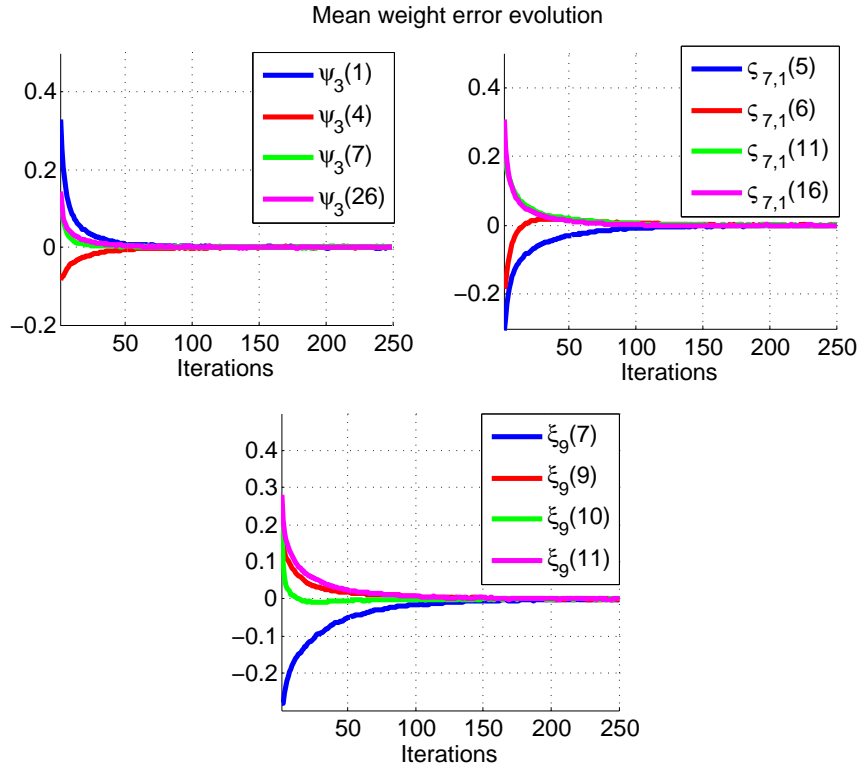
FIGURE 4.7: The mean error trajectories of some vector coefficients related to the global (upper left), common (upper right) and local parameters (bottom) at randomly selected nodes.

4.5 Concluding Remarks

In this chapter, we have proposed two distributed adaptive schemes where a local LMS is run at each node in order to estimate each set of local parameters. Coupled among themselves and with all these local estimation processes, the parameters of global and common interests are estimated by LMS-based schemes implemented under a diffusion mode of cooperation. After obtaining conditions under which the proposed strategies are asymptotically unbiased, the mean-square steady-state performance has been evaluated. Computer simulation results supporting our theoretical findings have been provided. Moreover, the performance of the proposed algorithms has been illustrated in the context of cooperative spectrum sensing in cognitive radio networks.

This chapter concludes discussion on the parameter estimation issue over networks. The following two chapters are devoted to the data gathering problem in wireless sensor networks for structural health monitoring.

Chapter 5

Lossless DSC approach for data gathering

This chapter addresses the problem of data gathering based on distributed source coding in wireless sensor networks for structural health monitoring. As explained in Section 1.2 of the introductory chapter, there are several difficulties in the sensor reachback problem arising in this kind of networks, such as massive data to be transmitted to the sink node as well as the fact that not all sensor nodes always have communication channels of good enough quality. To handle these difficulties, a DSC-based approach, which exploits spatial correlation among the measurements at different nodes, will be considered in this chapter.

A DSC technique achieves lossless compression of multiple correlated sensor outputs [36] without establishing any communication links between the nodes. A DSC algorithm for the reachback problem, based on pair matching of the nodes, was proposed in [37]. A significantly improved algorithm was proposed in [38], based on application of DSC strategy in a sequential manner.

In contrast to the work in [38], where each sensor node uses a direct communication channel with the sink node, in our work, presented in this chapter, we additionally allow cooperation among the nodes. Under the assumption that there exist unreliable channels between the sensor nodes and the sink, it can be shown that this approach achieves less total power consumption as well as reduced maximum power per sensor node required for a feasible power allocation to exist. Furthermore, these performance improvements are obtained at the cost of only a slight increase in computational complexity as compared to the complexity of the scheme in [38].

5.1   Problem statement

In general, we consider a network of sensors acquiring data from a civil structure in order to reproduce a physical phenomenon at the sink. The natural analog signals are first quantized and then compressed in order to minimize the total number of bits which will be sent to the sink.

Let us now formulate the problem more precisely. We consider a dense wireless sensor network consisting of N nodes, deployed in a civil structure that we wish to monitor. Each sensor node acquires a measurement X_n ($n \in \mathcal{N} = \{1, \ldots, N\}$) of some physical variable of its environment and transmits it to a single sink node, for further processing. We model each such measurement as an instance of a discrete random variable $\mathcal{X}_n$ whose number of possible values equals the number of quantization levels. Due to the nature of the event being monitored, we assume that the random variables $\mathcal{X}_n$ are correlated. In this setting, our scope is to devise an energy efficient method for the sensor reachback problem. Thus, a proper cost function would be the sum

$$\sum_{n=1}^{N} P_n \, , \tag{5.1}$$

where P_n denotes the power required for transmitting the data of node n to the sink. The minimization of (5.1), subject to some proper constraints, shall give an efficient transmission method for our problem. Apart from the power variables P_n, the transmission rates R_n of each sensor node, also constitute a set of variables that need to be defined in an optimal manner.

We assume uncorrelated flat fading channels between the sources and the sink, corrupted by additive white Gaussian noise (AWGN). The channel capacity is $C_n(P_n) = log(1 + \gamma_n P_n)$, where the noise power is normalized to one and channel gains γ_n are constants known to the sink. Since the sink is supposed to recover all measurements losslessly, the rate at which each sensor node transmits should satisfy $R_n \leq C_n(P_n)$. For the power which may be transmitted by each sensor node, we also impose a peak power constraint P_{max} due to the fact that every sensor node has limited transmission power in practice.

5.2   Information Theory background

Recall that the entropy of a discrete random variable X_1, denoted as $H(X_1)$, could be seen as the minimum number of bits required to encode X_1 without any loss of
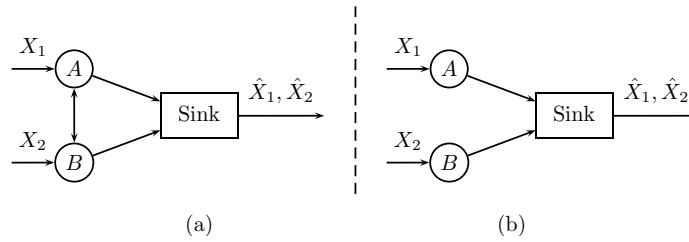
FIGURE 5.1: (a) Explicit Communication. (b) Distributed Source Coding.

information. Similarly, the joint entropy $H(X_1, X_2)$ of two discrete random variables X_1 and X_2 can be seen as the minimum number of bits required to encode X_1 and X_2 jointly. In case that X_1 contains some information about X_2, the following inequality holds $H(X_1, X_2) < H(X_1) + H(X_2)$.

First, let us consider the explicit communication scenario shown in Fig. 5.1a. A typical joint encoding of X_1 and X_2 could be achieved by first encoding X_2 to $H(X_2)$ bits (its individual entropy), then communicating these bits to the X_1 node, and finally encoding X_1 to $H(X_1|X_2)$ bits, which is the conditional entropy of X_1 if X_2 is known, and by definition, joint entropy could be achieved $H(X_1, X_2) = H(X_2) + H(X_1|X_2)$. Obviously, exploiting correlation in an efficient way by applying such a joint encoding scheme across the whole WSN is infeasible since it would require all nodes to participate in inter-node communication. Furthermore, the nodes would need to communicate their individual entropies among themselves which would prohibitively increase power consumption.

An alternative strategy, Distributed source coding (DSC), refers to separate compression and joint decompression of two or more physically separated sources. The sources are encoded independently (hence distributed) at the encoders and decompressed jointly at the decoder [36]. In other words, it is enough to use $H(X_1|X_2)$ bits to encode X_1 instead of $H(X_1)$, even without communication between two nodes, given that the decoder has full knowledge of X_2 (Fig. 5.1b). This was shown for the first time by Slepian and Wolf in 1973 [70]. They showed that two discrete sources X_1 and X_2 can be losslessly decoded as long as the rates of two sources are in the so-called Slepian-Wolf region (Fig. 5.2), which is defined by the following inequalities:

$$R_1 \geq H(X_1|X_2)$$

$$R_2 \geq H(X_2|X_1)$$

$$R = R_1 + R_2 \geq H(X_1, X_2).$$
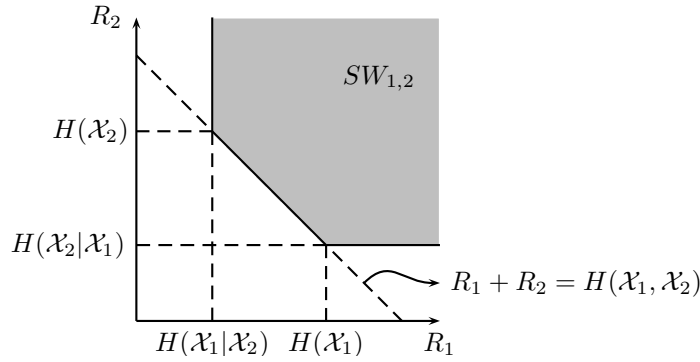
FIGURE 5.2: The Slepian-Wolf region $SW_{1,2}$ for two sources $\mathcal{X}_1$ and $\mathcal{X}_2$, defines the feasible rate pairs (R_1, R_2) for which joint lossless decoding can be performed at the destination.

To understand the concept of DSC and how correlation may be exploited let us consider a simple example in which the most significant bits in both sequences are the same, while some last bits, the least significant ones, differ. In fact, in this example the conditional entropy corresponds to these (different) least significant bits.

5.3   DSC in a network case

DSC-based optimal strategies for WSNs were proposed in [71], [72]. Despite the attempts to design codes for multiple sources [73], this problem still remains open due to the fact that these codes achieve suboptimal rates. Thus, in practice Slepian-Wolf (S-W) codes only for two sources are considered. These codes can operate at any rate in the S-W region and may adapt to any change in correlation between the sources [74].

Roumy and Gesbert [37] formulated the pairwise distributed source coding problem in the network setting. They presented algorithms for rate and power allocation for two scenarios while assuming the existence of the direct channels between each source node and the terminal.

In the first scenario, by assuming noiseless channels between nodes and the sink, they considered the problem of deciding which particular nodes should be jointly decoded at the sink and which rates should be allocated in order the total sum rate to be minimized. In the second scenario, they assumed orthogonal noisy channels between the nodes and the sink and considered minimization of total power consumption.

Also, it was assumed that the sink possesses full knowledge of the individual and the joint entropies as well as the channel capacities for each possible pair of nodes.
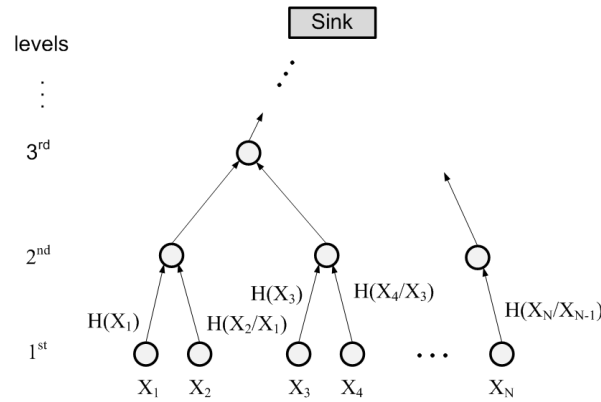
FIGURE 5.3: A hierarchical structure.

In short, the resource allocation problem is to determine the optimal pairing combinations of the nodes in the network and the corresponding rates for them such that the sum rate or the sum power is minimized. As a result, the problem was mapped onto the graph-theoretic problem of choosing the minimum weight matching of an appropriately defined weighted undirected graph.

It is of interest to underline that in general the chosen optimal pairs are not the same for both scenarios considered (i.e. sum rate or sum power minimization).

Although this approach has significantly smaller cost than the one which does not apply DSC (all nodes send the measurements at $H(X_n)$, regardless of their correlation), it is still far from the theoretically optimal case (DSC for N sources), especially in cases the correlation among the nodes is high. This is the result of considering only the correlation of the nodes in the pairs, and not among the pairs. Motivated by this, let us examine possible ways to exploit the correlation further.

5.4 Hierarchical and Sequential structures

Let us assume a hierarchical transmission structure (Fig. 5.3). Without loss of generality, let us assume that only 1st level nodes observe a phenomenon and take the measurements $X_1, X_2, X_3, X_4, ..., X_N$ and that these measurements are correlated. Since we are restricted to practical codes for pairwise DSC, let us examine whether hierarchical organization of these pairs could provide us with any benefit.

Let us assume that each pair applies pairwise DSC sending total bits equal to the joint entropy, e.g. $H(X_1, X_2) = H(X_1) + H(X_2|X_1)$. The question is whether the received sequences at the 2nd level nodes could be further compressed. From the information-theoretic perspective, if $H(X_1, X_2) + H(X_3, X_4) > H(X_1, X_2, X_3, X_4)$ holds, then it is

FIGURE 5.4: A sequential structure.

possible to have further gain. However, in the general case, it is not easy to find the correlation pattern of already coded sequences by S-W codes. Therefore, in order to further exploit the spatial correlation of $X_1, X_2, X_3, X_4...$, 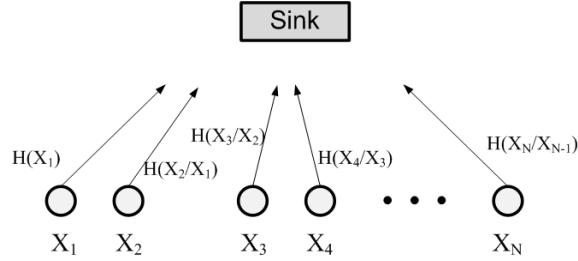the nodes at 2nd level would have to decode the received sequences. In other words, the sequences (X_1, X_2) and (X_3, X_4) should be recovered at the respective nodes at 2nd level.

Let us now consider how the correlation model affects our possible strategies. If the correlation between the joint sequences (X_1, X_2) and (X_3, X_4) was known, then even with pairwise DSC codes (which we are practically restricted to), it would be possible to achieve the optimum overall joint entropy for four sources, $H(X_1, X_2, X_3, X_4) = H(X_1, X_2) + H(X_3, X_4 | X_1, X_2)$. However, for a hierarchical structure of $N = 2^i$ sources, in order to achieve the optimum, the correlation between $(X_1, X_2, ..., X_{2^i/2})$ and $(X_{(2^i/2)+1}, ..., X_N)$ should be known for each i. This is too difficult to have in practice and thus we restrict ourselves to the pairwise correlation model $(X_1|X_2), (X_2|X_3), \ldots,$ $(X_{k-1}|X_k)$, where $k = 2, \ldots, N$. In this case, for a structure given in Fig. 5.3, the best achievable rate at 3rd level could be, for instance, $H(X_1|X_2) + H(X_2) + H(X_3|X_2) + H(X_4|X_3)$. Similarly, for N sources, the best achievable rate at the sink would be $H(X_1) + H(X_2|X_1) + \ldots + H(X_N|X_{N-1})$.

Alternatively, the previous rate could be obtained by applying the so-called sequential DSC [75] which is a non-hierarchical, 1-level structure (Fig. 5.4). The main idea is to use previously decoded data as side information for other sources. For instance, using X_1 as side information, and after receiving $H(X_2|X_1)$, the sink could decode X_2. Next, it could use X_2 as side information for decoding X_3, after receiving $H(X_3|X_2)$, and so on. Consequently, the transmission of only one node at the rate of the individual entropy is required, while all other nodes could transmit at the rates of conditional entropies resulting in the significant reduction of overall transmitted bits.

It should be noted that the process of decoding X_n is dependent on whether $H(X_n|X_{n-1})$ and all previous sequences have been correctly received or not. If any of these fails to be received, the chain is broken and X_n is not able to be decoded.
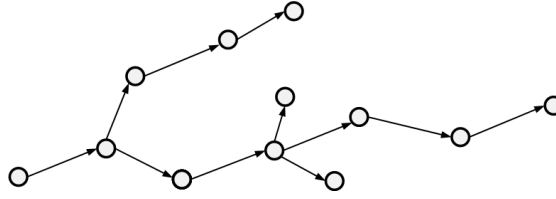
FIGURE 5.5: An example of a directed spanning tree.

Apart from this reliability issue, it could be concluded that hierarchical decoding/re-encoding scheme with both pairwise S-W codes and correlation model attain the same rate as the 1-level sequential DSC scheme. Moreover, the upper layer nodes could only be seen as simple relay nodes, and there is no benefit of their decoding/re-encoding capabilities. So, it can be concluded that, once properly implemented, DSC strategy is independent of the routing/transmission structure, which was also discussed in [35], in a more general context. The reliability issue, or dependence on receiving previous $H(X_n|X_{n-1})$ sequences correctly, will be tackled in the following section by allowing some cooperation between the nodes.

Also, it should be noted that the case in which all nodes in the hierarchical structure of Fig. 5.3 take measurements (not only at 1st level) is a special case of a 2-D sequential case, which is actually a directed spanning tree problem (Fig. 5.5), as described in [38]. A more general solution was proposed in [38] which outperforms the one in [37], especially in the case of high correlation among the measurements of the nodes. More specifically, the previously explained sequential strategy was applied so that a node may participate in joint decoding more than once, while in [37] it was assumed that each node participates in joint decoding strictly once. They proposed solutions for both noiseless and noisy channel cases. The former case was solved by applying an algorithm based on finding the minimum weight directed spanning tree of an appropriately defined directed graph. The latter case was solved by finding the minimum weight matching forest of an appropriately defined mixed graph.

5.5 Channel-aware cooperation-based extension of sequential decoding

To begin with, let us discuss the case that two nodes transmit their measurements to the sink as in Fig. 5.1b. In general, the nodes have to transmit at overall rate which equals to the joint entropy of their measurements $H(X_1, X_2)$.

For the noiseless channels, the nodes may transmit at any rates as long as their rates are on the boundary of Slepian-Wolf region (Fig. 5.2). However, it should be noted that there are two corner points defining the minimum rate each node may have while the other transmits at its individual entropy (maximum) rate. For noisy AWGN channels which are of similar quality ($\gamma_1 \approx \gamma_2$), in order to minimize the sum of powers, the sink should set the rates to be somewhere in the middle of the slope ($R_1 \approx R_2$). In practice, it may happen that there is a physical obstacle between a node and the sink causing a very small γ in the respective link. In such a case when a node experiences a deeply faded (bad) channel, the sink may compensate for that to a certain extent by allocating the maximum rate to the other node (corner point). Therefore, for a given channel, the minimum transmit power of a node is a function of its conditional entropy.

In a network setting, it is shown in [38] that the sequential scheme achieves optimal overall sum rate under pairwise DSC and pairwise correlation model constraints. Except for the first node, which transmits at the rate of its individual entropy, all other nodes transmit at the conditional entropy rates (corner points). As a result, in the case that a node has a very bad channel, the sink cannot compensate for it any further. Although a first node and one of its first neighbors will be encoded at a rate on the slope (and allocated power proportional to their channels quality), all other nodes will have a corner point rate allocation and corresponding power allocation.

Moreover, in [38] it is assumed that $P < P_{max}$. However, it should be noted it is required that all previous sequences have been correctly received in order to decode X_n, thus P_{max} threshold should be set really high to account for any possible very bad links in the network. This is not desirable since the sensor nodes are of limited power in practice.

To remedy this problem, we propose a cooperation scheme where a neighboring node could be used as a relay.

Let us first give a relation between the rate variables R_n and the respective power variables P_n. This relation will be given in terms of some functions $f_n(R_n)$ that are defined as the minimum powers required by the nodes of the network in order to transmit data at a rate R_n from node n to the sink node:

$$f_n(R_n) = \text{Minimum } P_n \text{ for rate } R_n . \tag{5.2}$$

In the case where the nodes of the network are only allowed to transmit to the sink node using a direct Additive White Gaussian Noise (AWGN) channel with Signal to Noise

Ratio (SNR) equal to $\gamma_n = |h_n|^2/\sigma_n^2$ we have that

$$f_n(R_n) = \frac{2^{R_n} - 1}{\gamma_n} \; ,$$

(5.3)

which is the inverse of the capacity function of the respective AWGN channel.

Let us assume that the sink performs all calculations regarding the rate and power allocation to the nodes. It possesses the full knowledge of the correlation between each possible pair of nodes, the individual entropies of the sources as well as the channel gains for all node-sink links. In addition, in our strategy, the sink should also know the internode channels among the nodes. Therefore, the sink may locate those nodes which have very bad channels according to some criteria. A criterion might be a threshold channel gain $\gamma_{threshold}$ below which a channel can be considered as a bad one. Another possibility would be to compare the power allocated to a node to a maximum power allowed (peak power constraint). We assume that there are several nodes with bad channels in the network, sparsely distributed. Next, for a node which has been denoted as a large power consumer, a test whether to cooperate with the best of its neighbors is performed as explained below.

Firstly, the sink chooses the best cooperating node from the subset of neighboring nodes, $\mathcal{S}_n$, which includes the predecessor and all successors of the node with a bad channel. In fact, in a sequential scheme where the rate allocation is computed by the directed spanning tree method, each node usually has one node as predecessor and one or more as successors. The exceptions are: i) the first node, which represents the root of the tree and do not have any predecessor ii) last nodes, which represent the leaves of the tree and do not have any successors. However, the method can be still applied since they have at least one neighbor.

Finally, the sink performs a test to decide whether it is beneficial to cooperate in terms of power consumption.

So, each node may transmit either through the direct channel to the sink, or use a relay (cooperate), so the decision will be made between these two available protocols:

1. Direct sink access: Each node n is given the option to communicate with the sink, via an AWGN channel with complex gain h_n and noise variance σ_n^2. Thus, using $\gamma_n = |h_n|^2/\sigma_n^2$, the capacity of this link is given by

$$C_n^{\text{Direct}}(P_n) = \log_2(1 + \gamma_n P_n) \; .$$

(5.4)

2. Cooperative decode and forward: Each node n is given the option to send all its data to a relay node, which will then forward it to the sink, in a two time-slot protocol. Let us assume that the channel between node n and node m, which will act as a relay, is an AWGN channel with complex gain h_{nm} and noise variance σ_{mn}^2, and define also $\gamma_{nm} = |h_{nm}|^2/\sigma_{nm}^2$. Then, the capacity of this protocol, taking also into account optimal power allocation between the nodes n and m, is given by [76]:

$$C_n^{\mathrm{DF}}(P_n) = \frac{1}{2} \log_2 \left(1 + \frac{\gamma_{nm}\gamma_m}{\gamma_{nm} + \gamma_m} P_n \right) . \tag{5.5}$$

As previously explained, the node transmission performance is tested against the performance of the best relay chosen from the subset $\mathcal{S}_n$. So, the capacity of the second option equals to

$$C_n^{\mathrm{DF}}(P_n) = \frac{1}{2} \log_2(1 + b_n P_n) , \tag{5.6}$$

where

$$b_n = \max_{m \epsilon \mathcal{S}_n} \left\{ \frac{\gamma_{nm}\gamma_m}{\gamma_{nm} + \gamma_m} \right\} . \tag{5.7}$$

Thus, according to the previous, the required functions $f_n(R_n)$ in this case are given by

$$f_n(R_n) = \min \left\{ \frac{2^{R_n} - 1}{\gamma_n}, \frac{4 \cdot 2^{R_n} - 1}{b_n} \right\} , \tag{5.8}$$

and according to the values of γ_n and b_n we have the following two cases:

(a) When $b_n \leq 4\gamma_n$, we have that

$$f_n(R_n) = \frac{2^{R_n} - 1}{\gamma_n} \tag{5.9}$$

(b) When $b_n > 4\gamma_n$ (which also implies that $b_n > \gamma_n$), we have that

$$f_n(R_n) = \begin{cases} \frac{2^{R_n} - 1}{\gamma_n}, & R_n \leq \log_2 \left(\frac{b_n - \gamma_n}{b_n - 4\gamma_n} \right) \\[2ex] \frac{4 \cdot 2^{R_n} - 1}{b_n}, & R_n > \log_2 \left(\frac{b_n - \gamma_n}{b_n - 4\gamma_n} \right) \end{cases} \tag{5.10}$$

Thus, from the above, we can see that energy savings, relative to the scheme in [38], are possible when $b_n > 4\gamma_n$ and $R_n > \log_2 \left(\frac{b_n - \gamma_n}{b_n - 4\gamma_n} \right)$. Fig. 5.6 depicts a plot of $f_n(R_n)$ for $\gamma_n = 1$ and $b_n = 5$, thus, since $b_n > 4\gamma_n$, equation (5.10) is used.
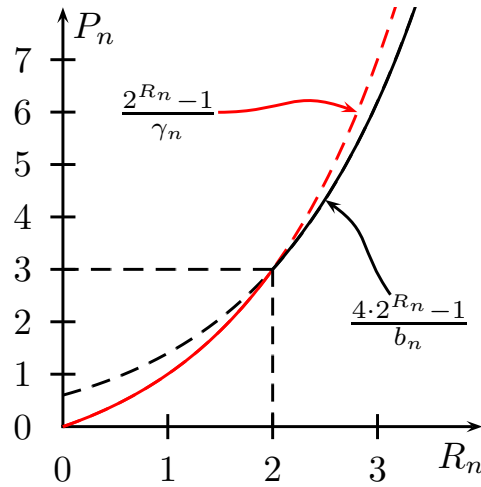
FIGURE 5.6: A demonstration of the function $f_n(R_n)$ for $\gamma_n = 1$ and $b_n = 5$. Function $f_n(R_n)$ appears as a solid line, while the two constituent functions according to (5.10) appear as dashed lines.

5.5.1 Numerical Results

In order to illustrate the gains achieved by applying a DSC approach as well as the proposed extension, let us consider a bridge scenario in which 10 sensors are equidistantly placed along the deck, similarly as in Fig. 1.1. The joint entropy model for any two sources, also used in [37], is a function of the individual entropy (which equals 8 for all nodes), of a correlation coefficient $c = 0.1$, and the distances d_{ij} between the sources i.e.,

$$H(X_i, X_j) = H(X_i) + (1 - 1/(1 + d_{ij}/c))H(X_i).$$

The distance between consecutive sensors is 0.1 and their distance to the sink is in the range from 0.5 to 1.0296. The channel gains are in general assumed equal to the inverse square distance. However, in some cases node-to-sink channels are further faded due to e.g. obstacles. Let us take as deeply faded channels those with $b_n/\gamma_n=10$. The total sum powers for non-DSC approach, sequential DSC and the proposed technique are presented in Table 5.1. Under the assumption that some nodes experience deeply faded channels comparing to their neighbors, the proposed strategy results in a decreased power consumption by the network as a whole. Under the same assumption, the peak power constraint is lowered as well.

TABLE 5.1: Comparison of sum powers.

	Number of deeply faded channels		
	1	2	3
Non-DSC	2571.40	3751.60	5092.90
Sequential	175.73	245.15	324.05
Proposed	129.02	152.76	179.81

5.6   Implementation-related issues

In our system model, we made several assumptions, such as independence of the channels and full knowledge of network topology, correlation model and the channel gains. There are several practical ways which can assure that the above assumptions come true. Let us outline some of the main ones:

1. Independent channels can be achieved by using multiple access techniques appropriate for WSNs.

2. All the involved channels may be estimated during a training period in which all nodes take part.

3. The sink could estimate the pairwise correlation model and corresponding joint entropies from the system model created during the design phase of a construction. Otherwise, it could be obtained after receiving the real measurements sent by the nodes at the rates of individual entropies (without applying DSC) during the training period.

4. In case that some node fails, the sink would need to establish new relations among the nodes only in the neighborhood of this node, without changing the whole sequential scheme.

5.7   Concluding Remarks

In this chapter, we studied the sensor reachback problem in wireless sensor network, where distributed source coding is used in order to losslessly compress the data. This approach is efficient when there is spatial correlation between the sensor nodes, as in structural health monitoring application. After showing that the source coding problem can be optimally separated from the transmission problem in the considered setting, the channel-aware extension of sequential decoding, based on cooperation between the nodes, has been proposed. In the case where some nodes experience deeply faded channels

toward the sink, it has been shown that the proposed scheme may achieve certain energy savings.

Throughout this chapter, we have considered only spatial correlation. However, temporal correlation can be utilized as well. Developing a strategy which exploits efficiently both types of correlation is the subject of the following chapter where we allow for a certain information loss that satisfies a predefined accuracy constraint.

Chapter 6

Lossy Prediction-based approach for data gathering

This chapter deals with a lossy prediction-based approach for data gathering problem in WSN for SHM. As mentioned in Section 1.2 of the introductory chapter, there are various strategies one may employ in order to perform lossy data compression in WSNs, e.g., the techniques based on compressed sensing, distributed transform coding, distributed source modeling, dictionary-based approaches, differential pulse code modulation etc [39]. Related to the prediction-based approach considered in this chapter, the works focus mostly on exploiting temporal correlation among the measurements, see [77–79]. For instance, the authors in [79] propose an approach to perform lossy compression on single node based on a differential pulse code modulation scheme with quantization of the differences between consecutive samples. They exploit a multi-objective evolutionary algorithm to generate a set of combinations of the quantization process parameters corresponding to different trade-offs between compression performance and information loss. Therefore, it is of interest to examine a practical prediction-based approach which may take advantage of both spatial and temporal correlations in WSN for SHM.

In this chapter, we develop a communication protocol which, based on a Time Division Multiple Access (TDMA) strategy and adaptive filtering techniques such as least mean squares and recursive least squares. In general, the RLS algorithm converges faster than its LMS counterpart for stationary processes, which has been also verified in Chapter 3, for the incremental-based network setting for parameter estimation. However, it does not inevitably hold that the tracking performance of RLS is similarly superior to that of LMS. A general conclusion about the relationship between their tracking behaviors is hard to make [61],[52]. Therefore, it is of interest to examine the performance of the proposed technique for both adaptive algorithms.

In either case, the proposed technique aims at overcoming the difficulties associated with the sensor reachback problem for SHM, previously described in Section 1.2. To do so, the protocol allows the sink node to keep an exact replica of the adaptive filters that, at each node, exploit the spatial and temporal correlations among sensor measurements to predict the current measurement from its own past measurements as well as past measurements obtained by its neighbors. Specifically, in the designed protocol each node is assigned a time slot that is divided into two sub-slots. During the first sub-slot, each sensor acquires a new measurement and computes the prediction error of its associated adaptive filter. If the prediction error is small enough (i.e. below a predefined threshold), then during the first sub-slot the considered sensor node sends the output of its filter to its neighbors, so that they can use this value as input for the prediction filters they operate. In the opposite case, i.e., when the prediction error is not small, the node updates its filter (i.e. using an LMS or RLS update step) and sends its actual measurement to its neighbors. Afterwards, if the prediction is not accurate, since a Multiple Input Single Output (MISO) channel is known to result in energy savings as compared to the Single Input Single Output (SISO) case [80], all the nodes which collaborated during the first sub-slot will form a MISO channel to simultaneously transmit the current measurement to the sink node. This way, with the aim of having an exact replica of all the filters implemented by the cooperating sensor nodes, the sink node is able to incorporate the transmitted measurement to the input of the aforementioned filters and update the filter associated with the considered sensor node. After deriving the communication protocol, both LMS-type and RLS-type implementations of the new technique have been tested extensively via real acceleration measurements from the Canton Tower in China.

6.1   Problem statement

Let us consider a dense wireless sensor network consisting of N nodes, deployed in a civil structure that we wish to monitor. Consider also that node n ($n = 1, 2 \ldots, N$) has N_n neighbors, in the sense that they are close enough to node n so that wireless communication with low power can be accomplished. We will denote the neighbors of node n as $k_{n,1}, k_{n,2}, \ldots, k_{n,N_n}$. With a slight abuse of notation, throughout this chapter we will use normal letters (both capital and small ones) to denote scalar quantities, while boldface capital letters refer to matrices and boldface small letters stand for vectors. Next, each sensor node n, at discrete time t, acquires the measurement $y_{n,t}$ which is related to an event that takes place in the area where the wireless sensor network has been deployed. Define also the vectors of m past measurements of each sensor node n

as

$$\mathbf{y}_{n,t} = \begin{bmatrix} y_{n,t-1} & y_{n,t-2} & \cdots & y_{n,t-m} \end{bmatrix}^T \, , \tag{6.1}$$

$$n = 1, 2, \ldots, N \, .$$

Also, let us define the stacked vectors

$$\mathbf{u}_{n,t} = \begin{bmatrix} \mathbf{y}_{n,t}^T & \mathbf{y}_{k_{n,1},t}^T & \mathbf{y}_{k_{n,2},t}^T \cdots & \mathbf{y}_{k_{n,N_n},t}^T \end{bmatrix}^T \, , \tag{6.2}$$

$$n = 1, 2, \ldots, N \, ,$$

that represent the past m measurements of all sensor nodes in the neighborhood of node n. Consider now the correlation matrices defined as

$$\mathbf{R}_n = E[\mathbf{u}_{n,t}\mathbf{u}_{n,t}^T] \, , \qquad n = 1, 2, \ldots, N \, . \tag{6.3}$$

Clearly, if the matrices $\mathbf{R}_n$ are diagonal, the sensor measurements within all neighborhoods are uncorrelated. In contrast, if the matrices $\mathbf{R}_n$ are only block-diagonal with block size m, the measurements are correlated in time but spatially uncorrelated. In this work, we will focus on the general case where $\mathbf{R}_n$ are of a general form, implying that the sensor measurements are correlated both in time and in space.

Thus, we are interested in deriving a network protocol able to transmit the sensor measurements to the data-collecting node in an energy-efficient way. Such a protocol should take advantage of the aforementioned correlations, in order to reduce the number of transmissions toward the sink. Furthermore, the protocol should provide accuracy guarantees for the received data.

6.2 A TDMA based cooperative protocol

6.2.1 Predictors and correlation of measurements

As mentioned in the previous section, we are interested in deriving an energy-efficient protocol for the transmission of the measurements to the data-collecting node. To this end, if we were able to reduce the number of information bits that need to be transmitted, this would have a considerable effect on the energy spent by the data-gathering process. Such a reduction in the number of information bits that need to be transmitted can be accomplished if we take advantage of the correlations among the measurements. In particular, if we are able to identify and send only the "new" information that lies in

the measurements, then significant energy savings would emerge. A way for identifying such "new information" employs the notion of signal predictors.

Due to the nature of the observed phenomenon the measurements' process $y_{n,t}$ is commonly a predictable one, at least to some extent. In particular, if the data-collecting node had knowledge of previous measurements acquired by sensor n (and possibly previous measurements of other nodes in the vicinity of node n), then it could compute an estimate of $y_{n,t}$. This estimate, of course, corresponds to information already known to the data-collecting node. In principle, we can distinguish between two different types of prediction functions, namely, (a) one that does not change with time, which implies that the correlation mechanism is constant or stationary, and (b) a time-varying prediction function, implying that the statistics of the signals measured by the nodes of the network have a dynamic behavior. Assuming the process to be stationary, the prediction function can be realized as a linear filter with coefficients obtained by minimizing the mean-squared error between the measurements $y_{n,t}$ and their predicted values.

However, in most real world applications the observation processes are non-stationary since their statistical characteristics are changing in time. As a result, the optimal coefficients of the predictor are changing in time as well. In order to track these changes, a practical approach is to iteratively calculate them by updating previous filter coefficients as it is done in adaptive filters [52]. Such an approach offers the additional benefit that the data-collecting node does not need to know the statistics of the underlying process. Rather these statistics are in effect estimated by the adaptive filter.

6.2.2   Simple cooperative TDMA protocol

As already mentioned in the introduction, another approach for reducing the energy required to transmit data relies on the concept of cooperative communications. In particular, in cooperative communications, a number of accurately synchronized nodes transmit data concurrently so that the system resembles a transmitter with multiple antennas. During the previous phase, the nodes have agreed upon the data that will be sent. In effect, benefits similar to Multiple Input Multiple Output (MIMO) systems can be achieved [80], hence the terms virtual MIMO or distributed MIMO are often used alternatively to denote cooperative communication systems.

For illustration purposes, let us consider now a straightforward cooperative communication protocol for the problem at hand, in which correlation among the measurements acquired by the nodes of the WSN is not taken into account. According to this protocol, each sensor node is assigned its own time-slot in order to transmit information, in a Time Division Multiple Access (TDMA) fashion. Cooperative communication can be
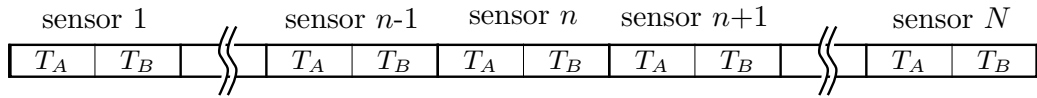
FIGURE 6.1: Each of the sensors is assigned its own time-slot to transmit, in a TDMA fashion. Furthermore, each time-slot is divided into two sub-slots. During the first sub-slot of duration T_A, each sensor n transmits to its k_n neighbors. During the second sub-slot of duration T_B, node n and its neighbors transmit to the sink node in a cooperative fashion.

incorporated into this protocol, by dividing each time-slot into two sub-slots as depicted in Figure 6.1. During the first sub-slot of duration T_A, each sensor n transmits its estimated (or observed) value to its k_n neighbors. During the second sub-slot of duration T_B, node n and its neighbors transmit to the sink node in a cooperative fashion. In such a scenario, both the Amplify and Forward (AF) as well as the Decode and Forward (DF) methods [76] can be adopted.

6.2.3   Cooperative TDMA exploiting correlation

Consider now an extension of the aforementioned protocol, where the correlation of the measurements is taken into account. Since the measurements are correlated in time and in space, the idea of using past measurements acquired by node n as well as past measurements from nearby sensor nodes, in order to predict new measurements, seems well justified. This fact can be used to save some of the transmissions to the sink node, in the case where the sink node can itself predict the required measurements within some predefined accuracy. Thus, let each sensor node n keep a time varying prediction filter $\mathbf{f}_{n,t}$ as well as a data vector

$$\tilde{\mathbf{u}}_{n,t} = \left[\begin{array}{cccc} \tilde{\mathbf{y}}_{n,t}^T & \tilde{\mathbf{y}}_{k_{n,1},t}^T & \tilde{\mathbf{y}}_{k_{n,2},t}^T \cdots & \tilde{\mathbf{y}}_{k_{n,N_n},t}^T \end{array} \right]^T , \tag{6.4}$$

so that the output of the filter, defined as

$$\hat{y}_{n,t} = \mathbf{f}_{n,t}^T \cdot \tilde{\mathbf{u}}_{n,t} \tag{6.5}$$

is an approximation of the actual measurement $y_{n,t}$ obtained by sensor n at time t. In particular, $\hat{y}_{n,t}$ is a prediction of the actual measurement $y_{n,t}$. In the above expressions, we have used the vectors

$$\tilde{\mathbf{y}}_{n,t} = \left[\begin{array}{cccc} \tilde{y}_{n,t-1} & \tilde{y}_{n,t-2} & \cdots & \tilde{y}_{n,t-m} \end{array} \right]^T ,$$

$$n = 1, 2, \ldots, N . \tag{6.6}$$

to represent approximate versions of the past m measurements obtained by sensor n. Thus, vectors $\tilde{\mathbf{u}}_{n,t}$ and $\mathbf{f}_{n,t}$ have dimensions $m \cdot k_n \times 1$. Let us now define a binary variable $b_{n,t}$ according to the prediction error, as

$$
b_{n,t} = \begin{cases} 0 & \text{if} \quad |\hat{y}_{n,t} - y_{n,t}| \leq e \\[2mm] 1 & \text{if} \quad |\hat{y}_{n,t} - y_{n,t}| > e \end{cases}, \tag{6.7}
$$

where e denotes a small positive constant. The approximate measurements $\tilde{y}_{n,t}$ are defined as,

$$
\tilde{y}_{n,t} = \begin{cases} \hat{y}_{n,t} & \text{if} \quad b_{n,t} = 0 \\[2mm] y_{n,t} & \text{if} \quad b_{n,t} = 1 \end{cases}. \tag{6.8}
$$

Based on the above definitions, the protocol of each sensor node n can be seen in Table 6.1. At a time instant t, each sensor acquires its new measurement $y_{n,t}$ and starts a synchronized loop to track the N time-slots that will follow. As seen from Table 6.1, node n is active in two cases: (a) when the current slot s is equal to its index n, and (b) when the current slot s is equal to the index of any of its neighbors. In case (a), the node computes the output of its prediction filter and compares it to the actual measurement $y_{n,t}$. Thus, it computes the binary variable $b_{n,t}$ that determines whether the prediction was accurate or not. In the case where the prediction was not accurate, the prediction filter is updated using an adaptive algorithm. Table 6.1 summarizes the steps followed in order to perform the update of the filter, for the cases of the LMS and the RLS update algorithms. As a general rule, the LMS algorithm should be used when reduced computational complexity is required, due to the fact that the RLS algorithm is an order of magnitude costlier than the LMS algorithm. On the other hand, in general, the RLS algorithm converges faster than its LMS counterpart and is less sensitive to eigenvalues disparities in the autocorrelation matrix of the data for stationary processes. However, it does not inevitably hold that the tracking performance of RLS is similarly superior to that of LMS. A general conclusion about the relationship between their tracking behaviors is hard to make [61],[52]. Therefore, it is of interest to study the performance of the derived protocol for both adaptive algorithms. As it will be shown in the experiments' section, RLS performs better when adaptation stalls and restarts very often during operations, as it is the case with the suggested technique. Regardless of the algorithm used for the update, $y_{n,t}$ is used as a desired response signal. Then, the sensor node n computes $\tilde{y}_{n,t}$, which is either the output of the prediction filter (accurate prediction) or the actual measurement (inaccurate prediction). Thus, sensor n updates its input vector $\tilde{\mathbf{u}}_{n,t+1}$ and sends $\tilde{y}_{n,t}$ and $b_{n,t}$ to its neighbors. Finally, $\tilde{y}_{n,t}$ is sent to the sink node only if the prediction was inaccurate, otherwise the sink node is able to

TABLE 6.1: The protocol executed by the sensor node n.

Initialize $\mathbf{f}_{n,0}$, $\tilde{\mathbf{u}}_{n,0}$ and e

Initialize μ (if LMS is used)

Initialize λ , $\mathbf{P}_{n,-1} = \delta^{-1}\mathbf{I}$ (if RLS is used)

For $t = 0$ to $+\infty$

   Acquire the measurement $y_{n,t}$

   For $s = 1$ to N

     If $s = n$ then

       $\hat{y}_{n,t} = \mathbf{f}_{n,t}^T \tilde{\mathbf{u}}_{n,t}$

$$b_{n,t} = \begin{cases} 0 & \text{if} \quad |\hat{y}_{n,t} - y_{n,t}| \leq e \\ \\ 1 & \text{if} \quad |\hat{y}_{n,t} - y_{n,t}| > e \end{cases}$$

$$\tilde{y}_{n,t} = \begin{cases} \hat{y}_{n,t} & \text{if} \quad b_{n,t} = 0 \\ \\ y_{n,t} & \text{if} \quad b_{n,t} = 1 \end{cases}$$

       If $b_{n,t} = 1$

$\mathbf{f}_{n,t+1} = \mathbf{f}_{n,t} + \mu(y_{n,t} - \hat{y}_{n,t})\tilde{\mathbf{u}}_{n,t}$ (LMS update)

OR

$\mathbf{k}_{n,t} = \dfrac{\lambda^{-1}\mathbf{P}_{n,t-1}\tilde{\mathbf{u}}_{n,t}}{1+\lambda^{-1}\tilde{\mathbf{u}}_{n,t}^T\mathbf{P}_{n,t-1}\tilde{\mathbf{u}}_{n,t}}$

$\xi_{n,t} = y_{n,t} - \hat{y}_{n,t}$ (RLS update)

$\mathbf{f}_{n,t+1} = \mathbf{f}_{n,t} + \mathbf{k}_{n,t}\xi_{n,t}$

$\mathbf{P}_{n,t} = \lambda^{-1}\mathbf{P}_{n,t-1} - \lambda^{-1}\mathbf{k}_{n,t}\tilde{\mathbf{u}}_{n,t}^T\mathbf{P}_{n,t-1}$

       End

       Update $\tilde{\mathbf{u}}_{n,t+1}$ using $\tilde{y}_{n,t}$

       Send $\tilde{y}_{n,t}$ and $b_{n,t}$ to the neighbors (T_A sub-slot)

       If $b_{n,t} = 1$

         Send $\tilde{y}_{n,t}$ to the sink (T_B sub-slot)

       End

     Elseif $s \in \{k_{n,1}, k_{n,2}, \ldots, k_{n,N_n}\}$

       Listen for $\tilde{y}_{s,t}$ and $b_{s,t}$ (T_A sub-slot)

       Update $\tilde{\mathbf{u}}_{n,t+1}$ using $\tilde{y}_{s,t}$

       If $b_{s,t} = 1$

         Send $\tilde{y}_{s,t}$ to the sink (T_B sub-slot)

       End

     Else

       Sleep($T_A + T_B$ seconds)

     End

   End

End

---

compute $\tilde{y}_{n,t}$ using a prediction filter. In case (b), i.e., when a neighbor of n is active, node n listens for the transmitted values $\tilde{y}_{s,t}$ and $b_{s,t}$. It then updates its input vector

$\tilde{\mathbf{u}}_{n,t+1}$ with the received value $\tilde{y}_{s,t}$ and, in the sequel, helps its neighbor transmit to the sink by relaying $\tilde{y}_{s,t}$ if $b_{s,t}$ was 1.

The protocol followed by the sink node is depicted in Table 6.2. At each time instant, the sink node also executes a loop so as to track the N time-slots, in a synchronized fashion. For the first T_A seconds of each slot, the sink node is inactive because sensor-to-sensor communication takes place. At the following T_B seconds however, the sink node is receiving the measurement $\tilde{y}_{s,t}$ of the node assigned to the current slot. Of course, in the case where the prediction at node s was accurate, such a message will not be transmitted. Thus, the sink node must implement a procedure to detect such "empty" messages. The result of the detection process is a binary variable $\hat{b}_{s,t}$ which will be equal to $b_{s,t}$ in the case where the detection is correct. In the sequel, the sink node is able to compute $\tilde{y}_{s,t}^{(S)}$, (that is, a copy of $\tilde{y}_{s,t}$ at the sink) either as the output of a local prediction filter, i.e.,

$$\tilde{y}_{s,t}^{(S)} = \mathbf{f}_{s,t}^{(S)T} \cdot \tilde{\mathbf{u}}_{s,t}^{(S)} \; , \tag{6.9}$$

in the case where $\hat{b}_{s,t} = 0$ (accurate prediction) or by setting it equal to the received measurement $\tilde{y}_{s,t}$ (inaccurate prediction). In the case of inaccurate prediction, the sink node must use the same adaptive algorithm as the sensor s to update its local prediction filter for sensor s, so that the two filters are equal (of course, if all channels are error free). Finally, the sink node must update the input vectors of all the prediction filters affected by $\tilde{y}_{s,t}$, that is the prediction filter for node s and the local prediction filters of all its neighbors.

It can be verified from the description of the proposed data collection protocol, that in the case where all channels are error-free, the reconstructed sequences $\tilde{y}_{n,t}^{(S)}$ at the sink node satisfy the distortion criterion

$$\max_{n,t} |\tilde{y}_{n,t}^{(S)} - y_{n,t}| \leq e \; . \tag{6.10}$$

In fact, the maximum allowed distortion parameter e offers a trade-off between accurate reconstruction of the measurements by the sink node, and the number of transmissions required. Also, some other factors, such as the degree to which the measured signals can be predicted and the specific characteristics of the adaptive algorithm used to update the coefficients of the prediction filters, may influence the performance of the proposed protocol.

TABLE 6.2: The protocol executed by the sink node.

Initialize $\mathbf{f}_{s,0}^{(S)}$, $\tilde{\mathbf{u}}_{s,0}^{(S)}$ $\quad(s = 1, 2, \ldots, N)$

Initialize μ $\hspace{5cm}$ (if LMS is used)

Initialize λ , $\mathbf{P}_{s,-1} = \delta^{-1}\mathbf{I}$ $\quad(s = 1, 2, \ldots, N)$ $\quad$ (if RLS is used)

For $t = 0$ to $+\infty$

$\quad$ For $s = 1$ to N

$\quad\quad$ Sleep(T_A seconds)

$\quad\quad$ Listen for $\tilde{y}_{s,t}$ (T_B sub-slot)

$$\hat{b}_{s,t} = \begin{cases} 0 & \text{if} \quad \tilde{y}_{s,t} \quad \text{was not detected} \\ \\ 1 & \text{if} \quad \tilde{y}_{s,t} \quad \text{was detected} \end{cases}$$

$\quad\quad$ If $\hat{b}_{s,t} = 0$

$\quad\quad\quad$ $\tilde{y}_{s,t}^{(S)} = \mathbf{f}_{s,t}^{(S)^T}\tilde{\mathbf{u}}_{s,t}^{(S)}$

$\quad\quad$ Else

$\quad\quad\quad$ $\tilde{y}_{s,t}^{(S)} = \tilde{y}_{s,t}$

$\quad\quad$ $\mathbf{f}_{s,t+1}^{(S)} = \mathbf{f}_{s,t}^{(S)} + \mu\,(y_{s,t} - \mathbf{f}_{s,t}^{(S)^T}\tilde{\mathbf{u}}_{s,t}^{(S)})\tilde{\mathbf{u}}_{s,t}^{(S)}$ $\quad$ (LMS update)

$\quad\quad\quad\quad\quad$ *OR*

$\quad\quad$ $\mathbf{k}_{s,t} = \dfrac{\lambda^{-1}\mathbf{P}_{s,t-1}\tilde{\mathbf{u}}_{s,t}^{(S)}}{1+\lambda^{-1}\tilde{\mathbf{u}}_{s,t}^{(S)^T}\mathbf{P}_{s,t-1}\tilde{\mathbf{u}}_{s,t}^{(S)}}$

$\quad\quad$ $\xi_{s,t} = \tilde{y}_{s,t}^{(S)} - \mathbf{f}_{s,t}^{(S)^T}\tilde{\mathbf{u}}_{s,t}^{(S)}$ $\hspace{3cm}$ (RLS update)

$\quad\quad$ $\mathbf{f}_{s,t+1}^{(S)} = \mathbf{f}_{s,t}^{(S)} + \mathbf{k}_{s,t}\xi_{s,t}$

$\quad\quad$ $\mathbf{P}_{s,t} = \lambda^{-1}\mathbf{P}_{s,t-1} - \lambda^{-1}\mathbf{k}_{s,t}\tilde{\mathbf{u}}_{s,t}^{(S)^T}\mathbf{P}_{s,t-1}$

$\quad\quad$ End

$\quad\quad$ Update $\tilde{\mathbf{u}}_{s,t+1}^{(S)}$ using $\tilde{y}_{s,t}^{(S)}$

$\quad\quad$ For $i=1$ to N_s

$\quad\quad\quad$ Update $\tilde{\mathbf{u}}_{k_{s,i},t+1}$ using $\tilde{y}_{s,t}^{(S)}$

$\quad\quad$ End

$\quad$ End

End

6.2.4 Cooperative neighborhood selection

Firstly, let us analyze the merits and drawbacks of having cooperation among the sensor nodes. For a given node n, cooperation with N_n neighbors actually requires N_n additional transmissions to these neighbors at each time instant. Although the energy cost of the additional inter-node transmissions can be low due to their proximity, one should also take into account the channel quality between the cooperating nodes which may introduce additional distortion to the data being sent.

On the other hand, the gains can overcome the cooperation costs in case that the

number of transmissions toward the sink is reduced due to the exploitation of high spatial correlation among the measurements in the cooperating neighborhood. Certainly, the cooperation gains are not the same for all the nodes. In fact, the relation between the values of temporal correlation among the measurements of node n on one side, and the values of their spatial correlation with the measurements of the cooperating nodes should determine how beneficial the cooperation may be. Furthermore, an additional benefit can be obtained once the transmissions toward the sink are required. As previously explained, the cooperating nodes may simultaneously transmit to the sink node; thus forming MISO channel and improving energy-efficiency.

Not surprisingly, in the simulation section it turns out that choosing the suitable cooperating neighborhood, in terms of its size and the actual nodes involved, plays a significant role in enhancing the performance of the protocol. Therefore, the optimization of the cooperating neighborhood requires (a) the knowledge of all channels among the nodes (including the sink) and (b) the knowledge of the auto- and cross- correlation functions of all nodes. Regarding the former issue, in a practical system, all the involved channels may be estimated during a training period in which all nodes participate. Initially, all the nodes would send the training sequence to the sink and all other nodes. Afterwards, the nodes would also transmit to the sink the sequences that are received from all other nodes. Consequently, the sink could estimate all the involved channels in a centralized manner. However, here we focus on the criterion (b). Hence, in Section 6.3.2, we show a simple neighborhood selection procedure based on the correlation functions of all the nodes.

6.2.5   Possible extensions

In the previous sub-sections the basic version of the new method was presented. The method can be extended to several directions with relative pros and cons. Below we provide some brief discussion of possible extensions.

1) In the proposed protocol, in case that the predicted value is not accurate enough, the node transmits the real measurement to the sink. However, since the sink has an exact replica of the filter that run on each sensor node, it may calculate the inaccurate predicted value by itself, as previously explained in Section 6.2.3. Some power can be saved by sending only this difference (the prediction error) instead of the whole measurement.

The actual measurement and its predicted value are highly correlated, so their difference has small variations and, therefore, in order to achieve a given distortion fewer bits are required.

It is well-known by basic Rate-Distortion Theory [81] that for a zero-mean Gaussian source with variance σ^2 and with squared-error distortion measure D, the rate-distortion function is given by

$$R(D) = \begin{cases} \frac{1}{2}log(\frac{\sigma^2}{D}) & 0 \leq D \leq \sigma^2 \\ \\ 0 & \text{otherwise} \end{cases} . \tag{6.11}$$

Assuming that the prediction error and the measurement signal are zero-mean Gaussian sources, for a given distortion $D \leq \sigma_e$, it can be easily shown that

$$R_e = R_y - \frac{1}{2}log(\frac{\sigma_y^2}{\sigma_e^2}) , \tag{6.12}$$

where R_e, R_y represent required bits to send the prediction error and the measurement signal, respectively, while σ_e^2, σ_y^2 are their variances.

This means that this modified protocol can achieve performance levels compared to the original at lower bit rates when the prediction errors are relatively small. The gain is expressed as the second term in (Eq. 6.12). However, providing the exact relation of this gain to the distortion criterium is not straightforward due to the following. Firstly, the distortion value influences the process of adaptive filtering and thus influences the prediction error and its variance. Secondly, only the prediction errors greater than specific distortion value are being transmitted which results in variance change of transmitted sequence comparing to the variance of the sequence of all prediction errors.

Let us provide an example of a possible gain using the measurements described in Section 6.3.2. In case that sensor 10 cooperates with sensors 9 and 4, and for the distortion value corresponding to sending the measurement with 8 bits (R_y =8 bits), the gain defined in Equation 6.12 equals to 4.1285 bits. Accordingly, 50% of power could be saved by sending only the prediction error (R_e =4 bits) instead of the measurement in this case.

2) Another approach to improving the protocol described in Section 6.2.3 could be to use only appropriately chosen delayed samples of a cooperating neighbor and not all samples in between. For instance, the highest cross-correlation between the measurements of nodes 2 and 4 arises for *delay* = 45. Therefore, it can be concluded that the adaptive predictor could have significantly less coefficients and still to be able to exploit the spatial correlation of a great delay. In order to optimize the performance of this protocol, some training period should be performed during forced vibration testing.

3) Finally, the protocol can be improved by allowing each sensor to update its own filter with the real measurement regardless of distortion criteria, but still to send the measurements to the sink and to also periodically send the filter coefficients to the sink only if the prediction was inaccurate (i.e., large change in input signal). In this scenario, the filter at the sink node has slightly worse prediction abilities. Hence, it is necessary to periodically receive the filter coefficients changes from the sensor node in order to adjust its own filter and satisfy the distortion criterion.

6.3   Numerical results

In order to demonstrate the effectiveness of the proposed algorithms, we have performed extensive experiments with real data. Although the dataset has not been designed for our protocol, the experiments show certain performance benefits of cooperation among the nodes. In particular, the acceleration measurements from the Canton Tower obtained during an earthquake have been used in order to present these cooperation gains. Toward this aim, we examine the number of transmissions toward the sink as a function of the maximum allowed absolute distortion, i.e., the value of the parameter e.

6.3.1   The Canton Tower

The Canton Tower (the Guangzhou TV and Sightseeing Tower) was constructed in 2010 in Guangzhou, China. It has already attracted the interest of several researchers [82]. It is a super-tall structure with a height of 610m. On the top level of the tower at height of 454m an antennary mast is mounted with 164m height (Fig. 6.2).

The tower is a tube-in-tube structure; the outer tube is made of steel and the inner one is a reinforced concrete tube. The two tubes are linked together by 36 floors and 4 levels of connection girders. The underground part of the tower is 10m height and consists of 2 floors with plan dimensions of 167m by 176m. The outer tube is shaped by concrete-filled-tube (CFT) columns, spaced in an oval shape, inclined vertically, and connected by hollow steel rings and braces. The oval shape dimensions varies from 60m by 80m at the underground level (altitude of -10m) to their minimum values of 20.65m by 27.5m at the altitude of 280m, and then they increase again to 40.5m by 54m at the top level of the tube (altitude of 450m). The oval shape of the top level is rotated 45 degrees horizontally relative to that of the bottom level. The top level plan is also inclined 15.5 degrees to the horizontal plane. The inner tube shape is an oval with constant dimensions along its height (14 m by 17 m), and its centroid is not that of the
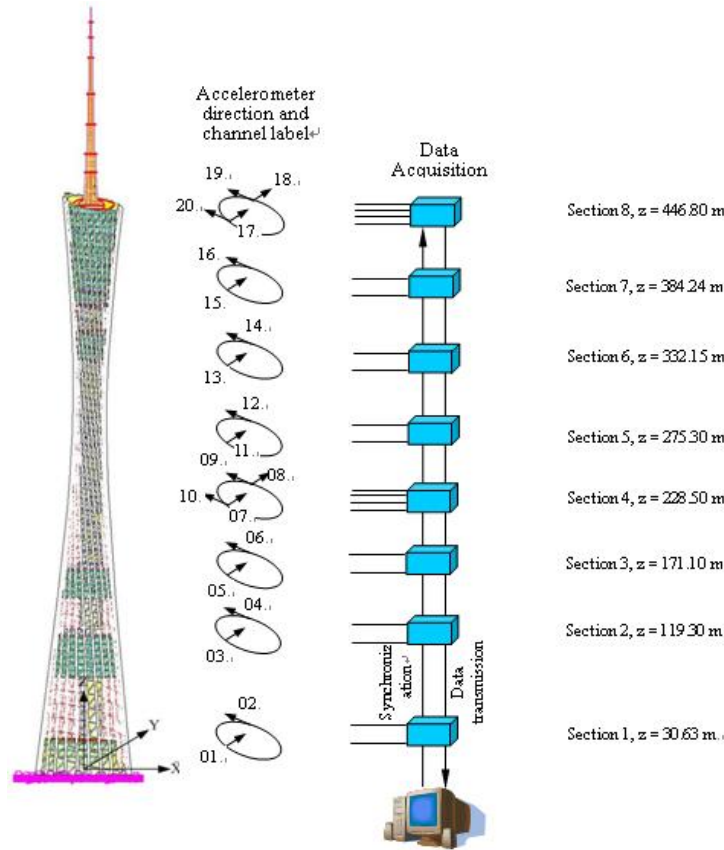
FIGURE 6.2: The distribution of accelerometers along the tower height.

outer tube. The thickness of the tube varies from 1m at the bottom to 0.4m at the top [83].

An SHM system consisting of over 600 sensors has been designed and implemented by the Hong Kong Polytechnic University for both in-construction and in-service real-time monitoring of the tower [84]. The distribution of accelerometers along the tower height is demonstrated in Figure 6.2. The dynamical response of the tower to an earthquake was recorded by 17 sensors. The measured acceleration data sequences obtained from several sensors are illustrated in Figure 6.3 for six minutes of response during an earthquake. The sampling frequency of the signal was 50 Hz.

6.3.2 Results for LMS- and RLS-based protocols

In this subsection, we analyze the gains that may be achieved by applying the two schemes described in Section 6.2.3, for different cooperation scenarios. It has been concluded that for a given distortion, the number of required transmissions from a certain node toward the sink varies with reference to (a) the number of the filter coefficients
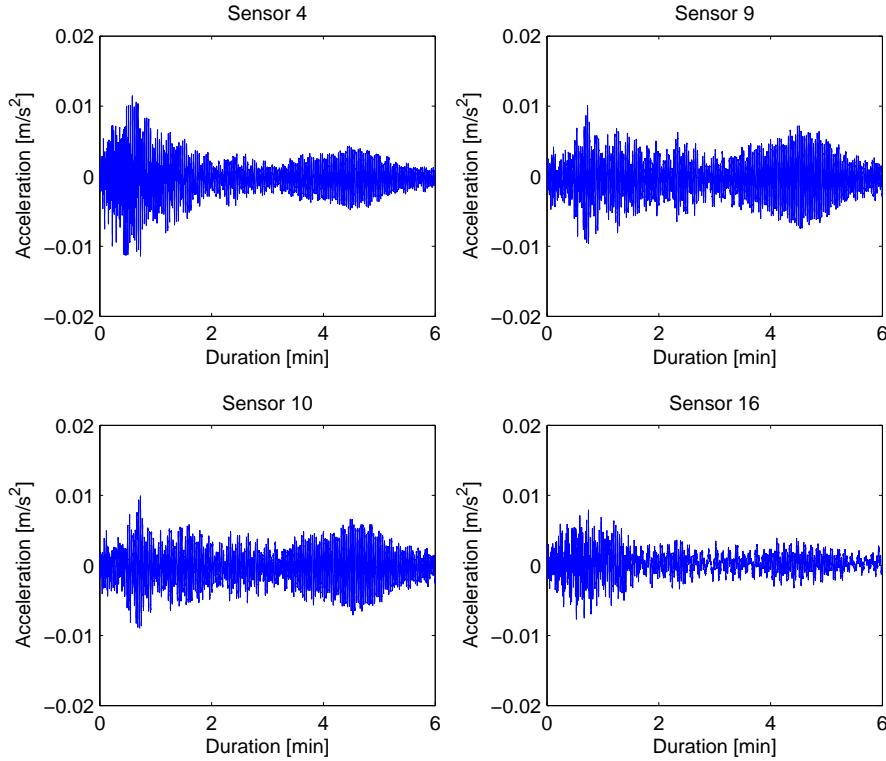
FIGURE 6.3: The measured acceleration data sequences.

of a sensor node, (b) the size of cooperating neighborhood and (c) which node(s) are selected for cooperation.

Let us first consider the performance of the proposed algorithms for different values of total filter coefficients. In particular, Figures 6.4-6.5 present the performance of the LMS-based and RLS-based algorithms employed for sensor node 9 having different filter lengths. It can be realized that the number of required transmissions varies for different filter size scenarios and that the performance does not necessarily improve as the filter length grows. Therefore, to perform a fair comparison among different cooperation scenarios, hereafter we assume the same filter lengths. For instance, for the filter length of 18, a non-cooperative node exploits its 18 past measurements. On the contrary, by cooperating with one neighbor, a node exploits 9 of its own past measurements and 9 of the neighbor's.

Next, Figure 6.6(a) compares the LMS-based and RLS-based schemes for sensor node 10, which is located in the middle of the tower, for a filter length equal to 6. In both schemes, we analyze a cooperative scenario, i.e., cooperation with sensor 9 (on the same floor as node 10), and non-cooperative, where node 10 relies only on its own measurements. It can be seen that for both protocols there is a benefit due to cooperation. Not surprisingly, for correlated input signals, the RLS-type implementation generally performs better due to the faster convergence speed and better ability to adapt
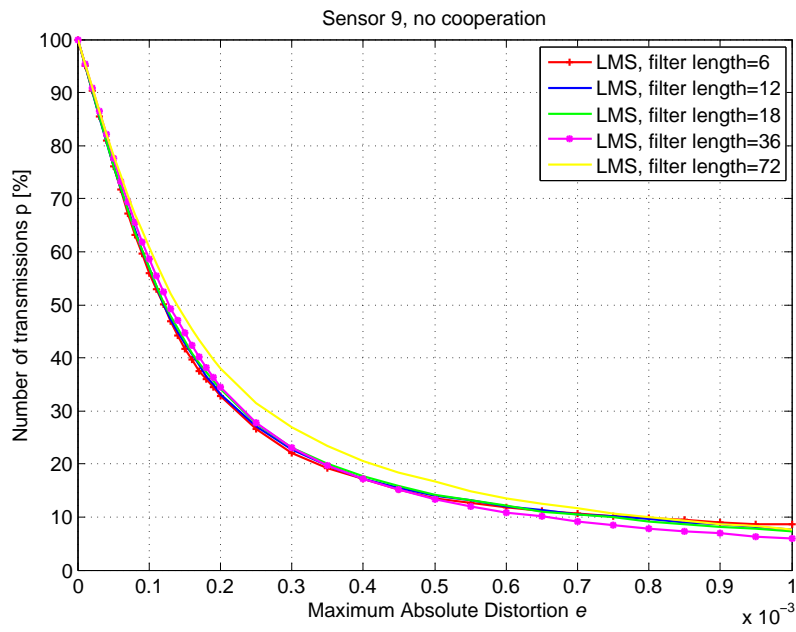
FIGURE 6.4: Performance of the LMS-based scheme for non-cooperative sensor 9 with different filter lengths.
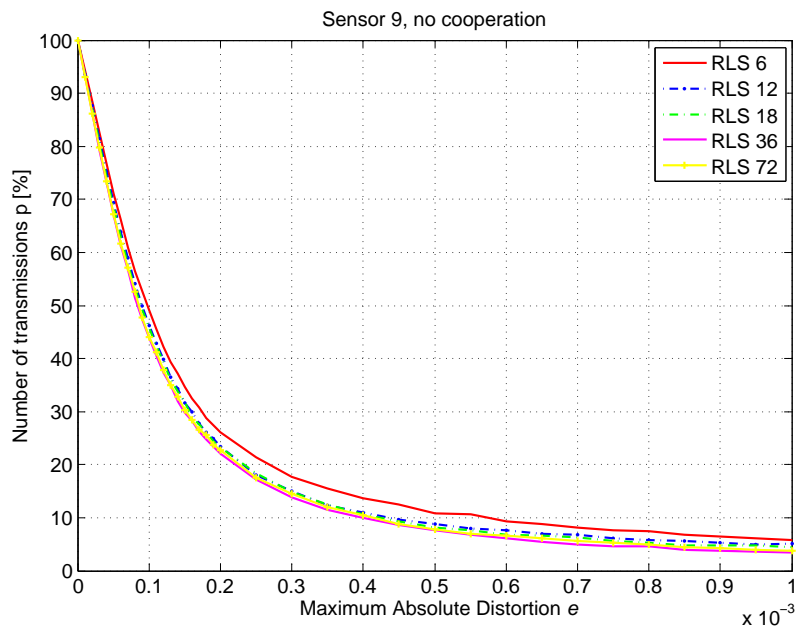

FIGURE 6.5: Performance of the RLS-based scheme for non-cooperative sensor 9 with different filter lengths.

(a) Number of filter coefficients=6   (b) Number of filter coefficients=12
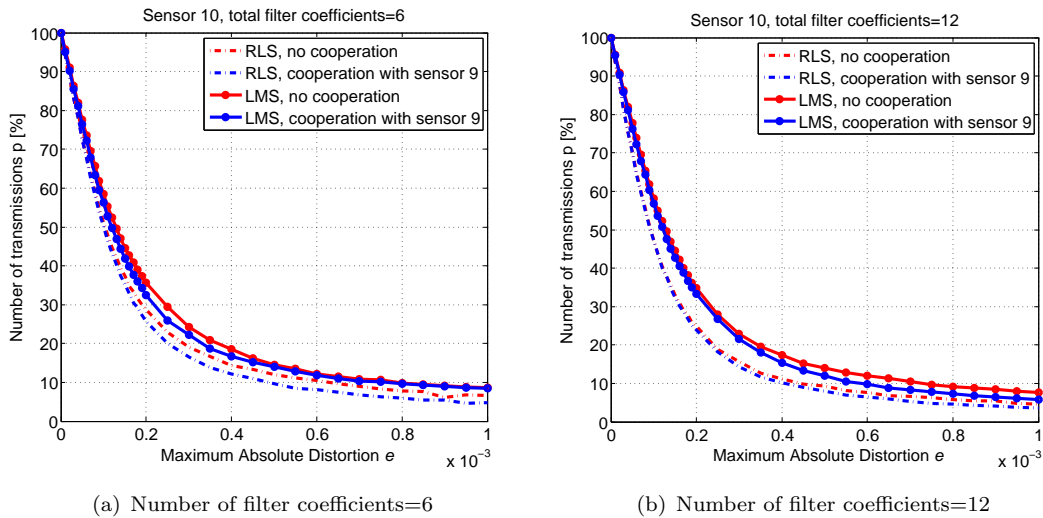
FIGURE 6.6: Performance comparison between the LMS-based and RLS-based schemes
for sensor 10.

in a fast time-varying environment. Also, RLS performs better when adaptation stalls
and restarts very often during operations. Of course, such performance gains come at an
increased computational complexity over the LMS-type implementation. In Fig. 6.6(b),
for the same simulation setting we use a greater filter length, and the cooperation gain
seems to be smaller (yet still existing).

In the following, we focus on the RLS-based scheme. Let us analyze how the per-
formance changes as the cooperating neighborhood size grows, for greater filter lengths.
The results for sensor 10 demonstrate that, in general, the performance can be improved
by increasing the number of cooperating nodes. However, in order to maximize the gains,
one should carefully select suitable cooperating neighbors; see Fig. 6.8(a).

A simple, yet effective, way to determine a suitable cooperating neighborhood is to
analyze the correlation coefficients for each node at the zero-th lag. Note that in this
setting, we do not take into account the criterion of channel quality among the nodes.
Thus, after ordering the absolute values of correlation coefficients, for a neighborhood
size of 6, one should just select the best 5 nodes ordered by this criterion. In Table 6.3,
we order 10 nodes according to their relevance to node 10. For the signals considered
in these experiments, this simple approach provides good results due to the fact that
the crosscorrelation functions are wide enough, so the zero-th lag correlation coefficients
gives enough information even for longer filter lengths. The autocorrelation function of
node 10 and its crosscorrelation with several sensors are plotted in Fig. 6.7. Note that
the cooperation gain for each sensor is dependent on the relation between the values of
its auto- and cross- correlations at the different lags. In case that its autocorrelation at
the limit lags, defined by the filter length, is greater than the crosscorrelation close to the

TABLE 6.3: The crosscorrelation coefficients between the data of sensor 10 and other sensors.

Correlation coefficient	Sensor number
0.9796	9
0.8756	12
0.8559	6
0.7688	20
0.7387	19
0.6286	4
0.4347	11
0.4199	2
0.3869	16
0.1520	17
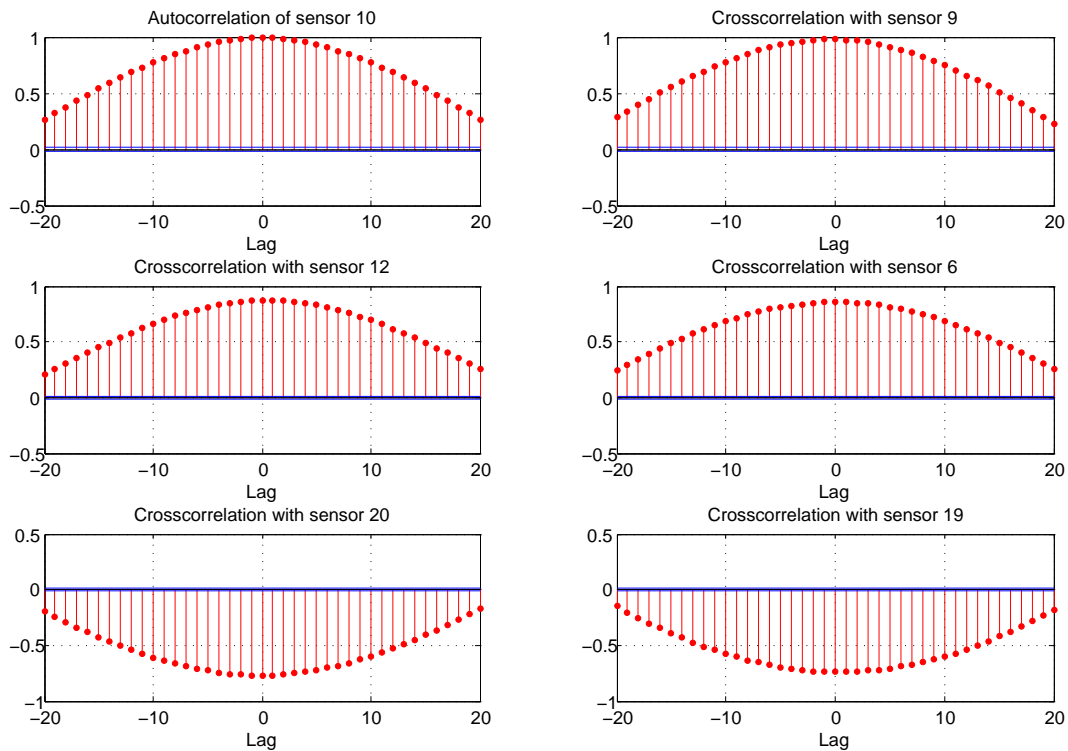
FIGURE 6.7: The autocorrelation function of node 10 (upper left) and its crosscorrelation with several sensor nodes.

zero-th lag, then for this node the cooperation will not be useful. On the other hand, when the crosscorrelation have greater values, then the neighbors add new information, so the predictor learns better the process. Furthermore, observe that the performance of the protocol may deteriorate by randomly adding nodes into the previously selected neighborhood. In Fig. 6.8(b), we illustrate this by plotting the curve obtained for a neighborhood consisting of 12 nodes (including node 10 itself). In fact, in addition to

(a) Number of filter coefficients=18   (b) Number of filter coefficients=36

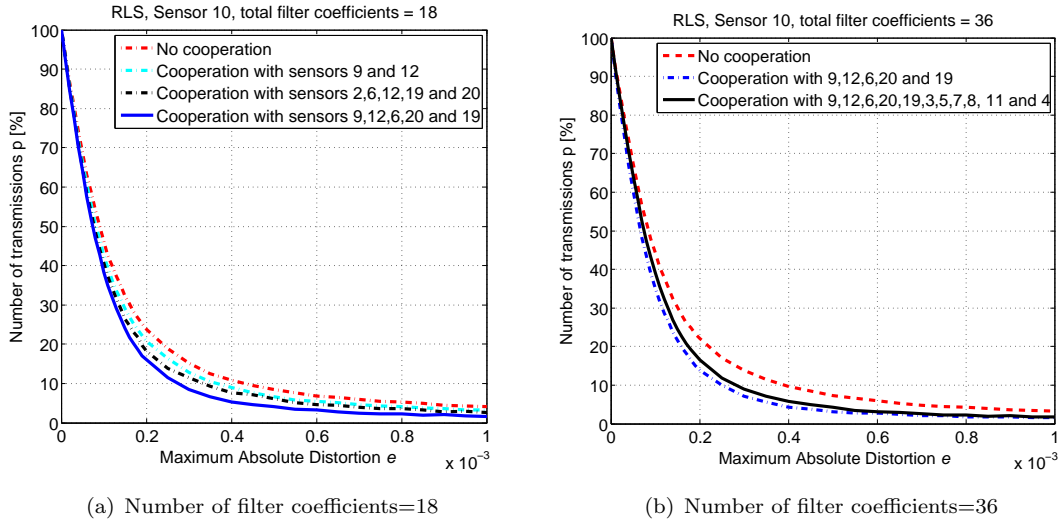FIGURE 6.8: Performance of the RLS-based scheme for different cooperating neighbor-
hoods for sensor 10.

the group of 6 nodes performing well (solid line), we added 6 other nodes which seemed
to be less relevant to sensor 10. Actually, five of these less relevant nodes measure a
different axis than node 10 (see Fig. 6.2). Due to the adaptive nature of the protocol,
they actually reduced the cooperation gains with respect to the scenario with properly
selected cooperation neighborhood.

Finally, in Fig. 6.9 we plot the averaged performance of 6 sensors where in the co-
operative case all of them cooperate with their best 5 neighbors. Although not all of
them experience cooperation benefits to the same extent, there is an average perfor-
mance improvement as compared to the non-cooperative case. To conclude, the power
consumption of a sensor node may be reduced by leveraging spatio-temporal correlations
among sensor nodes. To this end, a crucial issue to be considered is the selection of an
optimal cooperating neighborhood in terms of both its size and the nodes involved.

6.4   Concluding Remarks

A TDMA based protocol for sensor reachback in an SHM system has been described.
The proposed protocol, takes into account the fact that sensor measurements are cor-
related in space and time in order to reduce the amount of information bits needed to
transmit the measurements acquired by the sensor nodes back to a sink node, within
some prescribed distortion e. Also, the protocol does not need to know the statistics
of the event being monitored by the wireless sensor network, rather, these statistics are
learned via the use of adaptive algorithms. Furthermore, the protocol uses the idea
of cooperative communication in order to reduce the required transmission power. The
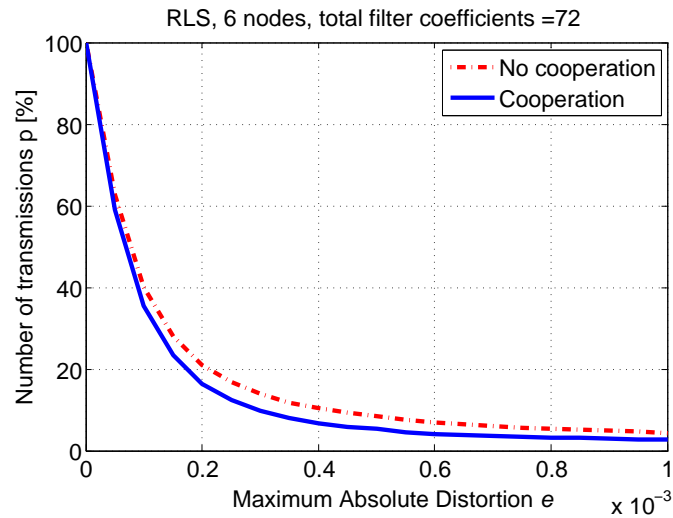
FIGURE 6.9: The averaged performance for 6 nodes.

new technique, based on both the RLS-type and the LMS-type implementation, has been tested extensively via real experimental data and it turns out that it may offer considerable saving in transmitted energy. It has been shown that the RLS-type implementation performs better when adaptation stalls and restarts very often during operations, as it is the case with the proposed technique. Furthermore, the appropriate selection of cooperating sensor nodes is of great importance.

Chapter 7

Conclusion and Open Issues

This final chapter firstly presents an overall conclusion of this dissertation. In addition, the current research lines that have been undertaken as well as recommendations for future research are provided.

This dissertation has studied the distributed processing techniques for parameter estimation and efficient data-gathering in wireless communication and sensor networks.

Regarding the parameter estimation problem, we have formulated a new node-specific paradigm where the nodes are interested in estimating parameters of local, common and/or global interest. This stands in sharp contrast with the literature in which most works are actually restricted to the cases where all the nodes have the same vectors of parameters to estimate. In the proposed node-specific parameter estimation formulation, the different nodes' interests are partially overlapping, while the non-overlapping parts can be arbitrarily different. To solve the formulated NSPE problems, several distributed adaptive strategies have been developed, i.e., Incremental NSPE LMS, Incremental NSPE RLS and Diffusion NSPE LMS. The RLS-based algorithm has been derived to solve NSPE problem where each node is interested in a set of parameters of local interest and a set of global parameters, while the LMS-based implementations have considered a more general case, where there also might be parameters common to a subsets of nodes. For the latter algorithms, the theoretical analyses related to the mean and the mean-square performance have been carried out. Simulation results have verified the theoretical findings and both algorithms have been applied to the problem of cooperative spectrum sensing in cognitive radio networks.

Concerning the data-gathering in wireless sensor networks applied for structural monitoring, the main challenges have been identified to be the huge amount of data and

possibly unreliable channels toward the sink. To mitigate these difficulties, two compression approaches, a lossless and a lossy one, have been analyzed. In the lossless approach, the potential of the distributed source coding have been studied from an information-theoretic point of view. By exploiting the optimal separation between the source coding problem and the transmission problem in the considered setting, we propose a channel-aware protocol for sequential decoding based on cooperation between the nodes. It has been shown that the proposed protocol achieves less total power consumption as well as reduced maximum power per sensor node required for a feasible power allocation to exist. In the lossy approach, the focus has been put on adaptive spatio-temporal prediction. The derived cooperative communication protocol relies on adaptive filtering techniques such as LMS and RLS. Experiments with real acceleration measurements, obtained from the Canton Tower in China during an earthquake, have demonstrated the effectiveness of the derived method for both adaptive filtering implementations.

Current work and open issues are described below:

- *A coalitional game theoretical approach to the NSPE problem*: Regarding the NSPE problem presented in Chapters 2-4, it is of interest to study it in the case where nodes or groups of nodes are allowed to be selfish using concepts from game theory.

  In general, game theory can be defined as the study of mathematical models of conflict and cooperation between intelligent rational decision-makers. It has been applied to various disciplines such as economics, political sciences, philosophy and more recently, to engineering [85]. Unlike non-cooperative game theory where the modeling unit is a single player, coalitional game theory seeks for optimal coalition structure of players in order to optimize the worth of each coalition. According to [86], coalitional games can be classified into the following three categories, i.e., canonical coalitional games, coalition formation games and coalitional graph games.

  The literature dealing with the analyses of the adaptive networks from the game theoretic perspective has been rather limited [87–90]. Most studies focus mainly, although not exclusively, on the game-theoretical approaches based on non-cooperative game theory. In all these studies, the aim of the nodes was to estimate a set of global parameters that were identical to all nodes.

  In our current work, we analyze a distributed adaptive parameter estimation problem in the framework of coalitional game theory. We consider the coalitional game to be the Non-Transferable Utility (NTU) game for which the choice of coalitional actions defines each player's payoff. The focus has been put on the following two directions: 1) to study the parameter estimation problem via diffusion strategy as a canonical game and to extend the analysis to NSPE setting, and 2) to propose

coalition formation game for NSPE setting when the coalition formation cost is considered.

- *DSC from a coalitional game theoretical perspective*: Distributed source coding has been analyzed in Chapter 5 as a lossless approach to data gathering in wireless sensor networks. As described in the previous bullet, it is of interest to examine scenarios where sources or group of sources are permitted to be selfish using the analytical tools from coalitional game theory. In some relevant works [91],[92], interpretation of rate regions in terms of users that are thought as players in a cooperative game has been done for the case where there is only one sink. Therefore, it is of interest to study scenarios with several sinks interested in different, but possibly overlapped groups of sources.

- *Algorithmic extensions*: The NSPE model and the algorithms derived in Chapters 2-4 provide a set of tools which one may readily employ in order to derive algorithms of the different types. Furthermore, a further study may encompass different extensions of the existing algorithms. For instance, the algorithms may be extended to incorporate adaptive topologies, dynamic nodes' interests, node relevance with respect to how many interests it has, different types of asynchronous behavior etc.

- *Applications*: As previously emphasized in Section 1.3 of the introductory chapter, a big part of the motivation for formulating the NSPE problem in this dissertation has been the fact that the scenario where there are only global parameters to be estimated, equal for all nodes, has a rather limited range of practical applications. In real world, it may rarely occur that there is only one phenomenon influencing the whole network or that different events can be modeled only with w^o since they have their influences over exactly the same geographical areas. Therefore, by removing this restriction, it is reasonable to expect that the range of possible applications becomes notably wider. For instance, in Chapters 2 and 4, an illustrative application has been provided, i.e., cooperative spectrum sensing in cognitive radio networks where we also account for possible local and common interferers. Next, the proposed NSPE setting seems also to fit well in scenarios related to cooperative adaptive processing among the devices that are affected by nuisance sources in a different way, e.g., cooperative active noise cancellation. Furthermore, we strongly believe that the world of multimedia represents an inexhaustible source of applications matching well with the proposed NSPE setting.

Therefore, future work should include examining all points raised above as well as other new applications that conform to the NSPE model introduced in this dissertation.

Bibliography

[1] D. Estrin, L. Girod, G. Pottie, and M. Srivastava. Instrumenting the world with wireless sensor networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2001. ICASSP 2001*, volume 4, pages 2033–2036, 2001. (Cited on pages 2, 4, and 49.)

[2] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayeed. Detection, classification, and tracking of targets in distributed sensor networks. *IEEE Signal Processing Magazine*, 19(2):17–29, 2002. (Cited on pages 2 and 4.)

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer networks*, 38(4):393–422, 2002. (Cited on pages 2 and 4.)

[4] F. Zhao, J. Liu, J. Liu, L. Guibas, and J. Reich. Collaborative signal and information processing: An information-directed approach. *Proceedings of the IEEE*, 91 (8):1199–1209, 2003. (Cited on pages 2 and 4.)

[5] S. Nikoletseas and J.D.P. Rolim. *Theoretical Aspects of Distributed Computing in Sensor Networks*. Monographs in Theoretical Computer Science. An EATCS Series. Springer, 2011. ISBN 9783642148491. (Cited on pages 2 and 4.)

[6] C. Y. Chong, S. P. Kumar, and B. A. Hamilton. Sensor networks: Evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003. (Cited on pages 2 and 4.)

[7] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49 (9):1520–1533, 2004. (Cited on page 2.)

[8] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and distributed computation: numerical methods*. Athena Scientific, Singapore, 1997.
(Cited on page 2.)

[9] I. D. Schizas, G. Mateos, and G. B. Giannakis. Distributed LMS for consensus-based in-network adaptive processing. *IEEE Transactions on Signal Processing*, 57 (6):2365–2382, 2009.
(Cited on page 2.)

[10] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione. Gossip algorithms for distributed signal processing. *Proceedings of the IEEE*, 98(11):1847–1864, 2010.
(Cited on page 2.)

[11] S. Kar and J. M. F. Moura. Consensus + innovations distributed inference over networks: cooperation and sensing in networked systems. *IEEE Signal Processing Magazine*, 30(3):99–109, 2013.
(Cited on page 2.)

[12] C. G. Lopes and A. H. Sayed. Diffusion Least-Mean Squares over adaptive networks: Formulation and performance analysis. *IEEE Transactions on Signal Processing*, 56(7):3122–3136, 2008.
(Cited on pages 3, 61, 67, and 77.)

[13] F. S. Cattivelli and A. H. Sayed. Diffusion LMS strategies for distributed estimation. *IEEE Transactions on Signal Processing*, 58(3):1035–1048, 2010.
(Cited on pages 3, 4, 11, 13, 44, 63, 66, and 77.)

[14] A. H. Sayed. Diffusion adaptation over networks. *To appear in E-Reference Signal Processing, R. Chellapa and S. Theodoridis, Eds., Elsevier, 2013*, abs/1205.4220, 2012. URL http://arxiv.org/abs/1205.4220.
(Cited on pages 3, 11, 42, 44, 69, 77, 79, and 84.)

[15] A. H. Sayed and C. G. Lopes. Distributed Recursive Least-Squares strategies over adaptive networks. In *IEEE 40th Asilomar Conference on Signals, Systems and Computers, 2006. ACSSC 2006*, pages 233–237, 2006.
(Cited on pages 3, 49, 50, 53, and 55.)

[16] S. Chouvardas, K. Slavakis, and S. Theodoridis. A novel adaptive algorithm for diffusion networks using projections onto hyperslabs. In *2010 2nd International Workshop on Cognitive Information Processing (CIP)*, pages 393–398. IEEE, 2010.
(Cited on pages 3 and 11.)

[17] S. Chouvardas, K. Slavakis, and S. Theodoridis. Adaptive robust distributed learning in diffusion sensor networks. *IEEE Transactions on Signal Processing*, 59(10): 4692–4707, 2011.
(Cited on pages 3, 11, and 44.)

[18] J. Chen and A. H. Sayed. Diffusion adaptation strategies for distributed optimization and learning over networks. *IEEE Transactions on Signal Processing*, 60(8): 4289–4305, 2012.
(Cited on pages 3, 11, and 44.)

[19] C. G. Lopes and A. H. Sayed. Incremental adaptive strategies over distributed networks. *IEEE Transactions on Signal Processing*, 55(8):4064–4077, 2007.
(Cited on pages 3, 11, 13, 15, 16, 23, 26, 44, 61, and 63.)

[20] F.S. Cattivelli and A.H. Sayed. Analysis of spatial and incremental LMS processing for distributed estimation. *IEEE Transactions on Signal Processing*, 59(4):1465–1480, 2011.
(Cited on pages 3 and 11.)

[21] L. Li, J. A. Chambers, C. G. Lopes, and A. H. Sayed. Distributed estimation over an adaptive incremental network based on the affine projection algorithm. *IEEE Transactions on Signal Processing*, 58(1):151–164, 2010.
(Cited on pages 3, 11, 13, 26, and 63.)

[22] Christos H Papadimitriou. *Computational complexity*. John Wiley and Sons Ltd., 2003.
(Cited on pages 3 and 11.)

[23] A. Bertrand and M. Moonen. Distributed adaptive node-specific signal estimation in fully connected sensor networks - part I: Sequential node updating. *IEEE Transactions on Signal Processing*, 58(10):5277–5291, 2010.
(Cited on page 3.)

[24] A. Bertrand and M. Moonen. Distributed adaptive node-specific signal estimation in fully connected sensor networks - part II: Simultaneous and asynchronous node updating. *IEEE Transactions on Signal Processing*, 58(10):5292–5306, 2010.
(Cited on page 3.)

[25] V. Kekatos and G. B. Giannakis. Distributed robust power system state estimation. *IEEE Transactions on Power Systems*, 28(2):1617–1626, 2013.
(Cited on page 3.)

[26] R. Abdolee, B. Champagne, and A. H. Sayed. Diffusion LMS for source and process estimation in sensor networks. In *IEEE/SP 17th Workshop on Statistical Signal*

Processing, 2012. SSP 2012, pages 165–168, 2012.
(Cited on pages 4 and 61.)

[27] J. Chen and A. H. Sayed. Distributed Pareto-optimal solutions via diffusion adaptation. In *IEEE Statistical Signal Processing Workshop, 2012. SSP 2012.*, pages 648–651, 2012.
(Cited on pages 4 and 61.)

[28] X. Zhao and A. H. Sayed. Clustering via diffusion adaptation over networks. In *3rd International Workshop on Cognitive Information Processing, 2012. (CIP 2012)*, pages 1–6, 2012.
(Cited on pages 4, 69, and 87.)

[29] J. Chen, C. Richard, and A. H. Sayed. Multitask diffusion adaptation over networks. *IEEE Transactions on Signal Processing*, 62(15):4129–4144, 2014.
(Cited on pages 4 and 61.)

[30] J. Chen, C. Richard, and A. H. Sayed. Diffusion LMS over multitask networks. *CoRR*, abs/1404.6813, 2014. URL http://arxiv.org/abs/1404.6813.
(Cited on page 4.)

[31] J. P. Lynch. A summary review of wireless sensors and sensor networks for structural health monitoring. *The Shock and Vibration Digest*, 38(2):91–128, 2006. URL http://svd.sagepub.com/cgi/doi/10.1177/0583102406061499.
(Cited on page 4.)

[32] J.A. Rice, K.A. Mechitov, S.H. Sim, B.F. Spencer, and G.A. Agha. Enabling framework for structural health monitoring using smart sensors. *Structural Control and Health Monitoring*, 18(5):574–587, 2011.
(Cited on page 4.)

[33] J. Barros, C. Peraki, and S.D. Servetto. Efficient network architectures for sensor reachback. In *2004 International Zurich Seminar on Communications*, pages 184 – 187, 2004. doi: 10.1109/IZS.2004.1287420.
(Cited on page 5.)

[34] T. Nagayama, P. Moinzadeh, K. Mechitov, M. Ushita, N. Makihata, M. Ieiri, G. Agha, Jr. B. F. Spencer, Fujino Y., and J. Seo. Reliable multi-hop communication for structural health monitoring, 2010.
(Cited on page 5.)

[35] J. Barros and S.D. Servetto. Network information flow with correlated sources. *IEEE Transactions on Information Theory*, 52(1):155 – 170, jan. 2006. ISSN 0018-9448. doi: 10.1109/TIT.2005.860435.
(Cited on pages 5 and 97.)

[36] V. Stankovic, L. Stankovic, and S. Cheng. Distributed source coding: Theory and applications. *Proc. Eusipco-2010 18th European Signal Processing Conference*, 2010.
(Cited on pages 6, 91, and 93.)

[37] A. Roumy and D. Gesbert. Optimal matching in wireless sensor networks. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):725–735, 2007.
(Cited on pages 6, 91, 94, 97, and 101.)

[38] S. Li and A. Ramamoorthy. Rate and power allocation under the pairwise distributed source coding constraint. *IEEE Transactions on Communications*, 57(12): 3771–3781, 2009.
(Cited on pages 6, 91, 97, 98, and 100.)

[39] T. Srisooksai, K. Keamarungsi, P. Lamsrichan, and K. Araki. Practical data compression in wireless sensor networks: A survey. *Journal of Network and Computer Applications*, 35(1):37–59, 2012.
(Cited on pages 6 and 105.)

[40] N. Bogdanović, J. Plata-Chaves, and K. Berberidis. Distributed incremental-based lms for node-specific parameter estimation over adaptive networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5425–5429, May 2013. doi: 10.1109/ICASSP.2013.6638700.
(Cited on pages 6, 7, 49, 50, 58, 63, and 66.)

[41] J. Plata-Chaves, N. Bogdanović, and K. Berberidis. Distributed incremental-based rls for node-specific parameter estimation over adaptive networks. In *2013 Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*, pages 1–5, Sept 2013.
(Cited on pages 6, 7, and 63.)

[42] N. Bogdanović, J. Plata-Chaves, and K. Berberidis. Distributed diffusion-based lms for node-specific parameter estimation over adaptive networks. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7223–7227, May 2014. doi: 10.1109/ICASSP.2014.6855002.
(Cited on pages 6 and 7.)

[43] N. Bogdanović, J. Plata-Chaves, and K. Berberidis. Distributed incremental-based lms for node-specific adaptive parameter estimation. *IEEE Transactions on Signal Processing*, 62(20):5382–5397, Oct 2014. ISSN 1053-587X. doi: 10.1109/TSP.2014. 2350965.
(Cited on pages 6, 7, 8, 63, 64, 65, 66, and 88.)

[44] J. Plata-Chaves, N. Bogdanović, and K. Berberidis. Distributed diffusion-based lms for node-specific adaptive parameter estimation. *Submitted to IEEE Transactions on Signal Processing*, abs/1408.3354, 2014. URL http://arxiv.org/abs/1408.3354.
(Cited on pages 6, 7, and 8.)

[45] N. Bogdanović, D. Ampeliotis, and K. Berberidis. Cooperative transmission of measurements in wsn for monitoring applications. In *2012 IEEE International Conference on Industrial Technology (ICIT)*, pages 356–361. IEEE, 2012.
(Cited on page 8.)

[46] D. Ampeliotis, N. Bogdanović, K. Berberidis, F. Casciati, and R. Al-Saleh. Power-efficient wireless sensor reachback for shm. In *Proceedings of the Sixth International IABMAS Conference, Stresa, Lake Maggiore, Italy*, 8-12 July 2012.
(Cited on page 8.)

[47] N. Bogdanović, D. Ampeliotis, K. Berberidis, F. Casciati, and J. Plata-Chaves. Structural health monitoring of a cable-stayed bridge using smart sensor technology: deployment and evaluation. *Smart Structures and Systems*, 14(2):1–16, 2014.
(Cited on page 8.)

[48] U. Yildirim, O. Oguz, and N. Bogdanović. A prediction-error-based method for data transmission and damage detection in wireless sensor networks for structural health monitoring. *Journal of Vibration and Control*, 19(15):2244–2254, 2013.
(Cited on page 8.)

[49] D. P. Bertsekas. A new class of incremental gradient methods for least squares problems. *SIAM Journal on Optimization*, 7(4):913–926, 1997.
(Cited on pages 11 and 16.)

[50] A. Nedic and D. P. Bertsekas. Incremental subgradient methods for nondifferentiable optimization. *SIAM Journal on Optimization*, 12(1):109–138, 2001.
(Cited on page 11.)

[51] M. G. Rabbat and R. D. Nowak. Quantized incremental algorithms for distributed optimization. *IEEE Journal on Selected Areas in Communications*, 23(4):798–808,

2005.

(Cited on pages 11 and 16.)

[52] A. H. Sayed. *Adaptive filters*. Wiley-IEEE Press, 2011.

(Cited on pages 15, 16, 23, 25, 26, 49, 53, 65, 79, 105, 108, and 110.)

[53] R. A. Horn and C. R. Johnson, editors. *Matrix Analysis*. Cambridge University Press, New York, NY, USA, 1986.

(Cited on pages 20 and 21.)

[54] M. Godavarti and A. O. Hero. Partial update LMS algorithms. *IEEE Transactions on Signal Processing*, 53(7):2382–2399, 2005.

(Cited on pages 20 and 21.)

[55] A.R. Meenakshi and C. Rajian. On a product of positive semidefinite matrices. *Linear algebra and its applications*, 295(1):3–6, 1999.

(Cited on page 22.)

[56] R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge university press, 2012.

(Cited on page 22.)

[57] P. Di Lorenzo, S. Barbarossa, and A. H. Sayed. Bio-inspired swarming for dynamic radio access based on diffusion adaptation. In *IEEE 19th European Signal Conference, 2011. EUSIPCO 2011*, pages 1–6, 2012.

(Cited on pages 42 and 43.)

[58] P. Di Lorenzo, S. Barbarossa, and A. H. Sayed. Bio-inspired decentralized radio access based on swarming mechanisms over adaptive networks. *IEEE Transactions on Signal Processing*, 61(12):3183–3197, 2013.

(Cited on pages 42 and 84.)

[59] J. A. Bazerque and G. B. Giannakis. Distributed spectrum sensing for cognitive radio networks by exploiting sparsity. *IEEE Transactions on Signal Processing*, 58 (3):1847–1862, 2010.

(Cited on pages 42 and 43.)

[60] P. Di Lorenzo, S. Barbarossa, and A. H. Sayed. Distributed spectrum estimation for small cell networks based on sparse diffusion adaptation. *IEEE Signal Processing Letters*, 20(12):1261–1265, 2013.

(Cited on pages 42, 43, 84, and 86.)

[61] M. H. Hayes. *Statistical digital signal processing and modeling*. John Wiley & Sons, 2009.

(Cited on pages 49, 105, and 110.)

[62] G. Mateos, I. D. Schizas, and G. B. Giannakis. Distributed Recursive Least-Squares for consensus-based in-network adaptive estimation. *IEEE Transactions on Signal Processing*, 57(11):4583–4588, 2009.
(Cited on page 49.)

[63] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed. Diffusion Recursive Least-Squares for distributed estimation over adaptive networks. *IEEE Transactions on Signal Processing*, 56(5):1865–1877, 2008.
(Cited on pages 49 and 50.)

[64] A.K. Gupta and D.K. Nagar. *Matrix variate distributions*. CRC Press, 1999.
(Cited on page 76.)

[65] N. Takahashi, I. Yamada, and A. H. Sayed. Diffusion Least-Mean Squares with adaptive combiners: Formulation and performance analysis. *IEEE Transactions on Signal Processing*, 58(9):4795–4810, 2010.
(Cited on pages 77 and 78.)

[66] H. Neudecker and T. Wansbeek. Fourth-order properties of normally distributed random matrices. *Linear Algebra and Its Applications*, 97:13–21, 1987.
(Cited on pages 80 and 81.)

[67] J.R. Magnus and H. Neudecker. The commutation matrix: some properties and applications. *The Annals of Statistics*, pages 381–394, 1979.
(Cited on page 81.)

[68] D. Von Rosen. Moments for matrix normal variables. *Statistics: A Journal of Theoretical and Applied Statistics*, 19(4):575–583, 1988.
(Cited on page 81.)

[69] S. Liu. Matrix results on the Khatri-Rao and Tracy-Singh products. *Linear Algebra and its Applications*, 289(1):267–277, 1999.
(Cited on page 82.)

[70] D. Slepian and J. K. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, 19(4):471–480, 1973.
(Cited on page 93.)

[71] R. Cristescu, B. Beferull-Lozano, and M. Vetterli. Networked slepian-wolf: theory, algorithms, and scaling laws. *IEEE Transactions on Information Theory*, 51(12): 4057–4073, 2005.
(Cited on page 94.)

[72] A. Ramamoorthy. Minimum cost distributed source coding over a network. *IEEE Transactions on Information Theory*, 57(1):461–475, 2011.
(Cited on page 94.)

[73] V. Stankovic, A. D. Liveris, Z. Xiong, and C. N. Georghiades. On code design for the slepian-wolf problem and lossless multiterminal networks. *IEEE Transactions on Information Theory*, 52(4):1495–1507, 2006.
(Cited on page 94.)

[74] V. Toto-Zarasoa, A. Roumy, and C. Guillemot. Rate-adaptive codes for the entire slepian-wolf region and arbitrarily correlated sources. In *2008. IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP*, pages 2965–2968. IEEE, 2008.
(Cited on page 94.)

[75] D. Marco and D. L. Neuhoff. Reliability vs. efficiency in distributed source coding for field-gathering sensor networks. In *Proceedings of the 3rd international symposium on Information processing in sensor networks*, pages 161–168. ACM, 2004.
(Cited on page 96.)

[76] Y-W. Hong, W.-J. Huang, F-H. Chiu, and C.-C.J. Kuo. Cooperative communications in resource-constrained wireless networks. *IEEE Signal Processing Magazine*, 24(3):47 –57, May 2007. ISSN 1053-5888. doi: 10.1109/MSP.2007.361601.
(Cited on pages 100 and 109.)

[77] F. Huang and Y. Liang. Towards energy optimization in environmental wireless sensor networks for lossless and reliable data gathering. In *MASS 2007. IEEE internatonal conference on Mobile adhoc and sensor systems*, pages 1–6. IEEE, 2007.
(Cited on page 105.)

[78] Y. Liang and W. Peng. Minimizing energy consumptions in wireless sensor networks via two-modal transmission. *ACM SIGCOMM Computer Communication Review*, 40(1):12–18, 2010.
(Cited on page 105.)

[79] F. Marcelloni and M. Vecchio. Enabling energy-efficient and lossy-aware data compression in wireless sensor networks by multi-objective evolutionary optimization. *Information Sciences*, 180(10):1924–1941, 2010.
(Cited on page 105.)

[80] S. Cui, A. J. Goldsmith, and A. Bahai. Energy-efficiency of mimo and co-
     operative mimo techniques in sensor networks. *IEEE Journal on Selected Ar-
     eas in Communications*, 22(6):1089 – 1098, aug. 2004. ISSN 0733-8716. doi:
     10.1109/JSAC.2004.830916.
     (Cited on pages 106 and 108.)

[81] J. G. Proakis and M. Salehi. *Communication Systems Engineering*. Prentice-Hall,
     Upper Saddle River, NJ, USA, second edition, August 2001. ISBN 0130617938.
     (Cited on page 115.)

[82] F. Casciati, R. Alsaleh, and C. Fuggini. Gps-based shm of a tall building: torsional
     effects. *Proceedings of the 7th International Workshop on Structural Health Moni-
     toring*, pages 340–347, 2009.
     (Cited on page 116.)

[83] Y. Q. Ni, Y. Xia, W. Y. Liao, and J. M. Ko. Technology innovation in developing
     the structural health monitoring system for guangzhou new tv tower. *Structural
     Control and Health Monitoring*, 16(1):73–98, 2009.
     (Cited on page 117.)

[84] Benchmark. A benchmark problem for the structural health monitoring of high-rise
     slender structures. *http://www.cse.polyu.edu.hk/benchmark/index.htm*, 2008.
     (Cited on page 117.)

[85] R. Myerson. *Game theory*. Harvard university press, 2013.
     (Cited on page 126.)

[86] W. Saad, Z. Han, M. Debbah, A. Hjorungnes, and T. Basar. Coalitional game
     theory for communication networks. *IEEE Signal Processing Magazine*, 26(5):77–
     97, 2009.
     (Cited on page 126.)

[87] C. Yu, M. van der Schaar, and A.H. Sayed. Reputation design for adaptive networks
     with selfish agents. *2013 IEEE 14th Workshop on Signal Processing Advances in
     Wireless Communications (SPAWC)*, pages 160–164, June 2013. doi: 10.1109/
     SPAWC.2013.6612032.
     (Cited on page 126.)

[88] C. Yu, M. van der Schaar, and A.H. Sayed. Cluster formation over adaptive net-
     works with selfish agents. In *2013 Proceedings of the 21st European Signal Process-
     ing Conference (EUSIPCO)*, pages 1–5, 2013.
     (Cited on page 126.)

[89] O. N. Gharehshiran, V. Krishnamurthy, and G. Yin. Distributed energy-aware diffusion least mean squares: Game-theoretic learning. *IEEE Journal of Selected Topics in Signal Processing*, 7(5):821–836, Oct 2013. ISSN 1932-4553. doi: 10.1109/JSTSP.2013.2266318.
(Cited on page 126.)

[90] C. Jiang, Y. Chen, and KJ R. Liu. Distributed adaptive networks: A graphical evolutionary game-theoretic view. *IEEE Transactions on Signal Processing*, 61 (22):5675–5688, 2013.
(Cited on page 126.)

[91] M. Madiman. Playing games: A fresh look at rate and capacity regions. In *ISIT 2008. IEEE International Symposium on Information Theory*, pages 2528–2532, 2008.
(Cited on page 127.)

[92] M. Madiman. Cores of cooperative games in information theory. *EURASIP Journal on Wireless Communications and Networking*, 2008.
(Cited on page 127.)